A multi-scale information geometry reveals the structure of mutual information in neural populations

Simone Azeglio
Institut de la Vision, INSERM, CNRS, & Laboratoire des Systèmes Perceptifs
Sorbonne University & École Normale Supérieure - PSL
17 Rue Moreau, Paris 75012 & 29 Rue d'Ulm Paris 75005
simone.azeglio@gmail.com

Steeve Laquitaine
Institut de la Vision, INSERM, CNRS
Sorbonne Université
17 Rue Moreau, Paris 75012

Ulisse Ferrari
Institut de la Vision, INSERM, CNRS
Sorbonne Université
17 Rue Moreau, Paris 75012

Matthew Chalk
Institut de la Vision, INSERM, CNRS
Sorbonne Université
17 Rue Moreau, Paris 75012
matthew.chalk@inserm.fr

Abstract

Understanding how neural population responses represent sensory information is a central problem in systems neuroscience. One approach is to define a representational geometry on stimulus space in which distances reflect how reliably stimuli can be distinguished from neural activity. However, different constructions of these distances can lead to qualitatively different conclusions about the neural code. Here, we show that a unique Riemannian representational geometry emerges from first principles governing how distances contract as stimulus resolution is lost through coarse-graining. This results in a multi-scale extension of the Fisher information metric, capturing encoding structure from fine stimulus details to coarse global distinctions. The resulting geometry is exactly related to the mutual information encoded by the population: well-encoded stimulus directions (those contributing more to mutual information) are expanded, whereas poorly encoded directions are contracted. The metric tensor can be estimated using diffusion models, making the framework practical for large neural populations and high-dimensional stimuli. Applied to visual cortical responses to natural images, the eigenvectors of the metric tensor identify stimulus variations that contribute most to the encoded stimulus information, yielding interpretable features that are robust to modelling choices. Together, these results provide a principled, information-theoretic framework for characterising neural population codes. Code and project page: multigeom.github.io

1 Introduction

How do neural populations represent sensory stimuli? A promising approach is to define a representational geometry over stimulus space, in which distances between stimuli reflect how reliably the population distinguishes them. Such geometries have been used to compare representations across

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

systems and stages of processing [1, 2], track changes in representation during learning [3, 4], and compare the internal representations of biological and artificial networks [5, 6]. However, many ways exist to define such a geometry from neural responses, and different choices can lead to qualitatively different conclusions about what a population encodes.

Here we show that a unique representational geometry emerges from first principles governing how representational distances transform as stimulus resolution is reduced. Because neural responses are noisy, we reason that the geometry should depend on the full response distribution rather than treating responses as deterministic. The canonical Riemannian geometry for distributions is constructed from the Fisher information metric, which characterizes how sensitive the response distribution is to infinitesimal stimulus perturbations [7]. However, Fisher information is purely local: two codes can have identical Fisher information everywhere yet differ markedly in their ability to distinguish distant stimuli [8]. We therefore seek a geometry that captures how populations encode stimulus differences at all scales, not only locally. This geometry is uniquely determined by simple requirements on how representational distances change under stimulus coarse-graining, whereby fine stimulus details become progressively harder to discriminate. The resulting geometry provides a multi-scale extension of the Fisher metric, capturing encoding structure from fine discriminations between nearby stimuli to coarse global distinctions.

Our geometry has an exact relationship to the mutual information encoded by the population: distances between stimuli are locally expanded along well-encoded directions, which contribute most to mutual information, and contracted along poorly encoded ones. We show, moreover, that the metric tensor can be estimated for high-dimensional natural stimuli using diffusion models [9, 10], making the framework practical for neural populations responding to complex natural stimuli.

Our construction builds on identities relating mutual information to the Fisher information, or equivalently the minimum mean squared error, of coarse-grained stimuli across diffusion scales [11–14]. Previous work has used these identities to decompose mutual information into scalar contributions from individual stimuli or stimulus-response pairs [14–16]. In contrast, here we define a representational geometry underlying mutual information, whose distances, geodesics and principal directions characterise how information is organised across stimulus space. Moreover, previous decompositions start from mutual information and must choose among many possible ways to decompose it. Instead, we show that our geometry emerges uniquely from simple axioms on how representational distances contract under coarse-graining. The link to mutual information, which is not mentioned in the axioms, thus emerges from the derivation rather than being put in by hand.

We apply our approach to recorded neural populations in visual cortex. The eigenvectors of the metric tensor identify the stimulus directions stretched by the geometry, revealing which stimulus variations contribute most to the mutual information. These directions yield interpretable image features that differ systematically across visual areas. Because the geometry emphasises stimulus directions that contribute to encoded information, and are therefore constrained by the data, these features reflect robust properties of the neural code rather than details of a fitted model. In contrast, we show that Fisher information can be sensitive to stimulus directions that contribute little to encoded information and are therefore poorly constrained. Together, our results establish a principled multi-scale geometry that reveals the structure of encoded information in neural population codes.

2 From Fisher information to a multi-scale geometry of neural coding

Consider a continuous stimulus $x \sim p(x)$, which elicits a neural population response r with likelihood $p(r \mid x)$. Our goal is to define a geometry on stimulus space in which distances reflect how reliably different stimuli can be distinguished from neural activity. This can be formalized by endowing stimulus space with a Riemannian metric, with infinitesimal line element

$$ds^2 = dx^\top G(x)\, dx, \tag{1}$$

where $G(x)$ is some positive semi-definite metric tensor.

A classical result from information geometry, Čencov's theorem [7], shows that under mild conditions on the family of distributions $p(r|x)$ indexed by x[1], the unique metric (up to scale) depending locally on $p(r \mid x)$ and its first derivatives at x, while also invariant to transformations of r that preserve

[1]The support of r does not depend on x, and $p(r \mid x)$ is everywhere differentiable in x with finite Fisher.

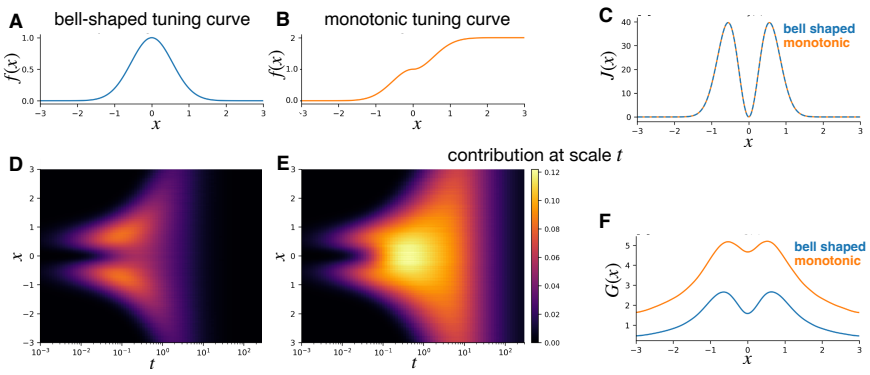

Figure 1: **Fisher and multi-scale Fisher for a one-dimensional stimulus.** (**A**–**B**) Bell-shaped and monotonic tuning curves ($\mathbb{E}[r \mid x] = f(x)$) for a single neuron with stimulus-independent Gaussian noise, encoding a scalar stimulus x with uniform prior. (**C**) The Fisher information is identical for both tuning curves, since they share the same local gradient magnitude and Gaussian noise variance. (**D**–**E**) Contribution at each diffusion scale, $[\phi_t * J_t](x)\,\delta t$, to the multi-scale Fisher, for the bell-shaped (D) and monotonic (E) tuning curves. (**F**) The multi-scale Fisher is larger everywhere for the monotonic tuning curve, reflecting that it encodes strictly more mutual information.

its sufficient statistics (i.e. to reparameterisations that retain all stimulus information), is the Fisher information matrix $G(x) \propto J(x)$,

$$J(x) = \mathbb{E}_{p(r|x)}\left[\nabla_x \log p(r \mid x)\, \nabla_x \log p(r \mid x)^\top\right]. \tag{2}$$

The Fisher information matrix thus defines the canonical Riemannian geometry over the family of response distributions, $p(r \mid x)$. It quantifies how sensitive $p(r \mid x)$ is to infinitesimal stimulus perturbations through the relation, $D_{\mathrm{KL}}(p(r \mid x) \| p(r \mid x + dx)) = \frac{1}{2} dx^\top J(x)\, dx + O(\|dx\|^3)$. Geometrically, the Fisher thus deforms stimulus space so that regions where nearby stimuli elicit more distinguishable responses are stretched further apart.

However, two neural codes can share identical local structure – equal sensitivity to infinitesimal stimulus perturbations – while differing arbitrarily in global structure, and thus in how well they distinguish distant stimuli [8]. For example, for a scalar stimulus with Gaussian observation noise, where $J(x) \propto f'(x)^2$ and $f(x) = \mathbb{E}[r \mid x]$ is the tuning curve, two tuning curves can share the same $|f'(x)|$ everywhere, yielding identical Fisher information at every stimulus, yet encode very different information about the stimulus globally (Fig. 1A–C). This global structure matters because what can be inferred from a response depends on the posterior, $p(x \mid r) \propto p(r \mid x)\, p(x)$, whose normalisation depends on how neurons respond to all stimuli, not only at x. Indeed, it is due to its global shape that the monotonic tuning curve (Fig. 1B) distinguishes whether x is larger or smaller than zero, while the bell-shaped tuning curve (Fig. 1A) does not.

Here we seek a geometry that reflects what the population encodes about the stimulus beyond fine discrimination captured by the Fisher. Requiring the metric itself to depend only on $p(r \mid x)$ and its derivatives at x would force it to be the Fisher information (Čencov's theorem); instead, we impose this locality on how the metric *changes* under coarse-graining. To uniquely define such a geometry, we consider how discriminability is affected by the loss of stimulus resolution induced by coarse-graining with isotropic Gaussian noise:

$$x_t = x + \sqrt{t}\, z, \qquad z \sim \mathcal{N}(0, I). \tag{3}$$

This coarse-graining is the unique choice that is invariant with respect to translations and rotations of stimulus space, and satisfies the semigroup property, whereby coarse-graining by t_1 and then t_2 is equivalent to coarse-graining once by $t_1 + t_2$ [17].

Note that, while we consider an unbounded stimulus space, $x \in \mathbb{R}^{n_x}$, the framework extends to periodic stimuli, such as orientation, by replacing Eq. (3) with diffusion on the circle (i.e. a wrapped Gaussian kernel), which is likewise rotation invariant and satisfies the semigroup property (Appendix E).

The Gaussian kernel mixes only an $O(\sqrt{t})$ neighbourhood around x, so in the limit $t \to 0$ it reduces the ability of the neural code to discriminate infinitesimally close stimuli while leaving discrimination of distant stimuli unaffected. This discrimination of nearby stimuli, eroded by coarse-graining, depends only on the local shape of $p(r \mid x)$. We therefore require that (i) distances in our geometry contract under coarse-graining, since coarse-graining reduces discriminability, (ii) in the limit $t \to 0$ this contraction depends locally on $p(r \mid x)$ and its derivatives, and (iii) the contraction depends only on what r encodes about the stimulus, and is thus invariant to sufficiency-preserving transformations of r. Next, we show formally how imposing these requirements on how the geometry *changes* under coarse-graining, rather than on the metric itself (as for the Fisher), uniquely determines a metric that integrates discriminability across stimulus scales.

3 Deriving the geometry from coarse-graining

Under the isotropic Gaussian diffusion of the stimulus described by Eq. (3), the joint density $p_t(r, x)$ evolves according to the heat equation,

$$\frac{\partial}{\partial t} p_t(r, x) = \frac{1}{2} \Delta_x p_t(r, x), \tag{4}$$

with initial condition $p_0(r, x) := p(r \mid x)\, p(x)$. This induces a scale-dependent family of likelihoods $p_t(r \mid x) = p_t(r, x) / \int p_t(r, x)\, dr$.

Let $G_t(x)$ denote some Riemannian metric on stimulus space constructed from the diffused joint distribution $p_t(r, x)$. Since coarse-graining renders x indistinguishable from stimuli within an $O(\sqrt{t})$ neighbourhood, the effective metric at x is the average of G_t over this uncertainty. The rate of change of this effective metric, evaluated at $t = 0$, is

$$Q(x) := \frac{d}{dt}\, \mathbb{E}_z\big[G_t(x + \sqrt{t}\, z)\big]\Big|_{t=0}, \quad z \sim \mathcal{N}(0, I), \tag{5}$$

so that $dx^\top Q(x)\, dx$ is the rate of change of the squared line element. Performing a second-order Taylor expansion of $G_t(x + \sqrt{t}\, z)$, taking the expectation over z, and then differentiating with respect to t (Appendix A.1),

$$Q(x) = \left(\frac{\partial}{\partial t} + \frac{1}{2}\Delta_x\right) G_t(x)\Big|_{t=0}, \tag{6}$$

where the Laplacian operator Δ_x acts component-wise on $G_t(x)$.

To constrain $Q(x)$, we impose three requirements:

1. **Contraction.** Since coarse-graining reduces what can be inferred from neural responses, local distances should contract, i.e. $dx^\top Q(x)\, dx \leq 0$ for all dx and x. Moreover, this contraction should not depend on the choice of stimulus coordinates.
2. **Locality.** $Q(x)$ depends only on $p(r \mid x)$ and its first derivatives at x.
3. **Sufficiency.** Contraction under coarse-graining should depend only on what neural responses convey about the stimulus, not on how the response variable is parameterized. Thus, $Q(x)$ is invariant under any transformation $r \to r'$ that preserves the sufficient statistics of $p(r \mid x)$.

Requirement 1, that $Q(x) \preceq 0$ and transforms covariantly under changes of coordinates (Appendix A.1), implies that the geometry defined by the metric tensor $G(x)$ is Riemannian. However, Gaussian diffusion cannot be isotropic in every coordinate system, so we define it in one set of base coordinates (image pixels in Section 6). Expressed in other coordinates, the geometry is still Riemannian, but the underlying diffusion process is no longer isotropic. The uniqueness of $G(x)$ shown in this section thus holds given a choice of coordinates in which the diffusion process is isotropic Gaussian.

Requirements 2 and 3, together with the positive semi-definiteness of $-Q(x)$ (and mild regularity assumptions on $p(r|x)$ stated earlier), are precisely the conditions of Čencov's theorem [7] (Appendix A.1). It follows that $-Q(x)$ must be proportional to the Fisher information,

$$-Q(x) = \alpha J(x), \tag{7}$$

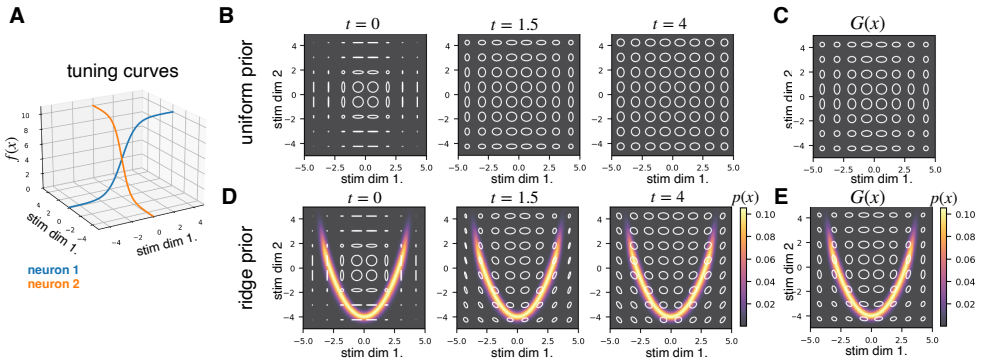

Figure 2: **Multi-scale geometry for a two-dimensional stimulus.** (**A**) Tuning curves $f(x)$ for two neurons with stimulus-independent Gaussian noise, encoding a two-dimensional stimulus. (**B**) Contribution at scales $t = 0, 1.5$ and 4 to the multi-scale Fisher, with uniform prior. Ellipses visualize the local metric $[\phi_t * J_t](x)\,\delta t$ at equally spaced stimulus values. (**C**) Multi-scale Fisher $G(x)$, integrated over all scales. (**D–E**) Same as B–C with a non-uniform ridge-shaped prior (shown in background). For $t > 0$, ellipses align with the ridge of the prior.

where $\alpha > 0$ is an arbitrary positive constant.

This relation holds for any likelihood satisfying the regularity conditions above. Since these are preserved under Gaussian diffusion, we can apply it to the diffused likelihood $p_t(r \mid x)$ at each scale t, yielding $-Q_t(x) = \alpha J_t(x)$, where $J_t(x)$ is the Fisher information constructed from $p_t(r \mid x)$. Thus, Eq. (6) can be rearranged to give, for all $t \geq 0$,

$$\frac{\partial}{\partial t} G_t(x) = -\frac{1}{2} \Delta_x G_t(x) - \alpha J_t(x). \tag{8}$$

This determines the evolution of the metric, but not the metric itself. To uniquely determine $G(x) := G_0(x)$, we must impose one additional requirement:

4. **Zero distance baseline.** If r is independent of x, then $G(x) = 0$ everywhere.

As $t \to \infty$, the response distribution becomes independent of the stimulus, so that $\lim_{t\to\infty} p_t(r \mid x) = p(r)$. From the above, this implies

$$\lim_{t\to\infty} G_t(x) = 0 \quad \forall x \in \mathcal{X} \tag{9}$$

Solving Eq. (8) backward from the above boundary condition (Appendix A.1) then yields a unique solution, up to an overall scaling factor,

$$\boxed{G(x) = \alpha \int_0^\infty [\phi_t * J_t](x)\, dt,} \tag{10}$$

where ϕ_t is an isotropic Gaussian kernel of variance t, and $*$ denotes convolution. We henceforth set $\alpha = 1$ without loss of generality, noting that this choice amounts only to an overall scaling of the metric. We call $G(x)$ the *multi-scale Fisher information matrix*, since it integrates the Fisher information across all coarse-graining scales.

Returning to our previous example, although the Fisher information is identical for both tuning curves (Fig. 1A–C), the Fisher information with respect to the *diffused* distribution $J_t(x)$ differs once $t > 0$ (Fig. 1D–E). Consequently, the multi-scale Fisher (Fig. 1F) is larger for the monotonic tuning curve, which, as noted earlier, encodes the sign of x.

To illustrate how the multi-scale geometry depends on both likelihood and prior, we consider a toy example in which two neurons encode a two-dimensional stimulus through sigmoidal tuning curves with Gaussian noise (Fig. 2A). We compare a uniform prior (Fig. 2B–C) with a prior concentrated along a curved ridge (Fig. 2D–E). At $t = 0$, the diffused Fisher $J_t(x)$ reduces to the standard Fisher information and is therefore independent of the prior (left panels of Fig. 2B,D). For $t > 0$,

however, the prior enters through the diffused likelihood, $p_t(r \mid x) \propto \int p(r \mid x')\, p(x')\, \phi_t(x - x')\, dx'$ (Appendix B), and the diffused Fisher becomes elongated along directions of greatest local stimulus variance (middle and right panels). Integrating over t then yields a multi-scale Fisher that reflects both the likelihood structure and the prior: ellipses are largest near $x = 0$, where the tuning curves are steepest, and in the non-uniform case align with the ridge of the prior (Fig. 2C,E).

4 The geometry directly captures mutual information

A central consequence of our construction is that the multi-scale geometry admits an exact mutual-information interpretation. Intuitively, stimulus directions that contribute most to the total mutual information encoded by the population are stretched more strongly by the geometry, pushing nearby stimuli farther apart. Formally, under mild regularity conditions on $p(r, x)$ (Appendix A.2), the metric tensor satisfies [14]

$$\boxed{I(R;X) = \frac{1}{2}\,\mathbb{E}_x\left[\mathrm{Tr}\left(G(x)\right)\right].} \tag{11}$$

This identity has a natural geometric interpretation: mutual information equals the expected rate at which squared distances increase under infinitesimal perturbations of the stimulus,

$$I(R;X) = \frac{1}{2}\,\frac{d}{d\varepsilon}\mathbb{E}_{x,z}\left[d^2\left(x; x + \sqrt{\varepsilon}\, z\right)\right]\Big|_{\varepsilon=0}, \qquad z \sim \mathcal{N}(0, I). \tag{12}$$

Equivalently, for N stimuli sampled from $p(x)$,

$$I(R;X) = \frac{n_x}{2} \lim_{N\to\infty} \frac{1}{N} \sum_{i=1}^{N} m_i^2, \tag{13}$$

where n_x is the number of stimulus dimensions, and $m_i = d_i^g / d_i$ is the local magnification at x_i: the ratio of the geometric distance d_i^g to the Euclidean distance d_i between x_i and its nearest neighbour (Appendix A.2). Mutual information is thus the mean squared magnification of stimulus space induced by the neural code. Crucially, this identity is not imposed by construction, but emerges as a consequence of our coarse-graining axioms.

This result also clarifies the interpretation of our earlier examples (Figs. 1–2). For a one-dimensional stimulus with uniform prior, mutual information reduces to the integral of the multi-scale Fisher (Fig. 1F). The fact that $G(x)$ is larger for the monotonic than the bell-shaped tuning curve therefore implies that the monotonic code encodes more mutual information. Likewise, the elongation of $G(x)$ along directions of greater local stimulus variance (Fig. 2E) reflects the stronger contribution of those directions to the total information carried by the population. More generally, the multi-scale geometry shows how local structure in the likelihood and stimulus prior shapes the encoded information.

5 Estimating the multi-scale geometry from diffusion models

We can estimate the squared infinitesimal line element $dx^\top G(x)\, dx$ by rewriting Eq. (10) as an expectation over z (with $\alpha = 1$, following our convention):

$$dx^\top G(x) dx = \int_0^\infty \mathbb{E}_{z\sim\mathcal{N}(0,I)}\left[dx^\top J_t(x + \sqrt{t}z)\, dx\right] dt. \tag{14}$$

The integral over t is approximated by the trapezoid rule in $\log t$ (writing $dt = t\, d\log t$) on a discrete grid of scales $\{t_k\}$ and the expectation over z is approximated by Monte Carlo sampling (Appendix H).

Direct evaluation of the squared line element, $dx^\top J_t dx$, is intractable in high dimensions, since it requires integrating over x to obtain the diffused conditional distribution $p_t(r \mid x_t) \propto \int \phi_t(x_t - x)\, p(r, x)\, dx$. Instead, we derive an estimator for $dx^\top J_t(x_t) dx$ in terms of posterior means that can be learned with a conditional diffusion model.

Expressing $J_t(x_t)$ as the covariance of the score $\nabla_{x_t} \log p_t(r \mid x_t)$, and applying Bayes' rule to relate the score to $\nabla_{x_t} \log p_t(x_t \mid r)$ (noting that $\nabla_{x_t} \log p_t(x_t)$ does not contribute to the covariance),

$$J_t(x_t) = \mathrm{Cov}_{p_t(r|x_t)}[\nabla_{x_t} \log p_t(x_t \mid r)]\,. \tag{15}$$

Under isotropic Gaussian diffusion, Tweedie's identity [18] gives

$$\nabla_{x_t} \log p_t(x_t \mid r) = \frac{1}{t}\big(\hat{x}(x_t, r) - x_t\big), \qquad \hat{x}(x_t, r) = \mathbb{E}[x \mid x_t, r]. \tag{16}$$

Since x_t does not depend on r, this yields

$$\begin{aligned} dx^\top J_t(x_t) dx &= \frac{1}{t^2} dx^\top \mathrm{Cov}_{p_t(r|x_t)}[\hat{x}(x_t, r)] dx \\ &= \frac{1}{2t^2} \mathbb{E}_{r,r' \sim p_t(r|x_t)} \left[\left((\hat{x}(x_t, r) - \hat{x}(x_t, r'))^\top dx \right)^2 \right]. \end{aligned} \tag{17}$$

Thus, the diffused Fisher along direction dx is the expected squared difference between two independently sampled posterior-mean estimates, projected onto dx.

Equation (17) expresses $dx^\top J_t(x_t) dx$ in terms of the posterior mean $\hat{x}(x_t, r) = \mathbb{E}[x \mid x_t, r]$, which we estimate using a conditional diffusion model [9, 19]. Because the estimator depends only on differences between two samples from the same conditional model, systematic errors cancel; this avoids the mismatch bias of approaches that subtract conditional from separately trained unconditional estimators, which can become potentially unstable as $t \to 0$ [14, 13, 15, 16].

We approximate Eq. (17) using Monte Carlo samples $r, r' \sim p_t(r \mid x_t)$. Exact sampling would require first drawing $x \sim p(x \mid x_t)$ and then $r \sim p(r \mid x)$, which entails a full backward diffusion pass and is prohibitively expensive. Following [14, 10], we instead use the approximation $p_t(r \mid x_t) \approx p(r \mid \hat{x}(x_t))$, where $\hat{x}(x_t) = \mathbb{E}[x \mid x_t]$ is obtained in a single forward pass through the denoising model. This approximation becomes increasingly accurate for small t, where the posterior concentrates around its mean. Full implementation details are provided in Appendix H.

6 Uncovering the multi-scale geometry of recorded visual neurons

We next examined what our multi-scale geometry reveals about how recorded visual populations encode natural images. In practice, computing the geometry requires fitting an encoding model to data, raising a key question: does the resulting geometry reflect intrinsic properties of the neural code, or artifacts of the fitted model?

This issue is particularly acute for high-dimensional stimuli, where encoding models are primarily constrained along directions supported by natural image statistics and remain weakly constrained elsewhere. Because the Fisher information depends directly on local response gradients, it can be sensitive to such poorly constrained directions, leading to model-dependent artifacts. In contrast, through its connection to mutual information, the multi-scale geometry is expected to emphasize stimulus variations that lie along the natural image manifold and contribute most to the encoded information (Fig. 2E). As a result, we hypothesized that it should be more strongly constrained by the data and less sensitive to modeling artifacts.

To test this, we analyzed the dataset of Papale et al. [20], consisting of large-scale electrophysiological recordings from awake macaque V1 and V4 during presentation of natural images. We selected the 50 most reliable neurons from each area and fit deep encoding models to predict their responses from image input (Appendix I). As is common for encoding models of population recordings [21–24], we modelled trial-to-trial variability as independent Gaussian noise of fixed variance (Eq. (38)). The geometries reported here therefore do not account for noise correlations, although the framework itself does not require this assumption (Appendix D). Our model, based on an InceptionV3 architecture [25] (Fig. 3A), achieved near state-of-the-art predictive performance, with 0.75 and 0.71 correlation with responses in V1 and V4 respectively (Fig. 3B–C).

We first examined the leading eigenvectors of the multi-scale metric tensor $G(x)$ for individual images x, which identify local stimulus variations contributing most strongly to encoded information. These eigenvectors differed markedly between cortical areas: in V1 they were spatially localized and edge-like, emphasizing fine image structure, whereas in V4 they were broader and more spatially distributed, capturing more global image structure (Fig. 3D–E). Across 200 held-out TinyImageNet test images, V1 eigenvectors had a smaller spatial extent and higher spatial frequency than V4 eigenvectors (Fig. 3G; $p < 0.001$, Wilcoxon signed-rank tests; Appendix F). In contrast, the leading eigenvectors of the Fisher information resembled high-frequency noise (Fig. 3F). Although they also differed significantly between areas (Fig. 3H; $p < 0.001$; Appendix F), these differences were much

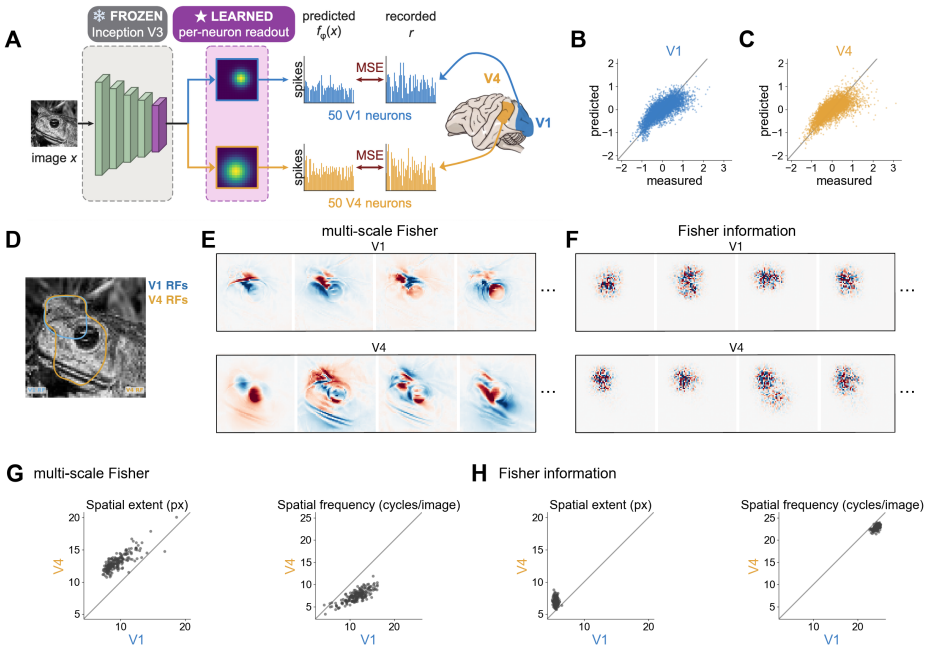

Figure 3: **Multi-scale geometry of neural population recordings in V1 and V4.** (**A**) Deep CNN encoding model used to predict V1 and V4 neural responses to natural images (Appendix I.2). (**B–C**) Measured versus predicted responses across all neurons and images, for V1 (B) and V4 (C). (**D**) Example image with V1 and V4 receptive field regions outlined in blue and yellow, respectively. (**E**) Leading four eigenvectors of the multi-scale metric tensor $G(x)$ for this image, for V1 (top) and V4 (bottom). (**F**) Same as E, for the Fisher information $J(x)$. (**G**) Top-10 eigenvectors of $G(x)$ across the 200 test images, V1 against V4: spatial extent and spatial frequency (one point per image, averaged over eigenvectors; line, identity). (**H**) Same as G, for $J(x)$.

smaller (mean absolute difference between V1 and V4, for $J(x)$ vs. $G(x)$: 1.4 vs. 3.7 pixels in spatial extent, 1.3 vs. 4.0 cycles/image in spatial frequency). This is consistent with the Fisher's sensitivity to response gradients along poorly constrained stimulus directions, which causes its leading structure to be dominated by model-dependent artifacts rather than by intrinsic properties of the neural code.

We next tested whether this difference was robust across encoding models. We fit a second architecture (ResNet-50 [26]; Appendix I) with similar predictive performance (Fig. 4A). The leading eigenvectors of the multi-scale geometry were qualitatively similar across models fit to the same cortical area (Fig. 4B). To quantify this, we measured alignment between the subspaces spanned by the top k eigenvectors of each metric tensor, averaged across images. For the multi-scale metric, alignment was higher for two models fit to the same area than for two models of the same architecture fit to different areas (Fig. 4D; at $k = 10$ this held in each of the 200 images, $p < 0.001$, Wilcoxon signed-rank test), indicating that it captured structure intrinsic to the recorded population. For the Fisher information, the ordering was reversed: the two architectures fit to the same area were less aligned with each other than each architecture was with itself across V1 and V4 (Fig. 4C & E; at $k = 10$ this held in each of the 200 images, $p < 0.001$, Wilcoxon signed-rank test), implying that its leading eigenvectors are driven more by the choice of encoding model than by the neural data (similar results were obtained using Bures–Wasserstein distance [27] rather than subspace alignment, Appendix I.5).

We asked whether the sensitivity of the Fisher geometry to poorly constrained directions could be reduced by using a simpler, less flexible encoding model. We therefore fit neural responses with an ON-OFF model [28] (Appendix I), whose front-end consists of only 12 physiologically motivated gain-control parameters, compared with $\sim$ 173k frozen convolutional parameters in the InceptionV3 backbone (truncated at `Conv2d_4a_3x3`). The leading Fisher eigenvectors then became more interpretable, revealing edge-like features (Fig. 4C). However, predictive performance dropped substantially to 0.20 and 0.13 correlation for V1 and V4 respectively (Fig. 4A), indicating that the

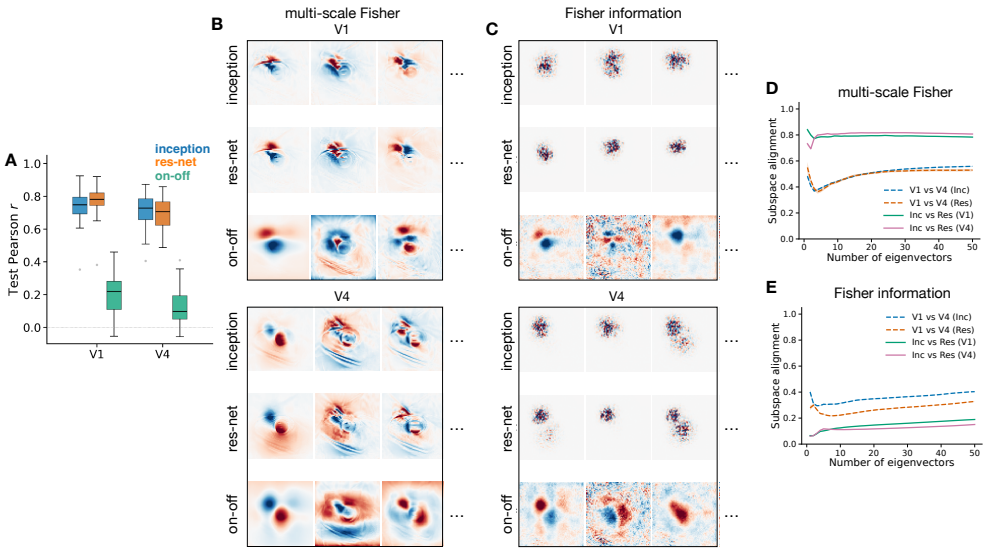

Figure 4: **Robustness across encoding model architectures.** (**A**) Predictive performance for Inception, ResNet, and ON-OFF models, for V1 and V4 (boxplots of test correlation across the 50 neurons of each area). (**B**) Leading eigenvectors of the multi-scale Fisher $G(x)$ for an example image (Fig. 3D), for each model architecture fitted to V1 (top) and V4 (bottom). (**C**) Same as B, for the Fisher information $J(x)$. (**D–E**) Subspace alignment of the top k eigenvectors of $G(x)$ (D) and $J(x)$ (E), comparing models fitted to the same area (solid) versus the same model fitted to different areas (dashed), averaged across images (shading, s.e.m.).

model no longer captured important aspects of the neural code. Thus, meaningful Fisher features could be recovered only by sacrificing model expressiveness, whereas the multi-scale geometry remained interpretable even for high-performing deep CNNs.

Finally, we examined how the local geometry evolved when images were perturbed along the leading two eigenvectors of $G(x)$ (Fig. 5). We projected the metric onto the corresponding two-dimensional subspace and visualized its local variation as a field of ellipses, with perturbation steps of fixed size, set to 1% of the Euclidean norm of the reference image, along each of the two leading eigenvectors, of $G(x)$ in Fig. 5A and of $J(x)$ in Fig. 5B, so that both fields cover the same neighbourhood of the reference image. For the multi-scale metric, these ellipses changed little across the perturbation space and were nearly identical between Inception and ResNet models (Fig. 5A). The Fisher ellipses, in contrast, varied rapidly and inconsistently across models (Fig. 5B), confirming that in this setting the Fisher geometry reflects properties of the encoding model more than structure in the neural code.

7 Discussion

We derived a Riemannian geometry with an exact relation to mutual information. This connection is rooted in identities relating mutual information to the Fisher information, or equivalently MMSE, across diffusion scales [13, 11, 12]. These have been used to decompose information into contributions from individual stimuli [14] and stimulus-response pairs [15, 16]. Here, instead, we use them to connect *representational geometry* to mutual information. This yields a richer framework than decomposition alone: metric structure, geodesic paths, and distances between stimuli characterize how information is organized across stimulus space, rather than reducing it to scalar contributions from individual stimuli and responses.

Recent work showed that a Riemannian geometry constructed purely from the stimulus distribution $p(x)$ can predict human perceptual judgments [29]. In contrast, because our geometry is derived from the full joint distribution $p(r, x)$, it characterizes how well neural populations encode stimuli, not just their statistical structure. It nonetheless yields testable predictions for perception. For any downstream readout y that depends on the stimulus only through r, the data-processing inequality for the Fisher information, applied at each scale, implies that distances can only shrink as information

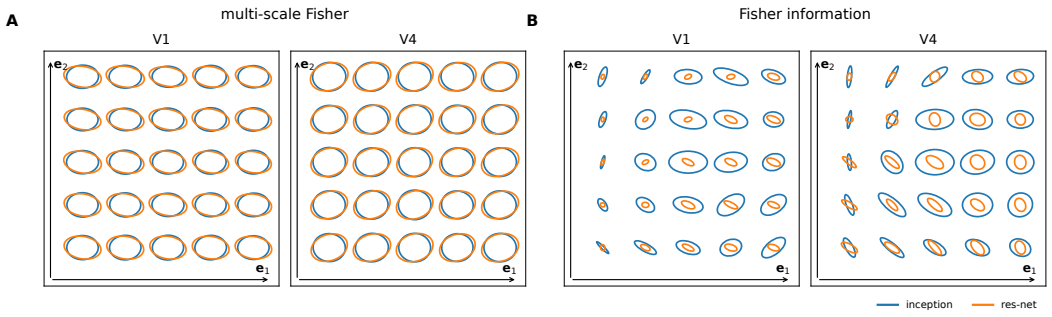

Figure 5: **Local metric field projected onto the leading eigenvectors of $G(x)$.** (**A**) Ellipses show the multi-scale Fisher $G(x)$ computed from Inception (blue) and ResNet (orange) encoding models, as the reference image is perturbed along the leading two eigenvectors of $G(x)$ from the Inception model. At each perturbed image, both metrics are projected onto the same two-dimensional subspace (leading Inception eigenvectors at the reference image). Results are shown for V1 (left) and V4 (right). (**B**) Same as A, for the Fisher information $J(x)$.

passes through successive stages of processing, so that $d_y(x, x') \leq d_r(x, x')$. The geometry defined by neural responses should therefore, in principle, constrain perceptual geometry.

A large body of work defines distances between stimuli based on the similarity of elicited neural responses. However, previous works rely on different definitions of representational similarity, under different assumptions: for example, ignoring response variability [3, 6], relying on mean responses and covariances [30, 31], or grounding distances in a deep model's latent space [32, 33]. Here we instead derive a unique, noise-aware geometry from first principles governing how local distances evolve under stimulus coarse-graining. As with mutual information, this avoids arbitrary decoding assumptions and provides a principled description of neural population codes that can be compared across models and datasets.

Eigenvectors of the Fisher information have been used to identify the stimulus variations best encoded in early visual areas, in shallow networks fitted to neural data [34] or in idealized models (with e.g. identical, spatially symmetric receptive fields) not fitted to recordings [28, 35, 36]. For expressive deep models fitted to V1 and V4, we found the Fisher geometry poorly constrained, with features that vary across architectures, whereas the multi-scale geometry emphasizes directions that are both informative and constrained by the data, and is stable across models fitted to the same data.

A limitation of our approach is that, like the Fisher information, it requires learning a probabilistic encoding model $p(r|x)$ describing how neurons respond to stimuli, and thus sufficient data to estimate this model reliably. Increasingly, however, large-scale datasets of neural responses to natural stimuli are becoming available [20, 24, 37–39], making such approaches feasible in practice. A second limitation is computational: estimating the geometry is more expensive than the Fisher information, as it requires training a diffusion model and performing multiple evaluations to obtain Monte Carlo estimates of the metric tensor. Nonetheless, diffusion models are widely used, so the approach remains practical despite its increased computational cost.

Our construction is closely related to renormalization-group (RG) approaches, which describe how systems evolve under coarse-graining [40]. Prior work has studied how Fisher information changes along such flows, showing that relevant directions are preserved while irrelevant ones become progressively less distinguishable [41, 42]. Our goal here is different: rather than tracking the evolution of a given metric, we derive a unique geometry from first principles and integrate across diffusion scales to obtain a multi-scale metric with an exact link to mutual information. Despite these differences, future work could build on connections to RG theory to better understand how coding structure across scales shapes sensory representations.

Acknowledgments and Disclosure of Funding

This work was funded by the Agence Nationale de la Recherche (ANR) ANR-RetNet4EC (to MC), PostGenAI@Paris (ANR-23-IACL-0007 to UF and SA), NatNetNoise (ANR-21-CE37-0024 to UF), IHU FOReSIGHT (ANR-18-IAHU-01 to UF), COCORICO (ANR-26-CE37-4899-01 to UF and MC), and the SCAI through IDEX Sorbonne Université (ANR-11-IDEX-0004, PhD position of SA); Qube Research & Technologies (QRT), DISSENSATION; and the Conseil Régional d'Île-de-France through DIM C-BRAINS. The authors declare no competing interests.

References

[1] Nikolaus Kriegeskorte, Marieke Mur, Douglas A Ruff, Roozbeh Kiani, Jerzy Bodurka, Hossein Esteky, Keiji Tanaka, and Peter A Bandettini. Matching categorical object representations in inferior temporal cortex of man and monkey. *Neuron*, 60(6):1126–1141, 2008.

[2] Joao Barbosa, Amin Nejatbakhsh, Lyndon Duong, Sarah E Harvey, Scott L Brincat, Markus Siegel, Earl K Miller, and Alex H Williams. Quantifying differences in neural population activity with shape metrics. *bioRxiv*, pages 2025–01, 2025.

[3] Maithra Raghu, Justin Gilmer, Jason Yosinski, and Jascha Sohl-Dickstein. SVCCA: Singular vector canonical correlation analysis for deep learning dynamics and interpretability. *Advances in Neural Information Processing Systems*, 30, 2017.

[4] Ari S Morcos, Maithra Raghu, and Samy Bengio. Insights on representational similarity in neural networks with canonical correlation. In *Advances in Neural Information Processing Systems*, volume 31, pages 5732–5741, 2018.

[5] Seyed-Mahdi Khaligh-Razavi and Nikolaus Kriegeskorte. Deep supervised, but not unsupervised, models may explain IT cortical representation. *PLoS computational biology*, 10(11): e1003915, 2014.

[6] Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *International Conference on Machine Learning*, pages 3519–3529. PMLR, 2019.

[7] Shun-ichi Amari. *Information geometry and its applications*. Springer, 2016.

[8] Nikolaus Kriegeskorte and Xue-Xin Wei. Neural tuning and representational geometry. *Nature Reviews Neuroscience*, 22(11):703–718, 2021.

[9] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

[10] Hyungjin Chung, Jeongsol Kim, Michael T McCann, Marc L Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. In *International Conference on Learning Representations*, 2023.

[11] Dongning Guo, Shlomo Shamai, and Sergio Verdú. Mutual information and minimum mean-square error in gaussian channels. *IEEE Transactions on Information Theory*, 51(4):1261–1282, 2005.

[12] Xianghao Kong, Rob Brekelmans, and Greg Ver Steeg. Information-theoretic diffusion. In *The Eleventh International Conference on Learning Representations*, 2023.

[13] Longxuan Yu, Xing Shi, Xianghao Kong, Tong Jia, and Greg Ver Steeg. Mmg: Mutual information estimation via the mmse gap in diffusion. *arXiv preprint arXiv:2509.20609*, 2025.

[14] Steeve Laquitaine, Simone Azeglio, Carlo Paris, Ulisse Ferrari, and Matthew Chalk. Decomposing stimulus-specific sensory neural information via diffusion models. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

[15] Xianghao Kong, Ollie Liu, Han Li, Dani Yogatama, and Greg Ver Steeg. Interpretable diffusion via information decomposition. In *The Twelfth International Conference on Learning Representations*, 2024.

[16] Shaurya Dewan, Rushikesh Zawar, Prakanshul Saxena, Yingshan Chang, Andrew Luo, and Yonatan Bisk. Diffusion pid: Interpreting diffusion via partial information decomposition. *Advances in Neural Information Processing Systems*, 37:2045–2079, 2024.

[17] David Applebaum. *Lévy Processes and Stochastic Calculus*, volume 116 of *Cambridge Studies in Advanced Mathematics*. Cambridge University Press, Cambridge, 2nd edition, 2009.

[18] Bradley Efron. Tweedie's formula and selection bias. *Journal of the American Statistical Association*, 106(496):1602–1614, 2011.

[19] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021.

[20] Paolo Papale, Feng Wang, Matthew W Self, and Pieter R Roelfsema. An extensive dataset of spiking activity to reveal the syntax of the ventral stream. *Neuron*, 113(4):539–553, 2025.

[21] David A. Klindt, Alexander S. Ecker, Thomas Euler, and Matthias Bethge. Neural system identification for large populations separating "what" and "where". In *Advances in Neural Information Processing Systems*, volume 30, 2017.

[22] Santiago A. Cadena, George H. Denfield, Edgar Y. Walker, Leon A. Gatys, Andreas S. Tolias, Matthias Bethge, and Alexander S. Ecker. Deep convolutional models improve predictions of macaque V1 responses to natural images. *PLOS Computational Biology*, 15(4):e1006897, 2019. doi: 10.1371/journal.pcbi.1006897.

[23] Konstantin-Klemens Lurz, Mohammad Bashiri, Konstantin Willeke, Akshay K. Jagadish, Eric Wang, Edgar Y. Walker, Santiago A. Cadena, Taliah Muhammad, Erick Cobos, Andreas S. Tolias, Alexander S. Ecker, and Fabian H. Sinz. Generalization in data-driven models of primary visual cortex. In *International Conference on Learning Representations*, 2021.

[24] Konstantin F Willeke, Paul G Fahey, Mohammad Bashiri, Laura Pede, Max F Burg, Christoph Blessing, Santiago A Cadena, Zhiwei Ding, Konstantin-Klemens Lurz, Kayla Ponder, et al. The sensorium competition on predicting large-scale mouse primary visual cortex activity. *arXiv preprint arXiv:2206.08666*, 2022.

[25] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2818–2826, 2016.

[26] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[27] Rajendra Bhatia, Tanvi Jain, and Yongdo Lim. On the bures–wasserstein distance between positive definite matrices. *Expositiones mathematicae*, 37(2):165–191, 2019.

[28] Alexander Berardino, Valero Laparra, Johannes Ballé, and Eero Simoncelli. Eigen-distortions of hierarchical representations. *Advances in Neural Information Processing Systems*, 30, 2017.

[29] Guy Ohayon, Pierre-Étienne H. Fiquet, Florentin Guth, Jona Ballé, and Eero P. Simoncelli. Learning a distance measure from the information-estimation geometry of data. In *The Fourteenth International Conference on Learning Representations*, 2026.

[30] Heiko H Schütt, Alexander D Kipnis, Jörn Diedrichsen, and Nikolaus Kriegeskorte. Statistical inference on representational geometries. *Elife*, 12:e82566, 2023.

[31] Alex H Williams, Erin Kunz, Simon Kornblith, and Scott Linderman. Generalized shape metrics on neural representations. *Advances in Neural Information Processing Systems*, 34:4738–4750, 2021.

[32] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 586–595, 2018.

[33] Binxu Wang and Carlos R Ponce. Tuning landscapes of the ventral stream. *Cell Reports*, 41(6), 2022.

[34] Xuehao Ding, Dongsoo Lee, Joshua Melander, George Sivulka, Surya Ganguli, and Stephen Baccus. Information geometry of the retinal representation manifold. *Advances in Neural Information Processing Systems*, 36:44310–44322, 2023.

[35] Jenelle Feather, David Lipshutz, Sarah E Harvey, Alex H Williams, and Eero P Simoncelli. Discriminating image representations with principal distortions. In *The Thirteenth International Conference on Learning Representations*, 2025.

[36] Jingyang Zhou, Chanwoo Chun, Ajay Subramanian, and Eero P Simoncelli. Comparing neural models using their perceptual discriminability predictions. In *Proceedings of UniReps: the First Workshop on Unifying Representations in Neural Models*, volume 243, pages 170–181. PMLR, 2024.

[37] Polina Turishcheva, Paul G Fahey, Michaela Vystrčilová, Laura Hansel, Rachel Froebe, Kayla Ponder, Yongrong Qiu, Konstantin F Willeke, Mohammad Bashiri, Eric Wang, et al. The dynamic sensorium competition for predicting large-scale mouse visual cortex activity from videos. *ArXiv*, pages arXiv–2305, 2024.

[38] 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? *bioRxiv*, page 407007, 2018.

[39] Konstantin F Willeke, Polina Turishcheva, Alex Gilbert, Goirik Chakrabarty, Hasan A Bedel, Paul G Fahey, Yongrong Qiu, Marissa A Weis, Michaela Vystrčilová, Taliah Muhammad, et al. Omnimouse: Scaling properties of multi-modal, multi-task brain models on 150b neural tokens. *arXiv preprint arXiv:2604.18827*, 2026.

[40] G Jona-Lasinio. The renormalization group: A probabilistic view. *Il Nuovo Cimento B (1971-1996)*, 26(1):99–119, 1975.

[41] Archishman Raju, Benjamin B Machta, and James P Sethna. Information loss under coarse graining: A geometric approach. *Physical Review E*, 98(5):052112, 2018.

[42] Reevu Maity, Subhash Mahapatra, and Tapobrata Sarkar. Information geometry and the renormalization group. *Physical Review E*, 92(5):052101, 2015.

[43] N. N. Čencov. *Statistical Decision Rules and Optimal Inference*, volume 53 of *Translations of Mathematical Monographs*. American Mathematical Society, 1982.

[44] Bruno B. Averbeck, Peter E. Latham, and Alexandre Pouget. Neural correlations, population coding and computation. *Nature Reviews Neuroscience*, 7(5):358–366, 2006. doi: 10.1038/nrn1888.

[45] Rubén Moreno-Bote, Jeffrey Beck, Ingmar Kanitscheider, Xaq Pitkow, Peter Latham, and Alexandre Pouget. Information-limiting correlations. *Nature Neuroscience*, 17(10):1410–1417, 2014. doi: 10.1038/nn.3807.

[46] Ya Le and Xuan S. Yang. Tiny imagenet visual recognition challenge. 2015. URL `https://api.semanticscholar.org/CorpusID:16664790`.

[47] Martin N Hebart, Adam H Dickter, Alexis Kidder, Wan Y Kwok, Anna Corriveau, Caitlin Van Wicklin, and Chris I Baker. Things: A database of 1,854 object concepts and more than 26,000 naturalistic object images. *PloS one*, 14(10):e0223792, 2019.

[48] Daniel LK Yamins and James J DiCarlo. Using goal-driven deep learning models to understand sensory cortex. *Nature Neuroscience*, 19(3):356–365, 2016.

[49] TorchVision maintainers and contributors. Torchvision: Pytorch's computer vision library. `https://github.com/pytorch/vision`, 2016.

[50] Lyndon Duong, Kathryn Bonnen, William Broderick, Pierre-Étienne Fiquet, Nikhil Parthasarathy, Thomas Yerxa, Xinyuan Zhao, and Eero Simoncelli. Plenoptic: A platform for synthesizing model-optimized visual stimuli. *Journal of Vision*, 23(9):5822–5822, 2023.

[51] Gene H Golub and Charles F Van Loan. *Matrix computations*. JHU press, 2013.

[52] Sarah E Harvey, David Lipshutz, and Alex H Williams. What representational similarity measures imply about decodable information. In *Proceedings of UniReps: the Second Edition of the Workshop on Unifying Representations in Neural Models*, volume 285, pages 140–151. PMLR, 2024.

[53] Patrick von Platen, Suraj Patil, Anton Lozhkov, Pedro Cuenca, Nathan Lambert, Kashif Rasul, Mishig Davaadorj, Dhruv Nair, Sayak Paul, William Berman, Yiyi Xu, Steven Liu, and Thomas Wolf. Diffusers: State-of-the-art diffusion models. `https://github.com/huggingface/diffusers`, 2022.

A Mathematical details

A.1 Derivations for Section 3

Taylor expansion (Eq. (6)). Expanding each entry of G_t to second order around x,

$$G_t(x + \sqrt{t}\, z) = G_t(x) + \sqrt{t} \sum_i z_i\, \partial_i G_t(x) + \frac{t}{2} \sum_{i,j} z_i z_j\, \partial_i \partial_j G_t(x) + O(t^{3/2}).$$

Averaging over z, using $\mathbb{E}[z_i] = 0$ and $\mathbb{E}[z_i z_j] = \delta_{ij}$,

$$G_t(x + \sqrt{t}z) = G_t(x) + \frac{t}{2} \Delta_x G_t(x) + O(t^{3/2}), \tag{18}$$

where Δ is the Laplace operator, applied to each component of $G_t(x)$. Taking the derivative with respect to t, then setting $t = 0$ gives,

$$\frac{d}{dt} \mathbb{E}_z[G_t(x + \sqrt{t}z)]\bigg|_{t=0} = \partial_t G_t(x)|_{t=0} + \frac{1}{2} \Delta_x G_t(x)\bigg|_{t=0} \tag{19}$$

as in Eq. (6) in the main text.

Čencov's theorem. Let $\{p(r \mid x)\}$ be a family of distributions whose support does not depend on x, differentiable in x with finite Fisher information. The only Riemannian metric on this family that (i) depends only on $p(r \mid x)$ and its first derivatives at x, and (ii) is invariant under transformations of r that preserve the sufficient statistics of $p(r \mid x)$ (Markov morphisms), is the Fisher information, up to a positive constant [43, 7].

Requirement 1 states that $dx^\top Q(x)\, dx \le 0$ for all x and dx, so $-Q(x)$ is positive semi-definite; $Q(x)$ is symmetric since $G(x)$ is. It also requires the contraction rate to be the same in any coordinates. Let $y = \psi(x)$ be a smooth reparametrisation with Jacobian $\Lambda(x) = \partial\psi/\partial x$, so that $dy = \Lambda\, dx$, and let $\tilde{Q}(y)$ be the contraction computed in y-coordinates for the same coarse-graining process. Then $dy^\top \tilde{Q}\, dy = dx^\top Q\, dx$ for all dx holds if and only if $Q = \Lambda^\top \tilde{Q}\, \Lambda$, the transformation law of a covariant rank-2 tensor. Together with the regularity conditions stated in the main text, $-Q(x)$ is therefore a Riemannian metric tensor.

Requirement 2 is condition (i), that $Q(x)$ depends only on $p(r \mid x)$ and its first derivatives at x, and Requirement 3 is condition (ii), that $Q(x)$ is invariant under transformations of r that preserve the sufficient statistics of $p(r \mid x)$. All the conditions of Čencov's theorem are therefore met, and $-Q(x) = \alpha J(x)$ with $\alpha \ge 0$. We exclude the trivial case $\alpha = 0$, which gives $G \equiv 0$.

Backward solution (Eq. (10)). Let $\phi_t * G_t$ denote the convolution of each entry of G_t with the isotropic Gaussian kernel of variance t. By the product rule,

$$\partial_t(\phi_t * G_t) = (\partial_t \phi_t) * G_t + \phi_t * (\partial_t G_t).$$

The heat kernel satisfies $\partial_t \phi_t = \frac{1}{2} \Delta_x \phi_t$, and differentiation commutes with convolution, $(\Delta_x \phi_t) * G_t = \phi_t * (\Delta_x G_t)$, so, using Eq. (8),

$$\begin{aligned} \partial_t(\phi_t * G_t) &= \phi_t * \left(\tfrac{1}{2} \Delta_x G_t + \partial_t G_t\right) \\ &= -\alpha\, \phi_t * J_t. \end{aligned}$$

Integrating from $t = 0$ to ∞ gives

$$[\phi_t * G_t]_0^\infty = -\alpha \int_0^\infty \phi_t * J_t dt.$$

By Eq. (9) the upper limit vanishes, and $\phi_0 * G_0 = G_0$, so

$$G(x) \equiv G_0(x) = \alpha \int_0^\infty [\phi_t * J_t](x)\, dt,$$

as stated in the text.

A.2 Relation between geometry and mutual information

Recently, Laquitaine et al. [14] showed that for a memoryless channel $x \to r$, and assuming that $p(x)$ and $p(x|r)$ have finite first and second moments (for all r), the mutual information can be written:

$$I(R;X) = \frac{1}{2}\mathbb{E}_x\left[\mathrm{Tr}\int_0^\infty (J_t * \phi_t)(x)\, dt\right] \tag{20}$$

where ϕ_t is an isotropic Gaussian kernel of variance t, $*$ denotes convolution, and J_t is the Fisher information computed with respect to the diffused likelihood $p_t(r|x)$. Since the term within the expectation is the trace of the multi-scale Fisher metric $G(x)$, we can write

$$I(R;X) = \frac{1}{2}\mathbb{E}_x\left[\mathrm{Tr}\left(G(x)\right)\right]. \tag{21}$$

We now show that this is equivalent to Eq. (12) in the main text. For a Riemannian metric, the squared geodesic distance between nearby points satisfies $d^2(x, x+\delta) = \delta^\top G(x)\,\delta + o(\|\delta\|^2)$. With $\delta = \sqrt{\varepsilon}z$, this gives $d^2 = \varepsilon\, z^\top G(x) z + o(\varepsilon)$, and the $o(\varepsilon)$ term does not contribute to the derivative at $\varepsilon = 0$. Using the local second-order expansion of the squared geodesic distance between x and $x + \sqrt{\varepsilon}z$, where $z \sim \mathcal{N}(0, I)$, z is statistically independent of x, and $\varepsilon \to 0$,

$$\frac{1}{2}\,\frac{d}{d\varepsilon}\mathbb{E}_{x,z}\left[d^2\big(x,\ x+\sqrt{\varepsilon}\,z\big)\right]\Big|_{\varepsilon=0} = \frac{1}{2}\,\frac{d}{d\varepsilon}\,\varepsilon\,\mathbb{E}_{x,z}\left[z^\top G(x)\,z\right]\Big|_{\varepsilon=0} = \frac{1}{2}\mathbb{E}_{x,z}\left[z^\top G(x)\,z\right]. \tag{22}$$

Applying the identity $z^\top A z = \mathrm{Tr}(A z z^\top)$ and using independence of z and x,

$$\begin{aligned}
\frac{1}{2}\,\frac{d}{d\varepsilon}\mathbb{E}_{x,z}\left[d^2\big(x,\ x+\sqrt{\varepsilon}\,z\big)\right]\Big|_{\varepsilon=0} &= \frac{1}{2}\,\mathrm{Tr}\Big(\mathbb{E}_x[G(x)]\,\mathbb{E}_z\left[zz^\top\right]\Big) && (23)\\
&= \frac{1}{2}\mathbb{E}_x\left[\mathrm{Tr}\left(G(x)\right)\right] && (24)\\
&= I(R;X). && (25)
\end{aligned}$$

We now prove Eq. (13) in the main text. Consider drawing N iid samples from the stimulus prior, $\{x_i\}_{i=1}^N \sim p(x)$, and let $NN(i)$ denote the Euclidean nearest neighbour of x_i. Define the local magnification induced by the metric:

$$m_i = \frac{d^g(x_i, x_{NN(i)})}{\|x_i - x_{NN(i)}\|_2}, \tag{26}$$

i.e. the ratio of the Riemannian distance between two nearby stimuli to their Euclidean separation. Writing the nearest-neighbour displacement as $x_{NN(i)} - x_i = \rho_i u_i$, where $\rho_i > 0$ and u_i is a unit vector, the squared magnification in the large-N limit is

$$\lim_{N\to\infty} m_i^2 = \frac{\rho_i^2\, u_i^\top G(x_i)\, u_i}{\rho_i^2\, u_i^\top u_i} = u_i^\top G(x_i)\, u_i. \tag{27}$$

Here $d^g(x_i, x_{NN(i)})^2 = \rho_i^2\, u_i^\top G(x_i)\, u_i + o(\rho_i^2)$, and $\rho_i \to 0$ as $N \to \infty$, so the correction vanishes in the limit. Taking expectations over samples and NN directions,

$$\begin{aligned}
\lim_{N\to\infty}\frac{1}{N}\sum_{i=1}^N m_i^2 &= \mathbb{E}_{x,u}[u^\top G(x) u] && (28)\\
&= \mathrm{Tr}\big(\mathbb{E}_x\big[G(x)\,\mathbb{E}_{u|x}[uu^\top]\big]\big). && (29)
\end{aligned}$$

As $N \to \infty$, the sample cloud in any vanishing neighbourhood of x converges locally to a homogeneous Poisson point process with intensity $Np(x)$. By the isotropy of the homogeneous Poisson process in the local tangent space, the nearest-neighbour direction is uniformly distributed on $\mathbb{S}^{n_x-1}$, giving

$$\mathbb{E}_{u|x}[uu^\top] = \frac{1}{n_x} I, \tag{30}$$

where n_x is the stimulus dimensionality. Substituting back into the expression for the average squared magnification,

$$\begin{aligned}
\frac{n_x}{2}\lim_{N\to\infty}\frac{1}{N}\sum_{i=1}^N m_i^2 &= \frac{1}{2}\mathrm{Tr}\big(\mathbb{E}_x[G(x)]\big) && (31)\\
&= I(R;X), && (32)
\end{aligned}$$

which is Eq. (13) in the main text.

B Gaussian stimuli and linear-Gaussian responses

For a Gaussian stimulus $x \sim \mathcal{N}(0, \Sigma)$ and responses $r = Ax + \eta$, $\eta \sim \mathcal{N}(0, \sigma^2 I)$, all quantities have closed forms. Given the coarse-grained stimulus $x_t = x + \sqrt{t}\, z$, the stimulus is Gaussian with mean $M_t x_t$, $M_t = \Sigma(\Sigma + tI)^{-1}$, and covariance $C_t = \Sigma - M_t\Sigma$, so $p_t(r \mid x) = \mathcal{N}(AM_t x,\ AC_tA^\top + \sigma^2 I)$ and

$$J_t = M_t^\top A^\top \left(AC_tA^\top + \sigma^2 I\right)^{-1} AM_t, \qquad G = \int_0^\infty J_t\, dt,$$

both constant in x. The prior enters through M_t and C_t: at $t = 0$, $M_0 = I$, $C_0 = 0$ and $J_0 = A^\top A/\sigma^2 = J$, the prior-free Fisher information. For $t > 0$, J_t is weighted towards directions of large prior variance. When Σ and J share eigenvectors e_i (eigenvalues s_i and j_i), $G = \sum_i \log(1 + s_i j_i)\, e_i e_i^\top$, and $\frac{1}{2}\mathrm{Tr}\, G = \frac{1}{2}\log|I + \Sigma J| = I(R; X)$.

For weak signals ($J \to 0$), G is linear in J: in the eigenbasis of Σ, $G_{ij} = J_{ij}\, s_i s_j \log(s_i/s_j)/(s_i - s_j)$, with $G_{ii} = s_i J_{ii}$. The leading eigenvectors of G therefore lie along directions that are both well encoded and of high prior variance, as in Fig. 2.

C Robustness to poorly constrained stimulus directions: a toy example

To illustrate the argument of Section 6, we use the linear-Gaussian setting of Appendix B with a two-dimensional stimulus whose prior has variance 1 along a "manifold" direction u_{on} and 0.01 along the orthogonal direction u_{off} (at $30°$), mimicking the low-dimensional structure of natural images. Ten neurons encode x through $A = 2\, a_{\mathrm{on}} u_{\mathrm{on}}^\top + 2\beta\, a_{\mathrm{off}} u_{\mathrm{off}}^\top$, with fixed orthonormal weight vectors $a_{\mathrm{on}}, a_{\mathrm{off}} \in \mathbb{R}^{10}$ and $\sigma^2 = 0.5$. The parameter β sets how strongly the neurons encode the off-manifold direction. We computed all quantities exactly, evaluating the integral over t by numerical quadrature.

Increasing β barely changes the distribution of responses, because the stimulus hardly varies off the manifold, yet it changes the Fisher information completely (Table 1). For $\beta = 1$–2, the response distribution changes by 0.002–0.02 nats and the off-manifold direction accounts for 1–4% of the response variance. Yet the Fisher information becomes as large as or larger off the manifold than on it, and its leading eigenvector turns off the manifold. The multi-scale metric changes by 5–16% and its leading eigenvector stays on the manifold. Off-manifold encoding does add information, and G reflects it in proportion to the stimulus variance in that direction.

Table 1: Off-manifold toy. KL: divergence of the response distribution $p(r)$ from its value at $\beta = 0$. Off-var.: fraction of response variance due to the off-manifold direction. Off/on: $u_{\mathrm{off}}^\top M u_{\mathrm{off}} / u_{\mathrm{on}}^\top M u_{\mathrm{on}}$ for $M = J$ or G. Change: relative change of the trace-normalised matrix from $\beta = 0$ (Frobenius norm). Angle: angle of the leading eigenvector from u_{on} (at $\beta = 1$ the two eigenvalues of J are equal).

β	KL (nats)	Off-var.	J off/on	G off/on	J change	G change	J angle	G angle	I (nats)
0	0	0%	0	0	0	0	0°	0°	1.099
0.5	0.0001	0.25%	0.25	0.009	0.28	0.013	0°	0°	1.109
1	0.0015	0.99%	1.00	0.035	0.71	0.048	–	0°	1.137
2	0.021	3.9%	4.00	0.13	1.13	0.16	90°	0°	1.237
4	0.23	13.8%	16.0	0.38	1.33	0.39	90°	0°	1.511
8	1.65	39.0%	64.0	0.82	1.39	0.64	90°	0°	2.004

D Correlated neural noise

Nothing in the construction of $G(x)$ assumes conditional independence, and the estimator of Section 5 applies to correlated responses if responses are sampled with the full noise covariance. We tested this on a population of $n = 6$ neurons encoding a two-dimensional Gaussian stimulus, $x \sim \mathcal{N}(0, \Sigma_x)$ with unit variances and covariance 0.3, through overlapping tuning, $\mathbb{E}[r \mid x] = Ax$, with A drawn once with i.i.d. $\mathcal{N}(0, 1)$ entries plus 0.6 times the population-mean tuning vector (rows $(0.49, 0.96)$, $(0.47, -1.16)$, $(1.05, 0.59)$, $(-0.39, 0.72)$, $(0.51, 0.43)$, $(0.17, 0.69)$). Responses are $r \sim \mathcal{N}(Ax, \Sigma)$ with baseline per-neuron variance $\sigma^2 = 0.3$. We introduced correlations in two ways. After encoding,

$\Sigma = \sigma^2[I + c\,uu^\top]$, adding ($c = 1, 4$) or removing ($c = -0.75$) variance along the unit-norm leading eigenvector u of the signal covariance $A\Sigma_x A^\top$. Before encoding, independent noise was added to the stimulus, $r = A(x + \xi_{\text{in}}) + \xi_{\text{out}}$ with $\xi_{\text{in}} \sim \mathcal{N}(0, fI)$, so that $\Sigma = fAA^\top + \sigma^2 I$: the correlations then arise because upstream noise is mapped through the weights that carry the signal, the mechanism described by Ding et al. [34]. The mutual information is $\frac{1}{2}\log|I + \Sigma_x A^\top \Sigma^{-1} A|$. We computed $\frac{1}{2}\langle \text{Tr}\, G\rangle$ with the conditional-pair form of Eq. (17), drawing $r = A\tilde{x} + L\xi$ with $\Sigma = LL^\top$ at stimuli $\tilde{x}$ sampled from $p(x \mid x_t)$, and using the exact Gaussian posterior means in place of a trained diffusion model. This simulation thus tests the identity and the correlated sampler independently of any trained diffusion network.

The estimate matched the mutual information for every covariance (Table 2, Fig. 6). Relative to a trial-shuffled surrogate with the same single-neuron variances, noise in the signal subspace reduced the information, whether prescribed after encoding (-7.9% and -13.5%) or propagated from upstream (-14.5% to -43.5%), and removing noise along the signal direction increased it ($+21.7\%$), as expected from the known effects of noise correlations [44, 45]. Rows 2 and 5 in Table 2 have the same mean correlation but differ almost twofold in information loss, so the effect depends on how the correlations are oriented relative to the signal more than on their magnitude.

Table 2: Correlated noise: mean pairwise noise correlation, mutual information (closed form), $\frac{1}{2}\langle \text{Tr}\, G\rangle$ (estimator, $\pm$ s.e.m.), their relative difference, and the change in information relative to the trial-shuffled surrogate.

Noise structure	corr.	$I(R;X)$	$\frac{1}{2}\langle \text{Tr}\, G\rangle$	rel. diff.	ΔI vs shuffled
Independent (control)	0.00	2.2774	2.2781 ± 0.0011	$+0.03\%$	–
After encoding, along signal ($c = 1$)	0.04	1.9596	1.9595 ± 0.0009	-0.00%	-7.9%
After encoding, along signal ($c = 4$)	0.11	1.5790	1.5796 ± 0.0008	$+0.04\%$	-13.5%
After encoding, reduced along signal	-0.05	2.9478	2.9448 ± 0.0014	-0.10%	$+21.7\%$
Before encoding, $f = 0.1$	0.04	1.7082	1.7073 ± 0.0008	-0.05%	-14.5%
Before encoding, $f = 0.3$	0.09	1.2047	1.2039 ± 0.0006	-0.07%	-27.2%
Before encoding, $f = 1.0$	0.17	0.6303	0.6294 ± 0.0003	-0.15%	-43.5%

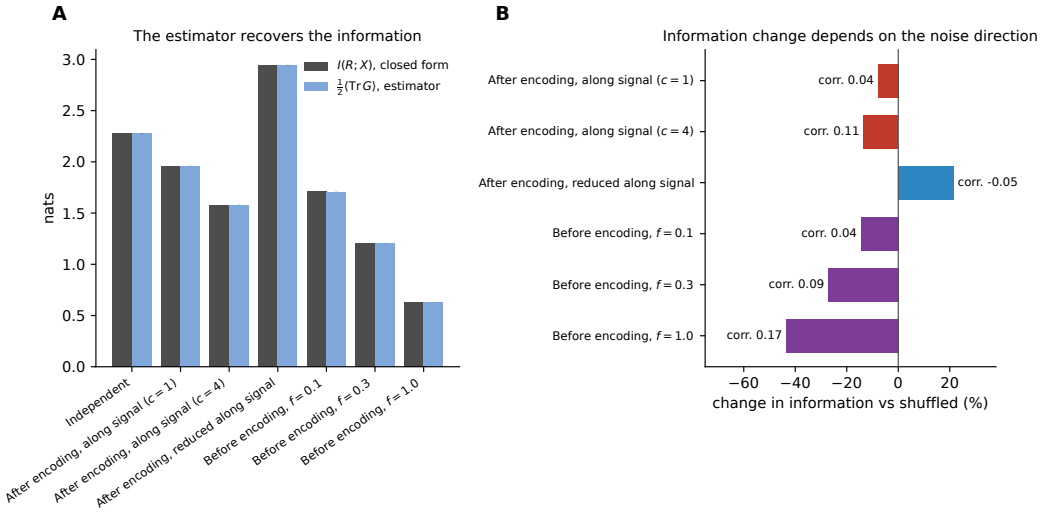

Figure 6: **The multi-scale geometry under correlated noise.** (**A**) Mutual information (closed form) and $\frac{1}{2}\langle \text{Tr}\, G\rangle$ (estimator) for the noise structures of Table 2; the error bars of the estimator (s.e.m.) are smaller than the line width. (**B**) Change in information relative to the trial-shuffled surrogate for each correlated noise structure, with its mean pairwise noise correlation (corr.). Red, noise added after encoding along the signal direction; blue, noise removed along the signal direction; purple, noise propagated from upstream.

E Periodic stimuli

The requirements of Section 3 concern only the local geometry of stimulus space, so the construction applies on the circle, with coarse-graining by Brownian motion on the circle: ϕ_t is the wrapped Gaussian heat kernel and the convolution in Eq. (10) is circular. We tested the identity of Section 4 for populations of Poisson neurons encoding a circular stimulus $x \in [-\pi, \pi)$ with uniform prior. Neuron i has von Mises tuning $f_i(x) = a_i \exp[(\cos(x - \theta_i) - 1)/w_i] + 0.01$. Rows 1–4 of Table 3 use identical, evenly spaced tuning curves ($a_i = 19$, $w_i = 0.29$). Rows 5–7 use 10 neurons with one heterogeneous property (preferred angles displaced by $0.5 \sin 2\theta_i$; amplitudes from 3.8 to 34.2; widths from 0.145 to 0.58). Row 8 uses 4 broad, low-rate neurons ($a_i = 5$, $w_i = 1.5$), for which the posterior covers much of the circle.

We computed the mutual information, $I(R; X) = H(X) - \mathbb{E}_r[H(X \mid r)]$, by quadrature of each posterior on a 2048-point grid, averaged over 10^5 sampled responses. For $\frac{1}{2}\langle \mathrm{Tr}\, G \rangle = \frac{1}{2}\int_0^\infty \mathbb{E}_x[J_t(x)]\, dt$, we used $\mathbb{E}_x[J_t] = \mathbb{E}_r\big[\mathbb{E}_{p_t(x|r)}[(\partial_x \log p_t(r \mid x))^2]\big]$, with $p_t(r \mid x)$ the circular convolution of the likelihood with ϕ_t and its derivative computed exactly as Fourier sums, on 39 log-spaced scales in $[10^{-5}, 40]$ with the trapezoid rule in $\log t$ (the contribution of $t < 10^{-5}$ added from the Fisher information at $t = 0$), and independent response samples. The score is computed exactly, without Tweedie's identity, so the test does not involve the Euclidean estimator of Section 5. With a uniform prior, $p_t(x \mid r)$ is the heat flow of the posterior on the circle, so the identity also holds response by response through de Bruijn's identity. Using independent samples makes the table a test of the numerical integration as well.

Table 3: Periodic stimulus: mutual information and $\frac{1}{2}\langle \mathrm{Tr}\, G \rangle$ (nats, $\pm$ s.e.m.) for Poisson populations encoding a circular variable.

Tuning configuration	$I(R; X)$	$\frac{1}{2}\langle \mathrm{Tr}\, G \rangle$	rel. diff.
1 neuron	0.9240 ± 0.0015	0.9227 ± 0.0016	-0.13%
4 neurons	2.2676 ± 0.0008	2.2663 ± 0.0008	-0.06%
10 neurons	2.8176 ± 0.0003	2.8180 ± 0.0003	$+0.01\%$
20 neurons	3.1680 ± 0.0002	3.1676 ± 0.0002	-0.01%
10 neurons, uneven spacing	2.8017 ± 0.0005	2.8026 ± 0.0005	$+0.03\%$
10 neurons, uneven amplitude	2.7989 ± 0.0005	2.7990 ± 0.0005	$+0.00\%$
10 neurons, uneven width	2.7580 ± 0.0005	2.7590 ± 0.0005	$+0.03\%$
4 neurons, broad tuning, low rate	0.7584 ± 0.0010	0.7583 ± 0.0010	-0.01%

F Eigenvector statistics across images

For each of the 200 test images, each deep encoder (Inception, ResNet-50) and each area, we took the top-10 eigenvectors v of $G(x)$ and of $J(x)$ and measured: the spatial extent, the root-mean-square distance of $|v|^2$ from its centroid (pixels); the spatial frequency, the root-mean-square frequency of the power spectrum of v (cycles/image); and the number of connected components of $\{|v| > 0.2 \max |v|\}$ (the sign of v is ignored). Because the eigenvectors of an image are not independent observations, we averaged each measure over the ranks and the two encoders, and compared V1 and V4 on the same image with a two-sided Wilcoxon signed-rank test across the 200 images (Fig. 3G). For $G(x)$, V1 eigenvectors were smaller (medians 9.0 vs. 12.9 pixels; 199 of 200 images; $p < 0.001$), of higher spatial frequency (11.7 vs. 7.5 cycles/image; $p < 0.001$) and had fewer components (8.1 vs. 9.5; $p < 0.001$). The extent difference held for each encoder separately and at every rank. The frequency difference held at all ranks. The difference in components held at thresholds up to 0.5. Eigenvectors of $J(x)$ also had fewer components in V1, so this measure, like the spatial extent, mainly reflects the smaller receptive fields in V1. For $J(x)$, V1 eigenvectors were also smaller in terms of spatial extent (5.6 vs. 7.0 pixels; $p < 0.001$), consistent with smaller receptive fields, but the difference in spatial frequency was small (24.3 vs. 22.9 cycles/image; $p < 0.001$; a ratio of 1.06, compared with 1.57 for $G(x)$). Shown separately for each architecture (Fig. 7), the difference between areas in $G(x)$ was larger than the difference between architectures (spatial frequency in V1 and V4: 10.6 and 7.0 cycles/image for Inception, 12.9 and 8.0 for ResNet-50). For $J(x)$, the architectures differed more than the areas (21.3 and 20.2 cycles/image for Inception, 27.3 and 25.7 for ResNet-50).

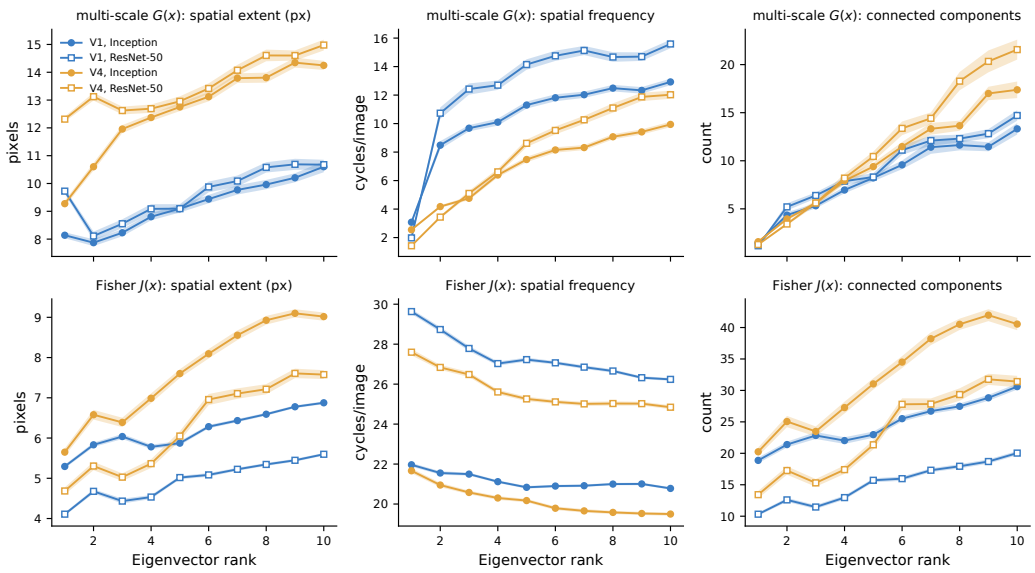

Figure 7: **Eigenvector statistics by rank.** Mean over the 200 test images (lines) $\pm$ s.e.m. (shading) of the spatial extent, spatial frequency and number of connected components of the k-th eigenvector of $G(x)$ (top) and $J(x)$ (bottom), for the Inception (filled circles) and ResNet-50 (open squares) models fitted to V1 (blue) and V4 (yellow).

G Data efficiency and Monte Carlo convergence

The only stage limited by recorded data is the encoding model $p(r \mid x)$, which $G(x)$ requires just as the Fisher information does. Our encoders reach 0.81 and 0.79 of the noise ceiling in V1 and V4 (Table 4). Given the encoder, the diffusion model is trained on synthetic pairs from an effectively unlimited image set, and the remaining error is Monte Carlo error. The metric is defined on stimulus space, so its dimension does not grow with the number of neurons. We did not measure how the estimated geometry changes with the amount of recorded data or the number of recorded neurons.

To quantify the Monte Carlo error, we ran Algorithm 1 five times with independent random seeds for each of 10 test images, for the Inception models of V1 and V4, at $N_{\mathrm{MC}} = 25$, 50, 100, 200 and 400 samples per scale (Fig. 8). The coefficient of variation of the leading eigenvalue across runs fell with N_{MC} up to $N_{\mathrm{MC}} = 200$, the value used throughout (log-log slope -0.62 in V1 and -0.58 in V4 between 25 and 200 samples, close to the $N_{\mathrm{MC}}^{-1/2}$ expected for independent samples), and did not decrease further at 400 (Fig. 8A). The top-5 eigenvectors were reproducible across runs (mean squared cosine of the principal angles 0.92 in V1 and 0.93 in V4 at $N_{\mathrm{MC}} = 200$) and became more so at 400 (0.97; Fig. 8B).

H Estimating the geometry using diffusion models

This appendix details the diffusion model used to approximate the posterior mean $\hat{x}(x_t, r) = \mathbb{E}[x \mid x_t, r]$ that appears in Eq. (17), and the Monte Carlo estimator we employ to obtain the leading eigenvectors of the multi-scale metric $G(x)$.

H.1 Diffusion model architecture and training

The goal of this subsection is to specify a conditional score network $\epsilon_\theta(x_t, t, r)$ from which we can recover the conditional posterior mean $\hat{x}(x_t, r) = \mathbb{E}[x \mid x_t, r]$ that enters Eq. (17).

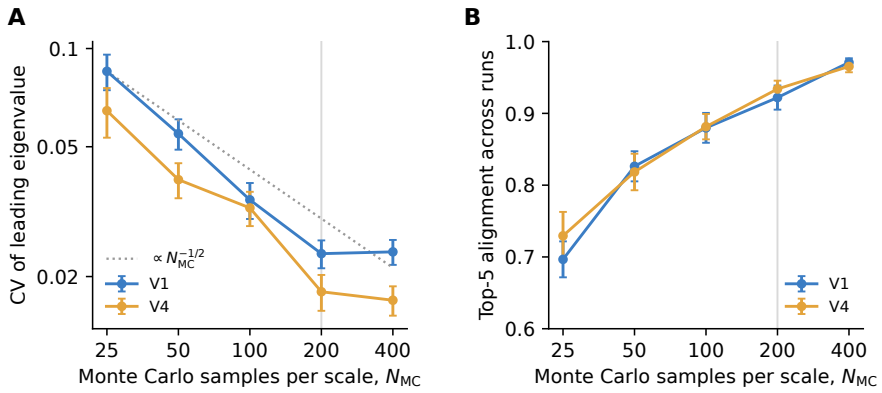

Figure 8: **Monte Carlo convergence of the estimator** (10 test images, 5 runs each, Inception models; mean ± s.e.m. over images). (**A**) Coefficient of variation of the leading eigenvalue of $G(x)$ across runs; dotted line, $N_{\mathrm{MC}}^{-1/2}$. (**B**) Mean pairwise alignment of the top-5 eigenvectors across runs (Eq. (42)). Grey vertical line, $N_{\mathrm{MC}} = 200$, the value used throughout.

Forward diffusion. The main-text coarse-graining of Eq. (3) is the *variance-exploding* process

$$x_\tau \;=\; x \;+\; \sqrt{\tau}\, z, \qquad z \sim \mathcal{N}(0, I), \quad \tau \geq 0, \tag{33}$$

where τ is the variance of the added noise; we use τ here to distinguish the continuous main-text scale from the discrete DDPM step t introduced below. To implement Eq. (33) with standard DDPM tooling, we adopt the *variance-preserving* parametrisation of Ho et al. [9]: for a clean image $x \in [-1, 1]^{64\times 64}$ (single channel), the noisy variable at DDPM step $t \in \{1, \dots, T\}$ is

$$x_t \;=\; \sqrt{\bar{\alpha}_t}\, x \;+\; \sqrt{1 - \bar{\alpha}_t}\, z, \qquad z \sim \mathcal{N}(0, I), \tag{34}$$

with $T = 1000$ and $\bar{\alpha}_t$ the cumulative product of the linear β-schedule of Ho et al. [9]. The two processes are equivalent up to a deterministic rescaling: defining $\tilde{x}_t := x_t / \sqrt{\bar{\alpha}_t}$ and substituting Eq. (34) gives

$$\tilde{x}_t \;=\; x \;+\; \sqrt{\tfrac{1-\bar{\alpha}_t}{\bar{\alpha}_t}}\, z, \tag{35}$$

which matches Eq. (33) under the identification

$$\tau \;=\; \frac{1 - \bar{\alpha}_t}{\bar{\alpha}_t}, \tag{36}$$

so the main-text scale τ equals the inverse signal-to-noise ratio at DDPM step t, sweeping $(0, \infty)$ monotonically as t runs from 1 to T. Since x_t and $\tilde{x}_t$ differ only by an invertible linear transformation, conditioning on either yields the same posterior, so $\hat{x}(x_t, r) = \mathbb{E}[x \mid x_t, r] = \mathbb{E}[x \mid \tilde{x}_t, r]$ is the same in both parametrisations, and the estimator in Eq. (17) is unchanged. The Jacobian of the reparametrisation $\tau \mapsto t$ is absorbed by integrating along $\log \tau$ rather than τ directly, which is precisely the log-SNR grid used in Appendix H.2.

Score network. A conditional score network $\epsilon_\theta(x_t, t, r)$ is trained to predict the noise realisation z from the noisy image, the diffusion step and the response vector r, using the standard DDPM mean-squared-error objective $\mathbb{E}\big[\|\epsilon_\theta(x_t, t, r) - z\|^2\big]$ [9]. At the minimum of this objective, $\epsilon_\theta(x_t, t, r) = \mathbb{E}[z \mid x_t, t, r]$. Taking $\mathbb{E}[\,\cdot \mid x_t, r]$ on both sides of Eq. (34) and using linearity of conditional expectation then yields the conditional posterior mean as a direct function of the learned noise prediction,

$$\hat{x}(x_t, r) \;=\; \frac{1}{\sqrt{\bar{\alpha}_t}}\Big(x_t \;-\; \sqrt{1 - \bar{\alpha}_t}\, \epsilon_\theta(x_t, t, r)\Big) \;=\; \mathbb{E}[x \mid x_t, r]. \tag{37}$$

The same relation with r replaced by the null token $r = \mathbf{0}$ recovers the *unconditional* posterior mean $\hat{x}(x_t) = \mathbb{E}[x \mid x_t]$ from the null-token prediction $\epsilon_\theta(x_t, t, \mathbf{0})$; we exploit this fact in the next subsection.

Observation model for responses. Throughout the paper we treat the recorded response vector r as a noisy observation of a deterministic image-to-response map $f_\phi(x)$ (the encoder, Appendix I),

$$p(r \mid x) \;=\; \mathcal{N}\big(f_\phi(x),\, \sigma^2 I\big), \qquad \sigma = 0.22. \tag{38}$$

The Gaussian form matches the statistics of the responses we actually use: the `normMUA` of Papale et al. [20] is trial-averaged and per-channel z-scored (Appendix I.1), producing an approximately continuous and symmetrically distributed signal rather than raw spike counts. This differs from the setting of Laquitaine et al. [14], whose conditioning vector is the (integer) spike count of a 49-neuron LNP-style model and is therefore drawn, at every training step, from a Poisson observation model $r \sim \mathrm{Poisson}(f_\phi(x))$. Both choices share the same structure—fix the encoder f_ϕ and match the training-time distribution of r to the observation model consistent with the data—but differ in which observation model is appropriate for the signal at hand. We calibrate σ to the residual trial-to-trial variability of the recordings: each held-out stimulus in Papale et al. [20] is presented 30 times, and for each electrode we compute the standard deviation of the z-scored per-trial response around the trial-averaged mean. The median of this quantity across electrodes is 0.22 in V1 and 0.25 in V4; we use $\sigma = 0.22$ for both areas, so that a single observation-noise scale is shared across all encoders and areas.

We stress that this σ is an actual property of the observation model in Eq. (38), not merely a regularisation trick: it appears unchanged in the training recipe below and in the response sampler of Appendix H.2.

Architecture. The score network ϵ_θ is implemented as a 2D UNet with cross-attention conditioning, using the standard `UNet2DConditionModel` configuration from the Hugging Face `diffusers` library. It operates on 64×64 grayscale images (one input channel, one output channel) with block-output widths $(64, 128, 256, 512)$, two residual layers per block, and cross-attention layers in the 16×16 down and up blocks and in the 8×8 middle block, following the implementation of Laquitaine et al. [14]. Neural responses enter through a three-layer MLP with SiLU activations that maps the 50-dimensional response vector r to a single 256-dimensional token, which is fed to the cross-attention layers as the `encoder_hidden_states` argument. Together with the conditioning embedder, the model has approximately 68 million parameters.

Training-time noise injection and classifier-free guidance. Two features of the training recipe are essential for the estimator of Eq. (17) to be numerically stable. First, we inject the observation noise of Eq. (38) into the conditioning vector at every training step: during each forward pass the network sees $\tilde{r} = r + \sigma\xi$, with $\xi \sim \mathcal{N}(0, I)$ and the same $\sigma = 0.22$ as in Eq. (38); note that this observation noise ξ is an independent random variable from the diffusion-process noise z in Eq. (34).

Second, we apply classifier-free guidance (CFG) dropout at rate $p_{\text{drop}} = 0.1$ [19]: with that probability, the entire conditioning vector is replaced by zero during a training step. A single network therefore learns both the conditional noise estimate $\epsilon_\theta(x_t, t, r)$ and, when queried with $r = \mathbf{0}$, an unconditional estimate $\epsilon_\theta(x_t, t, \mathbf{0}) \approx \epsilon^u_\theta(x_t, t)$. Without CFG training the model has no meaningful response at the null token, and the unconditional posterior mean $\hat{x}(x_t) = \mathbb{E}[x \mid x_t]$ used in the denoising step of the next subsection cannot be recovered from a single forward pass.

Optimisation and schedule. Training uses the DDPM MSE objective [9] with $T = 1000$ steps on the linear β-schedule. We optimise with AdamW at learning rate 10^{-4}, weight decay 10^{-4}, and a cosine schedule with 500 warm-up steps, for 200 epochs on $100{,}000$ TinyImageNet [46] training images resized to 64×64 and converted to single-channel gray. TinyImageNet has no associated neural recordings; the conditioning vector r for each training image is therefore generated synthetically by evaluating the deterministic encoder f_ϕ (Appendix I) on the image, yielding a 50-dimensional response vector that is then perturbed by the observation noise $\sigma\xi$, $\xi \sim \mathcal{N}(0, I)$, described above. The encoder f_ϕ is itself fitted on the macaque V1/V4 `normMUA` of Papale et al. [20], so the conditioning distribution seen by the diffusion model approximates that of the recorded population. Batches of 128 images are processed in `bfloat16` precision; we clip gradients at global norm 1.0. We use the checkpoints after 200 epochs throughout. One diffusion model is trained per (cortical area, encoder) pair in Figs. 3–4.

Compute resources. All training and inference ran on an internal cluster of NVIDIA A100-SXM4-40GB GPUs. Each conditional diffusion model was trained on a single A100, at ≈ 4.1 min per epoch,

so a 200-epoch run takes ≈ 13.5 hours. The six models (Inception, ResNet, ON-OFF, each fitted to V1 and V4) cost ≈ 81 A100-GPU-hours. The encoder readouts on top of frozen backbones are trained on pre-extracted features and are negligible in comparison (< 1 A100-GPU-hour in total). Estimating $G(x)$ with Algorithm 1, together with the Fisher baseline, took 33–85 s per image on a single A100 (median ≈ 40 s; nodes were shared with other jobs), so the 200 test images $\times$ 6 configurations add ≈ 15 A100-GPU-hours, for a total of ≈ 100 A100-GPU-hours for the reported experiments. The Monte Carlo convergence analysis (Appendix G) used a further ≈ 8 A100-GPU-hours, and the simulations of Appendices B–E ran on CPUs in minutes. Earlier and exploratory runs outside the reported sweep, including diffusion models trained for 500 epochs, used more than four times this compute.

H.2 Estimating metric tensor eigenvectors with the diffusion model

The goal of this subsection is to describe the Monte Carlo estimator used to obtain the leading eigenvectors of the multi-scale metric $G(x)$ from the trained score network ϵ_θ of Appendix H.1.

Numerical integration over diffusion scales. Given a clean image x, we estimate $G(x)$ by Monte Carlo integration of Eq. (17) over diffusion scales t and realisations of the forward-process noise $z \sim \mathcal{N}(0, I)$ in Eq. (34). We discretise t on a coarse grid $\{t_j\}_{j=1}^{J}$ with $t_1 = 40$, $t_J = 960$ and step 40 ($J = 24$ timesteps), and for each t_j draw $N_{\mathrm{MC}} = 200$ independent noise samples in mini-batches of 25. The line element of Eq. (17) is smooth in the log signal-to-noise variable $\log \tau$ (with $\tau = (1 - \bar{\alpha}_t)/\bar{\alpha}_t$ from Eq. (36)) rather than in t itself, so we integrate with respect to $\log \tau$ using the trapezoid weights $w_j = \frac{1}{2}(\log \tau_{j+1} - \log \tau_{j-1})$. This is a change of integration variable: the timesteps t_j are spaced uniformly on the DDPM index axis, and w_j is the corresponding discrete integration step $\Delta(\log \tau_j)$ in the log-SNR variable. The Jacobian of the reparametrisation, $\mathrm{d}\tau = \tau\, \mathrm{d}(\log \tau)$, has already been absorbed into the integrand: combining the $1/\tau^2$ factor that Tweedie's identity contributes to $J_t(\tilde{x}_t)$ in Eq. (17) with the relation $\hat{x}(\tilde{x}_t, r) - \tilde{x}_t = -\sqrt{\tau}\, \epsilon_\theta(x_t, t, r)$ leaves a $1/\tau$ factor, which is exactly cancelled by the τ from $\mathrm{d}\tau = \tau\, \mathrm{d}(\log \tau)$, so the $\log \tau$-integrand reduces to squared noise-prediction differences weighted by w_j (Algorithm 1). The resulting matrix $D \in \mathbb{R}^{(J \cdot N_{\mathrm{MC}}) \times P}$ ($P = 64^2$ pixels) satisfies $G(x) \approx D^\top D$, and its truncated singular value decomposition $D = U \Sigma V^\top$ yields $\lambda_k = \Sigma_{kk}^2$ and $v_k = V_{:,k}$ as the top-50 eigenvalues and eigenvectors of the estimated metric.

Sampling responses from the diffused likelihood. Evaluating Eq. (17) requires drawing $r \sim p_t(r \mid x_t)$, the response distribution conditional on the *diffused* image x_t. A direct Monte Carlo procedure would first draw $x \sim p(x \mid x_t)$ and then $r \sim p(r \mid x)$, but sampling $x \sim p(x \mid x_t)$ requires a full backward pass of the diffusion model at each (x_t, t) and is prohibitively expensive. We follow the approximation introduced by Chung et al. [10] and adopted for the information-decomposition setting by Laquitaine et al. [14]: replace the expensive $x \sim p(x \mid x_t)$ step by a point mass at the posterior mean $\hat{x}(x_t) = \mathbb{E}[x \mid x_t]$, which can be obtained in a single forward pass of the score network. Concretely, at each (x_t, t) we proceed in three steps. First, we read the unconditional posterior mean off Eq. (37) with $r = \mathbf{0}$,

$$\hat{x}(x_t) \;=\; \frac{x_t - \sqrt{1 - \bar{\alpha}_t}\, \epsilon_\theta^u(x_t, t)}{\sqrt{\bar{\alpha}_t}},$$

using $\epsilon_\theta^u(x_t, t) \approx \epsilon_\theta(x_t, t, \mathbf{0})$ from the null-token prediction of the same network (enabled by the CFG training of Appendix H.1). Second, we clip $\hat{x}(x_t)$ to $[-1, 1]$ since the clean image lives in this range. Third, we draw $r = f_\phi(\hat{x}(x_t)) + \sigma\xi$ with $\xi \sim \mathcal{N}(0, I)$ and $\sigma = 0.22$, i.e. we sample from the observation model of Eq. (38) evaluated at the denoised image. For the InceptionV3 and ResNet-50 encoders, f_ϕ accepts the $[-1, 1]$-rescaled image directly; for the ON-OFF encoder, which operates on luminance images in $[0, 1]$ (Appendix I.4), a thin wrapper rescales the clipped $\hat{x}(x_t) \in [-1, 1]$ to $[0, 1]$ before applying the front-end. The total cost is one extra forward pass of ϵ_θ per Monte Carlo batch to compute $\epsilon_\theta^u(x_t, t)$, plus one evaluation of f_ϕ on the denoised image.

Validity of the point-mass approximation at coarse scales. Replacing the broad diffused likelihood $p_t(r \mid x_t)$ by a point mass at the posterior mean $\hat{x}(x_t)$ collapses the response variance to the observation-noise floor $\sigma^2 I$; the gap is largest at coarse scales (large τ), where the true posterior $p(x \mid x_t)$ is broad. We accept this trade-off because the integrand we are approximating, $J_t(x_t)$,

is itself *small* at large τ: as $\tau \to \infty$ the diffused likelihood $p_t(r \mid x_t)$ converges to the marginal $p(r)$, so the true Fisher contribution at coarse scales tends to zero independently of the sampling approximation, and any point-mass bias is multiplied by a vanishing weight. An exact alternative — drawing $x \sim p(x \mid x_t)$ via a full backward diffusion pass at every Monte Carlo slot — would be prohibitive at our compute budget, and the point-mass approximation is established practice for diffusion posterior sampling [10] and in the closely related information-decomposition setting [14]. The cross-encoder robustness reported in Fig. 4 (Inception and ResNet-50 encoders fitted independently to the same data) provides indirect evidence that the bias does not dominate the leading $G(x)$ subspace, since the two encoders traverse different bias profiles and yet agree on the leading directions.

Conditional-pair estimator. For every (x_t, t) and every Monte Carlo slot we draw *two* independent response vectors $r_1, r_2 \sim p(r \mid \hat{x}(x_t))$ and take the difference of their conditional noise predictions,

$$D_{(j,s),:} \;=\; \sqrt{\tfrac{w_j}{2\,N_{\mathrm{MC}}}}\,\big(\epsilon_\theta(x_t, t, r_1) - \epsilon_\theta(x_t, t, r_2)\big), \tag{39}$$

where the factor $\frac{1}{2}$ inside the square root compensates for $\mathrm{Var}(A - B) = 2\,\mathrm{Var}(A)$ under i.i.d. draws and w_j is the log-τ trapezoid weight introduced above. Because both terms in Eq. (39) are evaluated at the same (x_t, t) and use the same conditional cross-section of ϵ_θ, any systematic error of the denoiser that is common to the two draws cancels in the subtraction. Eq. (39), the conditional–conditional (CC) form, is an alternative Monte Carlo realisation of Eq. (17) to the unconditional–conditional (UC) form $\epsilon_\theta(x_t, t, r) - \epsilon^u_\theta(x_t, t)$ used by Laquitaine et al. [14] with two separately trained networks; we use Eq. (39) throughout for consistency with our single-network CFG recipe of Appendix H.1. On a small held-out set of test images with multiple independent Monte Carlo realisations per image, and at matched cost (300 forward passes of the network per scale for each estimator), the top-5 eigenvalues of the CC estimate had a median 67% lower Monte Carlo variance than those of the UC form. The leading UC eigenvalue was 2.8–5.5 times the CC value, and this factor varied across images, so it cannot be corrected by a global rescaling of the metric. The main advantage of the CC form is that it avoids this mismatch between conditional and unconditional predictions. Per Monte Carlo sample, CC costs three forward passes (one null-token and two conditional), and UC two.

Algorithm. Algorithm 1 collects the steps of Appendix H.2 into a single pseudocode listing for reproducibility. A forward pass through ϵ_θ consists of passing the noisy image x_t and diffusion step t through the UNet2DConditionModel while feeding the response vector r (or the null token $\mathbf{0}$) through the 3-layer SiLU MLP of Appendix H.1; the resulting 256-dimensional token is supplied as the single-token `encoder_hidden_states` argument to the cross-attention layers of the UNet.

Fisher baseline. The classical Fisher information matrix $J(x)$ used as the comparison baseline in Figs. 3F, 3H, 4C, 4E and 5B is computed in closed form from the Gaussian observation model of Eq. (38). Since $p(r \mid x) = \mathcal{N}(f_\phi(x), \sigma^2 I)$ has constant covariance, the standard formula for the Fisher information of a multivariate Gaussian reduces to

$$J(x) \;=\; \frac{1}{\sigma^2}\, J_{f_\phi}(x)^\top J_{f_\phi}(x), \tag{40}$$

where $J_{f_\phi}(x) \in \mathbb{R}^{n \times P}$ is the Jacobian of the encoder mean $f_\phi : \mathbb{R}^P \to \mathbb{R}^n$ at the input image x, with $P = 64 \times 64 = 4096$ pixels and $n = 50$ target neurons. We evaluate $J_{f_\phi}(x)$ via PyTorch reverse-mode autograd, materialising it row-by-row (one backward pass per neuron). Because J_{f_ϕ} has rank at most $n \ll P$, we then extract the top-50 eigenpairs of $J(x)$ directly from the SVD of $J_{f_\phi} = USV^\top$ rather than forming the $P \times P$ outer product: the eigenvalues of $J(x)$ are S_i^2/σ^2 and the eigenvectors are the rows of $V^\top$. The same set of 200 test images is used for the multi-scale metric, so any Fisher-vs-$G(x)$ comparison fixes the encoder, the dataset, and the rank. The test images are the first 200 images of the TinyImageNet validation set, which the diffusion models never saw during training.

I Predictive neural models

The goal of the encoding models described in this appendix is to provide the image-to-response map f_ϕ of Eq. (38): given an image x, $f_\phi(x)$ should predict the trial-averaged `normMUA` of each recorded

Algorithm 1 Conditional-pair estimator of the top-k eigenpairs of $G(x)$.

Require: clean image $x \in [-1,1]^{64\times 64}$; trained score network ϵ_θ; encoder f_ϕ; DDPM schedule $\{\bar\alpha_t\}_{t=1}^T$; timestep grid $\{t_j\}_{j=1}^J$ with $J = 24$, $t_1 = 40$, $t_J = 960$; Monte Carlo count $N_{\text{MC}} = 200$; batch size $B = 25$; observation-noise scale $\sigma = 0.22$; rank k

Compute $\log \tau_j \leftarrow \log\big[(1-\bar\alpha_{t_j})/\bar\alpha_{t_j}\big]$ for $j = 1, \dots, J$
Compute trapezoid weights $w_j \leftarrow \frac{1}{2}(\log\tau_{j+1} - \log\tau_{j-1})$ (with half-weights at endpoints)
Initialise empty list $\mathcal{D}$
for $j \leftarrow 1$ to J **do**
  $s_j \leftarrow \sqrt{w_j/(2N_{\text{MC}})}$
  for $n \leftarrow 1$ to N_{MC} in batches of B **do**
    Draw $z \sim \mathcal{N}(0, I)$, shape $(B, 1, 64, 64)$
    $x_t \leftarrow \sqrt{\bar\alpha_{t_j}}\, x + \sqrt{1-\bar\alpha_{t_j}}\, z$ ▷ Eq. (34)
    $\epsilon^u \leftarrow \epsilon_\theta(x_t,\, t_j,\, \mathbf{0})$ ▷ null-token forward pass
    $\hat x \leftarrow (x_t - \sqrt{1-\bar\alpha_{t_j}}\, \epsilon^u)/\sqrt{\bar\alpha_{t_j}}$ ▷ Eq. (37) with $r = \mathbf{0}$
    $\hat x \leftarrow \text{clip}(\hat x,\, -1,\, 1)$
    $\mu \leftarrow f_\phi(\hat x)$ ▷ deterministic response prediction
    Draw $\xi_1, \xi_2 \sim \mathcal{N}(0, I)$, shape $(B, 50)$
    $r_1 \leftarrow \mu + \sigma\,\xi_1$; $r_2 \leftarrow \mu + \sigma\,\xi_2$ ▷ Eq. (38)
    $\epsilon_1^c \leftarrow \epsilon_\theta(x_t,\, t_j,\, r_1)$; $\epsilon_2^c \leftarrow \epsilon_\theta(x_t,\, t_j,\, r_2)$
    $\Delta \leftarrow \text{flatten}(\epsilon_1^c - \epsilon_2^c)$, shape $(B, 64^2)$
    Append $s_j\,\Delta$ to $\mathcal{D}$ ▷ Eq. (39)
  end for
end for
Stack $\mathcal{D}$ into a matrix $D \in \mathbb{R}^{(J\cdot N_{\text{MC}})\times 64^2}$
$(U, \Sigma, V^\top) \leftarrow \text{SVD}(D)$
return eigenvalues $\{\Sigma_{\ell\ell}^2\}_{\ell=1}^k$ and eigenvectors $\{V_{:,\ell}\}_{\ell=1}^k$ of $G(x) \approx D^\top D$

neuron so that the Gaussian observation model $p(r \mid x) = \mathcal{N}(f_\phi(x), \sigma^2 I)$ is a good fit to the data. We fit f_ϕ by maximising the corresponding Gaussian log-likelihood on paired stimulus–response data, which under our model reduces to minimising the per-neuron mean-squared error between predicted and trial-averaged responses. The fitted encoder then serves both as a readout of the underlying neural code (Figs. 3–4) and as the deterministic component f_ϕ used inside the response sampler of Appendix H.2. This appendix also describes the Bures–Wasserstein distance on PSD matrices that we use as an alternative measure of geometric similarity between reconstructed metric tensors.

I.1 Data processing

All encoding models are fitted on the *THINGS ventral-stream spiking dataset* (TVSD) from Papale et al. [20], which reports multi-unit activity from chronically implanted Utah-array recordings in an awake-behaving macaque while it fixated on 22,248 natural photographs drawn from the THINGS database [47]. The dataset provides a normalised multi-unit firing rate (`normMUA`) per recording channel, the train/test split used in Papale et al. [20], and a per-channel split-half *oracle* reliability that quantifies how much of the trial-averaged response is reproducible across repeats. Following the original paper, we rely on the provided electrode-to-area assignment (V1: channels 0–511; V4: 512–767) and, within each area, retain the 50 channels with the highest oracle reliability as the targets of our encoders. For Inception [25] and ResNet [26] encoders the stimuli are resized to 64×64, converted to a single grayscale channel, replicated across the three input channels expected by the pre-trained backbone, and rescaled to $[-1, 1]$ via the usual mean-0.5, std-0.5 normalisation. For the ON-OFF encoder, which operates directly on a single-channel luminance image in $[0, 1]$, we omit the final rescaling step.

Per-neuron oracle reliabilities (mean across the selected 50 neurons) are $\rho = 0.92$ for V1 and $\rho = 0.90$ for V4. These values put a ceiling on what *any* encoder can achieve against trial-averaged responses; the noise-corrected correlations in Table 4 should therefore be read as the fraction of that explainable signal actually captured by each encoder.

Table 4: Predictive performance of all encoders reported in Figs. 3–4. For each encoder and area we list the mean raw Pearson test correlation $\bar{r}$ between predicted and trial-averaged neural responses on the Papale test set, and the corresponding noise-corrected correlation $\bar{r}_{\text{nc}}$. The latter is computed per neuron as r_n/ρ_n, where ρ_n is the split-half oracle reliability of neuron n, and then averaged across the 50 target neurons; values close to 1 indicate performance near the intrinsic noise ceiling of the recording.

	V1		V4	
Encoder	$\bar{r}$	$\bar{r}_{\text{nc}}$	$\bar{r}$	$\bar{r}_{\text{nc}}$
Inception	0.75	0.81	**0.71**	**0.79**
ResNet-50	**0.78**	**0.85**	0.70	0.78
ON-OFF	0.20	0.22	0.13	0.14

I.2 InceptionV3 model

Both of our deep CNN encoders follow the factorised-readout template that is standard in the task-trained vision-to-neural-prediction literature [48, 38, 20]: a task-trained classifier backbone is truncated at an intermediate layer and frozen, and only a small per-neuron readout is fit to the neural data. For the encoder described in this subsection, the backbone is InceptionV3 [25] with the ImageNet-pretrained weights provided by `torchvision` [49], truncated at the `Conv2d_4a_3x3` layer so that a 64×64 input image produces a feature map of shape $192 \times 12 \times 12$.

We chose the truncation layer in a preliminary sweep over the candidate Inception blocks for each area. `Conv2d_4a_3x3` gave the highest correlation in *both* V1 and V4; deeper blocks did not help V4 and hurt V1. This departs from the common convention of assigning earlier layers to V1 and deeper layers to V4/IT [48, 38], but is itself common in practice: many brain-score leaderboard entries report a single best layer across multiple ventral-stream areas [38].

Using the same layer for V1 and V4 is important for the interpretation of our results: because both areas are read out from an identical feature basis, any systematic difference in $G(x)$ between V1 and V4 (Fig. 3) cannot be a layer-choice artefact and must reflect how the shared features are weighted to predict each area. The backbone is frozen throughout training.

On top of the frozen features we fit a per-neuron spatial-attention readout: each target neuron n is parameterised by a centre $\mu_n \in \mathbb{R}^2$, a width $\sigma_n > 0$, a feature-weight vector $w_n \in \mathbb{R}^C$ and a bias b_n. The predicted response is

$$\hat{r}_n(x) \;=\; \sum_c w_{n,c} \sum_{h,w} a_n(h,w)\, \phi_c(x)_{h,w} \;+\; b_n, \qquad a_n(h,w) \propto \exp\!\Big(-\tfrac{\|(h,w)-\mu_n\|^2}{2\sigma_n^2}\Big), \quad (41)$$

where a_n is normalised to sum to one over the feature-map grid and $\phi_c(x)$ is the c-th channel of the frozen backbone output. We optimise the per-neuron readout with AdamW at learning rate $3 \cdot 10^{-4}$, batch size 64, dropout 0.25 applied to the feature map, selecting the epoch with the highest validation correlation (epochs 79 and 93 for V1 and V4). Model selection uses a 10% hold-out validation split of the THINGS training images, kept disjoint from the Papale test set; all accuracy numbers used in Figs. 3–4 and Table 4 are computed on the held-out Papale test images, averaging the Pearson correlation between predicted and trial-averaged responses across the 50 target neurons. Our Inception encoder reaches 0.81 (V1) and 0.79 (V4) of the per-neuron noise ceiling (Table 4), matching the headline performance reported by Papale et al. [20] for comparable factorised readouts on TVSD.

I.3 Resnet50 model

To control for backbone-specific artefacts we also fit an encoder with the same readout structure on top of a pre-trained ResNet-50 [26]. Input images are standardised with the ImageNet mean and standard deviation inside the backbone wrapper; feature extraction stops at the first block of the first stage (`layer1.0`), which produces a $256 \times 16 \times 16$ feature map for our 64×64 input. The remainder of the readout, the loss, and the optimisation schedule are identical to those of the Inception encoder. On the Papale test set the ResNet-50 encoder reaches 0.85 (V1) and 0.78 (V4) of the per-neuron noise ceiling (Table 4), on par with or slightly above the Inception encoder on the same neurons.

This matched accuracy is what allows the cross-architecture comparison in Fig. 4: any systematic differences between the two encoders' geometries that survive cross-model averaging cannot be attributed to one model being a substantially better fit to the data.

I.4 On-off model

Our third encoder is a biologically motivated front-end in the spirit of Berardino et al. [28]. The backbone is the `OnOff` module from the `plenoptic` library [50], which produces two spatial channels encoding on-centre and off-centre luminance and contrast responses via centre-surround filters (31×31 kernels loaded with the pre-trained Berardino weights) and is kept frozen throughout training, exactly as for the Inception and ResNet backbones. The frozen front-end consists of 12 biologically interpretable scalars — on/off luminance and contrast gain controls and centre/surround radii of the difference-of-Gaussians filters — four orders of magnitude fewer than the $\sim$173k frozen convolutional parameters in the Inception backbone (Appendix I.2). On top of the frozen ON-OFF feature map we fit a dense linear readout that maps the $8{,}192$ features to the 50 target neurons ($409{,}650$ trainable parameters), by minimising the mean-squared error with model selection on validation correlation. Test correlations on the Papale test set are substantially lower than with the deep encoders: the ON-OFF encoder reaches about 22% of the per-neuron noise ceiling in V1 and 14% in V4 (Table 4), as expected from the dramatic reduction in front-end capacity. As discussed in the main text, this simpler encoder is informative not because it matches the data but because its Fisher geometry becomes interpretable once the model no longer has enough capacity to fit high-frequency idiosyncrasies of the neural code.

I.5 Comparing different geometries

Throughout Fig. 4D–E we quantify how similar two metrics A and B are by the alignment of their leading eigenvector subspaces. Given truncated eigendecompositions $A \approx V_A \Lambda_A V_A^\top$ and $B \approx V_B \Lambda_B V_B^\top$ with orthonormal $V_A, V_B \in \mathbb{R}^{P \times k}$, the *subspace alignment* of A and B at rank k is

$$\mathcal{A}_k(A, B) \;=\; \frac{1}{k} \, \|V_A^\top V_B\|_F^2 \;=\; \frac{1}{k} \sum_{i=1}^{k} \cos^2 \theta_i, \tag{42}$$

the mean squared cosine of the principal angles $\{\theta_i\}_{i=1}^k$ between the two subspaces [51]; $\mathcal{A}_k \in [0, 1]$, with $\mathcal{A}_k = 1$ iff the top-k subspaces coincide.

Subspace alignment (Fig. 4D–E) only constrains the angles between eigenvectors and is insensitive to the magnitude of the associated eigenvalues. To double-check the conclusions of Fig. 4 we therefore also compare the reconstructed metric tensors using the *Bures–Wasserstein* distance [27] on positive semi-definite (PSD) matrices, which simultaneously penalises mismatches in orientation and in scale. Given two PSD matrices A and B the Bures–Wasserstein distance is

$$d_{\mathrm{BW}}^2(A, B) \;=\; \mathrm{tr}(A) + \mathrm{tr}(B) \;-\; 2\,\mathrm{tr}\Big((A^{1/2} B A^{1/2})^{1/2} \Big), \tag{43}$$

and corresponds, for symmetric mean-zero Gaussians with covariances A and B, to the 2-Wasserstein distance between the two distributions. Of the many distances available on the cone of PSD matrices, Harvey et al. [52] argue that Bures–Wasserstein is the most appropriate choice for *decodable-information* notions of representational similarity, since it has a direct optimal-transport interpretation between the two representational geometries.

In practice we work with the truncated eigendecomposition $A \approx V_A \Lambda_A V_A^\top$, $B \approx V_B \Lambda_B V_B^\top$ returned by the diffusion estimator of Appendix H.2, with $V_A, V_B \in \mathbb{R}^{P \times k}$ orthonormal and Λ_A, Λ_B the diag of the top-k eigenvalues. Writing $C = V_A^\top V_B \in \mathbb{R}^{k \times k}$ for the overlap matrix between the two sets of eigenvectors, the cross term in Eq. (43) becomes $\mathrm{tr}\Big[\big(\Lambda_A^{1/2} \, C \, \Lambda_B \, C^\top \, \Lambda_A^{1/2} \big)^{1/2} \Big] = \sum_i \mu_i^{1/2}$ where $\{\mu_i\}$ are the eigenvalues of $M = \Lambda_A^{1/2} C \Lambda_B C^\top \Lambda_A^{1/2}$; we symmetrise M before diagonalisation to guard against numerical asymmetry and sweep $k = 1, \ldots, 50$.

Fig. 9 reports d_{BW} as a function of k, averaged across 200 test images, for the four encoder/area comparisons of Fig. 4. For the multi-scale metric, cross-model distances (same area, different backbone) lie below cross-area distances (same backbone, different area); for the Fisher information the ordering reverses on average (three of the four cross-model/cross-area pairings). This matches the

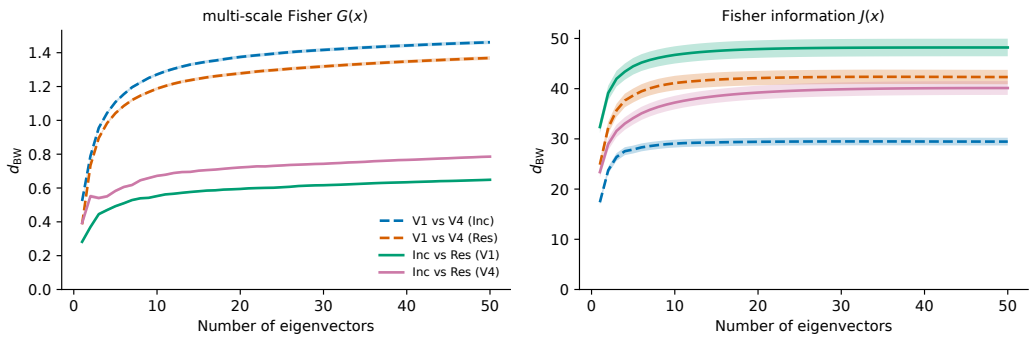

Figure 9: **Bures–Wasserstein distance between reconstructed geometries.** d_{BW} computed from the top-k eigenvalue/eigenvector pairs (Eq. (43)) as a function of k, averaged across images, for four encoder/area comparisons: V1 vs V4 within each backbone (Inception, ResNet) and Inception vs ResNet within each area (V1, V4). (**Left**) Multi-scale metric $G(x)$; cross-model distances (same area, different backbone) lie below cross-area distances (same backbone, different area). (**Right**) Fisher information $J(x)$; the ordering reverses, with cross-model distances exceeding cross-area distances. Shaded bands: s.e.m. across 200 test images.

subspace-alignment ordering of Fig. 4D–E, so our conclusions about the multi-scale metric do not depend on the specific choice of similarity measure.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and Introduction (§1) make three claims, each supported by a corresponding section: (i) requirements on how distances contract under coarse-graining determine a unique multi-scale extension of the Fisher metric (§3, which states that uniqueness holds for a given coarse-graining, with the derivations in Appendix A.1); (ii) this metric is exactly related to the mutual information (§4 and Appendix A.2); (iii) the metric can be estimated with a conditional diffusion model (§5 and Appendix H), and, for encoding models fitted to macaque V1 and V4, its eigenvectors differ between areas and are stable across encoding models, whereas the eigenvectors of the Fisher information depend on the model (§6, Figs. 3–5).

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper states its limitations. The Discussion (§7) gives two: (i) like the Fisher information, the method requires a probabilistic encoding model $p(r \mid x)$ and enough data to fit it; (ii) it is more expensive to compute than the Fisher information, as it requires training a diffusion model and Monte Carlo estimation. Three others are stated where they apply: (iii) the geometry is unique only for a given coarse-graining, here isotropic Gaussian noise in pixel space (§3); (iv) our V1 and V4 models treat trial-to-trial variability as independent across neurons, so the reported geometries do not account for noise correlations (§6; Appendix D shows that the estimator itself applies to correlated responses); (v) we did not characterise how the estimated geometry depends on the amount of recorded data or on the number of neurons (Appendix G, which also reports the Monte Carlo convergence).

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.

- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: The paper has two theoretical results. (i) For a given coarse-graining, the multiscale metric $G(x)$ is the unique solution of the requirements of §3. The requirements and the boundary condition are stated in §3, the regularity conditions on $p(r \mid x)$ in a footnote of §2, and the derivation, including the use of Čencov's theorem, in §3 and Appendix A.1. (ii) The relation to mutual information (§4). The identity $I(R;X) = \frac{1}{2}\mathbb{E}_x[\operatorname{Tr} G(x)]$ is due to Laquitaine et al. [14]; Appendix A.2 states it with its assumptions (a memoryless channel, and finite first and second moments of $p(x)$ and $p(x \mid r)$) and proves its two geometric forms, in terms of squared geodesic distances and of local magnification. Appendix B gives the closed form for Gaussian stimuli and linear-Gaussian responses.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Appendix H.1 specifies the conditional diffusion model (UNet2DConditionModel architecture, conditioning MLP, parameter count, noise schedule, observation model, classifier-free guidance training and optimisation). Appendix H.2 and Algorithm 1 give the Monte Carlo estimator of the leading eigenvectors of $G(x)$, including the timestep grid, the integration weights in $\log \tau$, the sampling of responses and the Fisher baseline. Appendix I documents the Inception, ResNet-50 and ON-OFF encoders, and Appendix I.1 the publicly released dataset of Papale et al. [20] and its preprocessing. Section 6 gives the perturbation steps of Fig. 5, Appendix F defines the eigenvector statistics and their tests, and the simulations of Appendices B–E are specified in those appendices.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.

- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The neural dataset of Papale et al. [20] is publicly released by its authors, and TinyImageNet [46] is a standard public dataset. The code for the conditional-pair estimator, the encoder and diffusion training pipelines and the figure scripts, together with the trained encoder and diffusion checkpoints, is released at `https://multigeom.github.io/`.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Appendix H.1 specifies the diffusion-model optimiser (AdamW, learning rate 10^{-4}, weight decay 10^{-4}, cosine schedule with 500 warm-up steps), training length (200 epochs on 100,000 TinyImageNet [46] images), batch size (128 in `bfloat16` with gradient clipping at 1.0), classifier-free guidance dropout (p_{drop}=0.1), and observation-noise scale (σ=0.22, calibrated to the residual trial-to-trial variability in the data). Appendix H.2 specifies the estimator hyperparameters (timestep grid t_1=40, t_J=960, step 40, J=24; N_{MC}=200 samples, batch size 25). Encoder train/test splits follow Papale et al. [20] and are documented in Appendix I.1; encoder accuracy uses the Papale test set, and the geometry analyses use 200 held-out TinyImageNet images. Section 6 gives the perturbation step of Fig. 5 (1% of the Euclidean norm of the reference image), and Appendix G the settings of the Monte Carlo convergence runs.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Fig. 4A shows per-neuron test correlations for each encoder as boxplots (median, interquartile range, whiskers, outliers) across the 50 neurons of each area. Fig. 4D–E show the subspace alignment as a function of k, as the mean across the 200 test images with s.e.m. shading, and Fig. 9 (Appendix I.5) shows the Bures–Wasserstein distance in the same way. The differences between V1 and V4 eigenvectors (Fig. 3G–H, Appendix F) and the ordering of alignments in Fig. 4D–E were tested per image with Wilcoxon signed-rank tests across the 200 test images, and Fig. 7 shows mean $\pm$ s.e.m. over images. The Monte Carlo error of the estimator is quantified in Appendix G (5 independent runs on each of 10 images; Fig. 8, mean $\pm$ s.e.m. over images) and compared between the conditional-pair and unconditional–conditional forms in Appendix H.2. The simulation results (Tables 2 and 3) report the estimate with its s.e.m.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).

- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Appendix H.1 ("Compute resources" paragraph) reports the hardware (internal cluster of NVIDIA A100-SXM4-40GB), per-epoch wall time (≈ 4.1 min on one A100), per-model cost (≈ 13.5 A100-GPU-hours for a 200-epoch run), and the total for the reported experiments: ≈ 81 A100-GPU-hours of diffusion training (6 models), < 1 A100-GPU-hour of encoder readout training, and ≈ 15 A100-GPU-hours of $G(x)$ and Fisher evaluation (200 images $\times$ 6 configurations), for ≈ 100 A100-GPU-hours, plus ≈ 8 A100-GPU-hours for the Monte Carlo convergence analysis. The simulations of Appendices B–E ran on CPUs in minutes. Earlier and exploratory runs, including diffusion models trained for 500 epochs, used more than four times this compute.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The work conforms to the NeurIPS Code of Ethics. We collect no new animal or human data; the neural recordings are reused from a previously published, publicly released macaque V1/V4 dataset [20] whose institutional ethics approvals are reported in the original publication.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: This is foundational theoretical and methodological work on the mathematical structure of mutual information in neural population codes. Its intended use is as an analysis tool for systems-neuroscience and representation-analysis research; we do not identify a direct path to negative societal impact or deployed decision-making systems.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: No high-risk assets are released. The trained diffusion models are single-channel 64×64 networks conditioned on 50-dimensional neural response vectors from one macaque dataset, and serve as denoisers for estimating the metric tensor. They have no identified dual-use risk.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All external assets are cited and used under their published licenses: the Papale *et al.* [20] macaque ventral-stream dataset; the THINGS image database [47]; TinyImageNet [46]; the `torchvision` library [49] and its ImageNet-pretrained InceptionV3 and ResNet-50 weights; the Hugging Face `diffusers` library [53] (Apache-2.0) for `UNet2DConditionModel`; and the `plenoptic` library [50] for the Berardino *et al.* [28] ON-OFF model.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.

- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We release the conditional-pair estimator code, the encoder and diffusion training pipelines, and the scripts that produce the figures and tables of the main text and appendices (except the schematic in Fig. 3A), together with the trained encoder and diffusion checkpoints, at `https://multigeom.github.io/`. The release documents the environment, the training commands, the checkpoint naming and the mapping from scripts to figures.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing or human-subjects research. The neural recordings are reused from awake-behaving macaques in the prior publication [20]; no new animal or human data are collected here.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: No new experiments with human or animal subjects were conducted for this paper. Animal-ethics approvals for the reused macaque dataset are documented in the original publication [20].

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: LLMs are not used as a component of the core methodology. The theoretical derivations, the conditional-diffusion estimator, the encoder training and the analysis pipeline do not invoke any LLM. LLMs were used to assist with writing, editing and code formatting, which per the NeurIPS policy does not require declaration.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.