Learning and Aligning Single-Neuron Invariance Manifolds in Visual Cortex

Anonymous authors
Paper under double-blind review

Abstract

Understanding how sensory neurons exhibit selectivity to certain features and invariance to others is central to uncovering the computational principles underlying robustness and generalization in visual perception. Most existing methods for characterizing selectivity and invariance identify single or finite discrete sets of stimuli. Since these are only isolated measurements from an underlying continuous manifold, characterizing invariance properties accurately and comparing them across neurons with varying receptive field size, position, and orientation, becomes challenging. Consequently, a systematic analysis of invariance types at the population level remains under-explored. Building on recent advances in learning continuous invariance manifolds, we introduce a novel method to accurately identify and align invariance manifolds of visual sensory neurons, overcoming these challenges. Our approach first learns the continuous invariance manifold of stimuli that maximally excite a neuron modeled by a response-predicting deep neural network. It then learns an affine transformation on the pixel coordinates such that the same manifold activates another neuron as strongly as possible, effectively aligning their invariance manifolds spatially. This alignment provides a principled way to quantify and compare neuronal invariances irrespective of receptive field differences. Using simulated neurons, we demonstrate that our method accurately learns and aligns known invariance manifolds, robustly identifying functional clusters. When applied to macaque V1 neurons, it reveals functional clusters of neurons, including simple and complex cells. Overall, our method enables systematic, quantitative exploration of the neural invariance landscape, to gain new insights into the functional properties of visual sensory neurons.

FIX

FIX

1 Introduction

To understand visual sensory processing, it is crucial to not only examine coding properties of individual neurons, but also how these coding properties are organized across the entire neural population. Two fundamental principles of sensory coding are selectivity and invariance: visual neurons exhibit selectivity to specific features such as orientation, while maintaining invariance to other transformations such as translation (Hubel & Wiesel, 1962; Riesenhuber & Poggio, 1999). For example, in the primary visual cortex (V1), simple cells are selective to the precise phase of a stimulus, whereas complex cells exhibit phase invariance, responding consistently regardless of phase shifts (Hubel & Wiesel, 1962). These computational properties enable neurons to maintain informative yet stable responses across diverse inputs, thus supporting reliable object recognition and visual processing.

Recently, data-driven approaches using Deep Neural Networks (DNNs), acting as *digital twins* of sensory neurons, have dramatically advanced our ability to model neural responses (Yamins & DiCarlo, 2016; Cadena et al., 2019; Lurz et al., 2020; Klindt et al., 2017). These approaches enable the identification of single maximally exciting inputs (MEIs) (Walker et al., 2019; Bashivan et al., 2019) or a diverse set of nearly maximally exciting stimuli that outline the invariance space of the neuron (Ding et al., 2023; Cadena et al., 2018). In this context, an invariance manifold can be defined as the set of stimuli that elicit near-maximal responses from a neuron. These stimuli are related by transformations to which the neuron is invariant, meaning the neuron's response remains constant (or nearly so) across these variations. For instance, the invariance manifold of a complex cell exhibiting phase invariance is the set of Gabor patterns that differ only in phase, and maximally excite it. While

NEW

invariances can also be explored at suboptimal response levels, focusing on peak response levels highlights the stimuli that the neuron is most selective to. Most existing methods, however, do not readily enable analyses to compare the invariance manifolds of neurons across populations and to identify clusters of neurons sharing common invariances. Ideally, such a method would fulfill the following requirements:

- It should accurately capture the invariance manifold of neurons, ensuring that subsequent analyses genuinely reflect neural coding.
- It should enable the identification of common invariance properties while disregarding differences in selected 'nuisance' receptive field (RF) attributes such as position, size, and orientation. FIX
- It should allow for a meaningful measure of the similarity of invariance properties across neurons.

Current approaches do not satisfy all of these requirements simultaneously.

In this paper, we propose a novel approach for aligning and comparing invariance manifolds across a population of neurons, fulfilling the above-mentioned requirements and allowing clustering of neurons based on their invariance properties. Our method first identifies the invariance manifold of individual neurons (Fig. 1 top panel), using the same approach as Baroni et al. (2023). It then learns an affine transformation on the pixel coordinates such that the same manifold activates another neuron as strongly as possible, effectively aligning their invariance manifolds spatially (Fig. 1 middle panel). Neurons whose invariance manifolds strongly activate each other likely share similar computational roles (Fig. 1 bottom panel). By performing this pairwise alignment across a population of neurons, our method systematically uncovers shared invariance properties, irrespective of differences in their receptive field attributes, allowing for clustering of neurons based on their invariances.

We test our method on both simulated and macaque V1 neurons, modeled by a response-predicting DNN. FIX The simulated neurons were carefully designed to display a range of invariance properties, and our method effectively learned and aligned their corresponding invariance manifolds, enabling accurate identification of the ground truth functional types. When applied to macaque V1 neurons, modeled by a response-predicting DNN, our approach captured and aligned the invariance manifolds of these neurons, uncovering multiple functional clusters, including, but not limited to, the expected clusters of canonical simple and complex cells.

Overall, our method generalizes effectively from synthetic to macaque V1 neurons, offering a robust framework for systematically analyzing invariance properties across sensory neurons, to deepen our understanding of the computational roles of individual neurons and gain new insights into the broader organizational principles governing neural populations.

2 Related Work

Our method is uniquely situated at the intersection of characterizing neuronal invariances and clustering neurons based on shared functional properties. The works discussed below are closely related to one or more aspects of our approach, either addressing neuronal invariances or functional clustering of neurons. NEW

Neuronal invariances Recent pioneering work has demonstrated the use of deep learning to characterize neural selectivity by identifying maximally exciting inputs (MEIs), which were subsequently validated through *in-vivo* experiments (Bashivan et al., 2019; Walker et al., 2019; Ponce et al., 2019). Using response-predicting DNNs, Cadena et al. (2018) and Ding et al. (2023) characterized neural invariances by generating multiple such MEIs for a neuron by directly optimizing the pixel values while encouraging diversity across the MEIs using an additional reguralization term.

Limitations of discrete approximations NEW While these methods mark a step forward in characterization of invariances, they approximate the underlying continuous invariance manifold through a finite set of points. This approach runs the risk of leaving gaps in the manifold, potentially resulting in a less precise representation of the invariance, or require a large number of samples, making it computationally challenging to capture its full complexity. This could result in challenges for single-neuron invariances that involve more complex transformations along the underlying manifold, as simply interpolating between discrete points may not reliably produce new samples on the manifold.

NEW **Learning continuous invariance manifolds** Recently, Baroni et al. (2023) proposed an alternative approach combining contrastive learning with implicit neural representations (INRs) to map a continuous latent input onto the invariance manifold. This method captures the manifold as a continuous entity, addressing the discretization issue from previous methods. NEW Similarly, Wang & Ponce (2022a;b) introduced a framework for modeling neural invariances at different activation levels, providing theoretical insights into their geometry and examining variations in tuning landscapes across the ventral stream. In contrast to Baroni et al. (2023), which learned continuous invariance manifolds using INRs, Wang & Ponce (2022a;b) leverage a pre-trained generative model to explore invariances in its latent space, capturing diverse patterns within the naturalistic image space learned by the generative model. FIX While these methods advance our understanding of invariance manifolds, they do not provide a mechanism to compare invariances across neurons at a population level, a gap we aim to address in this study.

Functional clustering of neurons Reliable comparison of neurons based on encoding properties requires separating shared nonlinear computations from variations in FIX nuisance RF properties, such as location, size, and orientation. For example, two simple cells with the same RF pattern but different locations and orientations should be grouped in the same cluster. While multiple methods have been developed for clustering neurons based on their encoding properties, they typically do not fully isolate the non-linear computations from the FIX nuisance RF properties or use clustering metrics that are often difficult to interpret.

NEW **Model-based clustering** Ustyuzhaninov et al. (2019) employed a model-based approach, clustering neurons based on feature weights in the readout layers of a rotationally-equivariant response-predicting model. This method effectively clusters neurons with similar encoding properties, disregarding differences in orientation and position. FIX However, this approach does not account for RF size variations, a property known to vary with eccentricity in higher mammals' visual systems (Wilson & Sherman, 1976; Cavanaugh et al., 2002; Harvey & Dumoulin, 2011). Additionally, this approach risks incorrectly separating neurons that share similar encoding properties as the same neuronal response function could be realized by different combinations of features. Finally, a metric on feature weights cannot readily be translated into aninterpretable metric of neuronal similarity.

NEW **Stimulus-based clustering** Burg et al. (2024) introduced a technique that clusters neurons by identifying most discriminative stimuli across groups of neurons, but does not account for variations in RF orientation or size. Similarly, Willeke et al. (2023) and Tong et al. (2023) employed contrastive learning to cluster neurons with similar maximally excitatory inputs (MEIs) in macaque V4 and mouse visual cortex, respectively. However, their method relies on dimensionality reduction of single-neuron MEIs, which may fail to cluster neurons exhibiting invariance properties. For example, two images from the invariance manifold of the same neuron could end up in different clusters, leading to inconsistent cluster assignment depending on the image used.

3 Method

Our approach consists of two stages: *Template Learning* and *Template Matching*. The first stage identifies the continuous invariance manifold of a single neuron, and the second stage aligns this learned manifold with another neuron to maximize its activity, evaluating their similarity (Fig. 1).

3.1 Template Learning

Implicit Neural Representation of the invariance manifold FIX In the template learning step, following Baroni et al. (2023), we use Implicit Neural Representations (INRs) to parameterize the continuous invariance manifold of a neuron modeled by a response-predicting DNN, generating diverse images that elicit maximal activity. INRs are neural networks that represent continuous signals, such as images or 3D shapes, by mapping coordinates to signal values without relying on discrete grid-based representations (Genova et al., 2019; Atzmon & Lipman, 2020; Sitzmann et al., 2019; Mildenhall et al., 2021). In the context of image generation, an INR maps 2D pixel coordinates (x, y) to pixel values $g_\phi(x, y)$, FIX producing a continuous, resolution-independent image from the input grid of pixel coordinates.

FIX Similar to Baroni et al. (2023), we introduce a periodic 1D latent variable $z \in [0, 2\pi)$ as an additional input to the INR, enabling it to generate multiple images from the same pixel grid and produce a

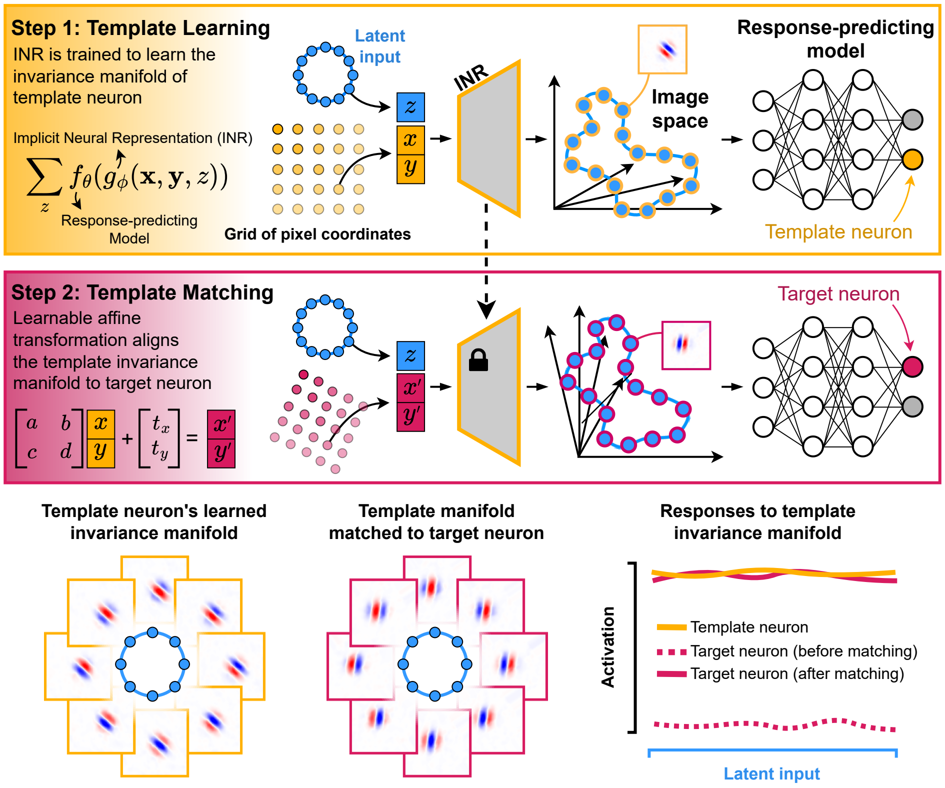
Step 1: Template Learning
INR is trained to learn the invariance manifold of template neuron
Implicit Neural Representation (INR)
$\sum_z f_\theta(g_\phi(\mathbf{x},\mathbf{y},z))$
Response-predicting Model
Latent input
Grid of pixel coordinates
z
x
y
INR
Image space
Response-predicting model
Template neuron
Step 2: Template Matching
Learnable affine transformation aligns the template invariance manifold to target neuron
$\begin{bmatrix} a & b \\ c & d \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} = \begin{bmatrix} x' \\ y' \end{bmatrix}$
z
x'
y'
Target neuron
Template neuron's learned invariance manifold
Template manifold matched to target neuron
Responses to template invariance manifold
Activation
Template neuron
Target neuron (before matching)
Target neuron (after matching)
Latent input

Figure 1: **Overview of the method: Top panel)** Template learning. An Implicit Neural Representation (INR), mapping from pixel coordinates and a latent space, generates a manifold of images that maximally, and equally, excite a neuron (referred to as the template neuron). **Middle panel)** Template matching. The same INR is used to generate the invariance manifold of the template neuron. An affine transformation on the pixel coordinates (x, y) is learned to align the template invariance manifold to a target neuron such that it activates the target neuron as strongly as possible. **Bottom panel)** Left: Template neuron's invariance manifold. Center: Template neuron's invariance manifold after template matching, *i.e.*, aligning to target neuron. Right: Schematic representation of the neurons' responses to different images of the template neuron invariance manifold. The template neuron's invariance manifold maximally activates the template neuron but does not activate the target neuron. However after matching, if the template neuron and target neuron present similar invariance properties, it maximally activates the target neuron.

family of images $\{g_\phi(\mathbf{x}, \mathbf{y}, z)\}_{z\in\mathcal{Z}}$ by varying z. For brevity, we might represent $g_\phi(\mathbf{x}, \mathbf{y}, z)$ as $g(z)$ in the following equations, omitting the dependence on $\mathbf{x}, \mathbf{y}$, and the parameters ϕ, which remain implicit.

While we generally follow the approach of Baroni et al. (2023), we introduced several modifications, including changes to the architecture and the use of positional encoding. Specifically, we implemented the INR as a fully connected neural network with 4 hidden layers, each containing 50 hidden units with a Tanh nonlinearity. Since we aim to generate grayscale images, the output layer consists of a single unit with a Tanh activation, producing values between -1 and 1. Additionally, we used positional encoding with random Fourier features (Tancik et al., 2020), as used by Baroni et al. (2023), on the pixel coordinates (x, y) and extended it to the latent variable z. When applied to the pixel coordinates, this encoding enables control over the spatial frequency of the generated images, whereas its application to the latent variable allows the INR to capture more complex variations along the manifold, even with small changes in z. **FIX**

Objective function The INR parameters ϕ are trained to generate a diverse set of images that **FIX**

collectively span the invariance manifold of a target neuron, ensuring all images elicit maximal activation. The images generated by the INR are fed to a response-predicting DNN $f_\theta(g_\phi(\mathbf{x}, \mathbf{y}, z_i))$ with fixed parameters θ, which evaluates the activation of the target neuron in response to the generated images (Fig. 1 top panel). Before being fed to the DNN, the generated images are linearly scaled to satisfy a specific norm constraint. Combined with the activity maximization objective, this ensures that the pixel variability in the generated images represents only features relevant to the neuron and prevents saturation of pixel values. To ensure diversity of patterns across the generated images, spanning the underlying invariance manifold, we use a contrastive objective (Chen et al., 2020) that encourages variations across the images. Combined with the objective of maximizing the neuron's response, the complete objective function (to maximize) for template learning is:

$$\mathcal{L}_{TL} = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{\alpha_i}{\alpha_{\text{MEI}}} + \lambda \cdot \log\left(\frac{\frac{1}{N_+}\sum_{z_j\in\mathcal{Z}_+^i}\exp\left(\frac{\text{sim}(g(z_i),g(z_j))}{\tau}\right)}{\frac{1}{N_-}\sum_{z_k\in\mathcal{Z}_-^i}\exp\left(\frac{\text{sim}(g(z_i),g(z_k))}{\tau}\right)}\right)\right), \tag{1}$$

where N is the total number of generated images corresponding to the number of latent points z. $\frac{\alpha_i}{\alpha_{\text{MEI}}}$ is the activation $\alpha_i = f_\theta(g_\phi(\mathbf{x}, \mathbf{y}, z_i))$ of the target neuron, normalized by its response to its MEI, obtained through standard pixel optimization. While different latent points were chosen to be equally spaced on a grid, they were jittered at every training step to allow the latent input to span the entire range. The strength of the contrastive objective is controlled by λ. The contrastive term uses cosine similarity $\text{sim}(g(z_i), g(z_j))$ between images generated from latent points z_i and z_j, scaled by temperature τ. Positive examples $\mathcal{Z}_+^i$ come from latent points near z_i, while negative examples $\mathcal{Z}_-^i$ come from distant points, with N_+ and N_- being their respective counts. This formulation encourages similarity between images corresponding to nearby latent points while making images corresponding to distant latent points different from each other, promoting a smooth and diverse learned manifold. Additional details about the template learning step can be found in Appendix B, and an analysis of the method's robustness to varying definitions of "near" and "distant" latent points is provided in Appendix G. **NEW**

3.2 Template Matching

Aligning the learned invariance manifold After the template learning step, where we identify the invariance manifold of the template neuron, the second stage focuses on aligning this template manifold with another neuron (referred to as the *target neuron*). We refer to this step as Template Matching (Fig. 1 middle panel). In this step, the template manifold is fixed by freezing the INR parameters $\phi = \phi^*$. To spatially transform the template, we learn an affine transformation A_ψ applied to the pixel coordinates (x, y) such that images sampled from the template maximally activate the target neuron $\tilde{f}_\theta(g_{\phi^*}(A_\psi(\mathbf{x}, \mathbf{y}), z))$.

Objective function The objective function for optimizing the parameters ψ of the affine transformation matrix is similar to Eq. 1 but without the contrastive learning objective since the INR parameters are not being optimized in this step and the invariance manifold is fixed:

$$\mathcal{L}_{TM} = \frac{1}{N}\sum_{i=1}^{N}\frac{\tilde{\alpha}_i}{\tilde{\alpha}_{\text{MEI}}}, \quad \tilde{\alpha}_i = \tilde{f}_\theta(g_{\phi^*}(A_\psi(\mathbf{x}, \mathbf{y}), z_i)), \tag{2}$$

where N is the total number of images generated from the manifold, $\tilde{\alpha}_i$ is the activation of the target neuron when presented with the transformed image $g_{\phi^*}(A_\psi(\mathbf{x}, \mathbf{y}), z)$, and $\tilde{\alpha}_{\text{MEI}}$ is the response of the target neuron to its MEI. By learning the affine transformation, we align the spatial structure of the template neuron's manifold to that of the target neuron. This alignment allows us to probe the degree of shared invariance between the two neurons, based on how well the transformed manifold activates the target neuron (Fig. 1 bottom panel). The activation metric used for alignment is immediately interpretable because it tells us how much the template manifold can drive another neuron. Additional details about the template matching step can be found in Appendix C.

While we focus on affine transformations here, our formulation permits extensions to non-linear (invertible) distortions of space or subsets of affine transformations, offering the ability to decide how flexible or constrained the invariance alignment should be. Additionally, the continuous nature of INRs ensures that transforming pixel coordinates does not introduce artifacts associated with discrete pixel manipulations.

3.3 Clustering based on invariance alignment

By applying the template learning and template matching steps to all pairs of neurons in the population, FIX we construct an activation matrix, where each row corresponds to one neuron (the template neuron) and each column to another neuron (the target neuron). Each entry (i, j) in this matrix represents the average response of neuron j to the images of the invariance manifold of neuron i after matching. To make sure that activations are comparable across neurons despite differences in their intrinsic excitability, we used relative activations as opposed to absolute responses. The relative response for any given neuron was computed as: $(\alpha - \alpha_0)/(\alpha_{\text{MEI}} - \alpha_0)$ where α is the absolute response, α_0 is the baseline response (*i.e.*, response to a gray image), and α_{MEI} is the MEI response. This yields relative activation values where 0 corresponds to the baseline response and 1 to the MEI response.

The activation matrix reflects the functional interaction between neurons by capturing how one neuron's learned invariance properties affect the activity of another neuron. Because neurons exhibit nonlinear response characteristics, clustering based on the activation matrix provides a better understanding of neurons' functional similarity than clustering directly using invariance manifolds or MEIs, mainly because distances in image space do not necessarily have a one-to-one correspondence with functional differences between neurons.

Additionally, notice that the matrix is not symmetric. Specifically, if $\mathcal{M}_1$ and $\mathcal{M}_2$ are the invariance manifolds of the template and target neuron, respectively, a high activation means that $\mathcal{M}_1 \subseteq \mathcal{M}_2$, but not necessarily vice versa. For example, if we consider a simple and a complex cell, after the FIX alignment of their invariance manifold, the Gabor pattern of a simple cell would maximally activate a complex cell, while the invariance manifold of a complex cell would not maximally activate a simple cell. Therefore, this asymmetry reflects a meaningful difference between neurons' computational roles and the relationship between their invariance manifolds.

Once the activation matrix is constructed, we cluster neurons by applying a standard agglomerative hierarchical clustering on the rows of the activation matrix. Each row corresponds to a single neuron and represents how strongly its invariance manifold activates other neurons after matching. We use FIX cosine dissimilarity as the distance metric for clustering, which helps focus on the relative activation patterns between neurons rather than the magnitude of the activation, ensuring that functional relationships are properly captured.

4 Experiments and Results

4.1 Validation on simulated neurons

As the first step, we validated our approach using a set of Gabor-based simulated neurons to both verify and quantify its ability to accurately capture and align neural invariances across different neuron types. The main advantage of this simulated environment is that we have access to the ground truth invariances for each neuron type, enabling precise benchmarking of the accuracy of the method.

The simulated neurons consisted of several types, including simple and complex cells, as well as FIX additional neurons exhibiting non-trivial, arbitrary invariance properties (Fig.2A, top row). Simple cells were modeled as linear-nonlinear systems using a single Gabor filter and complex cells were modeled using a set of Gabor filters with shared parameters but varying phases, achieving phase invariance by selecting the filter with the highest response. Neurons with more complex invariances were similarly modeled using combinations of Gabor filters with varying arbitrary parameters, selecting the highest response to achieve the desired invariance. For each neuron type, we created six instances, resulting in a total of 36 neurons. Neurons of the same type differed from one another by an affine transformation applied to their receptive fields, varying parameters such as position, orientation, and scale (Fig.2A, bottom row). To standardize the response of the neurons and facilitate the comparison of results, We normalized the Gabor filters to have a maximum activation of 1 in FIX response to stimuli of unitary norm. Further details on the simulation can be found in Appendix D.

We applied the template learning and template matching steps on all pairs of simulated neurons resulting in an activation matrix where row i shows how strongly the invariance manifold of neuron i activated all other neurons (Fig. 2C). While the neurons' order is randomized in this matrix, the high activation in the diagonal entries shows that our method successfully recovered the known invariance transformations for each neuron, generating invariance manifolds that activated the neurons close

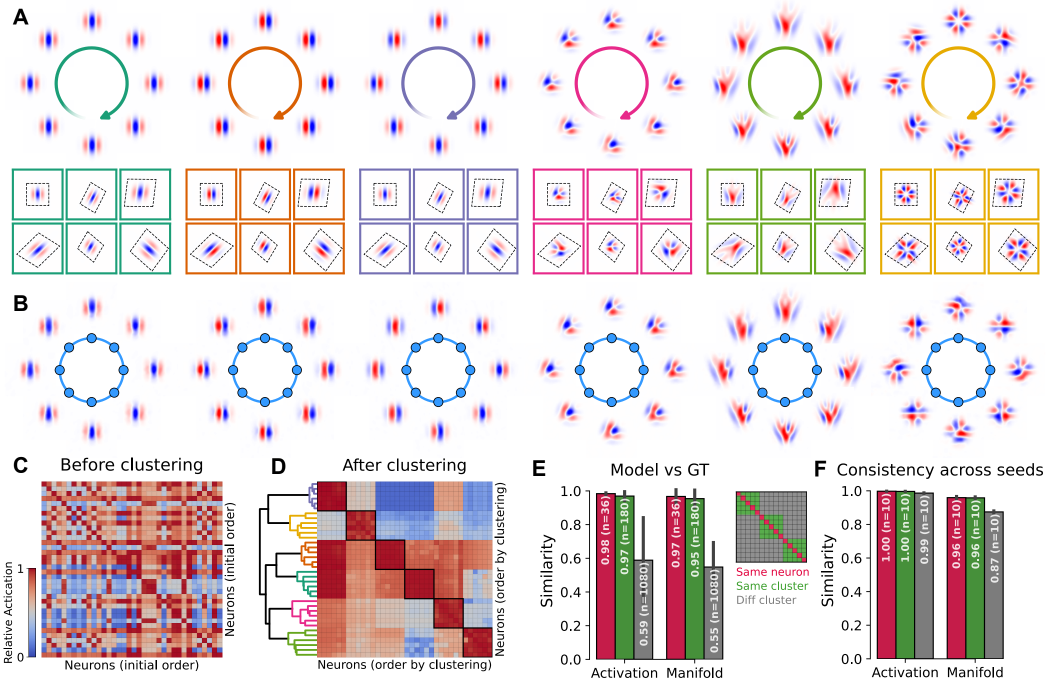

Figure 2: **Results on Simulated Neurons:** **A)** Types of Gabor-based simulated neurons, each associated with a distinct color. Top row: invariance manifold of a representative neuron for each type. From left to right: even simple cells, odd simple cells, complex cells, and three types of neurons with selectivity and invariance to more complex patterns. Note that images for simple cells are identical, indicating their lack of invariance. Bottom row: For each neuron type we model 6 neurons that differ by an arbitrary affine transformation illustrated by dotted lines around each neuron's receptive field. **B)** Learned invariance manifolds for representative neurons shown in the top row of Panel A. **C)** Activation matrix resulting from the pairwise alignment of neural invariances. Each row represents how the invariance manifold of one neuron activates others post-alignment, while each column shows how a specific neuron is activated by the manifolds of other neurons. **D)** Activation matrix ordered by applying a hierarchical clustering algorithm. The block diagonal pattern corresponds to identified clusters reflecting the division into different neuron types. Dendrogram colors reflect neuron types as colored in Panel A. **E)** Comparison of predictions versus ground truth. Left barplot: activations from learned manifolds (red), aligned manifolds of neurons within the same cluster (green), and aligned manifolds of neurons from different clusters (gray). A similarity of 1 means the invariance manifold activates the neuron as high as its MEI. Right barplot: similarity scores between the ground truth and the learned manifolds (red), aligned manifold of neurons within the same cluster (green), and aligned manifold of neurons from different clusters (gray). Details of the similarity score can be found in Appendix F. **F)** Consistency of results across 5 seeds, *i.e.*, $n = 10$ unique pairwise combinations. Left barplot: similarity scores of activation matrices across seeds. Similarity of diagonal entries is shown in red, off-diagonal same cluster is shown in green, and off-diagonal different cluster is shown in gray. Right barplot: similarity scores of learned manifolds (red), aligned manifolds of neurons within the same cluster (green) and from different clusters (gray), across seeds.

to their maximal activation (see Fig. 2B for examples of the learned invariance manifold). We then applied hierarchical clustering, using cosine dissimilarity as a distance metric, to the rows of the activation matrix, accurately recovering the predefined neuron types (Fig. 2D) and showing that our method successfully aligned the invariance manifolds of simulated neurons with similar type, despite differences in their RF properties. To further demonstrate the flexibility of our approach in accounting for affine transformations, we compared it with two model-based clustering methods previously introduced by Klindt et al. (2017) and Ustyuzhaninov et al. (2019). While these methods are effective for accounting for shifts (Klindt et al., 2017) and rotations (Ustyuzhaninov et al., 2019),

FIX

NEW

they do not generalize to the broader class of affine transformations. Details of this comparison are provided in Appendix H.

We further quantified deviations from the ground truth by conducting analyses on both the learned and aligned manifolds as well as the corresponding activations. For a given neuron, we defined the activation similarity between the ground truth act_{GT} and the learned manifold activation act as $1 - |act_{GT} - act|$. Similarly, we computed manifold similarity by measuring how well the images on one manifold match the closest images on the other manifold, averaged bidirectionally. This captures the degree to which the two manifolds overlap in terms of their most similar images (for more details refer to Appendix F). Our method effectively recovered activation values near the maximum not only in response to the learned invariance manifold of the given neuron, but also in responses to invariance manifolds of other neurons within the same type, after being matched (Fig. 2E left, red and green bar). Notably, when this alignment was performed between neurons from different types, the resulting activation was consistently lower, as expected (Fig. 2E left, gray bar). When comparing the learned and aligned manifolds with ground truth manifolds, we observed high similarity scores for manifolds aligned within the same cluster indicating that invariances were successfully aligned (Fig. 2E right, red and green bar). Alignments between different clusters yielded lower similarity scores, highlighting the differences in invariance properties across distinct neuron types (Fig. 2E right, gray bar).

NEW

NEW

FIX

FIX

Finally, we assessed the consistency of our method across different initializations (seeds), at both the activation and manifold levels. For comparing the activations across seeds, we defined an activation similarity metric as $1 - |act_i - act_j|$, which measures the closeness of activations between seed i and j, averaged across entries in the activation matrix. Similarly, we computed a similarity score of the corresponding learned and aligned manifolds across seeds to evaluate consistency in manifold learning and alignment, directly using images sampled from the manifolds. As depicted in Fig. 2F, our analyses show that our results on the synthetic data are robust, with high consistency observed both at the level of activation and in the alignment of invariance manifolds.

4.2 Application on macaque V1 neurons

We applied our method to a response-predicting model of a population of biological neurons recorded from the primary visual cortex of two macaques, using the publicly available dataset from Cadena et al. (2024) and the response-predicting model from Baroni et al. (2023). Briefly, the dataset contains neural responses from 458 V1 neurons recorded across 32 sessions as the macaques viewed up to 15,000 unique grayscale ImageNet images per session, while fixating. Images were shown for 120ms, and neuronal responses were measured by counting spikes from 40 to 160ms post-image onset. The model from Baroni et al. (2023) is an ensemble of three convolutional neural networks which we trained to predict these neural responses, approaching state-of-the-art prediction performance with a correlation of 0.73 between predicted and observed trial-averaged responses. We selected the 100 best-predicted neurons to ensure accurate modeling of their encoding properties (Fig. S3).

FIX

Applying the template learning step, our method learned invariance manifolds (Fig. 3A) that near maximally activate the neurons (Fig. 3B, diagonal entries). We then performed the template matching step to align the invariance manifolds across neurons, resulting in a 100-by-100 activation matrix. Similar to the procedure used for the simulated data, we applied a hierarchical clustering algorithm on the activation matrix to group neurons based on how similar their invariance manifold activates other neurons after being matched to them (Fig. 3B).

To identify meaningful clusters, we selected a dendrogram cut-off threshold that minimized the number of clusters while ensuring that neurons within each cluster shared similar invariance properties, as verified through visual inspection of their invariance manifolds. This resulted in five clusters containing multiple neurons (Fig. 3A). We identified two large clusters corresponding to simple and complex cells (Fig. 3 A-B, Clusters 1 and 2). Neurons in Cluster 1 displayed selectivity to Gabor patterns with no invariances, whereas neurons in Cluster 2 exhibited phase invariance. The remaining smaller clusters did not show uniform orientation tuning across their receptive fields, but instead exhibited selectivity to more complex patterns, diverging from the traditional classification of V1 neurons into simple and complex cells. For example, Cluster 5 contained neurons selective to checkerboard-like texture patterns and showed invariance to local alterations of these patterns.

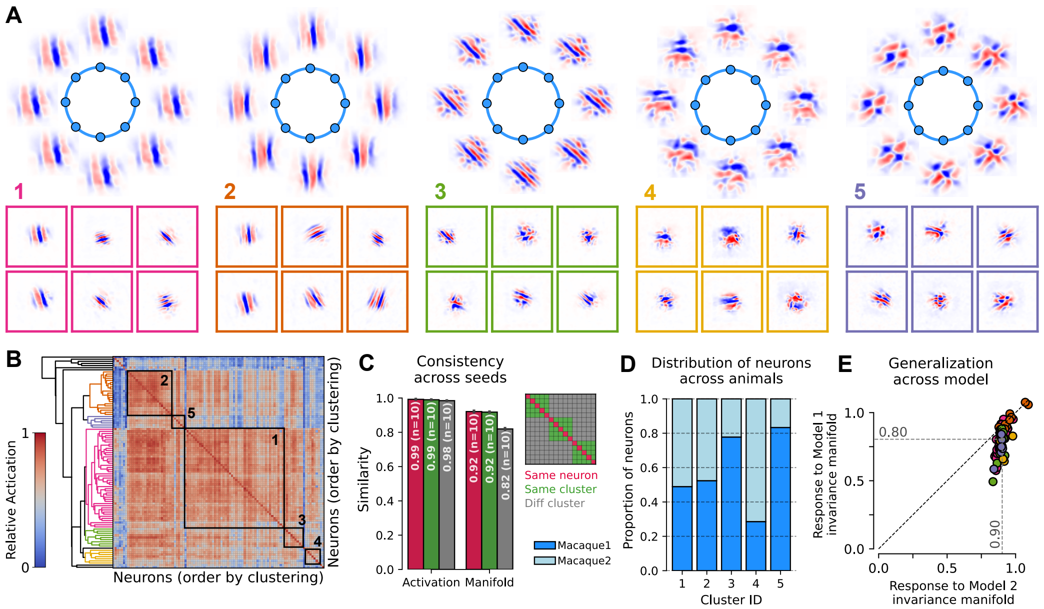

Figure 3: **Results on Macaque V1 Neurons: A)** Examples of learned invariance manifolds, organized by clusters identified in panel B. Each cluster is assigned a numerical ID and color. The top row shows invariance transformations (or their absence) for representative neurons from different clusters. The bottom row displays an example image within the invariance manifold for multiple neurons for each cluster. **B)** Activation matrix ordered by hierarchical clustering applied to rows, revealing clusters of neurons with shared invariance properties. **C)** Consistency of results across 5 seeds. Left barplot: similarity scores of activation matrices across seeds. Similarity of diagonal entries is shown in red, off-diagonal same cluster shown in green, and off-diagonal different cluster shown in gray. Right barplot: similarity scores of learned manifolds (red), aligned manifolds of neurons within same cluster (green) and from different clusters (gray), across seeds. **D)** Distribution of neurons within each cluster across different animals. **E)** Generalization across models. Invariance manifolds from model 1 (which was used for the previous analyses) were shown to model 2 (a model with different architecture predicting the same neurons). The y-axis shows the response of model 2 to model 1's invariance manifolds, and the x-axis shows the response of model 2 to its own invariance manifolds.

All identified clusters contained neurons recorded from both macaques (Fig. 3D), confirming that our analysis did not detect inter-subject differences, reflecting consistent sensory coding across the two animals. As we did with the simulated neurons, we tested the robustness of our method applied to the macaque V1 neurons across different instances. Our method produced consistent results in terms of the learned and aligned manifolds (Fig. 3C) and cluster consistency (Adjusted Rand Index = 0.75).

Finally, to rule out the possibility that our results were due to idiosyncrasies of the response-predicting model, we trained a second model with a different architecture (details in Appendix E) on the same set of neurons (Fig. S3B). We refer to the initial model as Model 1 and the newly introduced model as Model 2. We presented the invariance manifolds of Model 1 to the corresponding neurons in Model 2, and then learned invariance manifolds from Model 2 to compare the activations resulting from these two procedures. On average, the invariance manifold from Model 1 activated the corresponding neuron in Model 2 at 89% of the strength of its own manifold, confirming that the invariance manifolds from Model 1 strongly activate the same neurons in Model 2 and generalize well (Fig. 3E).

Together, these results demonstrate that our method can robustly identify common invariance properties of visual coding in biological neurons.

5 Discussion

We introduced a novel method to identify and align the continuous invariance manifolds of visual sensory neurons, enabling systematic characterization of visual coding across neural populations. By combining DNNs as models of biological neurons, an INR to learn their continuous invariance manifolds, and aligning neuronal manifolds through affine transformations, we provided a principled approach to quantify and compare invariances irrespective of receptive field differences. By analyzing the activation matrix resulting from invariance alignment, we gain insights into the functional differences across neurons potentially grouping them into distinct functional clusters based on their invariances.

When applied to Gabor-based simulated neurons, our method learned the true underlying invariance manifolds and successfully aligned them across neurons, identifying clusters corresponding to ground truth neuron types. Applying our method to 100 well-predicted macaque V1 neurons, we identified five distinct clusters two of which corresponded to traditional simple and complex cells, along with smaller clusters that exhibit invariances to more intricate patterns. For instance, one class of neurons was responsive to checkerboard-like texture patterns and demonstrated invariance to variations in the phase of these patterns in multiple directions, suggesting a complexity beyond the typical simple/complex dichotomy. We currently cannot fully exclude that these clusters result from an artifact of the recordings or the data pre-processing steps, such as spike sorting. Ultimately dedicated closed-loop neurophysiological experiments would need to be conducted to empirically validate the clusters.

However, there is some evidence that these clusters could reflect biological reality. First, all selected neurons had a high prediction performance on a set of images that the response-predicting model was not trained on (*i.e.*, a test set), hinting that the model has accurately captured the encoding properties of these neurons. Second, all clusters contained cells from both monkeys, indicating that the RF patterns and invariances generalize across animals. Third, the invariance manifolds learned by one model highly activated the same neurons of another model with a different architecture, indicating that the clusters, which are based on the learned invariances, are likely not an artifact of the network used to fit the cells, but reflect a property of the underlying data. Thus, our method could have potentially found novel types of invariances in macaque V1 neurons, similar to more complex invariances found in mouse visual cortex (Ding et al., 2023).

While our approach has merits, it is not without limitations. First, the method's computational complexity scales quadratically with the number of neurons, making it relatively expensive to apply on larger neural populations. Second, aligning complex invariance manifolds spatially is not necessarily a convex problem, which could complicate the alignment process for neurons with more intricate invariances. Finally, in this work we focused on one-dimensional invariance manifolds. While this appears to be sufficient for macaque V1 data, other cortical areas might require higher dimensional invariance manifolds. Nonetheless, Baroni et al. (2023) showed that INRs can be used to learn higher dimensional invariances.

Looking ahead, several promising directions can be explored to extend this work. Enhancing the scalability of the approach through more efficient algorithms could make it applicable to larger neural populations. Additionally, the flexibility of our framework allows exploring transformations beyond affine, enabling alignment of invariances up to more complex transformations. Moreover, our method can be easily applied to other species, potentially identifying novel invariance properties and enabling cross-species comparisons to uncover conserved or divergent functional principles across different neural systems.

In summary, the method we presented is a novel tool for exploring neural invariances in biological data at the population level, enabling the identification of neuron clusters with shared invariance properties, with the potential to uncover new principles of neural computations in visual cortex. These principles could also inform the design of artificial neural networks, helping to bridge the discrepancies between biological and artificial systems highlighted by (Feather et al., 2019; 2023). **NEW**

Reproducibility Statement

All code required to reproduce the experiments presented in this paper, including a Docker container for easy setup, will be made publicly available in a repository upon acceptance. All relevant training configurations, hyperparameters, and implementation details are provided in the appendix for full transparency. The experiments were conducted using Python 3.9, PyTorch 1.9.0, and CUDA 11.1. Running the complete pipeline on simulated neurons took approximately 2 hours on a single V100 GPU (one seed, total number of 36 simulated neurons). For the macaque V1 neurons (total of 100 neurons), each neuron's template learning and matching experiment was run separately and took approximately 15-20 minutes on a single A100 GPU (one seed).

REFERENCES

Matan Atzmon and Yaron Lipman. Sal: Sign agnostic learning of shapes from raw data. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2565–2574, 2020.

Luca Baroni, Mohammad Bashiri, Konstantin F Willeke, Ján Antolík, and Fabian H Sinz. Learning invariance manifolds of visual sensory neurons. In *NeurIPS Workshop on Symmetry and Geometry in Neural Representations*, pp. 301–326. PMLR, 2023.

Pouya Bashivan, Kohitij Kar, and James J DiCarlo. Neural population control via deep image synthesis. *Science*, 364(6439):eaav9436, 2019.

Max F Burg, Thomas Zenkel, Michaela Vystrčilová, Jonathan Oesterle, Larissa Höfling, Konstantin F Willeke, Jan Lause, Sarah Müller, Paul G Fahey, Zhiwei Ding, et al. Most discriminative stimuli for functional cell type clustering. *ArXiv*, 2024.

Santiago A Cadena, Marissa A Weis, Leon A Gatys, Matthias Bethge, and Alexander S Ecker. Diverse feature visualizations reveal invariances in early layers of deep neural networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 217–232, 2018.

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.

Santiago A Cadena, Konstantin F Willeke, Kelli Restivo, George Denfield, Fabian H Sinz, Matthias Bethge, Andreas S Tolias, and Alexander S Ecker. Diverse task-driven modeling of macaque v4 reveals functional specialization towards semantic tasks. *PLOS Computational Biology*, 20(5): e1012056, 2024.

James R Cavanaugh, Wyeth Bair, and J Anthony Movshon. Nature and interaction of signals from the receptive field center and surround in macaque v1 neurons. *Journal of neurophysiology*, 88(5): 2530–2546, 2002.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PMLR, 2020.

P Kingma Diederik. Adam: A method for stochastic optimization. *(No Title)*, 2014.

Zhiwei Ding, Dat T Tran, Kayla Ponder, Erick Cobos, Zhuokun Ding, Paul G Fahey, Eric Wang, Taliah Muhammad, Jiakun Fu, Santiago A Cadena, et al. Bipartite invariance in mouse primary visual cortex. *bioRxiv*, 2023.

Alexander S Ecker, Fabian H Sinz, Emmanouil Froudarakis, Paul G Fahey, Santiago A Cadena, Edgar Y Walker, Erick Cobos, Jacob Reimer, Andreas S Tolias, and Matthias Bethge. A rotation-equivariant convolutional neural network model of primary visual cortex. *arXiv preprint arXiv:1809.10504*, 2018.

Jenelle Feather, Alex Durango, Ray Gonzalez, and Josh McDermott. Metamers of neural networks reveal divergence from human perceptual systems. *Advances in Neural Information Processing Systems*, 32, 2019.

Jenelle Feather, Guillaume Leclerc, Aleksander Mądry, and Josh H McDermott. Model metamers reveal divergent invariances between biological and artificial neural networks. *Nature Neuroscience*, 26(11):2017–2034, 2023.

Kyle Genova, Forrester Cole, Daniel Vlasic, Aaron Sarna, William T Freeman, and Thomas Funkhouser. Learning shape templates with structured implicit functions. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 7154–7164, 2019.

Ben M Harvey and Serge O Dumoulin. The relationship between cortical magnification factor and population receptive field size in human visual cortex: constancies in cortical architecture. *Journal of Neuroscience*, 31(38):13604–13612, 2011.

David H Hubel and Torsten N Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of physiology*, 160(1):106, 1962.

David Klindt, Alexander S Ecker, Thomas Euler, and Matthias Bethge. Neural system identification for large populations separating "what"and "where". In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

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, et al. Generalization in data-driven models of primary visual cortex. *BioRxiv*, pp. 2020–10, 2020.

Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021.

Carlos R Ponce, Will Xiao, Peter F Schade, Till S Hartmann, Gabriel Kreiman, and Margaret S Livingstone. Evolving images for visual neurons using a deep generative network reveals coding principles and neuronal preferences. *Cell*, 177(4):999–1009, 2019.

Maximilian Riesenhuber and Tomaso Poggio. Hierarchical models of object recognition in cortex. *Nature neuroscience*, 2(11):1019–1025, 1999.

Vincent Sitzmann, Michael Zollhöfer, and Gordon Wetzstein. Scene representation networks: Continuous 3d-structure-aware neural scene representations. *Advances in Neural Information Processing Systems*, 32, 2019.

Matthew Tancik, Pratul P. Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan T. Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. *NeurIPS*, 2020.

Rudi Tong, Ronan da Silva, Dongyan Lin, Arna Ghosh, James Wilsenach, Erica Cianfarano, Pouya Bashivan, Blake Richards, and Stuart Trenholm. The feature landscape of visual cortex. *bioRxiv*, pp. 2023–11, 2023.

Ivan Ustyuzhaninov, Santiago A Cadena, Emmanouil Froudarakis, Paul G Fahey, Edgar Y Walker, Erick Cobos, Jacob Reimer, Fabian H Sinz, Andreas S Tolias, Matthias Bethge, et al. Rotation-invariant clustering of neuronal responses in primary visual cortex. In *International Conference on Learning Representations*, 2019.

Edgar Y Walker, Fabian H Sinz, Erick Cobos, Taliah Muhammad, Emmanouil Froudarakis, Paul G Fahey, Alexander S Ecker, Jacob Reimer, Xaq Pitkow, and Andreas S Tolias. Inception loops discover what excites neurons most using deep predictive models. *Nature neuroscience*, 22(12): 2060–2065, 2019.

Binxu Wang and Carlos R Ponce. On the level sets and invariance of neural tuning landscapes. *arXiv preprint arXiv:2212.13285*, 2022a.

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

Konstantin F Willeke, Kelli Restivo, Katrin Franke, Arne F Nix, Santiago A Cadena, Tori Shinn, Cate Nealley, Gabrielle Rodriguez, Saumil Patel, Alexander S Ecker, et al. Deep learning-driven characterization of single cell tuning in primate visual area v4 unveils topological organization. *bioRxiv*, pp. 2023–05, 2023.

JR Wilson and S Murray Sherman. Receptive-field characteristics of neurons in cat striate cortex: changes with visual field eccentricity. *Journal of neurophysiology*, 39(3):512–533, 1976.

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

A MEI GENERATION

For a given neuron, We obtained the Maximum Exciting Image (MEI) by directly optimizing image pixel values to maximize its response over 1000 steps with a step size of 10. Typically, the optimization converged well before reaching the maximum steps.

Generating single MEIs prior to learning invariance manifolds is critical for two reasons. First, it provides the maximal neuron response α_{MEI}, which is used to normalize the balance between the activation term and the contrastive term in the loss function, ensuring consistent importance across different neurons. Second, from an MEI, it is possible to compute an MEI mask that isolates the receptive field (RF) of biological neurons, as described in (Ding et al., 2023).

MEI masks were computed following Ding et al. (2023) by identifying the convex region where pixel values exceed 0.5 standard deviations. During template learning, using these masks ensures that artifacts do not appear outside the neuron's RF (refer to 1).

B TEMPLATE LEARNING DETAILS

The initialization weights for the fully connected layer in the Implicit Neural Representation (INR) were sampled from a Gaussian distribution with $\sigma = 0.1$. For positional encoding of pixel coordinates, we used 50 dimensions with a projection scale of 10. For the positional encoding of latent input, we used 50 dimensions with a projection scale of 0.1.

During template learning, to train the INR, we employed a jittering grid technique. A grid of $N = 20$ uniformly distributed points $z \in \mathcal{Z}$ along a 1D periodic latent dimension was passed to the INR at each step, generating images $\{g(z)\}_{z \in \mathcal{Z}}$. This grid was jittered at every step of the training procedure to allow the latent input to span the entire range. To define $\mathcal{Z}_+^i$ and $\mathcal{Z}_-^i$ for each point z_i, corresponding to near (positive) and distant (negative) points in the latent space, we chose a neighboring radius that spanned 10% of the latent space in each direction, resulting in two near (positive) grid points in each direction for every point z_i. All other points were designated as distant (negative). Contrastive regularization was applied solely within the points of the mask obtained from the MEI. The temperature parameter was set to $\tau = 0.3$.

The parameters of the INR were optimized using an Adam optimizer (Diederik, 2014) with a learning rate of 0.001, focusing on maximizing the activity and contrastive learning objectives as defined in 1. Training continued for a minimum of 500 steps and concluded once specific activity requirements were met by the images generated from the jittered grid sampled in the latent space. For experiments involving simulated neurons, the average required activity was set at $\alpha = 0.99\alpha_{MEI}$ and the minimum at $\alpha = 0.98\alpha_{MEI}$. For experiments involving macaque neurons, the average required activity was $\alpha = 0.9\alpha_{MEI}$, and the minimum was $\alpha = 0.85\alpha_{MEI}$, accommodating the expectation that the invariance manifold of biological neurons might not exhibit perfect iso-response properties. The strength of the contrastive regularization term, initially set to $\lambda = 2$, was reduced by a factor of 0.8 via a scheduler whenever the activity term plateaued. Activity was monitored every 50 grids, with a patience parameter of 5. This adaptive schedule, coupled with the condition to satisfy activity requirements before halting training, enabled a careful balance between activity maximization and contrastive regularization. As a result, the INR can successfully capture neural invariances across neurons exhibiting varying degrees of invariance.

C TEMPLATE MATCHING DETAILS

For each pair of template and target neurons, we performed template matching to maximize the target neuron's response to the template invariance manifold $\tilde{f}_\theta(g_{\phi^*}(A_\psi(\mathbf{x}, \mathbf{y}), z))$ through an affine transformation $A_\psi(\mathbf{x}, \mathbf{y})$.

To minimize the risk of converging on local minima, we initialized the affine transformation by centering the template's invariance manifold on the receptive field of the target neuron, as determined by the centroids of their MEI masks. We then uniformly rescaled the coordinates based on the size ratio between the MEI masks of the target and template neurons. Finally, we rotated the coordinates to the angle that yielded the maximal neural response.

We optimized the parameters of the affine transformation using the Adam optimizer with a learning rate of 0.001. Training was halted when the average activity of the target neurons ceased to increase, with a patience parameter set to 15.

To accommodate GPU memory constraints during template matching for a large number of neurons, we batched the procedures into groups of 20 or 50 neurons, depending on the available computational resources.

D Gabor-based simulated neuron implementation

We implemented simulated neuron models presenting various invariances using Gabor filters. We obtained simple cells as linear-nonlinear systems using a normalized Gabor filter and a final nonlinearity $(1 + \mathbf{ELU})/2$ which ensures non-negative responses and a unitary maximal response for images with unit norm. To model complex cells, we used a set of normalized Gabor filters with shared parameters but different phases. To achieve phase invariance, the model selects the filter that produces the highest output value, which is then passed through the same final nonlinearity. Similarly, for neurons exhibiting more complex invariances, we obtained sets of filters by combining multiple Gabor filters and normalizing them. To achieve the desired invariance, analogous to the process used for complex cells, the model selects the filter that produces the highest output value, which is then passed through the same final nonlinearity.

E Macaque response predictive models details

In our experiments with macaque V1 neurons, we employed a model published by Baroni et al. (2023) to predict neural responses to 93x93 pixel grayscale images, representing 2.65 degrees of the visual field. This model comprises an ensemble of three convolutional neural networks (CNNs), each trained separately to minimize the Poisson loss between predicted and recorded neural responses on a distinct test dataset. Predictions from the ensemble model are obtained by averaging the outputs of individual CNNs.

Each CNN consists of a convolutional core and a pyramidal readout (Sitzmann et al., 2019). The convolutional core features three layers of depth-separable convolutions (kernel sizes [24, 9,9]), each with 32 feature channels, batch normalization, ELU nonlinearity, and squeeze-and-excitation blocks.

To support the generalization of our method across different models, we trained a second ensemble model. This ensemble includes three separately trained CNNs, each featuring a convolutional core and Gaussian readout (Lurz et al., 2020). The cores of these networks consist of three layers of depth-separable convolutions (kernel sizes [24, 9,9]), 32 feature channels per layer, batch normalization, and ELU nonlinearity.

Following the training approach of Baroni et al. (2023), we z-score normalized the stimuli before training each model in the ensemble using the Adam optimizer (Diederik, 2014) with an initial learning rate of 0.004, a batch size of 128, and a learning rate reduction by a factor of 0.3 up to three times when there was no improvement in validation loss for three consecutive epochs. This secondary ensemble architecture achieved a correlation coefficient of 0.71 with trial-averaged responses.

F Similarity score between manifolds

We define the similarity score between two manifolds as

$$\text{score}(g_1, g_2) = \frac{1}{2}\left(\frac{1}{n}\sum_{i=1}^{n}\max_j\left(\text{sim}(g_1(z_i), g_2(z_j))\right) + \frac{1}{m}\sum_{j=1}^{m}\max_i\left(\text{sim}(g_1(z_i), g_2(z_j))\right)\right)$$

where $\mathbf{g}_1(z_i)$ and $\mathbf{g}_2(z_j)$ represent the images on the two manifolds, parametrized by the mappings of two INRs, $\mathbf{g}_1$ and $\mathbf{g}_2$. This measure computes the highest cosine similarity between each image on one manifold and among in the other manifold and vice-versa, then averages these quantities within manifolds and across manifolds to get a measure of similarity between two manifolds.

G Robustness to “near” and “distant” latent point configuration

NEW

The contrastive term in the template learning objective (Eq. 1) relies on specifying “near” (positive) and “distant” (negative) examples for a given latent point z_i. Positive examples $\mathcal{Z}_+^i$ are images corresponding to latent points near z_i, while negative examples $\mathcal{Z}_-^i$ correspond to distant points. This encourages similarity among nearby images and diversity among distant ones, promoting a smooth and diverse learned manifold. Here we assess the effect of “near” and “distant” latent point configuration on the smoothness of the learned manifolds, and their resulting activation. Specifically, we used a complex cell (Fig. S1A) and applied our method with five different settings of “near” vs “distant” range (Fig. S1C): 5%, 10%, 20%, 40%, and 45%. Each value represents the radius of the neighboring area as a percentage of the whole latent space. Here we sampled 20 points from a 1D latent space, meaning 5% represents 2 “near” points (1 on the left and 1 on the right) which is the minimum number of “near” points possible. Similarly, 45% represents 18 points (9 on the left and 9 on the right), which is the maximum number of “near” points possible. Our method successfully

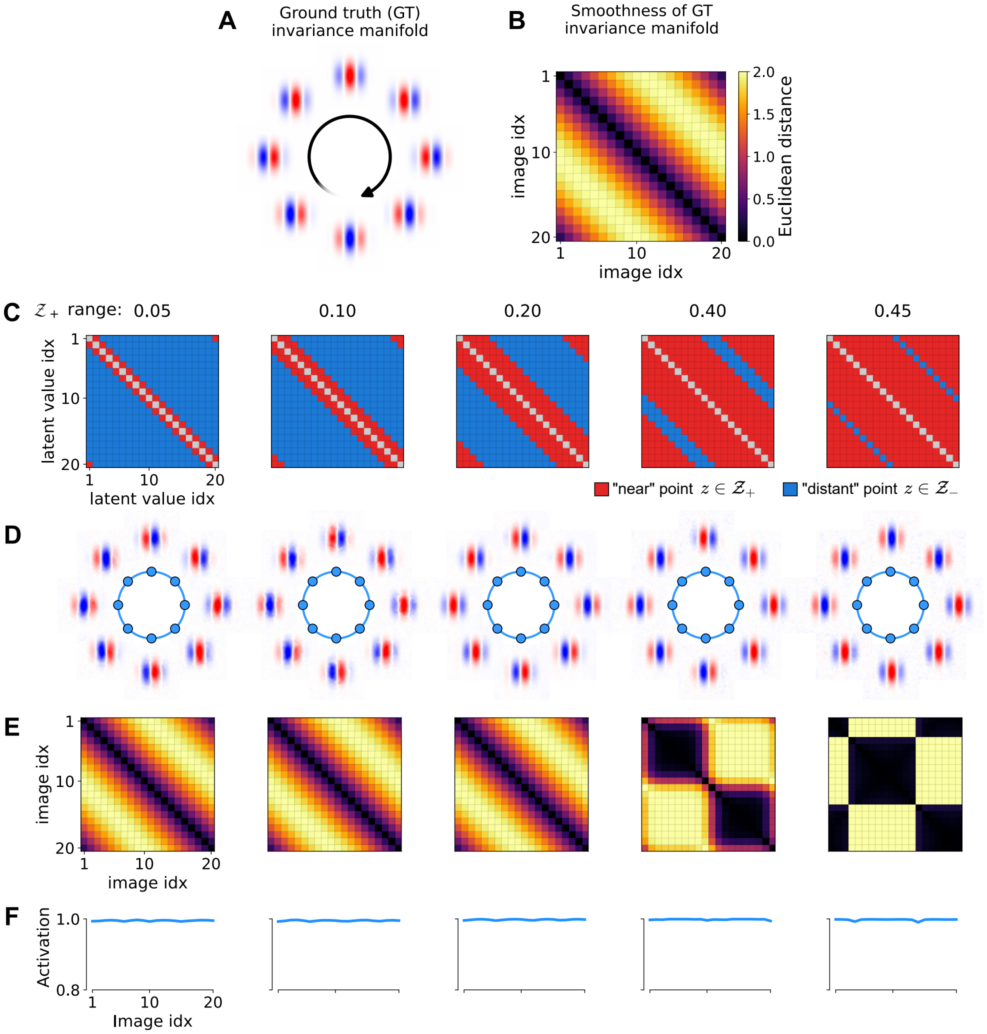

Figure S1: **A)** Ground truth invariance manifold of a complex cell. **B)** Euclidean distance matrix showing pairwise distances between images sampled from the GT manifold at equidistant phases. **C)** Neighboring mask illustrating “near” (red) and “distant” (blue) points based on varying definitions of proximity ($\mathcal{Z}_+$ range: 0.05–0.45). Each row corresponds to a different $\mathcal{Z}_+$ threshold. **D)** Learned invariance manifolds across different proximity definitions. **E)** Euclidean distance matrices between image pairs sampled from the learned manifolds at equidistant points in the latent space. **F)** Neural activation curves elicited by images sampled from the learned manifolds.

learns a smooth parametrization of the underlying manifold for most of the "near" percentages considered (Fig. S1D,E). Only in the extreme cases where "near" points represent the majority, smoothness is degraded (Fig. S1D,E last two columns). This is expected since the objective will encourage similarity much more strongly compared to diversity. Overall, these results suggest that our method presents robustness to the exact setting of "near" vs "distant" points.

H Comparison with model-based clustering algorithms

In this section, we compare our method to the model-based clustering approaches proposed by Klindt et al. (2017) and Ustyuzhaninov et al. (2019). Briefly, both methods rely on DNN-based models composed of two main parts: a core and a readout. The core is implemented as a neural network with 2D convolution layers, extracting nonlinear features across spatial locations. These features are shared across all neurons but are linearly combined by the readout at specific spatial locations to predict each neuron's response. While the core captures shared computations across the neuronal population, the readout's neuron-specific weights, which serve as proxies for neuron-specific computations, can be used to cluster neurons. **NEW**

Both methods can cluster neurons independently of differences in RF position, as the convolution operation is translation-equivariant. However, the model proposed by Ustyuzhaninov et al. (2019) extends this by incorporating rotation-equivariant convolution layers. This allows the core to learn nonlinear functions not only at different spatial locations but also across different orientations, enabling clustering invariant to both translation and rotation using the feature weights of the readout. While these approaches have proven effective for identifying functional clusters invariant to shifts (Klindt et al., 2017) and rotations (Ustyuzhaninov et al., 2019), our method generalizes these invariances to include all affine transformations, and can straightforwardly be generalized to more powerful transformations as well. Here, we provide an empirical comparison of the clustering results obtained with these different methods.

Simulated neurons To keep this experiment focused on assessing how well our approach accounts for transformations beyond translation (Klindt et al., 2017) and rotation (Ustyuzhaninov et al., 2019), we considered two types of neurons: odd and even simple Gabor cells. For each type, we considered two different RF sizes, two different orientations, and five randomly sampled locations, giving rise to 20 simulated neurons per cell type, and 40 in total (Fig. S2A). To keep the demonstration and its results clear, we focus on our method's ability to cluster neurons in a way that is invariant against transformations of pixel coordinates and did not demonstrate another ability of our model: to cluster by invariances.

Implementation of model-based methods We implemented the methods proposed by Klindt et al. (2017) and Ustyuzhaninov et al. (2019) using convolutional and rotation-equivariant convolutional models (using steerable filters as in Ecker et al. (2018)), respectively. Given the linear-nonlinear nature of the cells considered (implementation details in AppendixD), we used a single-layer core followed by a Gaussian readout as in Lurz et al. (2020) and a $(1 + \mathbf{ELU})/2$ final nonlinearity. The Gaussian readout is a special case of the factorized readout which was used by Klindt et al. (2017). Specifically, while the factorized readout learns a spatial mask for each neuron to learn relevant spatial locations (i.e., neuron's RF) for predicting neural responses, the Gaussian readout learns a single location in the spatial dimensions for each neuron and linearly combines features across channels at that specific location. Thus the Gaussian readout is a factorized readout where the spatial mask is reduced to a single pixel on the output tensor of the core. This results in a significant reduction in the number of readout parameters and has been shown to be more data-efficient (Lurz et al., 2020).

To minimize redundancies in features learned by the core, we enforced readout feature weights to be positive and adjusted the model capacity by fixing the number of channels to eight for the convolutional model (corresponding to cell type $\times$ number of sizes $\times$ number of orientations) and four for the rotation-equivariant convolutional model (corresponding to cell type $\times$ number of sizes). For the rotation-equivariant model, we uniformly discretized the full 2π range of orientations into four possible values $\in \{0, \pi/2, \pi, 3\pi/2\}$, matching the distance between the orientations considered in our simulated population (i.e., 0 and $\pi/2$). To further avoid redundancies and simplify training, we fixed the readout position for each neuron to its correct location, to focus the training of these models on learning the convolutional kernels and neuron-specific feature weights.

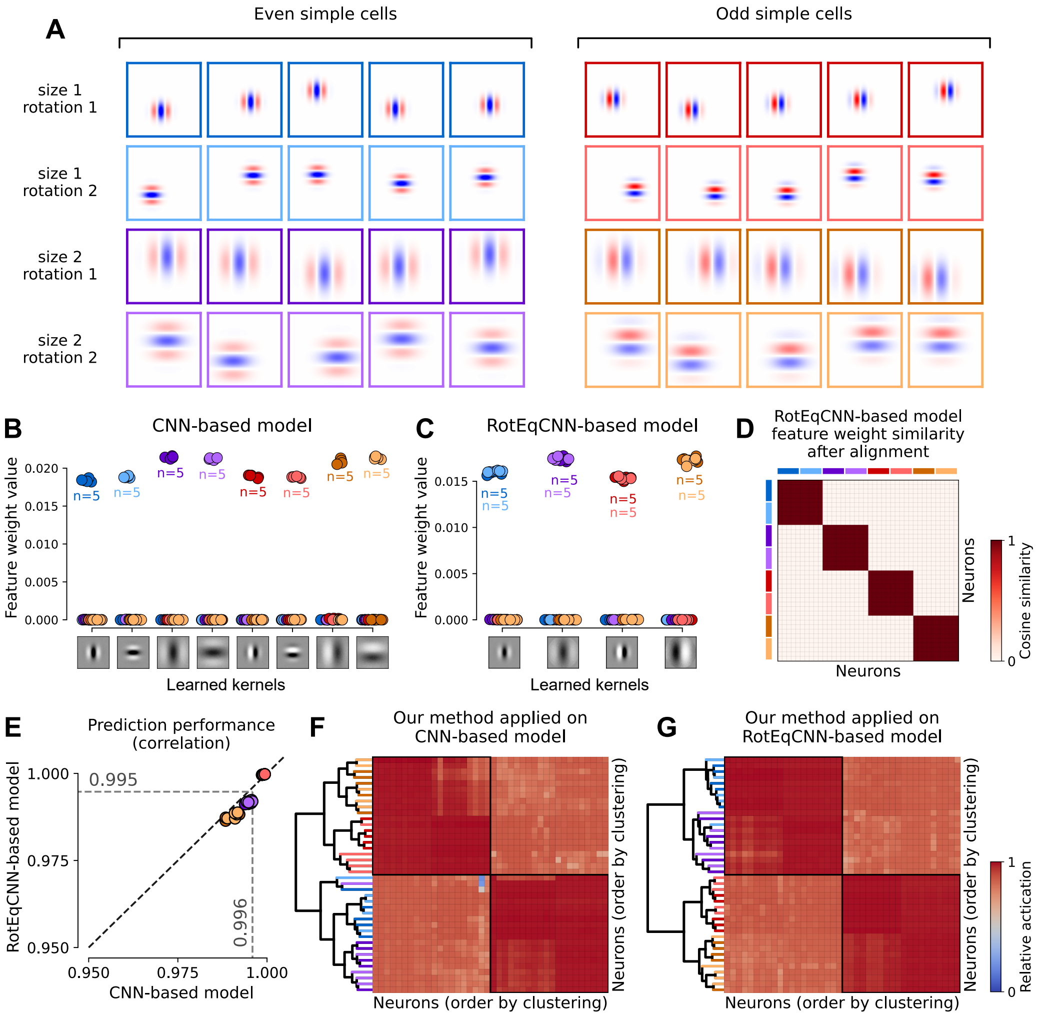

Figure S2: **Comparison with model-based clustering algorithms: A)** Simulated neuron population, containing odd and even simple cells with 2 different sizes, 2 different orientations, and 5 different randomly sampled positions. **B)** Feature weights of CNN-based model (Klindt et al., 2017). Neurons with the same neuron type, rotation, and size can be clustered together as they present similar feature weights. **C)** Feature weights of RotEqCNN-based model Ustyuzhaninov et al. (2019), while disregarding the specific orientation, by marginalizing (i.e., summing) the weights across different rotations of the same kernel. **D)** Cosine similarity between feature weights after aligning the weights regardless of their orientation. Neurons belonging to the same type and sharing size have high cosine similarity. Colors on the top and on the left are the same as in panel A. **E)** Scatter plot showing prediction accuracy (in terms of correlation to target responses) of CNN-based model vs RotEqCNN-based model. **F)** Resulting clustered activation matrix from the application of our method on the CNN-based model. Dendogram colors are the same as in panel A. **G)** Resulting clustered activation matrix from the application of our method on the RotEqCNN-based model. Dendogram colors are the same as in panel A.

For the remainder of this section, and in Fig. S2, we will refer to the convolutional (Klindt et al., 2017) and rotation-equivariant convolutional (Ustyuzhaninov et al., 2019) models as CNN-based and RotEqCNN-based models, respectively.

Training of model-based methods Both models were trained on 150,000 white noise images presented once, grouped in batches of size 10, minimizing the mean squared error between the ground truth responses of our simulated simple cell models and the predictions of the convolutional architectures. Similar to Klindt et al. (2017), to encourage the models to learn distinct kernels and allow clustering of neurons based on distinct functional properties, we applied an L1 regularization on the readout feature weights with a strength coefficient of 0.01. This setup enabled the models to effectively capture the functional diversity of the neuron population while maintaining both interpretability and efficiency. For training, we used the Adam optimizer with a learning rate of 0.01.

Clustering via model-based methods Both models achieved high prediction performance, computed as the correlation between the ground truth responses of our simulated simple cell models and the predictions of the trained models to newly sampled white noise images (see Fig. S2E), and learned to associate different neurons to different channels, grouping them according to the design of their architecture. Specifically Fig. S2B shows how, at the end of our training of the CNN-based model, neurons of the same type, size, and orientation learn similar feature weight vectors independently of their position. Consequently, while clustering based on these feature weights is invariant to differences in location, it does not disregard differences in size and orientation.

Similarly, Fig. S2C and D show how, at the end of our training of the RotEqCNN-based model, neurons can be clustered using their feature weights according to differences in type (odd vs even) and size, disregarding differences in orientation and position. Specifically, neurons performing the same computation but with different orientations picked the same kernel (i.e., the same computation), which in the RotEqCNN-based model is present with different rotations. In this case, a simple marginalization based on summing feature weights across the different rotations is sufficient to cluster neurons with unique type and RF size, disregarding RF location and orientation (Fig. S2C).

Additionally, we performed an alignment procedure similar to Ustyuzhaninov et al. (2019) to compare feature weights irrespective of rotational differences. By finding the optimal rotation that aligns the feature weights of two neurons, we can assess their computational similarity while disregarding orientation. The results, shown in Fig. S2D, highlight that neurons of the same type and size exhibit high cosine similarity in their aligned feature vectors, where the optimal rotation was found by minimizing the Euclidean distance between weights. This confirms the ability of the method introduced by Ustyuzhaninov et al. (2019) in clustering neurons despite variations in receptive field (RF) location and orientation. However, similar to Klindt et al. (2017), this method remains limited in its ability to cluster neurons irrespective of other nuisance RF properties such as size, which is known to vary with eccentricity in higher mammals' visual systems (Wilson & Sherman, 1976; Cavanaugh et al., 2002; Harvey & Dumoulin, 2011).

Comparison with our method Lastly, we applied our method, activation-based clustering via template learning and template matching, to both models. The clustering results are consistent across both architectures (Fig. S2F,G) and group neurons with the same type (odd vs even) together, successfully disregarding differences in RF position, orientation, and size (further confirming the results shown in Fig. 2). Importantly, the consistency of our results across both architectures highlights the model-agnostic nature of our method. That is, as long as an accurate response-predicting model is used our method yields consistent clusters, which is a crucial advantage in not constraining (or biasing) such findings or analyses to specific architectures or their implementation. An additional important observation is that the activation matrix, which contains the activations after template matching, presents relatively high values between neurons of different types. This reflects the fact that the neuron types considered, odd and even simple cells, differing only in phase, perform quite similar computations, as when properly shifted odd Gabor patterns can activate an even simple cell relatively high, and vice versa. In contrast, this computational similarity is not visible in the feature weights of the model-based methods.

I Supplementary figures

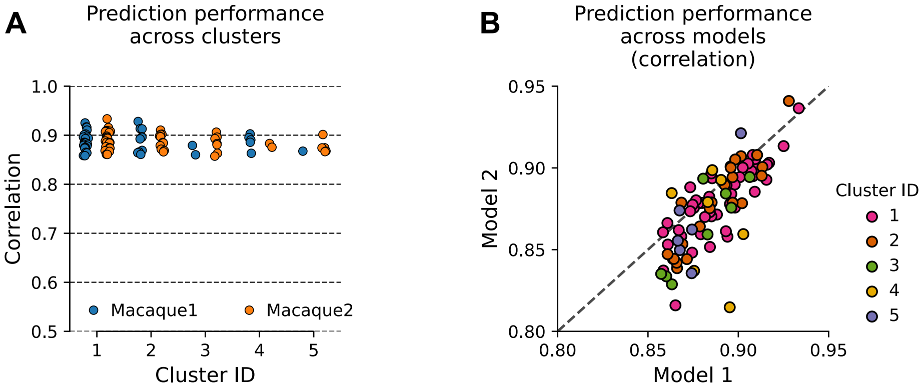

Figure S3: **Prediction performance of response-predicting models: A)** Prediction performance of Model 1 (which was used for the main results) in terms of correlation to trial-averaged responses on left out images (*i.e.*, test set). **B)** Prediction performance of Model 1 vs Model 2 in terms of correlation to trial-averaged responses on left out images (*i.e.*, test set). Model 2 has a different architecture but was trained on the same set of neurons as Model 1.