Modeling Visual Cortex with Maximum Capacity Representations

by

Thomas Edward Yerxa

A dissertation submitted in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

Center for Neural Science

New York University

January, 2026

Dr. Eero P. Simoncelli

Dr. SueYeon Chung

DEDICATION

For Danie, Dave, Sue, and Mika

Acknowledgments

I am eternally grateful to my advisors, Eero and SueYeon. Eero, your passion for and commitment to the pursuit of understanding are inspirational. Thank you for showing me that the value of a piece of knowledge is derived from its communication, and that communication is an act of empathy; the search for simplicity as well as kindness and consideration are therefore core tenets of scientific pursuits to which I will always aspire. SueYeon, without your willingness to take a chance on me, and your steadfast encouragement to pursue new ideas and new tools, none of this work would have been possible. With absolute sincerity, thank you.

I am very lucky to have spent the last decade a part many wonderful communities, without whose support this journey would have been more treacherous and less joyful. Both the Flatiron and CNS are special places to go to work. The assortment of expertise is formidable, but what stands out to me is the accessibility of this expertise. These are places where conversations over coffee turn into solutions at the blackboard or invitations to attend or present at group meetings as a matter of routine. This doesn't "just happen," but rather is facilitated by a team of diligent and talented administrators whose efforts foster such fruitful interactions; Jessica, Alicja, Matthew, your warmth will be missed.

LCV, what a home for a Ph.D.! Florentin, Pierre-Etienne, and Zahra, my three musketeers, you'll never know how much you're constant availability for a chuckle, a cheer, or commiseration has meant to me. Jenelle, you taught me so much about how to tackle hard problems, and especially the value of an honest conversation during a stressful time. Lyndon, you welcomed

me into the group when I joined remotely from California and I quickly realized I could never ask for a more steadfast and understanding friend. We knew we were in the good ol' days when we shared an office, and we made the most of it. LCV'ers past present (and future): Kate, Sreyas, Billy, Caroline, Colin, Nikhil, David, Hope, Jerry, Julie, Ben, and Guy, the lunches, dinners, and dances we shared are some of my most treasured memories. It has been the privilege of a lifetime to call such a brilliant and varied group of people my colleagues and friends.

I would be remiss to not mention those most directly responsible for starting me on this path: Emily Cooper, the most supportive mentor an aspiring undergraduate could have, and of course the "Physics Boys" (we really need to figure out a better group name). Cade, Caolan, Dan, Jackson, Patrick, Peter, Spencer, and Thomas, we started with each other and watching how everyone's paths have become their own has been a highlight of adult life. Here's to many more decades of friendship.

Finally to my family: Danie, Dave, and Sue, without your love, support, and caring I would not have been able to make the choices that led me here, and I am quite certain none of this work would have been worth doing. Through easy times and hard, you have always been there to challenge me when I needed it, and bolster me up I am down. I only hope to be able to do the same. Mika, what can I say but I love you. The depth and ease with which you understand me are nothing short of magic. I will always be there for you, as you have been for me.

Abstract

Providing normative explanations for the structure of neural representations of sensory stimuli is a longstanding goal of theoretical neuroscience. One influential framework is the efficient coding hypothesis, which posits that biological systems aim to encode as much information as possible, subject to constraints imposed by their physical and metabolic implementation. While this theory achieved significant success in explaining representations in early visual areas, it was eventually surpassed by task-trained deep neural networks as the leading models of higher-level cortical representations. In this thesis, we develop a modern formulation of the efficient coding hypothesis by introducing an objective based on manifold capacity, a measure of how many manifolds can be reliably separated by a given representation. We show that this framework is: (1) rich enough to produce representations that support sophisticated behavior and predict high-level cortical responses, (2) flexible enough to incorporate constraints and findings from visual and perceptual neuroscience, and (3) compatible with biologically plausible optimization schemes and canonical neural computations.

Contents

List of Figures

List of Tables

List of Appendices

1 Introduction

1.1 Overview

This thesis explores a modern instantiation of the efficient coding hypothesis based on manifold capacity. In this introduction I first motivate the search for a normative principle to explain the neural computations that support the diverse array of biological visual behaviors. Next I introduce and give historical context for the efficient coding hypothesis, before describing some of its successes and limitations as a model. Finally we arrive at the modern problem context of self-supervised learning, which I cast as yet another version of efficient coding. The introduction concludes with brief descriptions of each of the subsequent chapters.

1.2 Normative Models of Visual Processing

Providing normative explanations for the structure of biological representations of sensory stimuli is a longstanding goal of theoretical neuroscience. The formal origins of this line of inquiry can be traced back to at least the mid 19^{th} century and Hermann von Helmholtz, to whom it was clear even then that perception is an inferential process rather than an objective representation of physical reality (Von Helmholtz, 1867). This initial observation has subsequently been bolstered by a vast body of evidence demonstrating that, in many contexts, humans act as optimal Bayesian observers by incorporating both prior knowledge and information about the

uncertainty of their internal representations to make “best guesses," about the external state of the world (Jaynes, 1957; Knill and Pouget, 2004).

Given that biological sensory representations support this type of behavior, it is natural to suppose that they are designed for expressly this purpose. This is the principle on which “Optimal Inference" theories of neural computation are based. Notable members of this class of theory include those that optimize functions to predict human defined latent variables associated with some sensory signal. [1] Yet the human visual system must learn to solve these problems, raising the question of how feasible a direct inferential approach really is. Bayesian estimation provides an elegant framework for inference, with the central algorithmic or learning requirement being access to information about the relevant prior distribution. In simple low-dimensional cases, such priors can be learned efficiently from relatively few examples, and indeed this idea underlies the classical framework of signal detection theory. However, more complex inference tasks quickly encounter the “curse of dimensionality,” where the sample complexity of learning grows exponentially, rendering the acquisition of accurate priors prohibitively data-hungry in high dimensions.

In computer vision, this challenge was overcome by the curation of a massive dataset of natural images and fine-grained semantic category labels (Deng et al., 2009). This dataset, known as ImageNet, enabled hierarchical models with millions of parameters to be trained to to perform the difficult high dimensional task of object classification at a level far exceeding any previous approach, catalyzing the deep learning revolution (Krizhevsky et al., 2012). It was subsequently shown that these task-optimized deep neural networks learn representations that are predictive of neural responses to visuals stimuli in the ventral stream. Intriguingly that performance on the object classification task was strongly correlated with the degree of predictivity (Yamins et al., 2014).

[1] A strict definition of this class of models would only include those that explicitly combine prior beliefs with noisy measurements in order to make maximum a posteriori inferences. However, a more liberal interpretation that includes models trained to perform any inferential task will be instructive for my purposes.

Despite these considerable successes, this approach leaves much to be desired as a normative explanation for biological vision. For one, real organisms are unlikely to encounter the vast number of example signals and corresponding ground-truth latent values that would be required to facilitate such a "supervised" learning process. More fundamentally, supervised learning places the burden of task-choice on the modeler. What is the right latent variable to have a model extract? Given that "real" visual systems are interesting precisely because they support such a wide variety of robust behaviors, it seems likely that choosing any particular task (or even combination of tasks) is destined to produce an incomplete description. These concerns have, in many ways, been validated. Supervised object classifiers have been shown to be vulnerable to adversarial attacks, small human-imperceptible perturbations that reliably alter the model's output, revealing a significant divergence from biology in the perceptual capabilities of these models (Szegedy et al., 2013). Moreover, the early trend in which improvements in classification performance led to better alignment with neural data has since plateaued or even reversed: newer models may achieve higher accuracy on benchmarks, yet exhibit lower predictive power with respect to biological neural responses (Linsley et al., 2023; Schrimpf et al., 2018).

Where then do we turn in the search for an optimality principle to explain sensory processing systems? Frustratingly, the "right" answer is both obvious and at the same time computationally intractable to optimize. Evolutionary fitness is the only proximal force that actively shapes the features that appear in biological organisms, but accurately modeling fitness would require simulating the environment that creates selection pressures with sufficient fidelity as to facilitate learning. In light of this, we will instead specify two desiderata for candidate normative theories:

1. They should describe an optimality principle that is sufficiently general or *task agnostic*, in the hopes that any system that achieves such a goal would support optimal behaviors in a wide variety of tasks.

2. The aforementioned optimality principle should be specified in terms of sensory signals and

internal representations of those signals alone, i.e. without reference to "extra" information that may or may not be available to an organism as its neural representations are shaped on either evolutionary or developmental time scales.

These criteria place us firmly in the territory of un- or self- supervised learning, where the learning signal is derived from the input signal itself. Several such frameworks have been used to model biological vision with varying degrees of success. For instance Rao and Ballard (1999) proposed a hierarchical model of "Predictive Coding" where top-down feedback explicitly predicts lower level representations. Despite the elegance of the idea and a simple neural implementation, predictive coding has not proven a highly successful way to learn abstract representations. Földiák (1991) proposed that a network could learn to perform temporal prediction, which requires learning a rich representation that identifies which aspects of a signal are stable in time and how these features evolve. However, this thesis will focus on a framework that explicitly aspires to adhere as closely as possible to the spirit of criterion 1, while making concessions only in the name of providing a better model of some biological computation, namely, the efficient coding hypothesis.

1.3 Efficient Coding Theories of Sensory Representation

Natural signals contain rich structure, and as a result of these structures measurements of these signals have the potential to to encode redundant information. Thus, Barlow and Attneave's original formulations of efficient coding centered around the idea of *redundancy reduction* (Attneave, 1954; Barlow et al., 1961). A sketch of Barlow's argument is as follows. Consider a lossless channel that transmits sensory signals with an average information content of H bits by sending a sequence of impulses (i.e. spikes) down a finite number of fibers in response to each input (so each particular fiber makes some measurement and emits some number of spikes in response). Such a channel is said to be efficient when its capacity, C the maximal number of bits it is capable

of communicating, is matched to the signals information rate. Thus the goal of such an efficient coding system is to decide on an input-output mapping function that allows for the use of the smallest capacity channel. Even in this simple setting the hypothesis that sensory systems encode stimuli efficiently makes two qualitative predictions about neural representations:

1. Assuming that the channel's capacity is a monotonically increasing function of the total number of spikes emitted, it is clear that efficient codes should "economize" spikes. This can be achieved by assigning frequent inputs to output patterns that emit few spikes, and unlikely inputs to high-firing rate outputs. [2]

2. Individual fibers should encode some unique aspect of the input signal. Redundant responses betray the existence of excess representational capacity.

This simple formulation belies a richer framework. For example, it presupposes that it is possible to code sensory signals losslessly, which is on its face implausible. This is both because natural signals are massively high dimensional and the neural substrate of their encodings is inherently noisy. This immediately necessitates at least some reformulation, and in fact many different instantiations can be arrived at. So, a more general formulation of the efficient coding hypothesis is "neural representations of sensory stimuli should encode as much information about their inputs as possible, subject to the constraints imposed by their biological implementation." Concrete theories can be constructed via this framework by answering two key questions: (1) what do we mean by efficient (or, what constraints are we considering), and (2) what do and we mean by informative (or, how do we define redundancy)?

Consider for example, the simplest possible setting: a single neuron whose response encodes a one-dimensional stimulus. Under the assumption that the neuron's response is 1-to-1 (so no

[2] This result is highly reminiscent of Shannon's source coding theorem, where the optimal code length for a given input is proportional the log probability of that symbol (Shannon, 1948). Barlow was significantly influenced by Shannon, in fact his statement of the efficient coding hypothesis in the language of information theory is the crux of his contribution; Attneave made similar qualitative observations about representational efficiency years prior (Attneave, 1954).

information in lost), and impose the constraint that the neuron has a finite dynamic range, the maximally informative output distribution is simply the uniform distribution over the response range (which is the maximally entropic distribution on a compact space). As a result a neuron that is efficient in this sense will employ a tuning curve that is proportional to the cumulative distribution function of the stimulus distribution (Laughlin, 1981). Extending to D dimensional signals represented by a set of D linear measurements (neurons) yields a slightly richer problem setting. So long as the measurements are not linearly dependent no information will be lost, but we can still ask which set of directions in the signal space will lead to the least redundant measurements. If we measure redundancy as the degree to which two measurements are correlated with each other, the optimal measurements are given by the principal components of the signal (which by design produce decorrelated outputs). [3] If we expand the notion of redundancy to include all (higher order) dependencies, we arrive at the formulation of Independent Components Analysis, which explicitly seeks to find directions where linear measurements are as close as possible to statistically independent (Bell and Sejnowski, 1996). The difference between measurements optimized for these two notions of redundancy are visualized in Fig. 1.1.

One common choice for "how to measure informativeness" is the mutual information between neural responses and incoming stimuli. In settings where not all information can be preserved, this requires more than simply reducing the redundancy (or maximizing the entropy of responses). While a principled choice, unfortunately information theoretic quantities such as the mutual information are difficult to compute in general, and as a result many methods can be categorized according to how they attempt to do so. For instance, Atick and Redlich (1992); van Hateren (1992) and Doi et al. (2012) assume a simplified linear response model with Gaussian noise on inputs and outputs, which yields a closed form expression form for mutual information, in order to study coding efficiency in the retina. Karklin and Simoncelli (2011) assume a more

[3] Intriguingly, the principal components are also the measurement directions that emerge from networks whose connections are trained with "fire together, wire together" Hebbian learning rules (Hebb, 1949).

flexible nonlinear response model that is only locally Gaussian (i.e. responses are Gaussian with with covariance that varies between distinct inputs); the price of this flexibility is that they must solve for the optimal encodings numerically rather than analytically. Brunel and Nadal (1998) presented a lower bound on mutual information in terms of the expected value of a local information measure, which is in general much simpler to compute. Several subsequent works then proposed to maximize this bound rather than directly estimate the mutual information. Some examples include Ganguli (2012); Wei and Stocker (2012), which explore how populations of neurons equipped with nonlinear tuning curves and Poisson noise can optimally encode one-dimensional stimuli, and Yerxa et al. (2020) which extended these results to higher dimensions.

An alternative approach to measuring "informativeness" is to determine what can be decoded from a representation. For example the sparse coding framework of Olshausen and Field (1996) encourages that information about stimuli be preserved by minimizing the squared error of a linear reconstruction of the input signal. An alternative to decoding the stimulus itself, is to maximize the number of patterns or partitions of stimuli can be linearly separated in the representation space. The number of separable patterns is referred to as the "capacity" of a representation. Gardner and Derrida (1988) studied the capacity of representations to store partitions of individual data points and this work was extended to consider the capacity to store continuous groups of points or "manifolds" by Chung et al. (2018). One key advantage of capacity maximization over stimulus reconstruction is that, in the context of a biological implementation, it does not require circuitry that sends signals from the cortex back to the sensory periphery in order to be computed.

Besides a way to measure the quality of a representation, efficient coding theories must also specify what constraints to consider. One common constraint is the total amount of available neural resources whose collective activities comprise the representation. Neurons take up physical space and are costly to maintain, and many of the examples above explicitly consider the number of neurons as a constraint (Doi et al., 2012; Ganguli, 2012; Gardner and Derrida, 1988; Wei

and Stocker, 2012). Other constraints of this type include limitations on the amount of available neurotransmitters (i.e. the total strength of all synaptic connections), or a limit on the available length of wiring to connect neurons (Chklovskii et al., 2002; Doi et al., 2012). Another common consideration is the metabolic cost of neural activity. Many author's choose to penalize the total variance of neural firing rates (Atick and Redlich, 1992; Doi et al., 2012; van Hateren, 1992), and others operate under the assumption that the total number of spikes should be penalized (Olshausen and Field, 1996). However, careful accounting of energetic expenditure in neural circuits shows that the primary cost of ATP is borne by the postsynaptic units, which must restore their membrane potentials after excitatory postsynaptic currents. Thus, the metabolic cost of spiking is better modeled as the product of a neuron's firing rate and the total strength of it's connections to downstream units (Lennie, 2003). [4] Still other works have considered the constraint of biologically plausible optimization. Many optimization algorithms require precise credit assignment or the propagation of error signals across many layers of a network, which are not thought to be plausible computations for neural circuitry. There is no "correct" information metric or constraint set, and they are instead chosen depending on what particular neural phenomenon a given model aims to predict.

1.4 How Do we Measure Success?

Given a particular candidate model, i.e. a function f that produces some representation $\mathbf{r}$ of some image $\mathbf{x}$, how do we evaluate its ability to explain some neural computation? The most straightforward approach is to compare properties of the artificial image representation with measurements of neural responses to the same stimuli (i.e. physiological comparisons). A second family of comparison methods use access to the function f to predict how observers will behave

[4] It is also worth noting that real neurons seem to respect this principle. When faced with a metabolic deficit, populations do not reduce their firing rate but rather decrease their coding precision by raising their resting potentials and decreasing the conductance of ion channels (Padamsey et al., 2022; Padamsey and Rochefort, 2023).

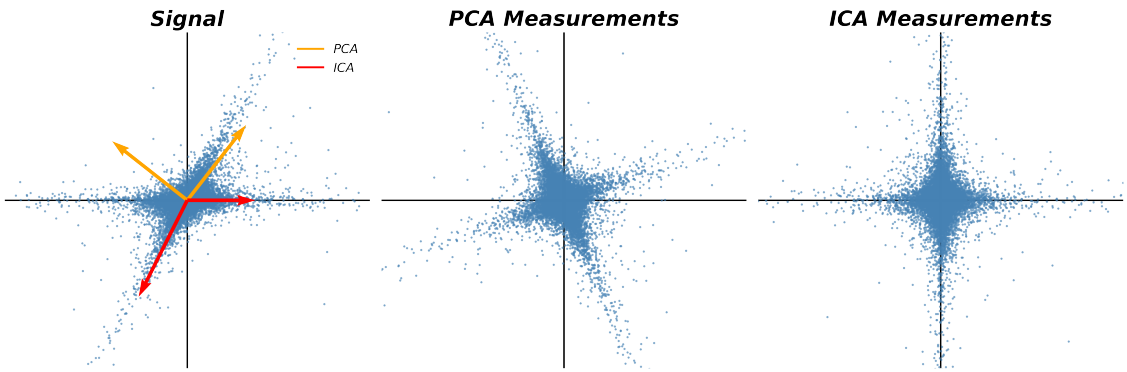

Figure 1.1: PCA vs. ICA: Different Notions of Redundancy. In the left panel, an example 2-D signal which is generated by taking a linear mixture of two heavy-tailed independent sources. Overlaid on the signal are the directions that are optimized to minimize (1) linear correlation between the two measurements, these are the first two principal components (labeled PCA) and (2) statistical dependence between the two measurements (the independent components, labeled ICA). In the right two panels we visualize the measurements made by either the principal components or the independent components. In both cases, the two measurements are decorrelated, which visually corresponds to the two "prongs" being approximately orthogonal to each other, but in the ICA space the two axis are perfectly independent (i.e. position on the abscissa gives no information about its position on ordinate in the ICA measurement space).

when queried with arbitrary inputs (i.e. perceptual comparisons). These approaches often have complementary strengths. For example, though the acquisition of neural data is expensive, once a set of measurements has been made it is often simple to ask how well any new model can explain these measurements. Conversely, behavioral or psychophysical measurements are relatively inexpensive, but these frameworks often require additional experiments to test the efficacy of each candidate model.

1.4.1 Perceptual Comparisons

If we accept the hypothesis that our perception of the world arises from neural representations of sensory measurements, we can learn a significant amount about the structure of these representations without ever directly observing them. A particularly elegant example comes from early investigations into human color perception. The color of a light source is determined by the distribution of energy across the different frequencies of electromagnetic radiation. Naively,

one would assume that to reproduce a particular color would require measuring at matching the energy at each frequency – which would imply the necessity of infinite measurements as frequency is a continuous variable! In the 19th century, Helmholtz sought to determine how many measurements were actually required, and found that participants could recreate the color of a source light by adjusting the weights of a combination of just three sources. The now famous color-matching experiments perceptual Evidences for Thomas Young's theory of trichromatic color perception, whose physiological basis would not be confirmed for another century (Baylor, 1987; Svaetichin, 1956; Von Helmholtz, 1867; Young, 1802).

Importantly, though the 3-component mixtures which were perceived by participants as having the same color as the test source, they did not share the exact same spectral content. The mixtures and the test source are examples of "metameric" stimuli: two signals that are physically different but are mapped to the same internal representation (or, perceived to be the same). Metamers reveal aspects of a signal that our internal representations are invariant to, and therefore can be leveraged to evaluate the quality of a candidate model by comparing stimuli that different models predict to be metameric (Feather et al., 2023). Models of neural responses also make predictions about perceptual sensitivity. For example, we can determine the extent to which an input perturbation changes a model's representation and compare this measurement with the detection threshold of a human observer when viewing the same perturbation (Berardino et al., 2017).

1.4.2 Physiological Comparisons

A more straightforward approach is to use a normative model to make predictions about response properties of neurons that have been directly measured in physiological experiments. The predictions any particular model can make will vary according to the setting considered by the modeler. For instance, the single unit efficient coding scheme of (Laughlin, 1981) derives an optimal response nonlinearity (namely, the cumulative distribution of the encoded stimulus, which

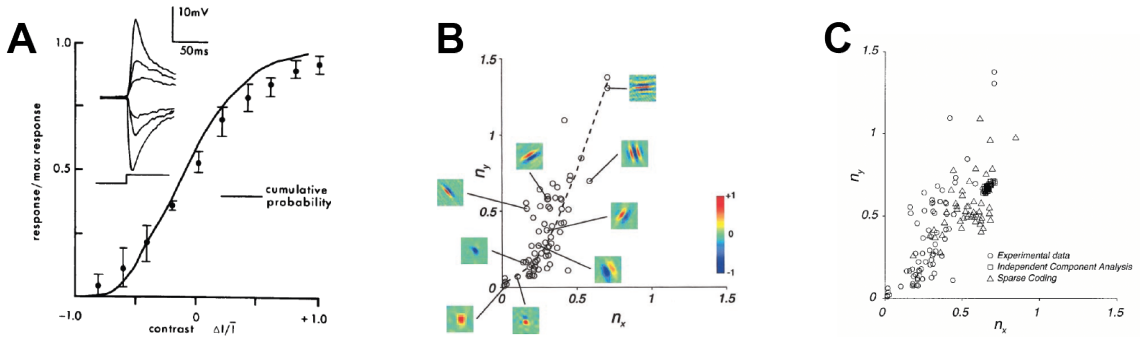

Figure 1.2: Physiological Comparisons of Efficient Coding Theories. **A**. The CDF of the distribution of contrast in natural visual inputs (solid line) is compared with the amplitude of an LMC cell's responses to various contrast level. This figure was reproduced from (Laughlin, 1981). **B**. The distribution of receptive field shapes in macauqe V1 simple cells. Along the x-axis the number of lobes (or cycles) grows, and along the y-axix the aspect ratio changes (i.e. the lobes become more elongated). **C**. The same distribution of receptive field shapes from V1 are compared to those obtained from either sparse coding or ICA, which both learn a gabor-like basis to represent images. We can see that the filters produced by ICA are nearly all similarly shaped, (the cluster of square dots), which stands in contrast to the basis produced by sparse coding and observed in the biological data.

maps the input signal to a domain where all response levels are equally likely). A comparison between this "histogram equalizing" strategy and the tuning curve of large monopolar cells in the fly visual system (which plots response amplitude vs. stimulus contrast) is reproduced in Fig. 1.2A, and reveals a reasonable match between the normative model and biological measurements. Sparse coding and ICA both predict that the optimal basis with which to represent natural image patches is a set of localized, oriented, and bandpass Gabor filters (Bell and Sejnowski, 1996; Olshausen and Field, 1996), a class of functions commonly used to characterize the receptive fields of simple cells in primate area V1. As both are population encoding models, we can compare properties of the optimal bases produced by each to the population of V1 units. Ringach (2002) characterized the distribution of receptive field shapes in simple cells and compared these with the distributions produced by sparse coding and ICA (see Fig. 1.2B/C). This comparison shows that, though both methods produce a qualitative match to primary visual cortex, sparse coding provides a better model of the diversity present in the biology.

While comparisons such as these are satisfying, they often rely on having a preexisting notion

of what a particular neuron or area is computing (i.e. the LMC neuron's encode contrast and V1 encodes localized oriented features). Unfortunately, we generally lack strong (or at least, universally accepted) hypotheses for the functions of later stages of processing and thus require a way for making comparisons between models and measurements that do not depend on strong assumptions. One common approach is to define a measure of similarity between $\mathbf{R}_{\text{model}} \in \mathbb{R}^{B \times d}$, the d-dimensional representation of a set of B inputs produced by a candidate model, and $\mathbf{R}_{\text{brain}} \in \mathbb{R}^{B \times n}$, the (typically trial averaged) firing rates of a population of n neurons in response to the same set of inputs. For example, representational similarity analysis (RSA), first computes the $B \times B$ pairwise distance matrix for each $\mathbf{R}_{\text{model}}$ and $\mathbf{R}_{\text{brain}}$, and then measures the correlation between the entries of these matrices (Kriegeskorte and Kievit, 2013). In this work we will primarily rely on a different approach that involves finding an optimal mapping function $M^* : \mathbb{R}^d \rightarrow \mathbb{R}^n$ that directly predicts neural responses from model outputs. The functional form of M^* can, of course, change the interpretation of the resulting similarlity between predictions and observations. One could choose a "strict," mapping function that simply selects the model unit most similar to an individual neural response, but it is unclear that such a one-to-one correspondence would exist even between pairs of individual brains. On the other hand, if we use a highly flexible map with many parameters it would be unclear whether predictivity is due to the normative hypothesis that shaped $\mathbf{R}_{\text{model}}$ or the power of the mapping function M^*.

The community has loosely settled on using a linear hypothesis class for the mapping function and finding the parameters of M^* using regularized regression. This is the setting in which task-trained deep networks rose to prominence as the most predictive models of neural responses in the visual system. Concretely, (Yamins and DiCarlo, 2016; Yamins et al., 2014) showed that by linearly mapping early layers of deep networks to early cortical areas such as V1 and V2, and deeper layers to later stages of processing such as V4 and IT, one could obtain higher predictivity than by using any existing approach. This "neural regression", paradigm (i.e. of selecting intermediate representations of hierarchical networks to map to different cortical areas), is employed

throughout the following chapters and schematized in Fig. 1.3.

Intriguingly, the advantage of task-optimized networks relative to more traditional models varies along the ventral stream hierarchy, with a negligible or non-existing improvement over handcrafted models in V1 (Parthasarathy et al., 2024a), and the most pronounced difference in IT. This leads us to speculate as to the cause for this observation: later stage neural representations must display complicated invariances in order to support flexible behaviors, like pose invariant object recognition, and typical efficient coding objectives struggle to provide guidance as to what information should a representation should preserve and what it should become invariant to. However, this is not a fundamental limitation of efficient coding or unsupervised learning; in the next section we describe a methodology that largely satisfies the key criteria described in Section 1.3 while also producing representations that are both useful for computer vision tasks and predictive of neural responses throughout the cortical hierarchy.

1.5 Self-Supervised Learning of Visual Representations

The goal of unsupervised learning is to discover structure in data using only the data itself. As a result most such techniques can be understood as some form of density estimation, the "mother of all unsupervised learning tasks". Self-supervised learning (SSL) on the other hand is characterized by the construction of an auxiliary task, typically by somehow modifying the observations towards this end. Some example auxiliary tasks include colorization (recovering color information from images that have been converted to grayscale) in-painting (predicting the values of pixels in an image that have been masked out. However, SSL came into its own when focus shifted from specifying what what features should be encoded, to specifying what information a representation should *discard*.

Concretely, methods that focus on learning functions that are invariant to some experimenter defined set of transformations on the input signal began to show significant promise. Let $\tau(\cdot, \rho)$

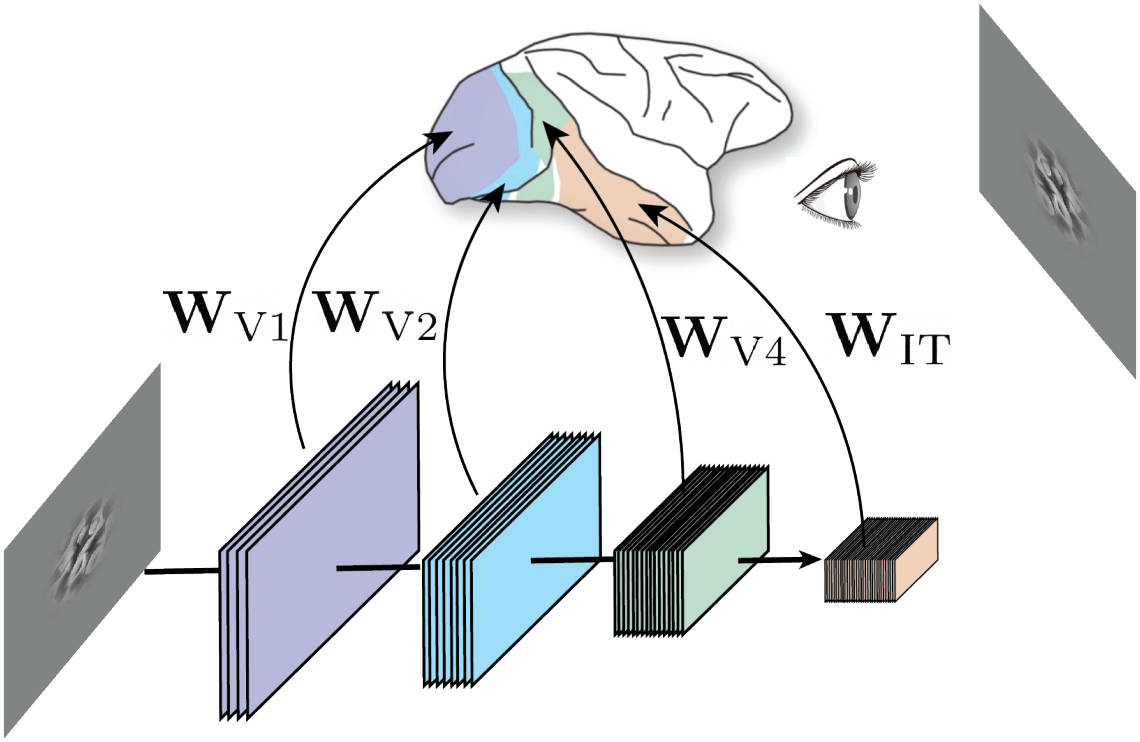

Figure 1.3: Neural Regression Schematic. The neural regression paradigm that is both common in the literature and used throughout the thesis. After the parameters of a hierarchical optimization are set according to some dataset and objective function, images are fed through the network and subsequently used to predict neural responses in visual cortex to the same images, typically via linear regression.

be an "augmentation" function parameterized by ρ that takes as input an image and produces a "view" of that image (i.e. τ could be a cropping operation and ρ could represent the coordinates of the crop). An invariance learning task would be to optimize some function f to minimize the expected difference in its outputs between pairs of random views of the same image: $\mathbb{E}_{\mathbf{x},\rho_1,\rho_2}[||\mathbf{z}_1 - \mathbf{z}_2||_2^2]$ where $\mathbf{z}_1 = f(\tau(\mathbf{x}, \rho_1))$, $\mathbf{z}_2 = f(\tau(\mathbf{x}, \rho_2))$. Unfortunately, this objective permits trivial solutions, if f is a constant function then perfect invariance is achieved. We therefore instead consider maximizing the mutual information between the representations of the two views:

$$I(\mathbf{z}_1; \mathbf{z}_2) = H(\mathbf{z}_1) - H(\mathbf{z}_1|\mathbf{z}_2). \tag{1.1}$$

The first term encourages f to encode as much information as possible about its inputs, while the second codifies a constraint that f should be as near as possible to invariant to the action of τ. As discussed in Section 1.3, this information theoretic quantity is difficult to optimize directly, and as a result many approaches that loosely share this goal emerged in the literature.

Many such methods utilize the InfoNCE loss function, first introduced in Oord et al. (2018), which is based on the idea of noise contrastive estimation from Gutmann and Hyvärinen (2010) and is a lower bound on the mutual information between views. Intuitively, this loss encourages different views of the same image to be nearby and repels the representations of distinct images; methods using this loss such as those from Chen et al. (2020b); HaoChen et al. (2021); He et al. (2020), are referred to as "sample contrastive" (Garrido et al., 2023a). Sample contrastive methods typically operate under the constraint that network outputs, $\mathbf{z}_{i,j}$ have unit norm. Thus the codomain of f is a compact space, and the maximally entropic $p(\mathbf{z})$ is the uniform distribution over the unit sphere (Wang and Isola, 2020).

If you instead constrain the outputs of f to have finite total variance, the maximally entropic distribution is an isotropic Gaussian. Indeed multiple methods aim to produce such decorrelated representations (Bardes et al., 2022; Zbontar et al., 2021). These methods make com-

parisons between pairs dimensions, aiming to diagonalize a representation's covariance matrix $\mathbb{E}[\mathbf{Z}_1^T \mathbf{Z}_2] \in \mathbb{R}^{d \times d}$ (where $\mathbf{Z}_{1,2}$ are matrices containing the network's outputs to many images, and d is the representation's dimensionality). [5] As such they are often called "dimension contrastive".

Critically both of these (and other similar) approaches, when paired with the appropriate set of input transformations τ, produce representations that rival direct those obtained with direct supervision in terms of both performance on downstream tasks such as object recognition and ability to predict neural responses throughout the ventral stream (Zhuang et al., 2021). Since semantic category knowledge can be effectively replaced by the knowledge that two inputs are related through some transformation, it is natural to ask whether transformations similar to τ occur during natural viewing conditions. Empirically the two most important types of transformations to include in τ are a random resized cropping operation and color jittering (which randomly modulates brightness, contrast, saturation and hue), which crudely approximate how visual inputs change with the position of an observer and changes in lighting conditions respectively. Taken together, these results suggest that training a model to preserve information while achieving viewpoint invariance and color constancy is sufficient to produce a representation that supports sophisticated behaviors and provides a strong quantitative account for the properties of units in the ventral stream. The promise of this direction serves as the direct inspiration for much of the work of the following chapters.

1.6 Thesis Structure and Overview

The thesis begins by taking one particular instantiation of coding efficiency, namely manifold capacity, and deriving a learning objective from this principle. While we are not the first to link SSL to efficient coding, our formulation leads to some advantageous computational properties. In the following two chapters address some limitations of standard SSL as an efficient coding

[5] Whereas the sample contrastive methods instead diagonalize the Gram matrix $\mathbb{E}[\mathbf{Z}_1 \mathbf{Z}_2^T] \in \mathbb{R}^{B \times B}$ (where B is the number of input images).

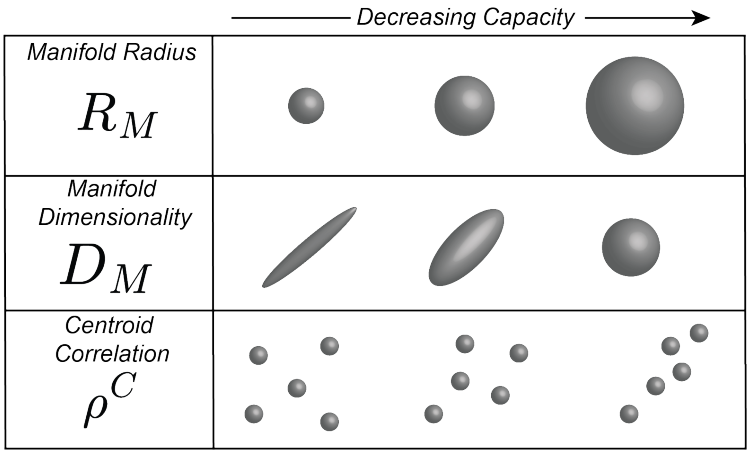

Figure 1.4: Geometric Components of Capacity. The geometric components that determine manifold capacity, visualized for manifolds exhibiting elliptical symmetry. As the scale of an ellipsoid grows, the storage capacity decreases (fewer large objects can be packed into a small space than larger ones); top row, the scale is measured by the manifold radius. Ellipsoids that extend in multiple dimensions are more difficult to pack than those whose extent is low dimensional (you can store more objects if they "lie flat"; middle row, this notion of size is quantified by the manifold dimensionality. Finally, the number of decodable patterns decreases when the positions of an ensemble of manifolds are correlated (as they become nearer to each other relative to their extent or their arrangement does not take advantage of all dimensions of the storage space); bottom row, this aspect of efficiency is measured by the correlation between manifold centroids.

framework (particularly when used to model biological vision). In the third chapter, I aim to design a more agnostic measure of informativeness by leveraging equivariance. In the fourth chapter I instead focus on a closer adherence to the constraints of biological learning: namely the ecological plausibility of "views" as temporal evolutions of images, and the constraint that optimization occurs in a cascade of isolated modules.

1.6.1 Posing SSL as Capacity Maximization

As discussed in Section 1.3, the capacity to store separable partitions of a dataset in a space with finite dimensionality is one notion of representational efficiency. Chung and colleagues extended Gardner's classic results on the perceptron capacity to store partitions of points by developing expressions to quantify the capacity to store *manifolds*, or sets of points that are self-similar in some sense (Chung et al., 2018; Gardner and Derrida, 1988). The theory revealed that,

even for manifolds with arbitrarily complicated shapes, the capacity could be calculated from three key geometric properties: the manifolds' "radius" and "dimensionality", as well as the correlation between the manifolds' positions. Each of these properties can be simply understood through a packing analogy. When packing a suitcase, you can fit many more size small shirts than you could size large; this notion of scale is measured by the raidus. However, you can fit more shirts of any given size when they are folded to lie flat than when crumpled haphazardly; this independent feature of size is captured by the manifold dimensionality. Finally, a full suitcase of course has articles spread equally throughout its extent, rather than piled up in one particular corner; this is measured by the correlation between the manifold's centroids. When manifold membership is defined by semantic category, the theory has been used as an analytical tool to help shed light on how neural computations in artificial networks and the ventral stream contribute to "category untangling" (Chou et al., 2024; Cohen et al., 2020; DiCarlo and Cox, 2007). In the second chapter, we ask whether defining neural manifolds to contain different views of a single image rather than exemplars from a specific category can give rise to a self-supervised learning principle based on manifold capacity. The primary difficulty is that computing the capacity for manifolds with arbitrary shapes is a computationally costly procedure that is prohibitively expensive for use in an optimization loop. However, when one approximates the geometries as elliptical each the geometric quantities discussed above are simple functions of the widths of each ellipse along its principal axes. A visualization of the characteristics is shown in Fig. 1.4 for three dimensional ellipsoids. Interestingly, this approach leads to a formulation of SSL that lies in-between sample-contrastive methods (which diagonalize the representation's Gram matrix) and dimension-contrastive methods (which diagonalizes the representation's covariance matrix), and centers the spectrum of a representation as the direct subject of optimization. The work described in Chapter 2 appeared previously as a publication in Advances in Neural Information Processing Systems (NeurIPS) 2023 (Yerxa et al., 2023).

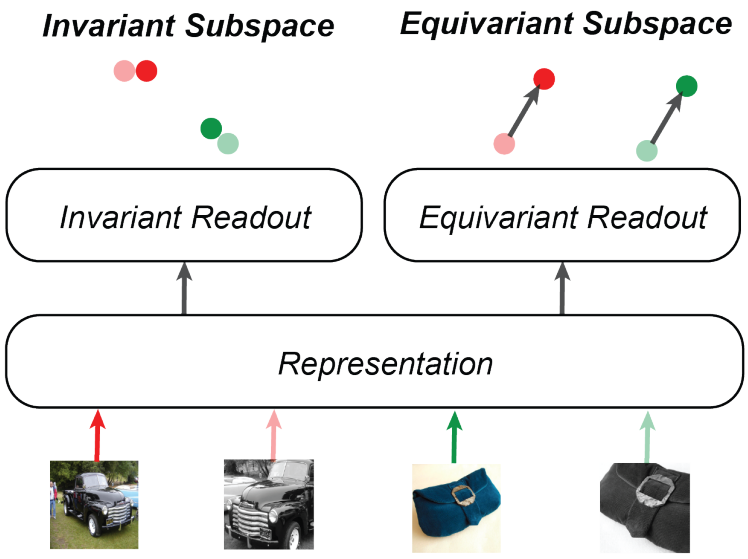

Figure 1.5: Dual Invariant-Equivariant Representations. Schematic for learning to encode content and transformation information jointly. Two images are subjected to the same transformation (i.e. flipping, cropping, and color distortion with the same parameters). We aim to learn a representation from which we can readout a transformation invariant space (where pairs of views of one image are nearby), and we can additionally readout an equivariant subspace (where the relative position of images related by the same transformation is the same).

1.6.2 Applying Efficient Coding to Transformations and Images Jointly

While invariance learning has proven to be a fruitful strategy both in terms of computer vision applications and as a framework for modeling biological vision, specifying what information a representation should and should not encode is in tension with the task-agnostic principle of efficient coding. To address this tension we introduced a method that aims to encode information *about transformations* alongside information that is *invariant to transformations*.

Concretely, we developed a "pair of pairs" data augmentation approach which allows us to encode that two images have been subjected to the same transformations in the same way that the standard SSL pipeline encodes that two views are derived from the same image. This allows us to define an equivariant task that requires that the same transformation, applied to distinct images, produces the same shift in representation space. The method is schematized in Fig. 1.5. Importantly this task can be learned alongside the invariance learning task, and produces net-

works that (1) preserve the organization of semantic categories characteristic of SSL, (2) display structured variability to image transformations that allows for a linear readout of transformation parameters, and (3) are systematically more predictive of neural responses deep in the macaque ventral stream (area IT). This work, described in Chapter 3, appeared previously as a publication in Advances in Neural Information Processing Systems (NeurIPS) 2024 (Yerxa et al., 2024b).

1.6.3 A Step Towards Biological Plausibility

In Chapter 4 we turn our attention to the "constraint side" of efficient coding. In particular, we are interested in the constraint of plausible optimization. While it is common practice to optimize all of the parameters of a hierarchical representation using the backpropagation of a global error signal, this learning algorithm is thought to be implausible for neural implementations because (1) it calls for exactly symmetric feedforward and feedback connections between areas and (2) the error signals for each parameter depend on the activities of units that are separated by multiple synaptic connections. We thus constrain learning to take place independently at each stage of computation by designing loss functions that operate on intermediate representations. Designing such losses is non-trivial, as the computational power of early stages is significantly lower than that of deeper stages, thus task difficulty must scale with depth in the network.

To find an ecologically plausible way of designing tasks with varying difficulty, we opt to take seriously the analogy between the "views" produced by τ in SSL schemes and the way that visual inputs transform over the course of time. In particular, the transformations between temporally nearby inputs are simpler than those that relate more distant inputs. Thus we can optimize the same manifold capacity loss proposed in Chapter 2 at each stage and modulate the task difficulty by growing the temporal extent of individual manifolds between successive stages. We demonstrate that our layerwise learning approach yields representations that are more predictive than those from full end-to-end learning up to area V4 of the visual cortex, and that these representations support object recognition in a manner more closely aligned with human behavior.

2 Learning Efficient Codes for Natural Images using Maximum Manifold Capacity Representations

2.1 Overview

The efficient coding hypothesis posits that sensory systems are adapted to the statistics of their inputs, maximizing mutual information between environmental signals and their representations, subject to biological constraints. While elegant, information theoretic quantities are notoriously difficult to measure or optimize, and most research on the hypothesis employs approximations, bounds, or substitutes (e.g., reconstruction error). A recently developed measure of coding efficiency, the "manifold capacity", quantifies the number of object categories that may be represented in a linearly separable fashion, but its calculation relies on a computationally intensive iterative procedure that precludes its use as an objective. Here, we simplify this measure to a form that facilitates direct optimization, use it to learn Maximum Manifold Capacity Representations (MMCRs), and demonstrate that these are competitive with state-of-the-art results on current self-supervised learning (SSL) recognition benchmarks. Empirical analyses reveal important differences between MMCRs and the representations learned by other SSL frameworks, and suggest a mechanism by which manifold compression gives rise to class separability. Finally, we

evaluate a set of SSL methods on a suite of neural predictivity benchmarks, and find MMCRs are highly competitive as models of the primate ventral stream.

2.2 Introduction

Biological visual systems learn complex representations of the world that support a wide range of cognitive behaviors, without relying on a large number of labeled examples. The efficient coding hypothesis (Barlow et al., 1961; Simoncelli and Olshausen, 2001) suggests that this is accomplished by adapting the sensory representation to the statistics of the input signal, so as to reduce redundancy or dimensionality. Visual signals have several clear sources of redundancy. They evolve slowly in time, since temporally adjacent inputs typically correspond to different views of the same scene, which in turn are usually more similar than views of distinct scenes. Moreover, the variations within individual scenes often correspond to variations in a small number of parameters, such as those controlling viewing and lighting conditions, and are thus inherently low dimensional. Many previous results have demonstrated how the computations of neural circuits can be seen as matched to such structures in naturalistic environments (Chung and Abbott, 2021; Fairhall et al., 2001; Laughlin, 1981; Simoncelli and Olshausen, 2001) Studies in various modalities have identified geometric structures in neural data that are associated with behavioral tasks (Bernardi et al., 2020; DiCarlo and Cox, 2007; Gallego et al., 2017; Hénaff et al., 2021; Nieh et al., 2021), and explored metrics for quantifying these structures.

The recent development of "manifold capacity theory" provides a more explicit connection between the geometry (size and dimensionality) of neural representations and their coding capacity (Chung et al., 2018). This theory has been used to evaluate efficiency of biological and artificial neural networks across modalities (Chung and Abbott, 2021; Dapello et al., 2021; Mamou et al., 2020; Stephenson et al., 2019). However, usage as a design principle for building model representations has not been explored.

Motivated by these observations, we seek to learn representations in which manifolds containing different views of the same scene are both compact and low-dimensional, while manifolds corresponding to distinct scene are maximally separated. Specifically:

- We develop a form of Manifold Capacity that can be used as an objective function for learning.
- We demonstrate that a Maximum Manifold Capacity Representation (MMCR) supports high-quality object recognition (matching the SoTA for self-supervised learning), when evaluated using the standard linear evaluation paradigm (i.e., applying an optimized linear classifier to the output of the self-supervised network) (Chen et al., 2020b).
- Through a analysis of internal representations and learning signals, we analyze the underlying mechanism responsible for the emergence of semantically relevant features from unsupervised objective functions.
- We validate the effectiveness of MMCR as a brain model by comparing its learned representations against neural data obtained from Macaque visual cortex.

Our work thus leverages normative goals of representational efficiency to obtain a novel model for visual representation that is both effective for recognition and consistent with neural responses in the primate ventral stream.

2.2.1 Related Work

Geometry of Neural Representations. Previous work has sought to characterize how representational geometry, often measured through spectral quantities like the participation ratio (the squared ratio of the l_1 and l_2 norms of the eigenvalues of the covariance matrix), shapes different aspects of performance on downstream tasks (Fusi et al., 2016). Elmoznino and Bonner (2024) found that high dimensionality in ANN representations was associated with ability to both

predict neural data and generalize to unseen categories. Stringer et al. (2019) observed that the spectrum of the representation of natural images in mouse cortex follows a power law with a decay coefficient near 1, and Agrawal et al. (2022) report that (in artifical representations) the proximity of the spectral decay coefficient to one is an effective predictor of how well a representation will generalize to downstream tasks.

Self-Supervised Learning. Our methodology is related to (and inspired by) recent advances in contrastive self-supervised representation learning (SSL), but has a distinctly different motivation and formulation. Many recent frameworks craft objectives that are designed to maximize the mutual information between representations of different views of the same object (Bachman et al., 2019; Chen et al., 2020b; Oord et al., 2018; Tian et al., 2020)). However, estimating mutual information in high dimensional feature spaces is difficult (Belghazi et al., 2018), and furthermore it is not clear that closer approximation of mutual information in the objective yields improved representations (Wang and Isola, 2020) By contrast, capacity measures developed in spin glass theory (Abbaras et al., 2020; Gardner and Derrida, 1988) are derived in the "large N (thermodynamic) limit"and thus are intended to operate in the regime of large ambient dimension (Bahri et al., 2020; Chung et al., 2018). We examine whether one such measure, which until now had been used only to evaluate representation quality, is also useful as an objective function for SSL.

Many SSL methods minimize the distance between representations of different augmented views of the same image while employing constraints to prevent collapse to trivial solutions (e.g., repulsion of negative pairs (Chen et al., 2020b), or feature space whitening (Bardes et al., 2022; Ermolov et al., 2021; Zbontar et al., 2021)). The limitations of using a single pairwise distance comparison have been demonstrated, notably in the development of the "multi-crop" strategy implemented in SwAV (Caron et al., 2020) and in the contrastive multiview coding approach Tian et al. (2020). Our approach is based on the assumption that different views of an image form a continuous manifold that we aim to compress. We characterize each set of image views with the spectrum of singular values of their representations, using the nuclear norm as a combined

measure of the manifold size and dimensionality.

The nuclear norm has been previously used to infer or induce low rank structure in the representation of data (Hénaff et al., 2015; Lezama et al., 2018; Wang et al., 2022). In particular, Wang et al. (2022) use it as a regularizer to supplement an InfoNCE loss. Our approach represents a more radical departure from the traditional InfoNCE loss, as we detail below. Rather than pair a low-rank prior with a logistic regression-based likelihood, we make the more symmetric choice of employing a *high rank* likelihood. This allows the objective to explicitly discourage dimensional collapse, a well known issue in SSL (Jing et al., 2022).

2.3 Maximum manifold capacity representations

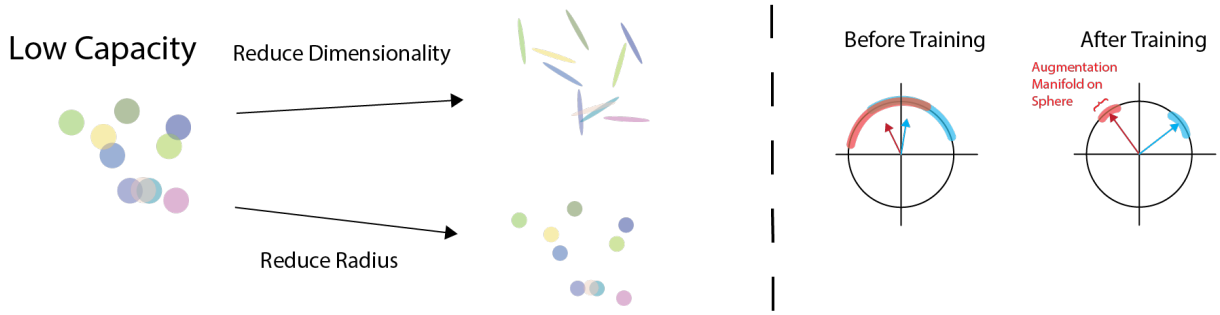

Figure 2.1: Two dimensional illustrations of high and low capacity representations. Left: the capacity (linear separability) of a random set of spherical regions can be improved, either by reducing their radii (while maintaining their dimensionalities), or by reducing their dimensionalities (while maintaining their average radii). Right: the objective proposed in this paper aims to minimize the nuclear norm (equal to the product of radius and sqrt dimensionality) of normalized data vectors (ie., lying on the unit sphere). Before training, the manifolds have a large extent and thus the matrix of their corresponding centroid vectors has low nuclear norm. After training, the capacity is increased: The manifolds are compressed and repelled from each other, resulting in centroid matrix with larger nuclear norm and lower similarity.

2.3.1 Manifold Capacity Theory

Consider a set of P manifolds embedded in a feature space of dimensionality D, each assigned a class label. Manifold capacity theory is concerned with the question: what is the largest value of $\frac{P}{D}$ such that there exists (with high probability) a hyperplane separating a random dichotomy (Cover, 1965; Gardner, 1988)? Recent theoretical work has demonstrated that there exists a critical value, dubbed the manifold capacity α_C, such that when $\frac{P}{D} < \alpha_C$ the probability of finding a separating hyperplane is approximately 1.0, and when $\frac{P}{D} > \alpha_C$ the probability is approximately 0.0 (Chung et al., 2018). The capacity α_C can be accurately predicted from three key quantities: (1) the manifold radius R_M, which measures the size of the manifold relative to the distance of its centroid from the origin, (2) the manifold dimensionality D_M which quanitifies the number of dimensions along which a manifold has significant extent, and (3) the correlation of the manifold centroids. When the centroid correlation is low the manifold capacity can be approximated by $\phi(R_M\sqrt{D_M})$ where $\phi(\cdot)$ is a monotonically decreasing function.

For manifolds of arbitrary geometry, the radius and dimensionality may be computed using an iterative process that alternates between determining the set of "anchor points" on each manifold that are relevant for the classification problem, and computing the statistics of random projections of these anchor points (Cohen et al., 2020). This process is both computationally costly and non-differentiable, and therefore unsuitable for use as an objective function. For more detail on the general theory see Appendix A.3. However, if the manifolds are assumed to be elliptical in shape, then both radius and dimensionality may be expressed analytically:

$$R_M = \sqrt{\textstyle\sum_i \lambda_i^2}, \qquad D_M = \frac{(\sum_i \lambda_i)^2}{\sum_i \lambda_i^2}, \tag{2.1}$$

where the λ_i^2 are the eigenvalues of the covariance matrix of the manifold. For comparison, when computing these values for a set of 100 128-D manifolds with 100 points sampled from each, the use of the analytical expression is approximately 500 times faster (in "wall-clock time") than the

general iterative procedure.

Using these definitions for manifold radius and dimensionality we can write the capacity as $\alpha_C \approx \phi(\sum_i \sigma_i)$ where σ_i are the singular values of a matrix containing points on the manifold (equivalently, the square roots of the eigenvalues of the covariance matrix). In this form, the sum is the L_1 norm of the singular values, known as the *Nuclear Norm* of the matrix. When used as an objective function, this measure favors sparse solutions (i.e., those with a small number of non-zero singular values) corresponding to low dimensionality. It is worth comparing this objective to another natural candidate for quantifying size: the determinant of the covariance matrix. The determinant is equal to the product of the eigenvalues (which captures the squared volume of the corresponding ellipsoid), but lacks the preference for lower dimensionality that comes with the Nuclear Norm. Specifically, since the determinant is zero whenever one (or more) eigenvalue is zero, it cannot distinguish zero-volume manifolds of different dimensionality. Lossy coding rate (entropy) has also been used as a measure of compactness (Yu et al., 2020), which simplifies to the log determinant of the covariance matrix under a Gaussian model (Ma et al., 2007). In this case, the identity matrix is added to a multiple of the feature covariance matrix before evaluating the determinant, which solves the dimensionality issue described above.

2.3.2 Optimizing Manifold Capacity

Now we construct an SSL objective function based on manifold capacity. For each input image (notated as a vector $\mathbf{x}_b \in \mathbb{R}^{\mathbb{D}}$) we generate k samples from the corresponding manifold by applying a set of random augmentations (each drawn from the same distribution), yielding manifold sample matrix $\tilde{\boldsymbol{X}}_b \in \mathbb{R}^{D \times k}$. Each augmented image is transformed by a Deep Neural Network, which computes nonlinear function $f(\mathbf{x}_b; \theta)$ parameterized by θ, and the d-dimensional responses are projected onto the unit sphere yielding manifold response matrix $\boldsymbol{Z}_b \in \mathbb{R}^{d \times k}$. The centroid $\boldsymbol{c}_b$ is approximated by averaging across the columns (response vectors). For a set of images $\{\mathbf{x}_1, ..., \mathbf{x}_B\}$ we compute normalized response matrices $\{\boldsymbol{Z}_1, ..., \boldsymbol{Z}_B\}$ and assemble their corresponding cen-

troids into matrix $C \in \mathbb{R}^{d \times B}$.

Given the responses and their centroids, the MMCR loss function can be written simply:

$$\mathcal{L} = -||C||_*, \tag{2.2}$$

where $||\cdot||_*$ indicates the nuclear norm. The loss explicitly maximize the extent of the "centroid manifold", which interestingly is sufficient to learn a useful representation. Concretely, maximizing $||C||_*$ implicitly minimizes the extent of each individual object manifold as measured by $||Z_b||_*$. We build intuition for this effect below.

Compression by Maximizing Centroid Nuclear Norm Alone. Each centroid vector is a mean of unit vectors, and thus has a norm that is linearly related to the average cosine similarity of those unit vectors. Specifically,

$$||\boldsymbol{c}_b||^2 = \frac{1}{K} + \frac{2}{K^2} \sum_{k=1}^{K} \sum_{l=1}^{k-1} \boldsymbol{z}_{b,k}^T \boldsymbol{z}_{b,l} \tag{2.3}$$

Here $\boldsymbol{z}_{b,i}$ denotes the representation of the i^{th} augmentation of $\boldsymbol{x}_b$. We can gain further insight by considering how the distribution of singular vectors of a matrix depends on the norms and pairwise similarities of the constituent column vectors. While no closed form solution exists for the singular values of an arbitrary matrix, the case where the matrix is composed of two column vectors can provide useful intuition. If $C = [\boldsymbol{c}_1, \boldsymbol{c}_2]$, $Z_1 = [\boldsymbol{z}_{1,1}, \boldsymbol{z}_{1,2}]$, $Z_2 = [\boldsymbol{z}_{2,1}, \boldsymbol{z}_{2,2}]$, the singular values of C and Z_i are:

$$\boldsymbol{\sigma}(C) = \sqrt{\frac{||\boldsymbol{c}_1||^2 + ||\boldsymbol{c}_2||^2 \pm ((||\boldsymbol{c}_1||^2 - ||\boldsymbol{c}_2||^2)^2 + 4(\boldsymbol{c}_1^T \boldsymbol{c}_2)^2)^{1/2}}{2}}, \tag{2.4}$$

$$\boldsymbol{\sigma}(Z_i) = \sqrt{1 \pm \boldsymbol{z}_{i,1}^T \boldsymbol{z}_{i,2}}. \tag{2.5}$$

So, $||\boldsymbol{\sigma}(C)||_1 = ||C||_*$ is maximized when the centroid vectors have maximal norms (bounded

above by 1, since they are the centroids of unit vectors), and are orthogonal to each other. As we saw above the centroid norm is a linear function of within-manifold similarity. Similarly, $||\boldsymbol{\sigma}(Z_i)||_1 = ||Z_i||_*$ is minimized when the within-manifold similarity is maximal. Thus the single term $||C||_*$ encapsulates both of the key ingredients of a contrastive learning framework, and we will demonstrate below that simply maximizing $||C||_*$ is sufficient to learn a useful representation. This is because the compressive role of "positives" in contrastive learning is carried out by forming the centroid vectors, so the objective is not positive-free. For example, if only a single view is used the objective lacks a compressive component and fails to produce a useful representation. In Appendix A.6 we demonstrate empirically that this implicit form effectively reduces $||Z_b||_*$ by comparing to the case where $||Z_b||_*$ is minimized explicitly. So, all three factors which determine the manifold capacity (radius, dimensionality, and centroid correlations) can be succinctly expressed in an objective function with a single term, $-||C||_*$.

Computational Complexity. Evaluating the loss for our method involves computing a singular value decomposition of $C \in \mathbb{R}^{d \times B}$ which has complexity $O(Bd \times \min(B, d))$, where B is the batch size and d is the dimensionality of the output. By comparison, contrastive methods that compute all pairwise distances in a batch have complexity $O(B^2 d)$ and non-contrastive methods that involve regularizing the covariance structure have complexity $O(Bd^2)$. Additionally, the complexity of our method is constant with respect to the number of views used (though the feature extraction phase is linear in the number of views), while pairwise similarity metrics have complexity that is quadratic in the number of views. It is also worth noting that doing implicit compression by maximizing $||C||_*$ offers an advantage in computational complexity relative to explicit compression. This is because evaluating a term such as $\sum_{b=1}^{B} ||Z_b||_*$ has computational complexity of $O(B^2 d \times \min(B, d))$.

2.3.3 Conditions for Optimal Embeddings

Recently HaoChen et al. (2021) developed a framework based on spectral decomposition of the "population augmentation graph", which provides theoretical guarantees for the performance of self-supervised learning on downstream tasks under linear probing. This work was extended to provide insights into various other SSL objectives by Balestriero and LeCun (2022), and we show below that leveraging this approach can lead to explicit conditions for the optimality of representation under our proposed objective as well.

Given a dataset $\boldsymbol{X'} = [\boldsymbol{x}_1, ..., \boldsymbol{x}_N]^T \in \mathbb{R}^{N \times D'}$ we construct a new dataset by creating k randomly augmented views of the original data, $\boldsymbol{X} = [\text{view}_1(\boldsymbol{X'}), ..., \text{view}_k(\boldsymbol{X'})] \in \mathbb{R}^{Nk \times D}$. The advantage of doing so is that we can now leverage the knowledge that different views of the same underlying datapoint are *semantically related*. We can express this notion of similarity in the symmetric matrix $\boldsymbol{G} \in \{0, 1\}^{Nk \times Nk}$ with $\boldsymbol{G}_{ij} = 1$ if augmented datapoints i and j are semantically related (and $\boldsymbol{G}_{ii} = 1$ as any datapoint is related to itself). We can normalize $\boldsymbol{G}$ such that its rows and columns sum to 1 (so rows of $\boldsymbol{G}$ are k-sparse with nonzero entries equal to $1/k$).

Now let $\boldsymbol{Z} \in \mathbb{R}^{Nk \times d}$ be an embedding of the augmented dataset. Then we have $\boldsymbol{GZ} = [\boldsymbol{C}, ..., \boldsymbol{C}]^T$ where $\boldsymbol{C}$ is the matrix of centroid vectors introduced above, and the number of repetitions of $\boldsymbol{C}$ is k. Then because $\boldsymbol{\sigma}([\boldsymbol{C}, ..., \boldsymbol{C}]) = \sqrt{k}\boldsymbol{\sigma}(\boldsymbol{C})$ we can write MMCR loss function as,

$$\mathcal{L} = -||\boldsymbol{GZ}||_* \tag{2.6}$$

This connection allows us to make the following statements about the optimal embeddings $\boldsymbol{Z}$ under our loss, which we prove in Appendix A.1:

Theorem: Under the proposed loss, the left singular vectors of an optimal embedding, $\boldsymbol{Z}^*$, are the eigenvectors of $\boldsymbol{G}$, and the singular values of $\boldsymbol{Z}^*$ are proportional to the top d eigenvalues of $\boldsymbol{G}$.

2.4 Results

Architecture. For all experiments we use ResNet-50 (He et al., 2016) as a backbone architecture (for variants trained on CIFAR we removed max pooling layers). Following Chen et al. (2020b), we append a small perceptron to the output of the average pooling layer of the ResNet so that $z_i = g(h(x_i))$, where h is the ResNet and g is the MLP. For ImageNet-1k/100 we used an MLP with dimensions $[8192, 8192, 512]$ and for smaller datasets we used $[512, 128]$.

Optimization. We employ the set of augmentations proposed in (Grill et al., 2020). For ImageNet we used the LARS optimizer with a learning rate of 4.8, linear warmup during the first 10 epochs and cosine decay thereafter with a batchsize of 2048, and pre-train for 100 epochs. Note that though we report results using a default batch size of 2048, a batch size as small as 256 can be used to obtain reasonable results (1.2% reduction compared to batch size 2048 – see Appendix A.10 for a sweep over batch size). We additionally employ a momentum encoder for ImageNet pre-training (all views are fed through an online network and an additional network whose parameters are a slowly moving average of the online network and all embeddings of each image are averaged to form centroids). We found the use of a momentum encoder provided a small advantage in terms of downstream classification performance, see Appendix A.15 for this ablation. For smaller CIFAR-10 we used a smaller batch size, many more views (40), and the Adam optimizer with fixed learning rate. See Appendix A.4 for exact details.

2.4.1 Transfer to Downstream Tasks

We used a standard linear evaluation technique, freezing the parameters of the encoding network and training a linear classifier with supervision (Chen et al., 2020b), to verify that our method extracts semantically relevant features from the data. We also perform semi-supervised evaluation, where all model parameters are fine tuned using a small number of labelled examples, and also check whether the learned features generalize to three out-of-distribution datasets:

Flowers-102, Food-101, and the Describable Textures Dataset (Bossard et al., 2014; Cimpoi et al., 2014; Nilsback and Zisserman, 2008) . The results are summarized in Table 2.1 and details of the training of the linear classifier are provided in Appendix A.7. Finally we also evaluate our best model in and each of the baselines on object detection on the VOC07 dataset. We followed (He et al., 2020; Zbontar et al., 2021), fine tuning the representation networks for detection with a Faster R-CNN head and C-4 backbone using the 1x schedule. MMCR achieved a mean average precision (mAP) of 54.6 and the baseline method performance ranged from 53.1 to 56.0, demonstrating that though our framework was inspired by a theory of classification the learned features do generalize to other vision tasks. See Appendix A.13 for detailed results.

Note that all included models were trained using the same backbone architecture (ResNet-50), dataset (ImageNet-1k), and number of pretraining epochs (100; we briefly explore the impact of longer pretraining in Appendix A.16). The results for networks pretrained on smaller datasets can be found in the Appendix A.9.

2.4.2 Analyses of Learned Representations

We next conduct a series of experiment to elucidate the differences between representations learned with different SSL procedures, and suggest a mechanism by which augmentation manifold compression gives rise to class separability. To reduce the computational requirements, these analyses are carried out on models trained on the CIFAR-10 dataset.

Mean Field Theory Manifold Capacity. In Fig. 2.2 we show that our representation, which is optimized using an objective that assumes elliptical manifold geometry, nevertheless yields representations with high values of the more general mean field manifold capacity (relative to baseline methods). For completeness we also analyzed the geometries of class manifolds, whose points are the representations of different examples from the same class. This analysis provided further evidence that learning to maximize augmentation manifold capacity compresses and separates class manifolds, leading to a useful representation. Interestingly MMCRs seem to use a

Table 2.1: MMCR Transfer Learning Evaluations. Evaluation of learned features on downstream classification tasks. The leftmost columns show results for the standard frozen-linear evaluation procedure on ImageNet (IN). Results for most methods in this setting are taken from Ozsoy et al. (2022) except for SwAV which is taken from the original paper (Caron et al., 2020). Columns 2 and 3 show semi-supervised evaluation on ImageNet (fine-tuning on 1% and 10% of labels). The results for this setting for VICReg and CorInfo Max are copied from Ozsoy et al. (2022). The final three columns show frozen-linear evaluation on other datasets. We evaluated models for which pretrained weights in the 100 epoch setting were available online; MoCo, Barlow Twins and BYOL were taken from solo-learn Da Costa et al. (2022) (https://github.com/vturrisi/solo-learn), while SwAV and SimCLR were taken from VISSL Goyal et al. (2021) (https://github.com/facebookresearch/vissl/blob/main/MODEL_ZOO.md). For all evaluations we performed we report the mean and standard deviation over 3 evaluation runs).

Method	IN	1%	10%	Food-101	Flowers-102	DTD
W-MSE (Ermolov et al., 2021)	69.4	-	-	-	-	-
NNCLR (Dwibedi et al., 2021)	69.4	-	-	-	-	-
SwAV (Caron et al., 2020)	64.6	-	-	-	-	-
SimSiam (Chen and He, 2021)	68.1	-	-	-	-	-
CorInfoMax (Ozsoy et al., 2022)	69.1	44.9	64.3	-	-	-
VICReg (Bardes et al., 2022)	68.7	44.8	62.2	-	-	-
BarlowTwins (Zbontar et al., 2021)	68.7	45.1 ± .12	61.7 ± .03	69.8 ± .03	86.2 ± .22	67.7 ± .20
SimCLR (Chen et al., 2020b)	66.5	42.6 ± .02	61.6 ± .09	67.2 ± .24	84.0 ± .19	64.8 ± .07
BYOL (Grill et al., 2020)	69.3	49.8 ± .05	65.0 ± .05	70.6 ± .1	84.8 ± .43	67.6 ± .14
MoCo-V2 (Chen et al., 2020c)	67.4	43.4 ± .07	63.2 ± .07	68.6 ± .03	82.4 ± .27	66.6 ± .16
SwAV (Caron et al., 2020)	**72.1**	49.8 ± .09	66.9 ± .05	72.1 ± .08	89.3 ± .1	68.2 ± .18
MMCR (2 views)	69.5 ± .02	46.6 ± .02	63.9 ± .02	72.0 ± .02	90.0 ± .24	68.5 ± .07
MMCR (4 views)	71.5 ± .04	49.4 ± .05	66.0 ± .05	73.2 ± .07	91.0 ± .04	**70.4** ± .46
MMCR (8 views)	**72.1** ± .04	**51.0** ± .02	**67.7** ± .11	**73.6** ± .04	**91.4** ± .07	**70.0** ± .24

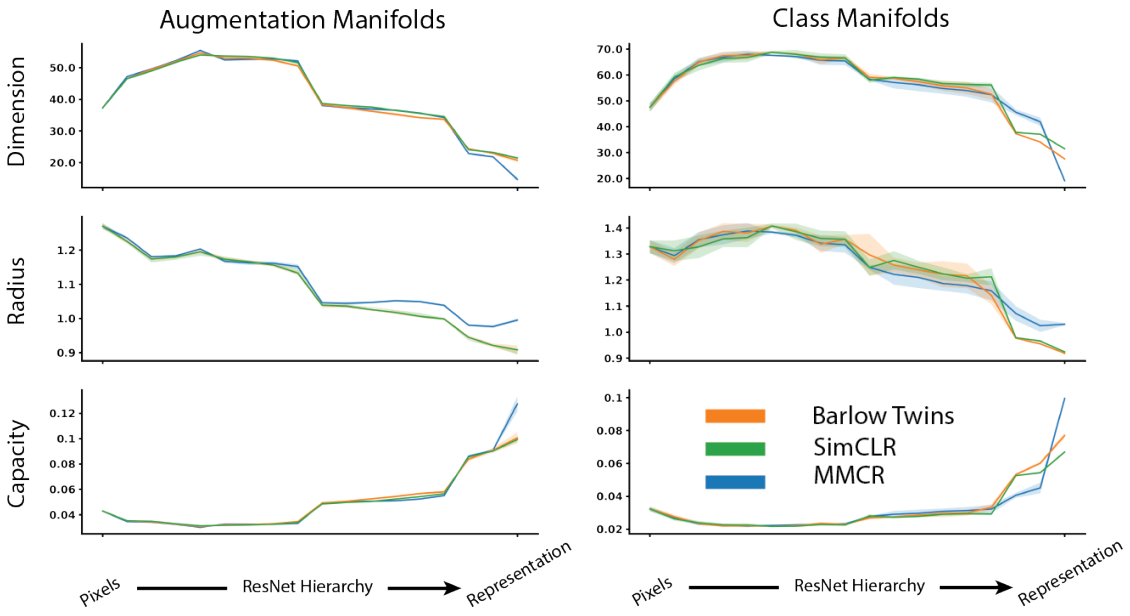

Figure 2.2: Mean field Manifold Capacity analysis. Manifold dimensionality (top row), radius (middle row), and capacity (bottom row), as a function of stage in the representational hierarchy (from pixel inputs to output of the encoder/learned representation). Shaded regions indicate 95% confidence intervals around the mean (analysis was conducted with 5 different random samples from the dataset, see Appendix A.5).

different strategy than the baseline methods to increase the capacity, yieldingclass/augmentation manifolds with larger radii, but lower dimensionality (Fig. 2.2) These geometrical differences do not emerge until the tail end of the hierarchy, suggesting early layers of each network are carrying out stereotyped transformations and loss function induced specialization does not emerge until later layers.

Emergence of neural manifolds via gradient coherence. We hypothesize that class separability in MMCRs arises because augmentation manifolds corresponding to examples from the same class are optimally compressed by more similar transformations than those stemming from distinct classes. To investigate this empirically, we evaluate the gradient of the objective function for inputs belonging to the same class. We can then check whether gradients obtained from (distinct) batches of the same class are more similar to each other than those obtained from different classes, which would suggest that the strategy for compressing augmentation manifolds from the

same class are relatively similar to each other. Fig. 2.3 demonstrates that this is the case: within class gradient coherence, as measured by cosine similarity, is consistently higher than across class coherence across both training epochs and model hierarchy.

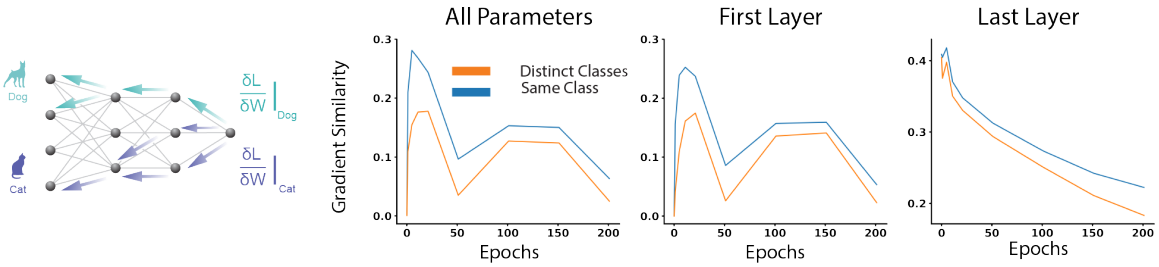

Figure 2.3: Intra- and Inter-class Gradient Similarity. Cosine similarity of gradients for pairs of single-class batches. We plot the mean pairwise similarity for pairs of gradients for different subsets of the model parameters (all parameters, and the first and last linear operators) obtained from single-class-batches coming from the same or distinct classes over the course of training. Because a large number of bathces were used 95% confidence intervals about these means are too small to be visible. To the left is a visualization of the fact that single-class gradients flow backward through the model in more similar directions.

Manifold subspace alignment. Within-class gradient coherence constitutes a plausible mechanistic explanation for the emergence of class separability, but it does not explain why members of the same class share similar compression strategies. To explore this question we examine the geometric properties of augmentation manifolds in the pixel domain. Here we observe small but measurable differences between the distributions of within-class similarity and across-class similarity, as demonstrated in the top row of Fig. 2.4. The subtle difference in the geometric properties of augmentation manifolds in the pixel domain in turn leads to the increased gradient coherence observed above, which leads to a representation that rearranges and reshapes augmentation manifolds from the same class in a similar fashion (bottom row of Fig. 2.4), thus allowing better linear separation of classes. Not only are centroids of same-class-manifolds in more similar regions of the representation space than those coming from distinct classes (Fig. 2.4 third column bottom row) but additionally same-class-manifolds have more similar shapes to each other (Fig. 2.4 bottom row columns 1 and 2 show same-class-manifolds occupy subspaces

with lower relative angles and share more variance).

We next ask how the representation learned with the MMCR objective differs from those optimized for other self-supervised loss functions. While MMCR encourages centroids to be orthogonal, the InfoNCE loss (Chen et al., 2020b) encourages negative pairs to be as dissimilar as possible, which is achieved when they lie in opposite regions of the *same* subspace . The Barlow Twins (Zbontar et al., 2021) loss is not an explicit function of feature vector similarities, but instead encourages individual features to be correlated and distinct features to be uncorrelated, across the batch dimension. Fig. 2.5 shows that these intuitions are borne out empirically: the MMCR representation produces augmentation manifold centroids that are significantly more orthogonal to each other than the two baseline methods.

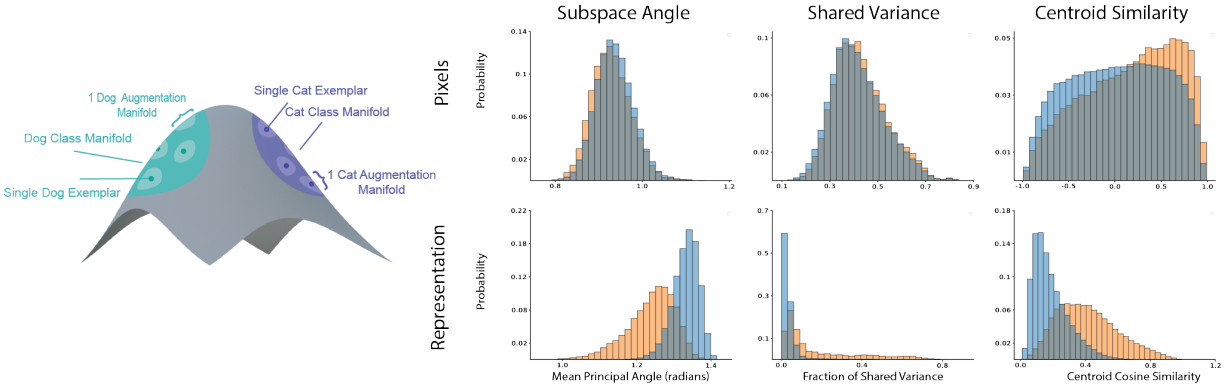

Figure 2.4: Intra- and Inter-class Representational Similarity. The distributions of various similarity metrics for augmentation manifolds from the same (orange) and distinct (blue) classes. These are shown for both the input images (top row), and the learned representation (bottom row). Left: schematic illustration of the exemplar-augmentation manifold-class manifold structure of the learned representation.

2.4.3 Biological relevance

Neuroscience has provided motivation for many of the developments in artifical neural networks, and it is of interest to ask whether SSL networks can characterize the measured behaviors of neurons in biological visual systems. As a simple test, Table 2.2 shows performance of our model compared with five other SSL models on the *BrainScore* repository (Freeman et al., 2013;

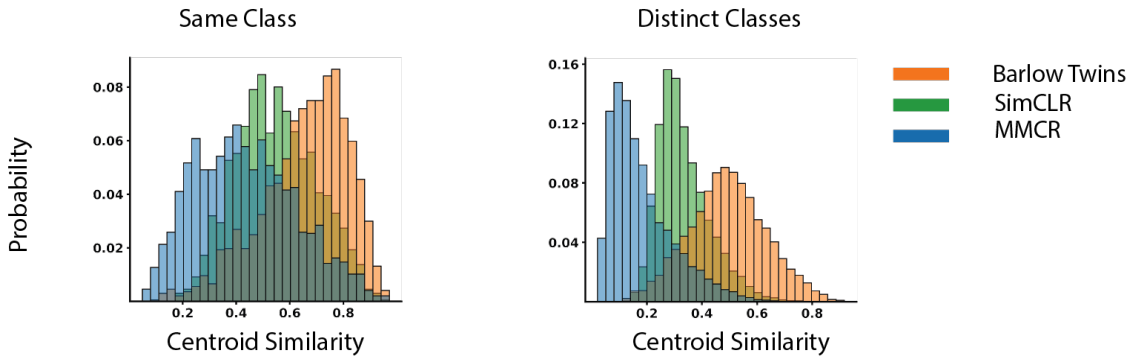

Figure 2.5: Intra- and Inter-class Centroid Similarity. Cosine similarities of centroids for models trained according to different SSL objectives. The left panel shows the distribution of cosine similarities for centroids of augmentation manifolds for examples of the same class, while the right shows the same distribution for examples from distinct classes. Note that because we are analyzing the outputs of the ResNet backbone (which are rectified), the minimum possible cosine similarity is 0.

Majaj et al., 2015; Marques et al., 2021; Schrimpf et al., 2018). We find that MMCR achieves the highest performance in explaining neural data from primate visual areas V2 and V4, second-highest for V1 and IT, and is the most or second most predictive for 8 out of the 11 individual datasets (see Appendix A.11).

In addition, we examine general response spectral characteristics that have been recently described for neural populations. In particular, Stringer et al. (2019) reported that the eigenspectrum of the covariance matrix of population activity in visual area V1 follows a power law decay with a decay coefficient of approximately 1 ($\lambda_n \propto n^{-\alpha}$ with $\alpha \approx 1$ where λ_n is the n^{th} eigenvalues). Subsequent studies in artifical networks have found that such a decay spectrum is associated with increased robustness to adversarial perturbations and favorable generalization properties (Agrawal et al., 2022; Nassar et al., 2020). Additionally, several recent works have investigated the connection between representational dimensionality and neural predictivity (Schaeffer et al., 2022; Tuckute et al., 2022). In particular, Elmoznino and Bonner (2024) report that high intrinsic dimensionality (as measured with the participation ratio of the representation covariance) is correlated with stronger ability to predict neural activity. Table 2.2 provides values for the participation ratio of each representation over the ImageNet validation set, as well as the decay

Table 2.2: Neural prediction and spectral properties of self-supervised models. First four columns provide average BrainScore correlation values (Schrimpf et al., 2018) for neurons recorded in four areas of primate ventral stream. Last two provide participation ratio (PR) and spectral decay coefficients (α), estimated for 10 bootstrapped samples of the features for the ImageNet validation set. When estimating the spectral decay coefficient we omitted the tails (where the power decays more rapidly), see A.12 for exact details. Entries indicate mean and standard error of the mean, and boldface indicates best performance within standard error. MMCR results are for a model trained with 8 views.

Model	V1	V2	V4	IT	PR	α
MMCR	**.494** ± .006	**.311** ± .005	**.481** ± .005	.416 ± .003	279.2 ± .3	1.04 ± 1e-4
SimCLR	**.500** ± .007	.288 ± .007	.475 ± .004	**.420** ± .003	124.6 ± .1	1.34 ± 3e-4
BYOL	**.500** ± .006	.291 ± .007	**.477** ± .005	.404 ± .003	248.0 ± .4	1.35 ± 1e-4
MoCo	**.499** ± .007	.293 ± .006	**.477** ± .005	**.417** ± .003	147.7 ± .2	1.44 ± 3e-4
Barlow	**.498** ± .007	.293 ± .008	**.477** ± .005	.404 ± .003	252.2 ± .2	1.21 ± 3e-4
SwAV	.488 ± .007	.296 ± .009	.463 ± .004	.398 ± .003	163.9 ± .2	1.15 ± 3e-4

coefficient of the covariance spectrum (see Appendix A.12 for more details on each experiment). We see that the MMCR has the highest participation ratio (dimensionality) - note that this differs from the quantity optimized in the objective function, which lies in the embedding space. In addition, MMCR also also yields features with a decay coefficient that is nearest to one.

We find that in this controlled setting each model explains a very similar fraction of neural variance through linear regression. This is consistent with recent and concurrent works (Conwell et al., 2022; Saxe et al., 2021) which have identified this lack of ability to discriminate between alternative models as a weakness of the dominant paradigm used for model-to-brain comparisons. However, our results demonstrate that different SSL algorithms produce representations with meaningfully different geometries (as evidenced by the large spread in the spectral properties such as the participation ratio and decay coefficient). This suggests the need for the development of new metrics for comparing models to data, such as geometrical measures, that capture these important differences between candidate models.

2.5 Discussion

We have presented a novel self-supervised learning algorithm inspired by manifold capacity theory. Many existing SSL methods can be categorized as either "contrastive" or "non-contrastive" depending on whether they avoid collapse by imposing constraints on the embedding gram or covariance matrix, respectively. Our framework strikes a compromise, optimizing the singular values of the embedding matrix itself. By directly optimizing this population level feature (the spectrum), we are able to encourage both alignment and uniformity (Wang and Isola, 2020) with a single-term objective. Additionally, this formulation circumvents the need for making a large number of pairwise comparisons, either between instances or dimensions. As a result learning MMCRs is efficient, requiring neither large batch size nor large embedding dimension. Finally, the method extends naturally to the multi-view case, offering improved performance with minimal increases in computational cost.

Our formulation approximates manifold geometries as elliptical, reducing computational requirements while still yielding a useful learning signal and networks with high manifold capacity. Specifically, we were able to leverage manifold capacity analysis in its full generality to gain insight into the geometry of MMCR networks after training. Further research could explore objectives based on more modest reductions of mean field manifold capacity that capture non-elliptical structure. Intriguingly, our method produces augmentation and class manifolds with lower dimensionality but larger radius than either Barlow Twins or SimCLR (Fig. 2.2). We do not understand why this is the case, but the differences indicate that capacity analysis can provide a useful tool for elucidating the different encoding strategies encouraged by various SSL paradigms.

Finally we investigated two recently proposed theories on ideal spectral properties for neural representations. For our considered set of models a spectral decay coefficient near 1 was associated with better performance on the within-distribution task and generalization to unseen

datasets (MMCR, SwAV, and Barlow were the top three models for each of the out-of-distribution classification tasks), a finding which is broadly aligned with both the empirical and theoretical findings of Agrawal et al. (2022). However, we also found that high dimensionality did not always correspond to strong neural predictivity: Despite having the lowest dimensionality, SimCLR performed strongly in terms of neural predictivity. This implies, perhaps unsurprisingly, that global dimensionality alone is not sufficient to explain the response properties of neurons in the primate ventral stream. In fact our experiments add to a growing body of work on the need for complementary approaches to linear predictivity for discriminating between candidate models of the visual system (Conwell et al., 2022). Happily the field is already moving to address this limitation, for instance concurrent work (Canatar et al., 2023) finds that decomposing neural predictivity error into distinct modes can yield insights into how different models fit different aspects of the neural data (even if they yield similar overall predictivity).

One promising direction for improving the quality of artificial networks as models of neural computations is to incorporate the constraints associated with biologically plausible learning (i.e. the need for local learning rules, Illing et al. (2021)). A complementary direction is to better align training data diets with ecological inputs. For example, temporal invariance rather than augmentation invariance seems a more plausible objective for optimization by a biological system (Aubret et al., 2023; Parthasarathy et al., 2023; Wiskott and Sejnowski, 2002). We speculate that a variant of the MMCR objective that operates over time may be well-suited to a neural circuit implementation as its computation would only require mechanisms for tracking (1) short timescale temporal means (to form centroids) and (2) the singular values of the population activity over longer timescales.

Neuroscience is brimming with newly collected datasets recorded from ever larger populations of neurons, and there is a growing set of methods that aim to make sense of these measurements through a normative lens. Here we have demonstrated that one such technique that has proven useful for gleaning insights from neural data (Froudarakis et al., 2020; Paraouty et al.,

2023; Yao et al., 2023) can be reformulated for use as an objective function to learn a useful abstract representation of images. Future work should aim to close the loop between modeling and analysis by using these learned models to generate experimentally testable predictions and constrain new experimental designs.

3 Contrastive-Equivariant Self-Supervised Learning Improves Alignment with Primate Visual Area IT

3.1 Overview

Models trained with self-supervised learning objectives have recently matched or surpassed models trained with traditional supervised object recognition in their ability to predict neural responses of object-selective neurons in the primate visual system. A self-supervised learning objective is arguably a more biologically plausible organizing principle, as the optimization does not require a large number of labeled examples. However, typical self-supervised objectives may result in network representations that are overly invariant to changes in the input. Here, we show that a representation with structured variability to input transformations is better aligned with known features of visual perception and neural computation. We introduce a novel framework for converting standard invariant SSL losses into “contrastive-equivariant" versions that encourage preservation of input transformations without supervised access to the transformation parameters. We demonstrate that our proposed method systematically increases the ability

of models to predict responses in macaque inferior temporal cortex. Our results demonstrate the promise of incorporating known features of neural computation into task-optimization for building better models of visual cortex.

3.2 Introduction

In the past decade, task-optimized deep neural networks (DNNs) have been used to predict responses of object-selective neurons in primates to natural image stimuli (Schrimpf et al., 2020; Willeke et al., 2023; Yamins et al., 2014). Such networks have a pronounced advantage over more traditional models for explaining responses in deeper areas with more abstract representations, such as inferior temporal cortex (IT). This observation naturally leads to the hypothesis that task optimization can provide a normative account for IT neuron tuning properties: late-stage visual representations are shaped by the need to perform ecologically relevant tasks.

However, the task that initially led to these advances was that of supervised object classification, a specific task that relies on an implausibly large number of labeled examples (Lindsay, 2021). More recently, computer vision has undergone a "self-supervised learning" (SSL) revolution. A variety of methods have been proposed to learn representations that match or surpass supervised training on multiple tasks by deriving sources of supervision from the data itself rather than relying on human annotations. For example, many popular SSL strategies aim to unify representations of different transformations of the same image (commonly referred to as "views"), while enforcing diversity among representations of distinct images. Additionally, self-supervised representations can predict primate neural responses with fidelity comparable to supervised representations (Konkle and Alvarez, 2022; Parthasarathy et al., 2024a; Zhuang et al., 2021).

Both of these training objectives are forms of invariance learning: responses of an ideal object classification model should be invariant across different objects from the same class, and self-supervised learning strives to achieve invariance to the transformations used to generate different

views. However biological visual representations are not fully invariant across views (DiCarlo and Cox, 2007; Kuoch et al., 2024). Indeed it has been demonstrated that training according to either of these two objectives leads to representations that are invariant to stimulus perturbations that are salient to human observers (Feather et al., 2023). Additionally, even in Area IT, which is thought to subserve invariant object recognition, neural populations encode a significant amount of "category orthogonal" information (e.g., object pose or viewing conditions that are unrelated to semantic category) (Hong et al., 2016). Furthermore, such selectivity for object-orthogonal attributes is meaningfully organized within Area IT (Hong et al., 2016) (i.e. object orthogonal attributes are linearly decodable from population responses). Whether such structured variability emerges in invariance-trained networks is likely determined by the uncontrolled inductive biases of the network architecture (Alleman et al., 2024).

Here, we develop an equivariant learning framework that encourages such structured variability in network representations. Our contributions are:

- We propose a novel framework that converts standard invariance-based self-supervised learning methods into "contrastive-equivariant" versions that produce structured, transformation-related variability. Unlike previous approaches, our method does not require supervised access to transformation parameters or costly modifications to the training procedure.

- We examine the tradeoff between invariance and structured variability through a series of representational analyses. We find that, relative to networks trained for invariance alone, our contrastive-equivariant network learns structured transformation variability that is shared across images and factorized with respect to variability related to changes in image content.

- We explore the impact of including an equivariant loss for predicting neural activity in IT, showing for the first time that explicitly encouraging structured variability via optimization leads to an improved ability to predict cortical responses to natural images.

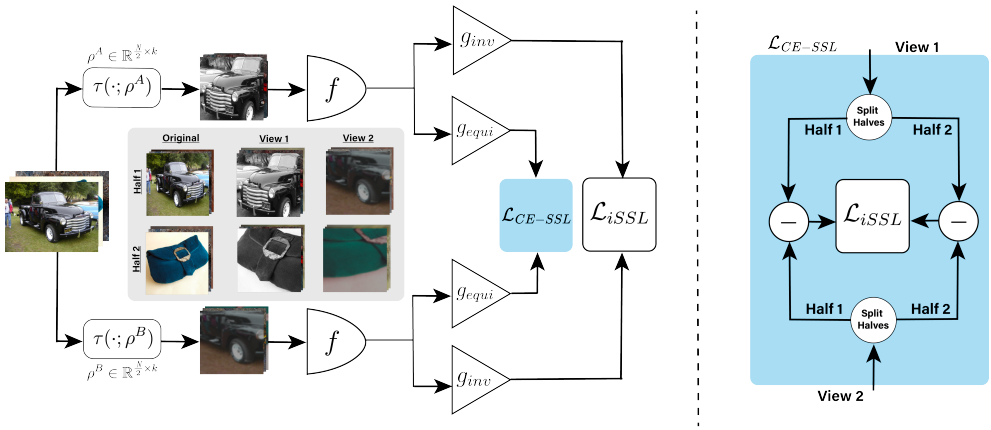

Figure 3.1: CE-SSL Schematic. Diagram of the proposed training method. At the beginning of each training epoch the dataset is randomly split into two non-overlapping halves. Left Gray Panel: corresponding images in each subset are augmented using the same set of two random transformations (so the total number of random transformations is halved relative to a standard iSSL training scheme). Every view is passed through a representation network f (ResNet-50 in this work) and the outputs are projected into two embedding spaces by different projector networks, g_{inv} and g_{equi}. In the invariant embedding space a standard iSSL loss is applied, while in the equivariant embedding space the same iSSL loss is applied to the difference vectors between transformation-positive pairs (visualized on the right).

3.3 Method

3.3.1 Transformation-Invariant Self-Supervised Learning (iSSL)

The influential work of (Chen et al., 2020b) showed that applying two random transformations (often called "augmentations") to a batch of images, then training a network to identify which pairs of transformed images originated from the same sample with a cross-entropy style loss (the InfoNCE loss, first formulated in (Gutmann and Hyvärinen, 2010)) yields representations that are competitive with supervised training for object classification. Many subsequent studies have developed alternative objective functions that produce similar results: Barlow Twins(Zbontar et al., 2021), VICReg (Bardes et al., 2022), and W-MSE (Ermolov et al., 2021) enforce augmentation invariance along with a constraint that the global covariance matrix is the identity; SimSiam (Chen

and He, 2021) and BYOL (Grill et al., 2020) employ architectural constraints that regularize towards uniform representations and simply optimize for transformation invariance. Other studies have formalized the problem in terms of maximizing information (Ozsoy et al., 2022) or capacity (Yerxa et al., 2024b), subject to an invariance constraint, which has enabled connections to normative theories of coding efficiency and manifold capacity (Barlow et al., 1961; Chung et al., 2018).

To formalize the definition of iSSL, we denote a dataset of images (e.g., ImageNet) by $X \in \mathbb{R}^{N \times D}$, where N is the number of images and D is their dimensionality (number of pixels). Let $\tau(\cdot; \rho) : \mathbb{R}^D \to \mathbb{R}^D$ be a function parameterized by ρ that maps images to images (for example, for τ a random crop operation, ρ specifies the region to be cropped). The goal of iSSL algorithms is to learn the parameters W of some function $f(\cdot; W) : \mathbb{R}^D \to \mathbb{R}^d$ such that the variability over ρ is minimal while preserving variability over X (to avoid trivial solutions such as $f(\cdot; W) = 0$ for all inputs). Many methods achieve this by observing pairs of randomly augmented views of a batch of images: $X^A = \tau(X; \rho_1)$, $X^B = \tau(X; \rho_2)$, with $\rho_1, \rho_2 \sim p(\rho)$ where $p(\rho)$ is a pre-chosen probability distribution over augmentation parameters. Generally iSSL frameworks employ an objective function that operates on the outputs of f, $Z^A = f(X^A; W)$, $Z^B = f(X^B; W)$. One popular framework is "Barlow Twins" (Zbontar et al., 2021), which uses the objective: $\mathcal{L}_{BT} = \Sigma_i (1 - C_{ii})^2 + \lambda \Sigma_{i, i \neq j} (C_{ij})^2$ where C is the cross-correlation matrix between Z^A and Z^B. The first term encourages the outputs in response to the same image subject to different augmentations to be correlated, while the second encourages the outputs in response to distinct images to be uncorrelated.

Because complete invariance to the transformations employed in iSSL is harmful for downstream tasks, most frameworks employ a learnable “projector network" that maps the outputs of the representation network to an embedding space before applying the loss. The nearly ubiquitous use of this “guillotine regularization" (Bordes et al., 2023), means that most iSSL methods aim to learn a function *from which an augmentation invariant subspace can be extracted.* While

this approach does permit some transformation-related variability in the representation, there is no explicit control or encouragement of that variability, and no incentive for that variability to be usefully structured.

3.3.2 Contrastive-Equivariant Self-Supervised Learning (CE-SSL)

To induce structured variability in learned representations, we require that an equivariant subspace can be extracted from f alongside the invariant subspace described above. A function is equivariant to a set of input transformations if there exists a corresponding set of output transformations that induce the same changes. In the self-supervised learning setting this property can be expressed as:

$$\forall \tau_\rho \in P, \;\; \forall x \in X, \;\; \exists T_\rho : f(\tau_\rho(x)) = T_\rho(f(x)), \tag{3.1}$$

where $\tau_\rho = \tau(\cdot; \rho)$ and P is the set of possible values of transformation parameters. Note that invariance is a special case of equivariance, in which T_ρ is an identity transformation for all ρ. To avoid this degenerate solution, we will require both that similarly transformed inputs be related by the same transformation in the output space, and that differently transformed inputs are related to each other by different transformations.

Our training methodlogy, summarized in Fig. 3.1, follows the principle first proposed by (Gupta et al., 2024): "Equivariance should be learned from pairs of data, as in invariant contrastive learning." First we split our dataset of images into two random non-overlapping equal-sized partitions $X_1, X_2 \in \mathbb{R}^{\frac{N}{2} \times D}$. Next we apply a randomly selected augmentation to both X_1 and X_2, so that corresponding rows of $X_1^{A/B}$ and $X_2^{A/B}$ contain distinct images that have been subjected to the same augmentations. Note that this reduces the total number of random samples of ρ by a factor of two relative to standard iSSL methods. Finally the resulting representation vectors are each fed through two distinct projector networks, $z_i^{A/B} = g_{inv}(r_i^{A/B})$ and $\tilde{z}_i^{A/B} = g_{equi}(r_i^{A/B})$. These two embeddings are optimized to be invariant to transformations and discriminative across base

images, or invariant to base images and discriminative across transformations, respectively. The overall objective (loss) functions is:

$$\begin{aligned} \mathcal{L}_{\text{overall}} &= (1-\lambda)\mathcal{L}_{iSSL} + \lambda\mathcal{L}_{CE-SSL} \\ \text{where} \quad \mathcal{L}_{iSSL} &= \mathcal{L}([z_1^A, z_2^A], [z_1^B, z_2^B]), \\ \text{and} \quad \mathcal{L}_{CE-SSL} &= \mathcal{L}(z_1^A - z_1^B, z_2^A - z_2^B), \end{aligned} \tag{3.2}$$

where both terms are written in terms of $\mathcal{L}$, a self-supervised learning loss function that encourages invariance and uniformity (Wang and Isola, 2020) (e.g., L_{BT}) and λ is a hyperparameter that determines the relative importance of extracting an invariant or equivariant subspace from the shared representation. In the notation of Eq. (3.1), by designing $\mathcal{L}_{CE-SSL}$ to encourage similar transformations to induce similar displacements in the output space, we are implicitly specifying that our output transformations are of the form $T_\rho(z) = z + z_\rho$. Thus we leverage the principles underpinning contrastive invariance learning to encourage representations that contain useful transformation-related information; this choice differentiates this formulation from previous equivariant self-supervised learning approaches.

3.4 Results

3.4.1 Implementation Details

Architecture and invariance ojective. For all experiments we use a ResNet-50 architecture (He et al., 2016) as the backbone representation network f. Our training scheme is compatible with any choice of iSSL framework, as specified by the choice of $\mathcal{L}_{iSSL}$. We experimented with three different base methods chosen to span the range from “instance contrastive" to “dimension contrastive" (Garrido et al., 2023a): SimCLR (Chen et al., 2020b), MMCR (Yerxa et al., 2024b), and Barlow Twins (Zbontar et al., 2021). In each case, we define g_{inv} using the projector network

architecture proposed in the original work. To retain the synergy between the normalization scheme, loss function, and projector architecture achieved by each framework we use the same architecture for both g_{inv} and g_{equi}.

Pretraining dataset and augmentations. We train using the ImageNet-1k dataset and the standard set of augmentations first introduced in (Grill et al., 2020), which includes random resized cropping, color jittering, Gaussian blurring, solarization, and horizontal flips. See Appendix B.1 for exact training details.

Invariance-equivariance tradeoff. For each of the three choices of $\mathcal{L}_{iSSL}$ we trained networks with hyperparameter values $\lambda \in \{0.0, 0.001, 0.1, 0.2, 0.3, 0.4, 0.5\}$, yielding a total of 21 learned representations (note: $\lambda = 0$ corresponds to standard iSSL). We found that classification performance becomes severely degraded for values of λ larger than 0.5 (see Appendix B.3).

3.4.2 Representational Analyses

Bures metric comparisons. We conducted a series of experiments to determine the extent to which various sources of variability in our dataset were meaningfully organized. The experiments utilized the Bures metric, which is the Wasserstein ("Earth Mover's") distance between mean-centered Gaussian distributions with covariance matrices C_1 and C_2:

$$D_B(C_1, C_2) = \text{trace}\left(C_1 + C_2 - 2\left(C_2^{1/2} C_1 C_2^{1/2}\right)^{1/2}\right). \tag{3.3}$$

When C_1 and C_2 are normalized to have a trace of 1, the maximal distance of 2.0 occurs when the variabilities lie in orthogonal subspaces (or are completely "factorized" from each other) and the minimum distance of 0.0 occurs when the covariances are equal. More generally, a large Bures distance indicates two sources of variability are factorized from each other and a low distance indicates shared structure. We first estimate the trace-normalized covariance of the outputs of

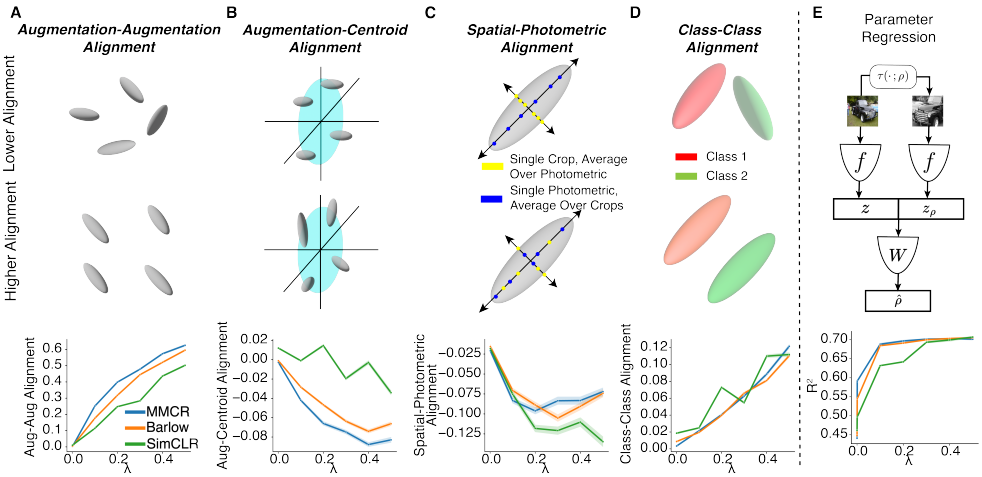

Figure 3.2: Effects of equivariance on representational geometry. Effects of equivariance on representational geometry. **A**: Alignment between augmentation manifolds (gray ellipsoids). **B**: Alignment between augmentation manifolds (gray ellipsoids) and the centroid manifold (blue disk). **C**: Alignment between spatial and photometric manifolds. Gray ellipsoids represent single augmentation manifolds, and blue/yellow points indicate the mean over the outputs from many transformations of a single view obtained via a photometric/spatial transformation, respectively. Expected distance is larger when the two sources of variability are factorized. **D**: Same as A., but for class manifolds. **E**: A schematic of the parameter regression experiment. In each panel, the bottom row depicts the results of each analysis described in the text of Sections 3.4.2 and 3.4.2. Shaded regions indicate 95% confidence intervals (estimated over the same comparisons the expected distance is estimated over for A-D and over 5 independent runs of the regression experiments for E). A summary of the sources of variability used to compute C_1 and C_2, and the ensemble used to estimate the expected alignment is measured can be found in Table B.3

some network f over two sources of variability and compute the Bures metric between the two.

Because we are mainly interested in the impact of the equivariance loss relative to the invariant baseline, each analysis below is carefully controlled to expose any structural differences. In particular, we estimate C_1 and C_2 over identical inputs for an invariant network and an equivariant network trained using the same base objective but a non-zero value of λ. We then can directly compare the measured Bures distance for the invariant network and each equivariant network ($\lambda \neq 0$): $\Delta D_B = D_B(C_1^{\lambda=0}, C_2^{\lambda=0}) - D_B(C_1^{\lambda\neq 0}, C_2^{\lambda\neq 0})$; this measure quantifies the amount of alignment between two sources of variability in an equivariant network relative to the invariant network baseline. In each of the panels in the bottom row of Fig. 3.2 we show how $\mathbb{E}[\Delta D_B]$ (denoted simply "Alignment") evolves as a function of λ when C_1 and C_2 are estimated over different sources of variability (y-axis labels indicate the ensembles over which the variabilities were estimated). We show the joint distribution of $(D_B(C_1^{\lambda=0}, C_2^{\lambda=0}), D_B(C_1^{\lambda\neq 0}, C_2^{\lambda\neq 0}))$ and summarize the sources of variability in each experiment described below in Appendix B.4.

Augmentation-Augmentation alignment. First we determine the extent to which augmentation variability is shared across base images in each network (Fig. 3.2A). For these experiments, both C_1 and C_2 are estimated over many random transformations of single images in the validation set (we will refer to the responses to such a group as an "augmentation manifold"). The expectation is then over randomly sampled pairs of augmentation manifolds. The positive expected difference of distance indicates the equivariant networks consistently produce lower distance between augmentation manifolds, indicating more shared augmentation variability across base images. This structure is closely related to what is encouraged by the equivariance loss term and the orderly increase as a function of λ suggests that we are optimizing effectively.

Augmentation-Centroid factorization. We next investigate the extent to which variability over augmentations is factorized from variability over base images (Fig. 3.2B). We use the "centroid manifold," (Yerxa et al., 2024b) to characterize the variability over base images, by measuring

the covariance over base images of the means over augmentations. That is, for these experiments (for each network respectively) C_1 is the covariance of the centroids of all augmentation manifolds and C_2 is the covariance of a randomly selected augmentation manifold. We observe that equivariant networks generally exhibit a larger distance between centroid and augmentation manifolds indicating increased factorization (or lower alignment) of image-content variability and image-augmentation variability. This structure was not explicitly encouraged by the objective and can be considered an emergent property of the equivariant learning procedure.

Spatial-Photometric factorization. Next we ask whether our equivariant training procedure induced increased factorization of variability to different types of input transformations (Fig. 3.2C). The standard augmentation procedure involves first taking a random crop (spatial variability) of a given image and then applying a series of pixel-level transformations (color-jittering, gaussian blurring, etc.) (photometric variability). To assess the impact of these two distinct classes of image transformations we first chose 20 random crops a given image, then applied the same set of 20 random photometric transformations to each individual crop, yielding 400 different views of each base image. C_1 and C_2 are then estimated over network responses that are averaged over different crops or different photometric transformations respectively, and the expectation is taken over different (single) base images. We observe the equivariant networks consistently exhibit increased factorization (i.e. larger Bures distances relative to the invariant trained network). This again is an emergent property of the equivariant learning procedure, and is particularly interesting in light of recent work that discovered that this form of transformation-factorization is more correlated with neural predictivity than transformation invariance (Lindsey and Issa, 2024).

Class-Class factorization. Finally we asked whether within-class variability was more or less shared between distinct classes in equivariant networks by estimating C_1 and C_2 over responses to all images in distinct classes in the validation set (the expectation is then taken over

different random pairs of classes) (Fig. 3.2D). Increased sharing of variability between class manifolds has been demonstrated to increase manifold capacity, and can make representations better suited for multi-task evaluations (Wakhloo et al., 2024; 2023). We observe higher alingment (lower expected pairwise Bures distances) in the equivariant networks indicating that the “class manifolds" relative to the invariant networks.

Linear embedding of augmentation-related information. While the above experiments demonstrate that equivariant training induces increased alignment of transformation-related variability between images, this does not necessarily imply that this variability is coherently organized. To assess this more directly, we measure the extent to which augmentation parameters can be linearly decoded from the networks' representations. Specificlly, we regress the concatenated outputs of a clean and transformed image onto the parameters of the applied augmentation. We report the resulting coefficient of determination (R^2) on a heldout set of validation images (Fig. 3.2). The equivariant training is seen to increase the amount of linearly accessible augmentation information relative to invariant training (the leftmost points plotted in Fig 3.2E). We further analyzed a set of equivariant models trained with weaker augmentation parameters (see B.5 for details). In these networks, we again observe that equivariant training increased the amount of linearly accessible augmentation information compared to invariant training. This holds not only for augmentation parameters within the training range (left panel B.3) but also for parameter values beyond the training range (right panel B.3). Thus, the equivariance properties of the models generalize beyond the training distribution. Future work could examine generalization to unseen types of augmentations.

3.4.3 Neural Predictivity

We utilized the BrainScore evaluation pipeline (Schrimpf et al., 2018) to measure the extent to which each learned representation can linearly predict neural responses measured in macaque

area IT, for four different experimental datasets. At the time of testing, our highest performing model (Barlow Twins objective, $\lambda = 0.2$) had the 10th highest average predictivity for area IT out of approximately 250 publicly available models on the Brain-Score leaderboard. Across a reasonably large range of values of λ, the equivariant models improved the neural predictivity relative to the invariant baseline ($\lambda = 0$) for all four datasets (Fig. 3.3). Many previous publications have noted that changes in training objective function have a small effect on neural predictivity, relative to other factors such as training dataset (Conwell et al., 2022; Tuckute et al., 2022; Yerxa et al., 2024b). In contrast, encouraging equivariance produced much larger gains than choosing between different base invariant objectives: the range of predictivities over the sweep of λ was around 4 times larger than the range of predictivities over objective functions for the invariant baseline. We further contextualize the scale of predictivity improvements in Fig 3.4 by comparing models to all public submissions on the BrainScore leaderboard; our equivariant training procedure improves performance of the already-strong invariant models to nearly state-of-the-art levels of IT predictivity. By training the most predictive model for 1000 epochs (rather than 100), we achieved 0.5355 mean fraction of explained variance, which makes this the top IT brain prediction model. To ensure that the observed alignment increases are not architecture specific, we trained a smaller set of models using different backbone architectures and observed similar trends when using both smaller and larger networks (see Appendix B.6 for details).

We quantified the correlation between our various representational measurements and the neural predictivity for each of the four electrophysiology datasets in Table 3.1. We observed that the only representational metric with a correlation greater than 0.4 across all four neural datasets was the Spatial-Photometric distance, which is the metric most closely related to the factorization score described in (Lindsey and Issa, 2024). While this previous study described a correlation between structured variability and neural predictivity measured from a large set of pre-trained models, our results demonstrate that explicitly encouraging such structures can improve alignment between artificial and biological representations. In addition to the previ-

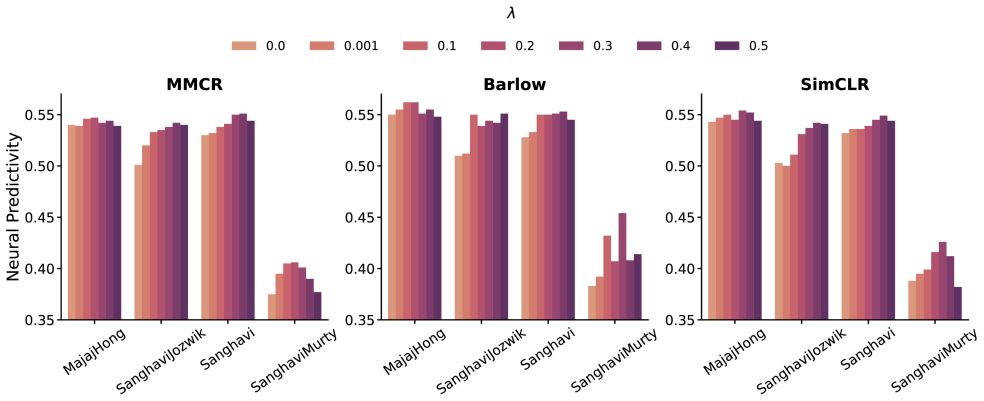

Figure 3.3: CE-SSL Brain-Score Evaluations. Brain-Score (noise-ceiled predictivity evaluated via ridge regression) for each value of λ (different colored bars) for each IT dataset (groups of columns) and base objective functions (different figure panels). For all datasets and base objectives the invariant network ($\lambda = 0$, lightest bars) is outperformed by at least one equivariant network, and the spread in predictivity over values of λ is significantly larger than the spread in predictivity over base objective functions for invariant networks.

ously described representational measurements, we also looked at the linear decoding of the hue modulation parameter in isolation. Hue modulation is one of 12 augmentation parameters that are linearly decoded in the parameter regression measures described in Section 3.4.2. We observed a strong correlation between neural predictivity and hue modulation, particularly with the Sanghavi-Jozwik dataset, which is the only response dataset that included color image stimuli (last column of Table 3.1).

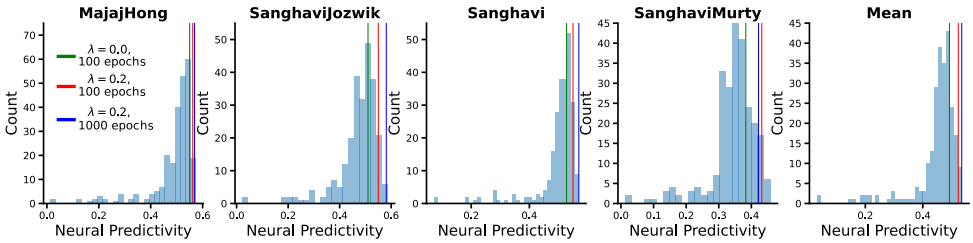

Figure 3.4: CE-SSL On Brain-Score Leaderboard. Histogram of neural predictivity scores for the 249 models on the public Brain-Score leaderboard at the time of testing, for each of the four considered IT datasets, as well as the mean over the four datasets. In each plot the vertical green line shows the score of the invariant Barlow Twins model, the red line shows the score for the equivariant Barlow Twins model with $\lambda = 0.2$, and the blue line shows a new model trained for 1000 epochs, also using Barlow Twins for the base loss and $\lambda = 0.2$.

Table 3.1: Correlating neural predictivity with representational measurements. Absolute values of Pearson correlation coefficients (R^2) between various representational measurements and the neural predictivity across each of the four IT datasets. The correlation was measured over each value of λ and base objective function for a total of 21 networks. Each column corresponds to a panel in Fig. 3.2, except for hue, which is the regression score obtained for the random hue modulation parameter in isolation.

Neural Dataset	Aug-Aug	Aug-Centroid	Spatial-Photometric	Class-Class	Param Regression	Hue Regression
Majaj-Hong	0.03	0.02	0.42	0.10	0.38	0.28
Sanghavi-Jozwik	0.86	0.7	0.83	0.77	0.91	0.91
Sanghavi	0.84	0.84	0.68	0.63	0.85	0.86
Sanghavi-Murty	0.33	0.41	0.42	0.14	0.56	0.48

3.4.4 Transfer Learning

Several previous studies that aim to reduce augmentation-invariance of self-supervised features have reported that the resulting representations generalize better to out-of-distribution classification datasets (Chavhan et al., 2023; Gupta et al., 2024; Suau et al., 2023; Xiao et al., 2021). However most of these studies focused on using the smaller ImageNet-100 dataset for training, in one case reporting that the transfer learning gains diminish or disappear when using ImageNet-1k (Chavhan et al., 2023). We tested our set of networks on 6 different downstream tasks and found limited evidence that the equivariant features confer an advantage in terms of out-of-distribution generalization when training on a sufficiently large and diverse dataset (see Table 2). To address this discrepancy with the literature we conducted additional experiments on networks trained using the ImageNet-100 dataset, and in this case observed improvement in generalization to diverse downstream tasks.

It is also worth noting that CE-SSL trained networks do not outperform their invariant counterparts on in-distribution generalization (see B.1.1 and Fig. B.1). This is not surprising in light of the fact that the suite of augmentations and architectures employed in SSL have been in some

sense optimized by the community in order to improve performance on this task (by aligning the transformation invariance task with the standard in-distribution classification task). However, for out-of-distribution classification tasks where the task-alignment is worse, the equivariance task could mitigate this mis-match. A concrete example is the Flowers-102 dataset, where the color of petals is a much stronger predictor of class than color is in, say, the ImageNet-1k dataset (so the color insensitivity induced by the standard augmentations could be detrimental). For this dataset we do see marginal improvements, but note that the improvements are much more pronounced when pretraining on smaller datasets (ImageNet-100). There are at least 2 possible explanations for this: (1) for ImageNet-1k pretraining the performance of the networks is already quite high, and the task is saturated, or (2) there is a more fundamental reason that the improvements in transfer learning induced by equivariance decrease as the size and diversity of the pretraining dataset grows. Future work could explicitly disambiguate between these hypotheses to determine why the benefits of transformation-related variability for out of distribution generalization are outweighed by the gains of scaling the dataset. Furthermore this result shows that the increased neural predictivity we observe in ImageNet-1k trained networks cannot be explained by a need to perform better on a variety of invariant-classification tasks.

3.5 Relationship to Existing Augmentation-Sensitive SSL Methods

A key feature that differentiates our approach is that it encourages equivariant structure without explicit access to augmentation parameters. This is enabled by the “paired augmentation" data generation procedure, and to the best of our knowledge CARE (Gupta et al., 2024) is the only existing work that shares this feature. Our method has two advantages over CARE: (1) in CARE the equivariance loss is applied in the same space as the invariance constraint, and because there are no “negative equivariant pairs" in the CARE framework, learning an invariant representation

Table 3.2: Equivariant Network Transfer Learning Evaluations. Frozen-Linear Evaluation for invariant and equivariant trained networks on 6 different downstream datasets: Cifar-10/100 (Krizhevsky et al., 2009), Oxford-Pets (Parkhi et al., 2012), Describable Textures Database (Cimpoi et al., 2014), Flowers-102 (Nilsback and Zisserman, 2008), and Food-101 (Bossard et al., 2014). We closely follow the evaluation procedure from (Lee et al., 2021) (see Appendix B.3 for details) and report top1 accuracy for each objective/dataset. In all cases we report the mean over 5 runs of the evaluation procedure, we observed very little variability (maximum of .2%, over all evaluations, we report the standard deviation over runs in Appendix B.3. The equivariant networks are denoted by prepending a "CE" before the objective and were trained using $\lambda = 0.1$, which enabled a substantial amount of structure variability without significantly impacting frozen-linear classification on the SSL training dataset (see Appendix B.1). For ImageNet-1k trained networks out of distribution performance decreased for most evaluations, while for ImageNet-100 trained networks performance was improved in 15 of 18 cases.

ImageNet-100 Training						
Objective	Cifar-10	Cifar-100	Pets	DTD	Flowers-102	Food-101
MMCR	84.3	63.3	67.0	66.1	83.1	60.9
Barlow	87.7	68.7	74.8	67.0	88.3	63.4
SimCLR	87.8	68.8	74.3	66.6	88.5	64.8
CE-MMCR	87.3	69.4	68.9	65.7	87.5	64.1
CE-Barlow	88.0	69.1	73.6	67.3	89.5	65.5
CE-SimCLR	87.9	68.2	72.6	67.5	88.6	65.2
ImageNet-1k Training						
Objective	Cifar-10	Cifar-100	Pets	DTD	Flowers-102	Food-101
MMCR	92.2	76.9	85.3	75.4	93.9	73.8
Barlow	91.8	75.8	86.5	73.0	93.8	72.2
SimCLR	91.8	74.7	85.1	74.5	92.7	70.5
CE-MMCR	92.2	76.6	84.3	75.7	94.0	73.8
CE-Barlow	91.8	75.3	85.7	75.6	94.2	73.0
CE-SimCLR	91.0	73.6	82.3	73.9	92.2	70.5

would perfectly satisfy the equivariant constraint; and (2) in CARE the standard augmentation pipeline is used to optimize the base $\mathcal{L}_{iSSL}$ loss and paired augmentation are used in parallel to optimize the equivariance term (so an increased number of passes through the network is necessary relative to standard training).

In EquiMod (Devillers and Lefort, 2023) the projector network is conditioned on the augmentation parameters by appending the parameters to the output of f. The authors theorize that knowledge of the augmentation parameters could allow the projector network to better extract an invariant subspace tailored to each transformation, thus allowing for more structured variability in the representation space. Alternatively, the projector network could simply ignore the augmentation parameters, resulting in a structure that is identical to invariant SSL. In practice (Garrido et al., 2023b) have found this to be the case. Split-Invariant-Equivariant (SIE) and Amortised Invariance (AI) learning (Garrido et al., 2023b) each improve on this principle by using a hypernetwork approach: a separate network takes as inputs the augmentation parameters and outputs the parameters of either g or both f and g respectively. While the collapse issue of EquiMod is avoided, this comes at the expense of significantly complicating the network computation, and in the case of AI introduces new parameters that need to be tuned for every downstream task (when augmentation information is not available). Still other methods supplement the standard invariant SSL loss with an auxillary term that involves predicting the parameters of the input transformation (Dangovski et al., 2022; Lee et al., 2021). The relationship of our method to these is analogous to the relationship of transformation-invariant self-supervised learning to supervised classification.

3.6 Discussion

We've developed a new self-supervised objective that explicitly encourages structured variability in networks, and demonstrated that it can produce increased alignment with responses of

neurons in primate visual area IT. While we are not the first to incorporate a notion of equivariance to self-supervised learning, our method improves on existing work in several ways: it require no extra passes through the network relative to invariance-based learning, it encourages diversity in the representation of transformation-related information by leveraging advances in invariance-based learning, and it does not rely on supervised access to transformation parameters. The parsimony of our approach (applying the same objective to both outputs of individual images and to displacements between similarly transformed images) allows our technique to be easily adapted to other settings such as temporal self-supervised learning (discussed below). Although in this work we focused on the visual domain, similar equivariant and invariant objectives could be investigated for other domains such as audio and langauge representation learning.

Our approach induced several interesting features in the learned representations: transformation variability is shared across base images and factorized with respect to variability over base images, the variability induced by distinct types of transformations are factorized from each other, there is increased alignment between class manifolds, and transformation related information is linearly encoded. Some of these properties are closely related to the imposed objective and some are emergent. We also confirmed that several of these representational properties are correlated with increased neural predictivity. Future work can extend these correlative observations to better understand how increasing transformation sensitivity improves neural alignment. For example, one could analyze the residuals of predicted neural firing rates of distinct models to determine how “overlapping" the variance predicted by each is (or alternatively, attempt to fit the residual variance of one model with another). Such analyses are becoming more feasible with the collection and release of larger scale datasets of neural responses to natural images (e.g., (Madan et al., 2024)). We view this result as demonstrating the promise of incorporating knowledge gained from experimental observations and large scale comparative studies into optimization procedures to produce better models.

Although our experimnents reveal both induced and emergent benefits, the inclusion of an

additive equivariance term in the objective does lead to fewer guarantees regarding the learned structure. For example in schemes where the output transformations (T_p's) are explicitly represented or learned, the resulting representation is "steerable" by default. It is of interest to investigate whether the output transformations could be reliably recovered from our learned representations. Additionally it would be interesting to consider other types of output transformations (CARE (Gupta et al., 2024) focuses on orthogonal transformations, and in the case where output transformations are learned they can be computed with nonlinear neural networks).

Finally, it is of interest to explore the use of more ecologically relevant sources of training data, e.g., by replacing synthetically transformed views of images with temporally adjacent frames of natural videos. This approach is particularly appealing from the perspective of biological plausibility, as the pairing of such training examples is readily available from natural visual experience. Several recent publications have shown that such a strategy can produce representations with competitive neural predictivity and performance on computer vision tasks (Parthasarathy et al., 2023; Venkataramanan et al., 2024; Zhuang et al., 2021). In this context, the typical invariance loss can be thought of as incentivizing representational slowness (Földiák, 1991; Wiskott and Sejnowski, 2002). The equivariance mechanism described in this work could be implemented by applying the same invariance-based loss function to the first temporal derivative of the responses, i.e. by encouraging the displacement between successive pairs of frames to be constant. Such a temporal-equivariance objective would incentivize representational straightness, which has been used to describe features of both human perception and neural activity in the ventral stream (Hénaff et al., 2021; 2019). Straightness in artificial representations has been found to be correlated with both neural predictivity and adversarial robustness (Harrington et al., 2023; Lindsey and Issa, 2024; Niu et al., 2024). These connections provide an array of promising research directions.

4 Modeling Visual Cortex by Maximizing Layerwise Multiscale Manifold Capacity

4.1 Overview

Task-optimized deep neural networks have risen to prominence as the most predictive phenomenological models of responses in primate visual cortex, but leave much to be desired from the perspective of biological plausibility. One such limitation is the reliance on precise credit assignment through global backpropagation of error signals. Recent work has shown that this weakness can be circumvented by requiring each subsequent stage to solve a distinct and increasingly complex task, allowing for layerwise local learning signals. We propose a novel strategy for crafting such intermediate losses that uses an efficient coding framework formulated in terms of manifold capacity, which can be computed using a sequence of canonical cortical computations. In particular, we leverage the relationship between the multiscale nature of visual signals and the dilation of receptive field sizes in cascaded visual representations to modulate complexity, allowing for the reapplication of these common loss computations at each stage of the hierarchy. We evaluate our approach on its ability to predict neural datasets spanning three areas of the ventral stream hierarchy in macaques, as well as human psychophysical data on an object classification

task. We find that our unsupervised layerwise model matches or exceeds the performance of competitive architecture-matched baselines on all evaluations considered.

4.2 Introduction

The ventral stream of the primate visual cortex has served as a reference for object recognition for many decades. Traditional models, based on linear filtering, rectification, and gain control have accounted for response properties of neurons in the early stages of this stream (e.g., areas V1 and V2, (Felsen et al., 2005; Heeger, 1992; Liu et al., 2016; Vintch et al., 2015; Willmore et al., 2010)), but have proven difficult to generalize to later stages (e.g., areas V4 and IT). Deep neural networks (DNNs) currently offer the strongest predictive models of neural responses in these later stages (Willeke et al., 2023; Yamins et al., 2014; Zhuang et al., 2021), but these results come with a number of limitations. For one, their predictions are obtained by regressing a large number of model neurons (typically thousands) onto each biological neuron, risking overfitting and allowing for a generic nonlinear approximation with minimal interpretability or insight into biological properties. Additionally, most DNN training aims to optimize an end-to-end objective function, which requires precise credit assignment via global backpropagation of error signals. This computation is generally thought to be implausible for biological implementation, and it may also provide insufficient constraint on the internal representation of the network, especially at the earlier stages of processing. In light of this, it is perhaps unsurprising that many of the best recent DNN models of early cortical representations are those that incorporate additional constraints. In particular, both adversarial training (which enforces robustness to small image perturbations - (Madry et al., 2018)), and layerwise learning (which directly applies objective functions at multiple stages of a network's hierarchy - (Parthasarathy et al., 2024b)), have offered improvements.

Here, we develop a layerwise self-supervised learning strategy that addresses the limitations

of end-to-end trained DNNs, while avoiding the high computational cost associated with adversarial learning. The objective for each layer is a particular form of coding efficiency, which seeks a compromise between representational quality and resource limitations. For the former, we use *manifold capacity*, which quantifies the number of neural manifolds that can be linearly separated in a population of neurons (Chung et al., 2018). Recent work has successfully adapted this framework into a self-supervised learning objective(Yerxa et al., 2023). For the latter, we constrain the computational capacity of each stage of processing through the network architecture. Early layers have fewer neurons with smaller receptive fields, and are also less expressive in terms of the complexity of their response functions (because they are computed with a smaller set of rectification operations). We also seek to match the complexity of the learning task (or measure of representational quality) to the computational capacity at each stage where the loss is applied (Parthasarathy et al., 2024b), by applying each layer's objective to a localized region whose size is proportional to the architecturally-constrained receptive field size. This results in a canonical implementation of the loss function at each stage of the network hierarchy, which can be computed using operations consistent with response properties observed in cortical neurons.

We implement a three-stage network based on the AlexNet architecture, and train each stage independently using synthetic videos derived by sampling still images from the ImageNet dataset and simulating their evolution over time with a randomly chosen mixture of translation, dilation, and intensity shifting operations. The self-supervised MMCR objective is applied to the temporal responses of each layer, over intervals whose duration also scales with the corresponding receptive field size. We show that the resulting network outperforms an architecturally-matched object-recognition network, as well as its adversarial robust extension, in explaining responses of primate neurons recorded in areas V1, V2, and V4. Moreover, this improvement in performance holds when restricting the regression fit to a smaller set of model neurons. Finally, we compare our trained network to human performance on classification tasks, and find that it supports downstream object classification behavior that is more robust to distribution shifts and better

aligned with human behavior than networks trained supervised or adversarial learning.

4.2.1 Related Work

In the context of modern machine learning, layerwise or "greedy" learning algorithms were initially developed as pre-training methods whose goal was to provide a good initialization for eventual end-to-end optimization (Bengio et al., 2006; Hinton et al., 2006). More recent work has demonstrated that the composition of gradient-isolated modules can achieve respectable levels of performance on downstream tasks when employing either supervised or self-supervised learning objectives (Belilovsky et al., 2019; Löwe et al., 2019; Siddiqui et al., 2024). In computational neuroscience, there has been a longstanding interest in developing and analyzing learning algorithms that update synaptic strengths based on local signals (e.g., Hebbian learning rules). Two recent examples that apply such rules to hierarchical representation learning are Contrastive Local and Predictive Plasticity (CLAPP - (Illing et al., 2021)) and Layerwise Predictive Learning (LPL - (Halvagal and Zenke, 2023)). By and large these algorithmic advances have fallen short of quantitatively improving our ability to model the responses of real neurons.

Parthasarathy et al. (2024b) argue that one key limitation of prior approaches is the failure to align task complexity with the representational capacity of each layer, which naturally increases with depth. By explicitly matching complexity to capacity, they achieve state-of-the-art neural predictivity in area V2. However, their implementation relies on providing each layer with a distinct input stream, modulating the augmentation strength used to train each stage via self-supervision. While intuitive, this strategy strains the analogy between model-training and any biological optimization algorithm which must of course use a single stream of visual inputs to learn all stages.

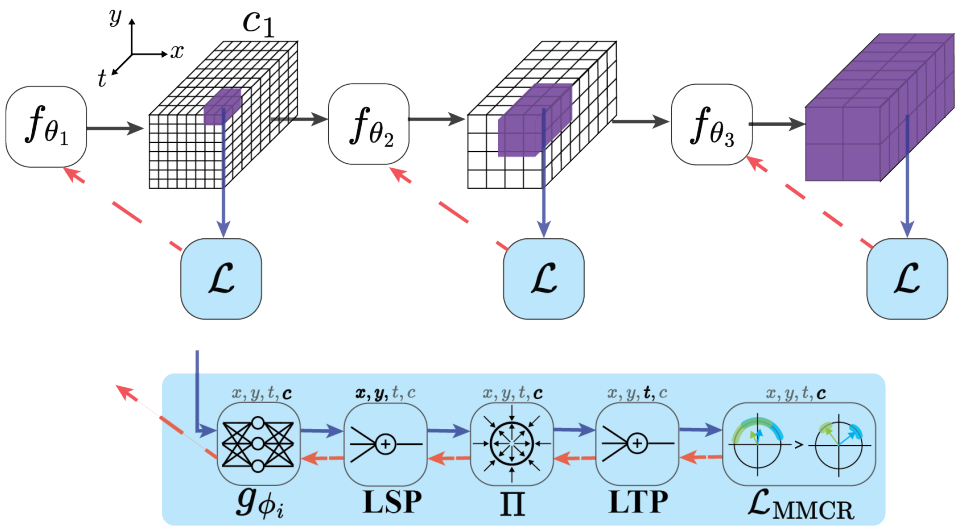

Figure 4.1: ST-MMCR schematic. In response to a video input, each stage of the network produces spatio-temporal feature maps ("channels", c_j, one shown for each stage, as a 3D block), with successively reduced spatial resolution, but fixed temporal resolution. The objective that is used to optimize parameters θ_i for the ith stage operates on a spatio-temporal region (indicated in purple) whose extent in both space and time successively increases by factors of two at each stage. This objective is of identical form at each stage, and consists of a sequence of computations (exploded diagram at bottom), each operating over one or two indices (indicated in bold above each computation): 1) a multi-layer perceptron g_{ϕ_i} with a single hidden layer is applied across channels, at each spatio-temporal location; 2) these responses are spatially pooled (averaged) over a local central region (purple block); 3) these are normalized (i.e., divided by their Euclidean length) over channels; 4) these are temporally pooled to yield manifold centroids; 5) the MMCR loss (negative nuclear norm of the matrix of centroids) is computed.

4.3 Methods

Deep convolutional networks transform an input image $x \in \mathbb{R}^{c \times h \times w}$ into a sequence of internal activation maps that preserve the 2-D topology of the input $\{a^{(1)} \in \mathbb{R}^{c_1 \times h_1 \times w_1}, ..., a^{(n)} \in \mathbb{R}^{c_n \times h_n \times w_n}\}$ through a cascade of linear-nonlinear operations (i.e. $a^{(1)} = f_{\theta_1}(x)$, $a^{(2)} = f_{\theta_2}(a^{(1)})$, and so on). At each stage the receptive field of a single unit (the region of the input x to which $a^{(i)}$ is sensitive) grows due to the repeated application of convolutional filters and downsampling operations in each stage. The response of neurons in deeper stages are thus sensitive to larger and more complex visual features, a property also found in neurons of successively deeper stages of the primate ventral stream. This is well matched to the *expressivity* growth of subsequent stages: $f_{\theta_2} \circ f_{\theta_1}$ can

express more complicated functions than f_{θ_1} alone by virtue of the additional trainable parameters in f_{θ_2} and a second point-wise nonlinearity.

Dilating receptive field sizes and computational expressivity can be leveraged to learn invariances to more complicated features and transformations at subsequent stages through the application of distinct loss functions at each layer. This is the key idea behind the Layerwise Complexity-matched Learning (LCL) scheme of Parthasarathy et al. (2024b), where image patches of different sizes and subjected to a variety of transformations of different strength are used to train two stages of a network independently (with layerwise losses and without backpropogation between stages). However in biological vision, the inputs to each stage of the hierarchy arise from a common visual stimulus, and that input arrives in a continuous temporal stream, rather than as samples randomly transformed relative to a base image. In this work we address this mismatch by introducing an objective based on local spatial pooling, and integration over time. Both the spatial pooling region and the temporal duration increase with each successive layer, in proportion to the receptive field sizes.

4.3.1 An Artificial Video Transformation

Real-world video data would provide the most natural and ecologically valid approach for training our models, but they present challenges for fair comparison and experimental control. A large body of models – particularly those pretrained on ImageNet – derive their inductive biases from the statistics of static natural images, and shifting to real video datasets introduces substantial variation in the distribution of visual content (which can confound downstream model comparisons (Conwell et al., 2022)). Instead, we train our models on simulated video sequences with smoothly evolving transformations. Starting with a single image randomly selected from the ImageNet dataset, we take two random (resized) crops and linearly interpolate their crop parameters to produce a sequence of frames with gradually shifting viewpoints. Following Parthasarathy et al. (2024b) we additionally apply mild random photometric transformations

(contrast and brightness modulation, stochastic conversion between color and grayscale, and the addition of gaussian noise), to each frame, and omit the more aggressive augmentations (e.g., hue shifts, solarization etc.) used in many SSL training schemes (Grill et al., 2020). For full details see C.2. These stimuli preserve alignment with ImageNet-pretrained models, allowing for controlled comparisons across training paradigms. Furthermore, this strategy enables fine-grained parametric control over the complexity of spatiotemporal transformations, providing a powerful tool for probing the relationship between transformation structure and representational learning.

4.3.2 Architecture

Following Parthasarathy et al. (2024b), we adopt AlexNet with batch normalization as our backbone architecture (Ioffe and Szegedy, 2015; Krizhevsky et al., 2012). While more recent architectures allow for large gains in performance on computer vision tasks, they provide minimal improvement in terms of neural predictivity, and the simplicity of the AlexNet architecture (a cascade of linear-nonlinear downsampling operations) strengthens the analogy between the model's computations and the initial feedforward cascade of transformations observed in the ventral stream (El-Shamayleh et al., 2013; Ziemba et al., 2016).

Our network is constructed as a cascade of three stages $\{f_{\theta_1}, f_{\theta_2}, f_{\theta_3}\}$, corresponding to primate visual areas V1, V2, and V4, respectively (see Fig. 4.1). The first two stages consist of a convolution, a halfwave rectifying (ReLU) nonlinearity, batch normalization, and a Max Pooling operation (which includes spatial downsampling by a factor of two). Thus the computational capacity of $f_{\theta_2} \circ f_{\theta_1}$ is approximately double that of f_{θ_1}. To double the relative capacity again, f_{θ_3} includes 2 stages of linear-nonlinear operations and concludes with a final downsampling operation. Note that this slightly deviates from the original AlexNet architecture which has 3 linear-nonlinear blocks before the third downsampling layer. The MMCR loss function is computed through additional nonlinear "projection heads", as is standard in SSL. Following Parthasarathy et al. (2024b), each of these projection heads (denoted g_{ϕ_1}, g_{ϕ_2} and g_{ϕ_3}) are implemented with 1-hidden layer

MLPs.

4.3.3 Objective

The MMCR objective function. Contrastive self-supervised learning methods aim to learn a representation, f_θ, that is invariant to some set of random input transformations, τ_ρ, and simultaneously discriminative across distinct inputs (Bardes et al., 2022; Chen et al., 2020b; Yerxa et al., 2023; Zbontar et al., 2021). Maximum manifold capacity representations (MMCR) cast SSL as capacity optimization and aim to maximize the number of linearly separable "transformation manifolds," that can be stored in the representation space produced by f_θ (Yerxa et al., 2023). For each input image in a dataset (notated as a vector $\mathbf{x}_b \in \mathbb{R}^{\mathbb{D}}$) we generate samples from the corresponding transformation manifold by applying the random transformation k times, yielding manifold sample matrix $\tilde{\boldsymbol{X}}_b \in \mathbb{R}^{D\times k}$. Each transformed image is mapped to a d-dimensional response space by f_θ and projected onto the unit sphere yielding manifold response matrix $\boldsymbol{Z}_b \in \mathbb{R}^{d\times k}$. The centroid $\boldsymbol{c}_b$ of the transformation manifold is then approximated by taking the sample mean (averaging across the k columns). For a set of images $\{\mathbf{x}_1, ..., \mathbf{x}_B\}$ we compute normalized response matrices $\{\boldsymbol{Z}_1, ..., \boldsymbol{Z}_B\}$ and assemble their corresponding centroids into matrix $\boldsymbol{C} \in \mathbb{R}^{d\times B}$.

Given the responses and their centroids, the MMCR loss function can be written simply:

$$\mathcal{L}_{\text{MMCR}} = -||\boldsymbol{C}||_*, \tag{4.1}$$

where $||\cdot||_*$ indicates the nuclear norm. Minimizing this objective simultaneously encourages transformation invariance and image-to-image discriminability, the two key conceptual ingredients in SSL, through a single term (Yerxa et al., 2023). This is because individual centroids, as averages of unit vectors, have norms that are bound above by 1, and are larger when the samples from the transformation manifold are more aligned (i.e., the objective encourages invariance). Additionally, maximizing the nuclear norm encourages distinct centroids to be as near orthogo-

nal to each other as possible (encouraging discriminability). While many SSL objective functions have been proposed, most require pairwise comparisons between the transformed inputs, and thus have complexity that is quadratic in the number of number of samples from the transformation manifold, k. MMCR has the unique distinction of having constant complexity: the size of $\mathbf{C}$ is independent of k. While this is of minimal importance in standard SSL settings where $k = 2$, our learning scheme relies on the use of large k, for which MMCR is uniquely well suited. It is worth noting that MMCR is not a direct optimization of capacity in general (Chou et al., 2024; Chung et al., 2018), but introduces a set of simplifying assumptions, such as elliptically symmetric manifold geometries, in order to arrive at a tractable objective function that bears some similarity to biological computations.

Local spatiotemporal MMCR. Rather than random samples from a transformation manifold, we train our network using transformations that smoothly vary across T frames (see Section 4.3.1). This choice allows us to derive a more ecological learning signal arising from *temporal and spatial proximity*. For input videos $\mathbf{x}_b \in \mathbb{R}^{T\times c\times h\times w}$ we compute corresponding feature maps produced by our 3 network stages $\mathbf{a}^{(i)} \in \mathbb{R}^{T\times c_i\times h_i\times w_i}$. To modulate the complexity of the learning task assigned to each network stage, we adjust the spatial and temporal extent of transformation manifolds on which the MMCR objective operates (see Fig. 4.1). We define $\mathrm{LSP}(s)$ as a local spatial pooling operation that averages over the central s pixels of a feature map. Similarly, let $\mathrm{LTP}(t)$ be a local temporal pooling operation that averages over the central t frames. For each input video $\mathbf{x}_b$, each network stage produces a single centroid:

$$\mathbf{c}^{(1)} = \mathrm{LTP}(\frac{t_f}{4}) \circ \Pi \circ \mathrm{LSP}(s_f) \circ g_{\phi_1} \circ f_{\theta_1}(\mathbf{x}_b) \tag{4.2}$$

$$\mathbf{c}^{(2)} = \mathrm{LTP}(\frac{t_f}{2}) \circ \Pi \circ \mathrm{LSP}(s_f) \circ g_{\phi_2} \circ f_{\theta_2} \circ f_{\theta_1}(\mathbf{x}_b) \tag{4.3}$$

$$\mathbf{c}^{(3)} = \mathrm{LTP}(t_f) \circ \Pi \circ \mathrm{LSP}(s_f) \circ g_{\phi_3} \circ f_{\theta_3} \circ f_{\theta_2} \circ f_{\theta_1}(\mathbf{x}_b) \tag{4.4}$$

where $\circ$ denotes function composition, and Π normalizes vectors over channels (projecting them onto the unit hypersphere). Note that the temporal pooling in each stage operates over successively longer durations. The spatial pooling operates over different *effective* window sizes, since the channels of each successive stage are sampled at half the resolution of the previous stage. Finally, the MMCR losses for each stage are computed from the nuclear norm of the centroid matrix obtained from a batch of input videos, and summed: $\mathcal{L} = \mathcal{L}_{\text{MMCR}}(\mathbf{C}^{(1)}) + \mathcal{L}_{\text{MMCR}}(\mathbf{C}^{(2)}) + \mathcal{L}_{\text{MMCR}}(\mathbf{C}^{(3)})$. The parameters of each layer $\{\theta_i, \phi_i\}$ are adjusted using only gradients of the corresponding stage, $\nabla_{\theta_i,\phi_i}\mathcal{L}_{\text{MMCR}}(\mathbf{C}^{(\mathbf{i})})$, with no backpropagation of error signals across stages.

We optimized the aforementioned architecture using this objective function, with videos generated from base images drawn from the ImageNet-1k dataset. For the sake of computational efficiency we used the minimum allowable 8 frames and choose the pooling region sizes to be global at the output of the final stage ($T = t_f = 8$ and $s_f = 7$). For more optimization details see Appendix C.3.

4.3.4 Evaluating Neural Predictivity

Evaluation methodology. We evaluate models on their ability to predict trial averaged firing rates via linear regression from their internal representations. Following Schrimpf et al. (2018), we use partial least squares (PLS) regression with 25 components to map from model feature maps to predicted mean firing rates. We compute the Pearson correlation coefficient between the predicted and observed responses for each neuron, and use the median over neurons as the "score" for a given evaluation. Finally, we use k-fold cross validation and report the average score over test splits (for details see C.1).

Physiology datasets. For areas V1 and V2 we use neural recordings from Freeman et al. (2013); Ziemba et al. (2016). The dataset contains single unit responses for 102 V1 neurons and 103 V2 neurons to texture images and their spectrally matched noise counterparts (with a total of 450

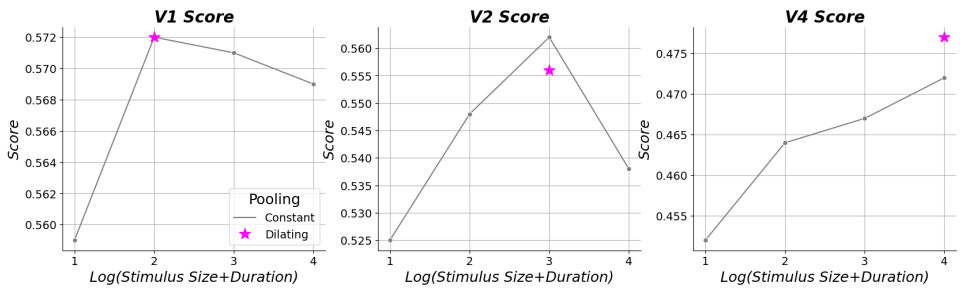

Figure 4.2: Neural predictivity as a function of task complexity. Following Parthasarathy et al. (2024b) we examine perfomance of a network trained using "constant" (global) spatial and temporal pooling, and modulate task difficulty by varying the size of input images and the strength of the spatial transformations they are subjected to (which is analgous to video duration). Our layerwise model, with pooling size/duration proportional to receptive field sizes in each stage, reaches near-peak performance in explaining data in all three cortical areas (red star), when trained over a single set of full-size video inputs.

images). For area V4 we use recordings from Lieber et al. (2024). The dataset contains responses of 211 single units in V4 that were evoked by natural images subjected to a parametrically controlled degree of "texturization," (with a total of 480 images). For further details on each dataset, see C.1.

4.4 Neural Alignment

We begin by replicating and extending the core result of Parthasarathy et al. (2024b), to verify the complexity-matching justification for our architecture and objective. We trained networks using global average pooling at each stage, while varying the spatial region (stimulus size) and the duration of the training videos (which determines the amplitude of the applied spatial transformations - see C.2 for details on the augmentation procedure). We find that the successive stages of the model are best at explaining data from their corresponding cortical area when the stimulus size and duration are proportional to their receptive field size (Fig. 4.2). In comparison, when a single network is trained on a single set of full-size videos using stage-adapted ("dilated") pooling, which modulates the task complexity at each layer directly through the architecture, it achieves (approximately) peak performance in explaining data in all three cortical areas.

Next, we compare the neural predictivity of our model with that of a set of architecture and

data-diet matched models, as well as handcrafted baseline models of V1 and V2 (see Fig. 4.3). Specifically, we consider the following baselines:

- **Supervised:** An AlexNet model trained with standard supervised cross entropy loss on ImageNet-1k. We use the standard weights from the pytorch library.
- **Robust:** An AlexNet model trained with L2-adversarial learning (Madry et al., 2018) that has been shown to provide a strong model of early cortical responses (ranking 3^{rd} of 450 models in the Brain-Score leaderboard for this V2 dataset) (Parthasarathy et al., 2024b; Schrimpf et al., 2018). We use model weights obtained from (Chen et al., 2020a).
- **Random:** An AlexNet with randomly initialized weights.
- **SteerPyrV1:** A 5-scale 4-orientation steerable pyramid decomposition (Simoncelli and Freeman, 1995), augmented with simple and complex cell nonlinearities which rectify the (components of) the complex-valued coefficients and compute the complex modulus, respectively.
- **SteerPyrV2:** Complex cell outputs of the SteerPyrV1 module are subjected to a second stage of steerable pyramid filtering and complex-cell nonlinearities. This is an instantiation of a second-order scattering transform (Bruna and Mallat, 2013).

First we note that these results reproduce several key results from previous literature: (1) handcrafted filters that are tuned for orientation and spatial frequency substantially outperform learned models on the V1 dataset; (2) for V1, supervised training provides only a modest increase in predictivity relative to the baseline model with random weights; and (3) L2 adversarial training induces significantly more alignment than standard supervised training in both V1 and V2 (Parthasarathy et al., 2024b). Finally, we see that our layerwise-trained ST-MMCR model performs similarly to the robust (adversarially trained) baseline and outperforms end-to-end supervised learning in all three areas.

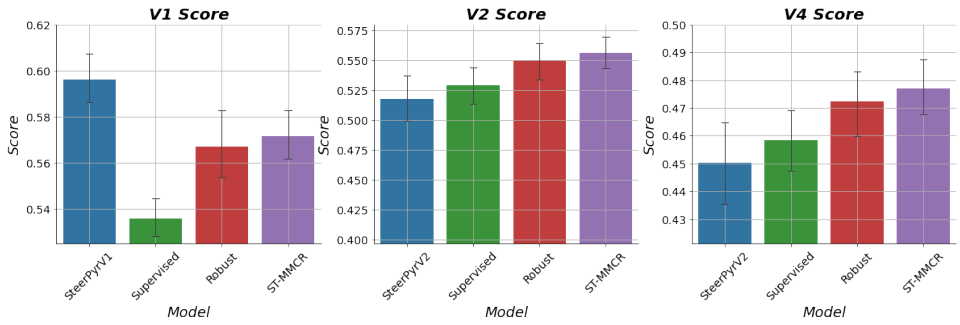

Figure 4.3: Neural Predictivity Across Baseline Models. For each model and neural dataset, we apply the procedure described in 4.3.4 for all model layers and report the highest score. For each panel, the lowest score on the ordinate corresopnds to the mean performance of an architecture-matched model with random weights. Error bars indicate 95% confidence intervals over train-test splits.

We additionally ablate two key aspects of our training algorithm: the stage-wise isolation of gradients, and the application of losses at internal layers of the network. To do so, we trained one network with an identical loss function and architecture, but without "gradient cutting," so for example $\mathcal{L}_{MMCR}(\mathbf{C}^{(2)})$ impacts the parameters of the first stage θ_1; we denote this ablation "E2E Gradient" in Fig. C.2. We additionally train a third model that uses end-to-end gradients, but further ablate all internal losses, so $\mathcal{L}_{MMCR}(\mathbf{C}^{(3)})$ alone is used to train all three stages ("E2E Loss" in Fig. C.2). Both modifications produce modest reductions in neural predictivities relative to our default settings across all three brain areas, suggesting at the very least that our simplified credit assignment procedure is not harming model-brain alignment (see Appendix C.1 Fig. C.2 for full results).

4.4.1 Partitioning Predictivity

As demonstrated in Fig. 4.3, and noted in previous literature (Cadena et al., 2024; Conwell et al., 2022), models obtained using the same architecture and training dataset tend to have similar predictivity of cortical responses.This does not necessarily imply that all such models are capturing the same aspects of neural responses, or encoding those aspects in the same format as the biological neurons (since predictions are based on linear regression from a large number of

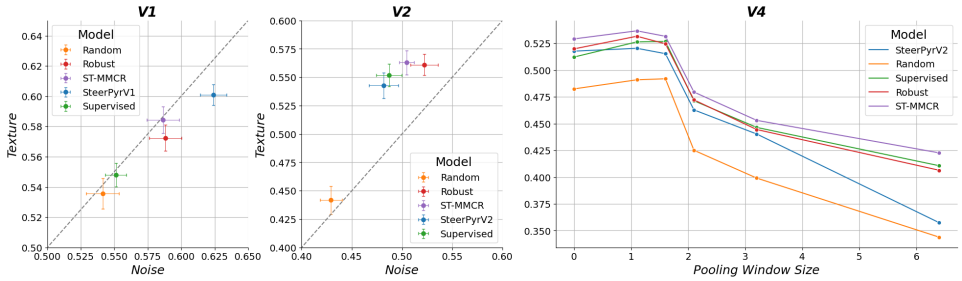

Figure 4.4: Partitioning Predictivity by Stimulus Parameters. Model predictions for each neural dataset separated by stimulus type. In V1 and V2 (left two plots) images are split into naturalistic textures and spectrally matched noise images. In V4, images are parameterized by the size of the windows over which they are "texturized."

model neurons). To address this, we examined two means of "decomposing" each model's neural predictivity.

Partitioning by stimulus parameters. Each neural dataset under consideration leverages images that are parametrically generated or distorted. Freeman et al. (2013) presented synthesized naturalistic textures and spectrally matched noise images (having identical 2nd-order spectral energy but without higher-order structure). Lieber et al. (2024) generate a continuum ranging from natural images to their fully texturized counterparts by drawing samples that are statistically matched over progressively larger "pooling windows" (a pooling window size of 0 corresponds to an unperturbed natural image and the largest window size corresponds to a fully texturized counterpart – see C.1 for examples).

Figure 4.4 shows comparisons of neural predictivity for texture and noise images in V1 and V2, and across pooling window sizes in V4. For V2 units (but not V1 units) responses to naturalistic textures are significantly better predicted than their spectrally matched noise counterparts (left two panels, Fig. 4.4, and for V4 units predictivity falls sharply for pooling window sizes greater than 1.6 visual degrees. These behaviors are generally shared between candidate models.

Partitioning by number of model units, using sparse regression. Even in the case where two models make identical predictions for neural responses to *all images*, they may encode the

information used to make these predictions in disparate formats. As an extreme example, consider an activation map produced by a candidate model $a \in \mathbb{R}^{c \times h \times w}$ where only a single element is predictive of an individual neuron's firing rate. In such a case the optimal linear mapping uses a weight vector that is one-hot; an alternative model might produce a mapping with all non-zero coefficients, yet the composition of network backbones with their respective linear maps can compute the same function. To assess the extent to which linearly predictive information is isolated or distributed across our suite of candidate representations, we trained linear mappings that are restricted to use k non-zero coefficients. Specifically, we perform feature selection on the training set of each split by adjusting the strength of an $L1$ (LASSO) regularizer until a sparse set of k coefficients is selected, then re-fit neural responses with the selected features using ridge regression (i.e., the "relaxed LASSO" – see Appendix C.5 for details).

The results of this analysis are shown in Fig. 4.5. First we note that trained models are generally better separated from their random-weight counterparts in the highly sparse regime. For example in V1 (left panel), the random model only narrows the gap to the supervised model when it has access to hundreds of model units to interpolate individual neural firing rates. In V2 (center panel) models are primarily separated by the amount of information offered by the few most predictive units, with the predictivity gains per unit (slopes of each curve) being approximately matched. These results suggest that bottlenecked regressions can provide a useful signal, particularly for disentangling alignment induced by training from alignment arising due to the architecture.

4.5 Behavioral Alignment

Several studies have provided evidence that representations whose early stages are more aligned with primary visual cortex support more human-like behavior in object representation tasks (Dapello et al., 2020; Feather et al., 2023; Marques et al., 2021). Recently, Parthasarathy et al.

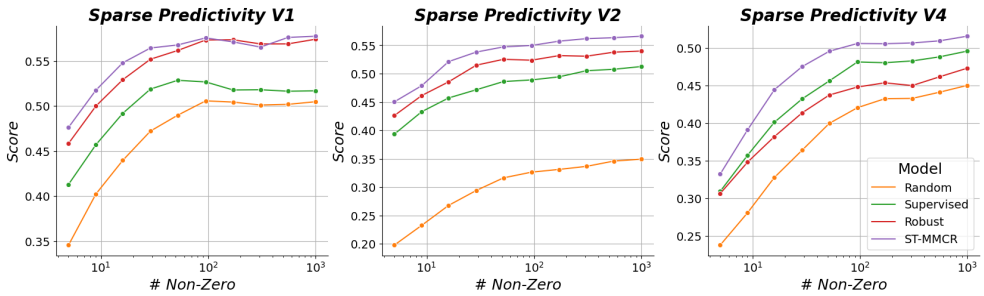

Figure 4.5: Sparse Neural Predictivity. For each model-benchmark pair we use lasso to select k model responses to predict each individual neuron's firing rate via ridge regression. We vary k over 10 logarithmically spaced values between 5 and 1,000.

(2024b) extended this result by training a downstream classifier on a self-supervised V2 front end model. However, it remains unclear whether this trend will hold in deeper cortical areas, or even whether V1 or V2 predictivity is more important for behavioral alignment.

We explicitly investigate this question by training classifiers to operate on responses of models of each brain-area in Section C.4 with results shown in Fig. C.4. We find that the classifier trained on top of the V2-like layer has the highest out-of-distribution generalization and alignment to human behavior, and we thus compare its performance with our baseline models in Fig. 4.6. Specifically, we utilized the suite of out-of-distribution (OOD) stimuli introduced by Geirhos et al. (2021) to test the models' ability to generalize as well as their propensity to make category judgments similar to human observers. The ST-MMCR model exhibits substantially lower accuracy on the standard ImageNet recognition task (Fig. 4.6 left panel) than the fully supervised model. But the ST-MMCR with a V2 frontend substantially outperforms *all* architecture-matched baselines in terms of OOD generalization, human choice, and error consistency (Fig. 4.6 right three panels).

4.6 Model Sensitivities Yield Divergent Predictions

Just as models that produce identical predicted firing rates can encode predictive information differently, even models with *identical feature maps* (for all images in a neural dataset) can exhibit

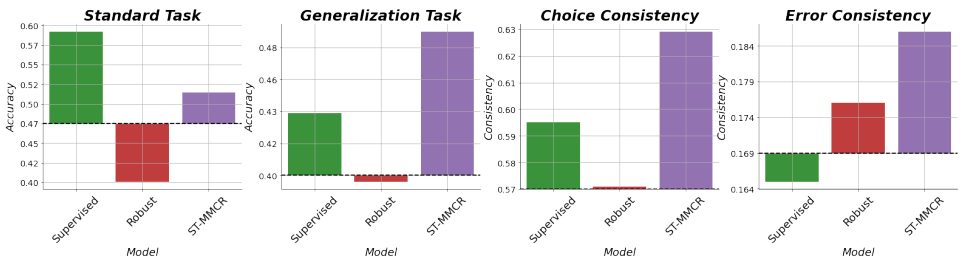

Figure 4.6: Behavioral Evaluations. In the first two panels we show the top-1 accuracy on the standard ImageNet-1k validation set and on the out-of-distribution images from Geirhos et al. (2021), respectively. In the right two panels we show the rate at which models make the same choice as human observers, and the alignment between model and human error patterns (as described in Geirhos et al. (2021)). In all panels, the black dashed lines indicates the performance of a classifier trained on top of a front end with random weights, with the same number of frozen layers as the ST-MMCR model.

significantly different local geometries. Consider a candidate model composed with a mapping function used to predict neural responses, $\hat{r}(x) = (m_l \circ f_l)(x)$. When m_l and f_l are both differentiable (as is the case when f_l computes the activations of some layer of a DNN and m_l is affine), models need not only predict neural responses, but can also predict response sensitivities via the Jacobian, $J(x) = \frac{\partial r}{\partial x}$. In particular, sensitivity to local perturbations is governed by a representations Fisher information matrix (FIM), $\mathcal{I}(x) \in \mathbb{R}^{D \times D}$ where D is the dimensionality of the input signal x (Seriès et al., 2009).

In the case of an additive white gaussian noise model the FIM is simply, $\mathcal{I}(x) = J^T(x)J(x)$. Berardino et al. (2017) show (1) the most and least noticable local perturbation directions around an input x are given by the eigenvectors associated with the largest and smallest eigenvalues of $\mathcal{I}(x)$ and (2) while the eigendecomposition of $\mathcal{I}(x)$ is prohibitively compute intensive when x is a high resolution image, the extremal distortions can be reliably estimated using the power iteration method (Mises and Pollaczek-Geiringer, 1929). We synthesize maximal and minimal "eigendistortions" for each of the three candidate models of V4 for a single image using the visualize the results in Fig. 4.7. Despite similar predictivity levels, the three models make visually different predictions about the population's sensitivities: the ST-MMCR model is maximally perturbed by increasing the contrast of a key feature, while the robust model seems to remove this feature,

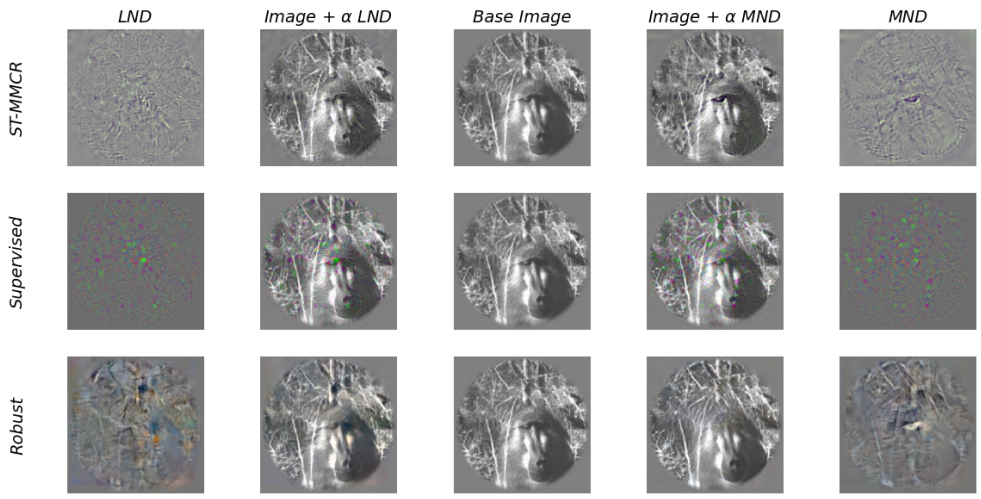

Figure 4.7: Eigendistortions of Neural Firing Rate Predictors. In the right/leftmost columns we visualizes the maximal and minimal eigendistortions of the V4 firing rate predictor derived from each candidate model respectively. These are unit vectors in the pixel space, and are scaled to fill the dynamic range of the display. In the second right/leftmost columns we show the eigendistortions added to the base image (middle column) with strength $\alpha = 10.0$. Here we map the lowest value over these three columns to 0 and the largest to 1 independently for each row (so within one row, the central three columns all use the same color map).

while the standard supervised model produces a less interpretable “polka-dot" like perturbation. See Appendix C.6 for more details and examples.

4.7 Discussion

We have developed a novel local layerwise self-supervised learning scheme, termed ST-MMCR, which naturally implements “complexity matching” by leveraging dilating spatial and temporal receptive fields, rather than relying on hand-crafted augmentation schemes as was done in previous work (Parthasarathy et al., 2024b). Our formulation is appealing from the perspective of biological plausibility. First, backpropagation of error signals arising from a single global objective is widely considered implausible as a model for biological brains (Stork, 1989), and the stagewise local computation of our model offers a potential alternative. Moreover, many converging lines of evidence indicate that primate neocortex is constructed of local circuits that are replicated -

to first approximation the entire cortical sheet consists of identical computational units (Douglas and Martin, 2004). Specific computations arise through parallel, sequential and recurrent connectivity of these circuits, with details largely learned during development and in response to life experience. This provides motivation for convolutional neural networks in general, and for a local common objective function computation. Both the architecture and objective of our model aim to satisfy this description: the three stages are implemented with identical computations, apart from the change in temporal pooling duration. Rectified linear operations, normalization of responses over groups of "cells", and spatial and temporal averaging are common elements of many cortical models (Carandini and Heeger, 2012).

To better understand how candidate models differ beyond overall predictivity, we performed a series of fine-grained analyses. By partitioning neural data according to stimulus parameters, we revealed systematic differences in which response components are predictable from model activations. Notably, V2 representations showed stronger alignment with naturalistic texture responses, while V4 predictivity degraded as natural images were texturized. Sparse regression analyses revealed that models initialized with random weights can achieve modest predictivity by interpolating neural responses using a large number of units, whereas trained models achieve comparable or superior predictivity using a much smaller set of informative features. These results suggest that bottlenecked regressions can surface alignment signals that are otherwise masked by overparameterization. Finally we found that our biologically motivated pretraining procedure induced increased alignment to human behavior in object classification tasks. Taken together, these results suggest that simplified credit assignment procedures are not only capable of producing brain-like representations, but the biological constraints they employ may play a key role in shaping the structure of real visual systems.

Future work can improve upon our approach by increasing the ecological relevance of the training stimuli, i.e. by using a suitably large and diverse set of natural videos. Furthermore, the final MMCR loss, which relies on nuclear norm computation, does not currently have an

obvious counterpart in the world of biological modeling, but dynamic circuits for optimizing other objectives may offer a path for future development (Pehlevan et al., 2017). Finally it would be interesting to consider alternative objectives that incorporate either more nuanced features of capacity maximization (Chou et al., 2024; Yerxa et al., 2024a) or more general non-ellipsoidal manifold geometries.

5 Discussion

5.1 Summary

This thesis explores the design and use of representations optimized for the capacity to store ellipsoidal manifolds representing sets of related inputs on the surface of a high dimensional sphere. Stated as such this seems a strange choice, but this problem setting is richer than it seems; the freedom to specify what principle determines if two data points are related (or belong to the same neural manifold) provides a surprising amount of flexibility to specify desired properties of the representation. [1]

We began by considering the setting of transformation based self-supervised learning, which amounted to defining sets of points to be related (i.e. belonging to the same elliptical manifold) if and only if they correspond to distinct views of the same base image. We found that casting the SSL problem in this light led to a remarkably simple single-term objective function, alleviating the need to carefully set a tradeoff parameter specifying the relative importance of local invariance and global informativity. The resulting loss function also had the unique property of having constant complexity in terms of the number of views used during training. This allowed us to effectively and efficiently scale the number of transformations applied, which led to state of the art performance for SSL on the ImageNet-1k classification task.

[1] It is also worth noting that, though throughout this work we approximated manifold geometries with ellipses, finding a more general formulation of manifold capacity that is amenable to optimization would be an interesting direction for future work.

The following chapters each addressed a limitation of the framework introduced in the first. For one, it is obvious that a representation should not discard all information about the transformations that relate a pair of images, after all many tasks may rely on information about the transformation rather than what features are conserved across views. This motivated us to propose a dual objective, with a standard invariance based term and a second equivariance based term that requires transformation information be preserved. Importantly, we did so in a manner that respects the core principle of SSL, teaching the network using paired examples rather than through direct supervision. In the capacity parlance, we achieved this by defining a new type of manifold where membership is defined by the application of a matching transformation, and requiring a high capacity to store both "manifolds of transformations" and "manifolds of images" simultaneously. Encouragingly, incorporating this second term led to systematic increases in neural predictivity, and even exceeded the current state of the art for predicting responses in area IT.

Finally, we incorporated the constraint of layerwise learning into capacity maximization. When each module of a hierarchical representation has its own task, it is important to scale the task difficulty as the network expressivity grows with depth. We introduced a natural way to achieve this "complexity matching" by simply changing the criteria for manifold membership with depth; the deeper more expressive parts of the network must compress manifolds consisting of points within a larger spatiotemporal neighborhood. We demonstrated that, despite using a suboptimal optimization scheme, layerwise capacity maximization produces strong representations that predict responses in V2 with nearly state of the art accuracy, and can even serve as a strong model as deep into cortex as V4.

5.2 Future Directions

5.2.1 Increasing the Ecological Relevance of Inputs

Each of the previous chapters, to various extents, participated in the conceit that the image transformations central to modern SSL are a reasonable substitute for the dynamics of natural videos. A natural next step is to test whether the algorithms developed here would benefit from exposure to this more realistic type of signal. Achieving strong performance on image based tasks (including predicting neural responses to static images) depends critically on the scale and diversity of the dataset used to train a model (Parthasarathy et al., 2023), and the community has yet to settle on a standard and publicly available dataset of videos in the same way that ImageNet quickly emerged as a standard choice. However, video understanding is a quickly becoming a frontier topic in machine learning and this gap will surely be filled. The standard MMCR objective employed in Chapters 2 and 4 could of course be adapted by using temporal proximity to define manifold membership. The equivariant setup of Chapter 3 could also have interesting applications. Applying the MMCR loss to the temporal derivative of the representation would be analagous to the equivariant term, encourage that information about dynamics be encoded alongside information about the features that are constant in time, and could be thought of as an instantiation of the "neural straightening" hypothesis (Hénaff et al., 2021).

Curating model inputs to better match cortical inputs seems an important step towards producing more accurate models. Besides being dynamic, the visual experience of a human is egocentric, foveated, and punctuated by saccadic eye movements. Surely evolution does not play dice, and so the inclusion of these features in the input data or developing theories that suggest why these mechanisms emerge seem a ripe topic for future research.

5.2.2 Increasing and Investigating the Plausibility of Optimization

A growing body of evidence, including Chapter 4 suggests that self-supervised learning objectives can be optimized in isolated stages without severely degrading the quality of the resulting representation. This is not the case for many other tasks, such as supervised object recognition. We speculate that this is because SSL tasks contain natural "sub-tasks," whose solutions contribute to the overall goal (for instance invariance to small translations can explicitly serve as step towards invariance to larger translations). Future work could investigate and formalize what properties of a learning goal lead to objectives with this property, which could place constraints on future normative theories at an algorithmic level.

It would also be interesting to develop mechanistic models for the optimization capacity based objectives. For example, it is possible to derive neural circuits that when paired with plausible synaptic learning rules (i.e. Hebbian updates) optimize "Similarity Matching" objectives that are conceptually similar to the objective functions common in transformation based SSL (Pehlevan et al., 2017). After all, some such circuit must exist if real neuron's are optimized for capacity throughout development.

A | Learning Efficient Codes for Natural Images using Maximum Manifold Capacity Representations

A.1 Optimal Embeddings

Recall the setting of self-supervised learning as described in Balestriero and LeCun (2022): given a dataset $\boldsymbol{X'} = [\boldsymbol{x}_1, ..., \boldsymbol{x}_N]^T \in \mathbb{R}^{N \times D'}$ we construct a new dataset by creating k randomly augmented views of the original data, $\boldsymbol{X} = [\text{view}_1(\boldsymbol{X'}), ..., \text{view}_k(\boldsymbol{X'})] \in \mathbb{R}^{Nk \times D}$. The advantage of doing so is that we can now leverage the knowledge that different views of the same underlying datapoint are *semantically related*. We can express this notion of similarity in the symmetric matrix $\boldsymbol{G} \in \{0, 1\}^{Nk \times Nk}$ with $\boldsymbol{G}_{ij} = 1$ if augmented datapoints i and j are semantically related (and $\boldsymbol{G}_{ii} = 1$ as any datapoint is related to itself). We can normalize $\boldsymbol{G}$ such that its rows and columns sum to 1 (so rows of $\boldsymbol{G}$ are k-sparse with nonzero entries equal to $1/k$).

Now let $\boldsymbol{Z} \in \mathbb{R}^{Nk \times d}$ be an embedding of the augmented dataset. Then we have $\boldsymbol{GZ} = [\boldsymbol{C}, ..., \boldsymbol{C}]^T$ where $\boldsymbol{C}$ is the matrix of centroid vectors introduced above, and the number of repetitions of $\boldsymbol{C}$ is k. Then because $\boldsymbol{\sigma}([\boldsymbol{C}, ..., \boldsymbol{C}]) = \sqrt{k}\boldsymbol{\sigma}(\boldsymbol{C})$ we can write MMCR loss function as,

$$\begin{aligned}\mathcal{L} &= -||\boldsymbol{GZ}||_* \\ &= -||\boldsymbol{Q\Lambda Q}^T\boldsymbol{USV}^T||_* \\ &= -||\boldsymbol{\Lambda Q}^T\boldsymbol{US}||_*\end{aligned} \tag{A.1}$$

Where we have taken the eigendecomposition of $\boldsymbol{G}$ which is real and symmetric and the SVD of $\boldsymbol{Z}$, and then used the fact that the singular value spectrum is invariant under left or right orthogonal transformations. We now show that a global optima of this objective is achieved when the left singular vectors of $\boldsymbol{Z}$ are the eigenvectors of $\boldsymbol{G}$ and the singular values of $\boldsymbol{Z}$ are proportional to the eigenvalues of $\boldsymbol{G}$. Throughout we will assume that the size of the dataset is greater than the dimensionality of the embedddings, $N > d$, as is the case in practical applications. First we prove a simple lemma about the spectrum of matrices who are extended by zeros (i.e. embedded in a higher dimensional space).

Lemma A.1: For $\boldsymbol{A} \in \mathbb{R}^{N\times N}$, $\boldsymbol{B} \in \mathbb{R}^{N\times d}$ with $d < N$, $||\boldsymbol{AB}||_* = ||\boldsymbol{A}\tilde{\boldsymbol{B}}||_*$ where $\tilde{\boldsymbol{B}} = [\boldsymbol{B}, \boldsymbol{0}] \in \mathbb{R}^{N\times N}$.

Proof: First note that $\boldsymbol{A}\tilde{\boldsymbol{B}} = [\boldsymbol{AB}, \boldsymbol{0}]$ so it suffices to show that for arbitrary $\boldsymbol{X}$ that $\boldsymbol{\sigma}(X) = \boldsymbol{\sigma}([\boldsymbol{X}, \boldsymbol{0}])$. Taking the SVD of $\boldsymbol{X}$,

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{U} & \widetilde{\boldsymbol{U}} \end{bmatrix} \begin{bmatrix} \Sigma \\ \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{V}^T \end{bmatrix} = \boldsymbol{U\Sigma V}^T$$

Then a valid singular value decomposition for $\tilde{X}$ is

$$\tilde{\boldsymbol{X}} = \begin{bmatrix} \boldsymbol{U} & \widetilde{\boldsymbol{U}} \end{bmatrix} \begin{bmatrix} \Sigma & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{V}^T & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{I} \end{bmatrix}$$

Clearly then, $||\boldsymbol{X}||_* = ||\tilde{\boldsymbol{X}}||_*$

Theorem: The proposed loss achieves a global minimum when the left singular vectors of $\boldsymbol{Z}$

are the eigenvectors of G, and the singular values of Z are proportional to the top d eigenvalues of G.

Proof: Let $\tilde{Z} = [Z, \mathbf{0}] \in \mathbb{R}^{N \times N}$. By Lemma A.1 we have $||GZ||_* = ||G\tilde{Z}||_*$. Von Neumann's trace inequality can be used to show $||G\tilde{Z}||_* \leq \sum_{i=1}^{Nk} \sigma_i(G)\sigma_i(\tilde{Z})$ (see Marshall et al. (1979) for proof). Examining (4) it is clear that this bound is achieved when $U = Q$. The problem can therefore be reduced to the constrained optimization problem,

$$\min_{\sigma_i(\tilde{Z})} \sum_{i=1}^{Nk} \sigma_i(G)\sigma_i(\tilde{Z})$$
$$\text{subject to } \sum_{i=1}^{Nk} \sigma_i(\tilde{Z})^2 = Nk$$

where the constraint comes from the fact that columns of Z are unit vectors. Intuitively, we are maximizing the inner product between a fixed vector $\sigma(G)$ and a vector with fixed L2 norm. The solution of course is to align the two vectors as closely as possible, i.e. when $\sigma_i(\tilde{Z}) \propto \sigma_i(G)$ for $i = 1, ..., d$. It is worth noting that by construction $\sigma_i(\tilde{Z}) = 0$ for $i > d$ and the columns of U associated with these zero valued singular values are unconstrained.

A.2 Pytorch Style Pseudocode for MMCR

```
# h: encoder
# g: projection head
# T: momentum temperature
# B: batch size
# K: number of augmentations
# D: projector output dimensionality
#
# lmbda: trade-off parameter
```

```
f_o, g_o = ResNet50(), MLP() # online networks

# initialize momentum network with identical params
f_m, g_m = f_o.copy, g_o.copy()

# momentum networks are not updated via gradient descent
f_m.requires_grad = False
g_m.requires_grad = False

for x in loader:
    # K randomly augmented views
    x = multi_augment(x) # B x K x H x W

    # push through encoder and projector
    z_o = g_o(h_o(x)) # B x K x D
    z_m = g_m(h_m(x)) # B x K x D
    z = concatenate(z_o, z_m, dim=1) # append outputs

    # project onto unit sphere
    z = normalize(z, dim=-1)

    # calculate centroids (mean over augmentation axis)
    c = z.mean(dim=1) # B x D

    # calculate singular values
```

```
U_z, S_z, V_z = svd(z) # batch svd
U_c, S_c, V_c = svd(c)

# calculate loss
loss = -1.0 * sum(S_c) + lmbda * sum(S_z) / B

# backward pass and optimization step
loss.backward()
optim.step()

# perform momentum update
with torch.no_grad():
    f_m.parameters() = (1 - T) * f_o.parameters()
                       + T * f_m.parameters()
    g_m.parameters() = (1 - T) * g_o.parameters()
                       + T * g_m.parameters()
```

A.3 Mean Field Theory Manifold Capacity Background Information

For completeness we summarize some of the central arguments from Chung et al. (2018), which develops the general form of manifold capactiy theory.

Mean Field Theory Recall the problem setting for manifold capacity analysis: given a set of P manifolds embedded in a feature space of dimensionality D, each assigned a random binary class label (Chung et al., 2018). Manifold capacity theory is concerned with the question: what

is the largest value of $\frac{P}{D}$ such that there exists (with high probability) a hyperplane separating the two classes? In the thermodynamic limit, where $P, D \to \infty$ but $\frac{P}{D}$ remains finite, the inverse capacity can be written exactly,

$$\alpha_M^{-1} = \mathbb{E}_{\vec{T}}[F(\vec{T})] \tag{A.2}$$

where, $F(\vec{T}) = \min_{\vec{V}} \left\{ \|\vec{V} - \vec{T}\|^2 \mid g_{\mathcal{S}}(\vec{V}) \geq 0 \right\}$, $\mathcal{S}$ is the set defining the manifold geometry (i.e. the set of vectors $\vec{S}$ that are points on an individual manifold), $\vec{T}$ are random vectors drawn from a white multivariate Gaussian distribution, and $g_{\mathcal{S}}(\vec{V}) = \min_{\vec{S}}\{\vec{V} \cdot \vec{S} \mid \vec{S} \in \mathcal{S}\}$, is the concave support function.

The KKT equations for this convex optimization problem are:

$$\begin{aligned} \vec{V} - \vec{T} - \lambda \tilde{S}(\vec{T}) &= 0 \\ \lambda &\geq 0 \\ g_{\mathcal{S}}(\vec{V}) - \kappa &\geq 0 \\ \lambda \left[g_{\mathcal{S}}(\vec{V}) - \kappa \right] &= 0. \end{aligned} \tag{A.3}$$

, where $\tilde{S}(\vec{T})$ is a subgradient of the support function. When the support function is differentiable, the subgradient is unique and equal to the gradient,

$$\tilde{S}(\vec{T}) = \nabla g_{\mathcal{S}}(\vec{V}) = \arg\min_{\vec{S} \in \mathcal{S}} \vec{V} \cdot \vec{S} \tag{A.4}$$

$\tilde{S}(\vec{T})$ is the unique point in the convex hull of $\mathcal{S}$ that satisfies the first KKT equation, and is called the "anchor point" for $\mathcal{S}$ induced by the random vector $\vec{T}$.

Equivalent Interpretation of Anchor Points For a given dichotomy (random binary class labelling) the weight vector of the maximum margin separating hyperplane can be decomposed into a sum of at most P vectors, with each manifold contributing a single vector, which lies within

the convex hull of the manifold. The position of said point point is a function of the manifolds position relative to all of the other manifolds in the space and depends on the particular set of random labels. Thus there exists a distribution of separating-hyperplane-determining-points for each individual manifold. Using the cavity method it can be shown that these points are none other than the anchor points that are involved in solving the optimization problem described above (Gerl and Krey, 1994).

Numerical Solution To solve the mean field equations numerically, one samples several random Gaussian vectors $\vec{T}$, and then for each $\vec{T}$, $\vec{V}$ and $\vec{S}$ are determined by solving the quadratic programming program given above. The capacity is then estimated as the mean value of F or the samples $\vec{T}$.

Manifold Geometries The way the capacity varies in terms of the statistics of the anchor points can be simplified by introducing two key quantities, the manifold radius R_M and manifold dimensionality R_M:

$$\begin{aligned} R_M^2 &= \mathbb{E}_{\vec{T}}[||\tilde{S}(\vec{T})||^2] \\ D_M &= \mathbb{E}_{\vec{T}}[\vec{T} \cdot \hat{S}(\vec{T})] \end{aligned} \tag{A.5}$$

where $\hat{S}(\vec{T})$ is a unit-vector in the direction of the anchor point $\tilde{S}$. In particular as discussed in the main text, the manifold capacity can be approximated by $\phi(R_M\sqrt{D_M})$ where ϕ is a monotonically decreasing function.

Elliptical Geometries In the case where the manifolds exhibit elliptical symmetries, the manifold radius and dimensionality can be written in terms of the eigenvalues of the covariance matrix of the anchor points:

$$\begin{aligned} R_M^2 &= \sum_i \lambda_i^2 \\ D_M &= \frac{(\sum_i \lambda_i)^2}{\sum_i \lambda_i^2} \end{aligned} \tag{A.6}$$

So, in this case R_M is the total variability of the anchor points, and D_M is a generalized par-

ticipation ratio of the anchor point covariance, a well known soft measure of dimensionality.

A.4 Additional Pre-training information

Settings for CIFAR/STL-10 We take the parameters of each augmentation directly from Zbontar et al. (2021), but for these lower resolution images we omitted Gaussian blurring and solarization augmentations. All models were trained for 500 epochs using the Adam optimizer (Kingma and Ba, 2014) with a learning rate of $1e-3$ and weight decay of $1e-6$. For all three methods we used a one hidden layer MLP with hidden dimension of 512 and output dimension of 128 for the projector head g. We swept batch size for each method and chose the one that resulted in the highest downstream task performance. For both SimCLR and Barlow Twins we found that a batch size of 128 was optimal (among 32, 64, 128, 256, and 512) for all 3 datasets. For MMCR there is a trade-off between batch size and the number of augmentations used, and the optimal value of that trade-off is highly dataset dependent. For CIFAR-10 and CIFAR-100 we used batch size of 32 and 40 views, and for STL-10 we used a batch of 64 with 20 views For Barlow Twins we used $\lambda = \frac{1}{128}$ which normalizes for the number of elements in the on-diagonal and off-diagonal terms in the loss. For SimCLR we used the recommended setting of $\tau = 0.5$. The overall performance of both baseline methods (and likely MMCR as well) could be increased with a more thorough hyperparameter search and by employing methodology that more closely matches the original works. For example, both methods would likely benefit from the combination of larger batch size, the use of the LARS optimizer (which is designed for large batch optimization), a learning rate scheduler consisting of linear warm-up followed by cosine annealing, longer training, and the use of more diverse augmentations (i.e. including solarization and gaussian blur). Additionally Barlow Twins reports that the representation can benefit from using a much larger projector network than we use. Because our goal was primarily to demonstrate that MMCR can produce representations that are comparable to these baselines rather than to produce state-of-the-art

results on small scale datasets we opted for simplifications wherever possible (using off the shelf Adam for optimization with a fixed learning rate, and fixing architectural hyperparameters like the projector dimensionality).

Settings for ImageNet-100 For ImageNet we more closely match the pre-training procedures of previous works. We use a batch size of 2048 and a smaller number of views for MMCR (4), and also use the full suite of augmentations from Zbontar et al. (2021). For the sake of efficiency we train for a reduced number of epochs (200). For MMCR and SimCLR we modified the projector hidden dimensionality to be 4096 for the projector head, following the original work (Chen et al., 2020b). For Barlow Twins we used the recommended 2-layer MLP with hidden and output dimensions of 8192, and set $\lambda = 5e - 3$, however these hyperparameters were optimal for the full ImageNet dataset, and not neccesarrily for ImageNet-100. We were unable to achieve better downstream performance using a ResNet-50 backbone than what has previously been reported in the literature for this dataset with a ResNet-18 backbone, therefore we report the ResNet-18 performance reported in (Da Costa et al., 2022). For SimCLR we use $\tau = 0.1$ which is the recommended setting for larger batch sizes.

Settings for ImageNet-1k: For ImageNet-1k we use mostly identical settings to ImageNet-100, but we increased the capacity of the projector network (using a 2 hidden layer MLP with hidden dimenisons of 8192 and output dimension of 512). We scaled the learning rate linearly with batch size: $\text{lr} = 0.6 \times \frac{\text{batch size}}{256}$. Additionally we reduce the number of pretraining epochs to 100. Finally for ImageNet-1k we found that employing a momentum encoder slightly boosted downstream performance (around +1% on ImageNet frozen-linear evaluation). Specifically, an identical encoding network and projector architecture is initialized with the same parameters as the initial "online" network, and during training the weights of this "momentum" network track a slowly moving average of the online network parameters (only the online network parameters are updated via gradient descent). Each augmented view is passed through both the online and momentum networks, and the resultant embeddings are all averaged to form the centroid vector

for a particular image in the batch. We used a momentum coefficient of 0.99.

Pre-training on 16 A100 GPUs using 8 views (our most compute intensive setting) takes approximately 32 hours.

A.5 Details of Representational Analyses

A.5.1 Manifold Capacity Analysis

For each pre-trained model, we extract layer activations across the ResNet hierarchy after a forward pass of a set of images. For class manifold analysis, the set of images contain 10 classes, where each class has 100 examples. Augmentation manifolds instead have 100 exemplars with 100 examples each. Following (Cohen et al., 2020), we take activations from all convolutional layers in ResNet-50 after a ReLU non-linearity. The specific extracted layers highlighted in bold fonts are given by Table A.1. The final analysis results are averaged over five data samplings with different random seeds and random projections of intermediate features to lower-dimension spaces (default 5000 dimensions).

A.5.2 Gradient Coherence Analysis

In Fig. 2.3, for each of the classes of CIFAR-10, we generate 100 batches of 32 augmentation manifolds of samples from a specific class (with 40 augmentations each). We then measure the gradient of the loss function for each batch during different stages of training, and compute the cosine similarity between every pair of gradients. Across all stages of training the mean cosine similarity between gradients generated from batches of the same class is larger than those from distinct classes (left column). This observation remains true when isolating the gradients of parameters from different stages of in the resnet-50 hierarchy (center and right columns, respectively).

A.5.3 Manifold Subspace Alignment

For Fig. 2.4 we generated 100 samples from the augmentation manfiolds of 500 images in the CIFAR-10 dataset. We then measure the mean subspsace angle (left column), fraction of shared variance (middle column) and centroid cosine similarity between each pair of manifolds. The same procedure was used for generating the data for Fig. 2.5.

Subspace Angle. Besides measuring the size and dimensionality of individual object manifolds we also wish to characterize the degree of overlap between pairs of manifolds. For this, we measure the angle between their subspaces (Knyazev and Argentati, 2002), which is a generalization of the notion of angles that applies to subspaces of arbitrary dimension.

Shared Variance. Object manifolds will generally have a lower intrinsic dimensionality then the space in which they are embedded. Therefore, the data will have low variance along several of the principal vectors used to calculate the set of subspace angles, and so many of the principal angles will have little meaning. To address this limitation we also compute the shared variance between the linear subspaces that contain object manifolds.

A.6 Implicit MMCR Effectively Reduces Augmentation Manifold Nuclear Norm

To test whether or not implicit manifold compression actually reduces the mean augmentation manifold nuclear norm, we can vary the value of λ. Below we see the evolution of both terms of the loss for several different values of lambda during training on CIFAR-10. For these experiments the batch size was 64 and the number of augmentations per image was 4.0. As shown in Fig. A.1, the level of compression of individual manifolds is nearly the same across all values of the parameter.

Table A.1: Layer Mapping Details for MFTMA Analyses. A Total of 18 Extracted ResNet-50 Layers (in **Bold**) for MFTMA Analysis

Layer	Type	Conv2d Size (H × W × C)
pixel	**Input**	None
conv1	$\begin{bmatrix} [l]\text{Conv2d} \\ \text{BatchNorm} \\ \textbf{ReLU} \end{bmatrix} \times 1$	$\left[[l]7 \times 7 \times 64\right] \times 1$
conv2_x	$\left[[l] \begin{bmatrix} [l]\text{Conv2d} \\ \text{BatchNorm} \end{bmatrix} \times 3 \right] \times 3$	$\begin{bmatrix} [l]1 \times 1 \times 64 \\ 3 \times 3 \times 64 \\ 1 \times 1 \times 256 \end{bmatrix} \times 3$
conv3_x	$\left[[l] \begin{bmatrix} [l]\text{Conv2d} \\ \text{BatchNorm} \end{bmatrix} \times 3 \right] \times 4$	$\begin{bmatrix} [l]1 \times 1 \times 128 \\ 3 \times 3 \times 128 \\ 1 \times 1 \times 512 \end{bmatrix} \times 4$
conv4_x	$\left[[l] \begin{bmatrix} [l]\text{Conv2d} \\ \text{BatchNorm} \end{bmatrix} \times 3 \right] \times 6$	$\begin{bmatrix} [l]1 \times 1 \times 256 \\ 3 \times 3 \times 256 \\ 1 \times 1 \times 1024 \end{bmatrix} \times 6$
conv5_x	$\left[[l] \begin{bmatrix} [l]\text{Conv2d} \\ \text{BatchNorm} \end{bmatrix} \times 3 \right] \times 3$	$\begin{bmatrix} [l]1 \times 1 \times 512 \\ 3 \times 3 \times 512 \\ 1 \times 1 \times 2048 \end{bmatrix} \times 3$

A.7 Classification Evaluation Procedure

CIFAR and STL-10: During pre-training all models were monitored with a k-nearest neighbor classifier (k=200) and checkpointed every 5 epochs. After pre-training, we trained linear classifiers on all checkpoints whose monitor accuracy was within 1% of the highest observed accuracy, and select the model that achieves the highest linear classification accuracy. Linear classifiers were trained using the Adam optimizer with batch size of 1024 and an initial learning rate of 0.1, which decayed according to a cosine scheduler over the course of 50 epochs. For the

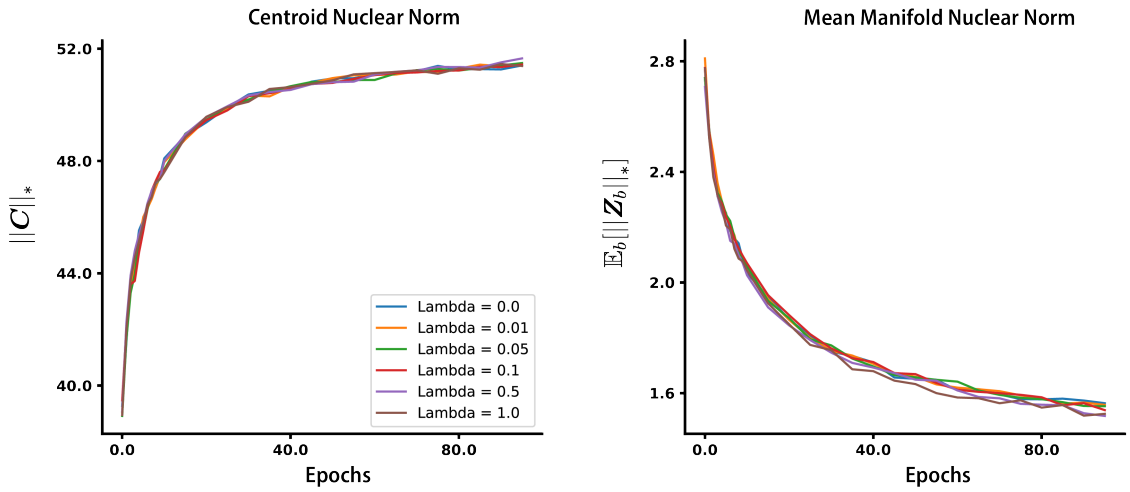

Figure A.1: Validation loss values for different values of λ.

linear classifier training, at train time we use the same set of augmentations as during unsupervised pretraining, at test time we only use center cropping and random horizontal flipping.

ImageNet-1k/100: For ImageNet datasets we closely followed the most widely adopted evaluation procedure. Following pre-training we freeze the encoder weights and train a linear layer in a supervised fashion using SGD with a batch size of 2048, learning rate of 1.6, and weight decay of 1e-6 for 50 epochs. During linear classifier training the only data augmentations are random cropping and random horizontal flips, and during evaluation inputs are center cropped.

Semi-Supervised: For semi-supervised evaluation we mostly follow the procedure outlined in Bardes et al. (2022). We use the SGD optimizer with momentum of 0.9 and weight decay of 1e-6 and the standard cross entropy loss. The augmentation procedure was the same as described above. Because the linear classifier is being trained from scratch and the representation is being fine tuned the learning rate for the parameters of the backbone is scaled down by a factor of 10, and both learning rates (backbone and classifier) followed a cosine decay schedule for 20 epochs. We used a batch size of 256 and swept the learning rate over [0.1, 0.3, 1.0] for each model.

Other Downstream Classification Tasks: Classifiers on these datasets were trained in a

similar fashion to those trained on the CIFAR and STL-10 datasets. The only differences being that for each method and datasets we swept the batch size over [128, 256, 512, 1024] and the the initial learning rate over [3e-2, 3e-3, 3e-4], and the augmentation procedure matched the standard setup for ImageNet training (only random cropping and horizontal flipping for training, resizing and center cropping for evaluation).

A.8 Training Metrics

In the Fig. A.2 below we monitor the evolution of both the objective (second panel), the mean augmentation manifold nuclear norm, the centroid norm, and the mean centroid similarity evaluated on the test set over the course of training.

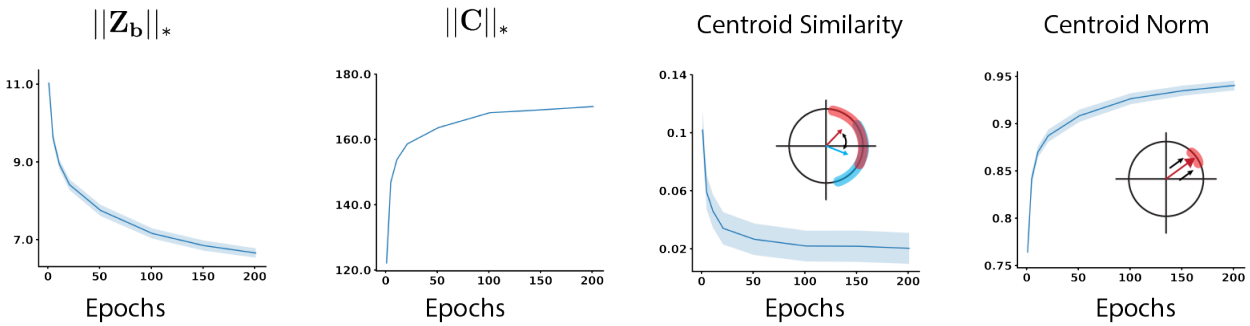

Figure A.2: Evolution of Metrics During Training. Geometric measures are evaluated on a set of 200 manifolds, each defined by an image drawn from the CIFAR-10 dataset, along with 16 augmentations. Shaded regions indicate a 95% confidence interval around the mean.

A.9 Classification Performance on Smaller Datasets

In Table A.2 below we report the performance of both our method as well as Barlow Twins and SimCLR when trained using a ResNet-50 backbone on smaller datasets.

Table A.2: *MMCR Classification Performance with Smaller Datasets.* Top-1 classification accuracies of linear classifiers for representations trained with various datasets and objective functions. Note: for Barlow Twins on ImageNet-100 we report the result from Da Costa et al. (2022) which uses a ResNet-18 backbone, as we were unable to obtain better performance. For MMCR on ImageNet-100 we tested both 2 views (matched to baselines) and 4 views, results are formatted (2-view)/(4-view)

Method	CIFAR-10	CIFAR-100	STL-10	ImageNet-100
Barlow Twins (our repro.)	90.91	67.91	89.96	80.38*
SimCLR (our repro.)	92.22	70.04	**91.11**	79.64
MMCR ($\lambda = 0.0$)	**93.53**	69.87	90.62	81.52/**82.88**
MMCR ($\lambda = 0.01$)	93.39	**70.94**	90.77	81.28/82.56

A.10 Batch Size Dependence

One of the most cited drawbacks of contrastive SSL methods has been that strong performance on downstream tasks requires training with large batch sizes, while non-contrastive methods (e.g., VICReg or Barlow Twins (Bardes et al., 2022; Zbontar et al., 2021)) that place constraints on the cross-correlation/covariance matrices of the embeddings are much more amenable to smaller batch training. It is also worth noting that the need for large batch sizes in contrastive methods can be alleviated in various ways, such as maintaining a memory bank (Wu et al., 2018) or employing a slowly updating momentum encoder (He et al., 2020). Given that our method is neither wholly contrastive nor non-contrastive (since it acts on the spectrum of the embedding matrix directly), we wondered howwould depend on training batch size. We pretrained on ImageNet-1k using batch sizes of 256, 512, 1024, 2048, 4096 and evaluate the linear classification accuracy for each. Encouragingly we observed only a modest decrease in performance for the smallest batch size tested. The results of this sweep, in comparison to Barlow Twins and SimCLR, is shown in in Fig. A.3 Note that for these runs we used two views and the linear learning rate scaling as described in Appendix A.4. Future work should endeavor to better understand the impact of various hyperparameters on the quality of learned representations.

An important detail is that for this experiment we did not employ a momentum encoder when training MMCR. It is argued in He et al. (2020) that the momentum encoder increases the effective minibatch size as the slowly moving weights encode information from preceding batches. However here we are interested explicitly interested in batch size dependence so we ablate this architectural confound (neither Chen et al. (2020b); Zbontar et al. (2021) employ momentum encoders).

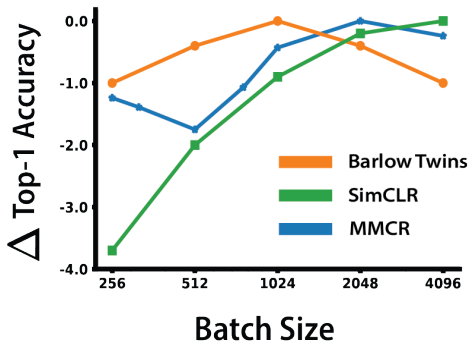

Figure A.3: Performance as a Function of Batch Size. Drop in top-1 performance relative to that of the best setting for three methods. Data for both Barlow Twins and SimCLR are copied from Zbontar et al. (2021).

A.11 Additional Details on BrainScore

Brain-Score evaluates a model in terms of its ability to predict the measured responses of neurons to images. We provide a brief introduction to the metric here (see (Schrimpf et al., 2018) for a complete description). Let $\boldsymbol{y} \in \mathbb{R}^N$ denote the average response of a single (biological) neuron to a set of N training images and $\boldsymbol{X} \in \mathbb{R}^{N \times K}$ be the response of K model neurons to the same images. Brain Score first solves the linear regression problem $\boldsymbol{y} = \boldsymbol{X}\boldsymbol{w}$ for weights $\boldsymbol{w}$. The model then predicts responses $\boldsymbol{y}'$ to a set of held out images. Next the Pearson correlation coefficient between $\boldsymbol{y}'$ and y is calculated, and the score for a particular dataset is the median of the individual neuron predicitivities in said dataset. The mean of these scores is taken over

Model	V1.0	V1.1	V2.0	V4.0	V4.1	V4.2
MMCR	0.270	0.718	0.311	0.577	0.627	0.492
SimCLR	0.224	0.776	0.288	0.576	0.626	0.48
BYOL	0.274	0.727	0.291	0.585	0.626	0.48
MoCo	0.273	.726	0.293	0.57	0.629	0.492
Barlow	0.276	.721	0.293	0.568	0.626	0.493
SwAV	0.252	.723	0.296	0.568	0.614	0.469
Model	**V4.3**	**IT.0**	**IT.1**	**IT.2**	**IT.3**	
MMCR	0.226	0.554	0.558	0.545	0.424	
SimCLR	0.224	0.552	0.545	0.518	0.456	
BYOL	0.216	0.55	0.545	0.516	0.41	
MoCo	0.215	0.54	0.560	0.550	0.437	
Barlow	0.221	0.545	0.547	0.518	0.412	
SwAV	0.202	0.533	0.537	0.518	0.405	

Table A.3: Detailed MMCR Brain-Score Evaluations. Brain-score comparison of six self-supervised models, over 11 different electrophysiological data sets recorded from macaque monkeys (Schrimpf et al., 2018). Datasets V1.0 and V2.0 are from Freeman et al. (2013), V1.1 is from Marques et al. (2021), and V4.0, V4.1, IT.0, and IT.1 are from Majaj et al. (2015). (V4/IT).(2/3) are the SanghaviJozwik2020 and SanghaviMurty2020 datasets as denoted by brainscore.

different train-test splits of each dataset. The score for a total brain area (as shown in Table 2.2 is the mean over train-test splits and distinct datasets. The standard error of the means for each dataset (which are typically between 1e-3 and 1e-2) within an area are summed quadrature and divided by the number of datasets to produce the errors reported in table 2.2. In table A.3 we split the brain area scores into individual datasets.

A.12 Additional Details on Spectral Properties

Participation Ratio We first extracted the 2048 dimensional feature vectors for each model in response to the images in the ImageNet validation set. Images were resized to 256×256 and then center cropped to 224×224 following the setting in which the classifiers are tested. We resampled the resultant feature matrices with replacement 10 times independently for each model For each resampled dataset (of features) we calculate the empirical covariance matrix, associated

eigenspectra, and associated participation ratios (squared ratio of L_1 to L_2 norm of the eigenvcetors).

Decay Coefficient The spectra obtained using the procedure outlined above all decayed rapidly near the tails (the least significant eigenvalues). To avoid undo bias from these tails when estimating the decay coefficients we only considered the top 2000 eigenvalues. To estimate the decay coefficient we fit a regression line to the logarithm of the eigenvalues as a function of the logarithm of their indexes. This fitting procedure was repeated across the bootstrapped spectra for each model to obtain standard errors of the mean. For all models the linear regression produced a strong fit to the data, with the minimum observed R^2 value being 0.95.

A.13 Object Detection

To ensure that the representations obtained with MMCR are not hyperspecialized to classification tasks we also evaluated our highest performing model on object detection. We follow (He et al., 2020; Zbontar et al., 2021), fine tuning the representation network with a Faster R-CNN head and C-4 backbone on the VOC07+12 dataset (training with the train+val split and evaluating on the VOC07 test split). All settings except for the initial learning rate (which we set to 0.12) were identical to those from He et al. (2020), and we similarly used the detectron2 library for this evaluation. MMCR with 8 views and 100 epochs of pretraining produced an AP50 of 81.9 (this is the mean over three independent fine tunes, the standard deviation was 0.2), demonstrating that the representeations generated by our method are not limited to object recognition; Table A.4 gives full results. For reference, a representation trained using MoCo v2 for 200 epochs achieves an AP50 (the most common evaluation metric for this dataset) of 82.4 (in the table below we report the result we obtained for MoCo pretrained for 100 epochs for consistency).

Method	mAP	AP50	AP75
Barlow	53.1	80.9	57.7
SwAV	54.4	81.6	61.0
MMCR	54.6	81.9	60.6
SimCLR	54.7	81.7	60.2
MoCo v2	55.6	**82.3**	61.7
BYOL	**56.0**	**82.3**	**62.0**

Table A.4: MMCR Object Detection Evaluation. We follow (He et al., 2020; Zbontar et al., 2021), fine tuning the representation network for detection with a Faster R-CNN head and C-4 backbone on the VOC07+12 dataset.

A.14 Limitations

Time and compute limitations prevented us from conducting exhaustive optimization of all design choices. For example we do not explore effects of varying the projector network width and depth. We additionally restrict the model to a ResNet-50 encoding network pretrained on the ImageNet-1k dataset for 100 epochs. This choice allows us to make fair comparisons to several recently developed SSL methods with a modest compute budget, but precludes answering questions about how the method scales up to larger problems.

We also note that although our objective function has favorable computational complexity compared to existing methods, the evaluation of our objective is not straightforward to distribute across machines. This is because we need to compute the SVD over the centroid matrix, which requires gathering the outputs of a distributed forward pass onto a single machine. Efficient methods for the distributed computation of the SVD could help alleviate this issue in the future.

A.15 Impact of Momentum Encoder

With all other settings fixed, we find that the use of a momentum encoder confers a small advantage in terms of classification performance. Below we report frozen linear evaluation ac-

curacy for MMCR training 2, 4, and 8 views with and without a momentum encoder.

	2 Views	4 Views	8 Views
With Momentum Encoder	69.5	71.4	72.1
Without Momentum Encoder	68.4	70.2	71.5

Table A.5: Impact of Momentum Encoder in MMCR. Shown are Top-1 accuracies under the linear evaluation protocol for representations trained with or without a momentum encoder using the settings described above. It appears that as the number of views increases the added diversity of positive sample representations conferred by the momentum encoder diminishes.

A.16 MMCR Benefits from Longer Pretraining

To verify that MMCR benefits from longer pretraining we modified our setup to pretrain with two views and reduced our base learning rate to 0.4 and train for 1000 epochs. We additionally changed the hyperparameters for pretraining the linear classifier when evaluating this model (the classifier was trained with an learning rate of 0.3 and batch size of 2048 for 100 epochs). We did not perform any hyperparameter tuning for this setting, so these results represent a performance floor for 2-view MMCR.

Method	Top-1 Accuracy
SimCLR	69.3
MoCov2	71.1
SimSiam	71.3
SwAV (without multicrop)	71.3
MMCR (2 views)	73.0
Barlow Twins	73.2
BYOL	74.3
SwAV (with multicrop)	75.3
NNCLR	75.6
ReLICv2 (Tomasev et al., 2022)	**77.1**

Table A.6: MMCR Performance With Longer Pretraining. Shown are Top-1 accuracies under the linear evaluation protocol for representations trained for 1000 epochs using various self supervised learning frameworks.

B | Contrastive-Equivariant Self-Supervised Learning Improves Alignment with Primate Visual Area IT

B.1 Additional Pretraining Details

Here we report some additional hyperparameters not included in the main text.

Optimization: For all experiments we trained for 100 epochs using a batch size of 2048 and used the LARS optimizer (You et al., 2017) with weight decay of 1e-6 and momentum of 0.9. Note that the $\mathcal{L}_{CE-SSL}$ loss is evaluated on pairs of augmented views and thus had an effective batch size of 1024. We use a base learning rate of 4.8 and a learning rate schedule consisting of linear warm-up for the first 10 epochs followed by cosine decay throughout training.

Projector Architectures Each trained network uses two projectors with matching architectures. For Barlow Twins we used the architecture proposed in the original work (Zbontar et al., 2021) (3 layer MLP with hidden layer and output layer widths of 8192). For MMCR we also used a 3 layer MLP with 8192 hidden width but 512 output units (also in line with the original work (Yerxa et al., 2024b)). For SimCLR we used the same projector architecture as MMCR, which is

larger than the MLP described originally because subsequent work (Garrido et al., 2023a) has found that SimCLR benefits from a more expressive projector.

B.1.1 ImageNet-1k

For Barlow Twins we set the λ_{BT}, which balances the on and off diagonal loss terms, hyperparameter to $5e-3$. For SimCLR we used a temperature of $\tau = 0.15$.

B.1.2 ImageNet-100

Besides the change of dataset, the only hyperparameter change in this setting is that we increased the number of pretraining epochs from 100 to 200 to be more in line with previous work.

B.2 Online-Linear Evaluation for the Pretraining Dataset

Because frozen-linear evaluation on large datasets is computationally intensive we instead opt for online-linear classification. During pretraining the representation network outputs are detached from the gradient propogation graph and fed through a linear layer that is optimized with the standard supervised cross entropy loss. Previous work has shown that online-evaluation is very strongly correlated with frozen-linear evaluation and incurs only a minimal cost on top of self-supervised pretraining. We report the accuracies for ImageNet-1k trained networks in Fig. B.1 and the smaller set of ImageNet-100 trained models in Table B.1.

B.3 Transfer Learning Evaluation Procedure

We closely follow the evaluation procedure from (Lee et al., 2021), we repeat the details here for completeness. First images are resized such that the shortest edge is 224 pixels, then center cropped to 224x224 resolution. Then features are extracted from train, validation, and test splits of

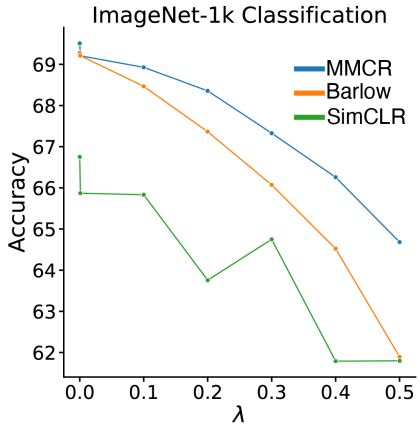

Figure B.1: CE-SSL ImageNet-1k Performance. In distribution accuracy on the validation set of ImageNet-1k evaluated by online-linear classification, for each objective and as a function of λ.

Model	Accuracy
MMCR	79.0%
Barlow	81.4%
SimCLR	83.5%
CE-MMCR	79.6%
CE-Barlow	81.6%
CE-SimCLR	82.5%

Table B.1: CE-SSL ImageNet-100 Evaluation. In distribution accuracy on the validation set of ImageNet-100 evaluated by online-linear classification.

each dataset. L-BFGS is used to optimize the standard cross entropy loss with $\mathcal{L}_2$ regularization, the value of the ridge parameter is swept over selected via performance on the validation set. Subsequently the linear classifier is retrained using both the train and validation sets, and we report the final accuracy on the held out test set.

This classification procedure was run 5 times with different random initializations, we reported the mean performance in 3.2, and report the standard deviation over runs below.

Table B.2: CE-SSL Transfer Learning Performance Variability. Standard deviation of top 1 accuracies over 5 independent runs of the transfer learning evaluation procedure.

ImageNet-100 Training						
Objective	Cifar-10	Cifar-100	Pets	DTD	Flowers-102	Food-101
MMCR	1e-2	7e-3	4e-2	1e-5	7e-2	2e-1
Barlow	1e-4	2e-4	2e-2	3e-1	3e-2	9e-3
SimCLR	1e-2	7e-3	1e-2	4e-2	7e-2	5e-3
CE-MMCR	1e-2	2e-2	4e-2	3e-2	1e-1	1e-2
CE-Barlow	5e-2	2e-2	3e-2	3e-2	8e-2	1e-2
CE-SimCLR	5e-3	2e-2	2e-2	2e-2	5e-2	1e-2
ImageNet-1k Training						
Objective	Cifar-10	Cifar-100	Pets	DTD	Flowers-102	Food-101
MMCR	1e-2	2e-2	8e-2	4e-2	2e-2	7e-3
Barlow	1e-2	1e-1	1e-1	4e-2	3e-2	3e-3
SimCLR	2e-2	7e-3	1e-4	4e-2	2e-2	8e-3
CE-MMCR	9e-3	2e-2	2e-1	2e-1	6e-2	2e-2
CE-Barlow	1e-5	1e-2	9e-2	1e-5	5e-2	9e-2
CE-SimCLR	1e-2	1e-2	2e-2	2e-2	7e-2	8e-3

B.4 Additional Details for Representational Analyses

Below we depict the joint distributions of the distances described in 3.4.2. For each setting (unique objective function and value of λ there are 800 unique measurements (i.e. points of a unique hue on an individual plot). Meaning, for example, there are 800 random pairs of augmentation manifolds compared in both invariant representation space and equivariant representation space for each equivariant network in the left most column. We summarize describe the sources of variability over which covariance matrices are estimated and the variables over which the expected Bures distance is calculated for the experiments in Fig. 3.2 A-D.

B.5 Out-of-distribution Equivariance

We aim to test whether learning equivariances using weak augmentations induces structured variability in response to stronger augmentations (the extent to which learned equivariances generalize beyond the range of transformations seen during training). We trained models using the Barlow Twins objective and the same sweep over values of λ using “weak" augmentations of (1) double the minimum crop size, (2) half the maximum value of color jittering, and (3) half the maximum size of Gaussian blurring kernel. We then repeat the parameter decoding experiments from the main paper Fig. 3.2E on these weak augmentations (left panel Fig. B.3), and on the non-overlapping part of the parameter space between the weak and strong augmentation distributions, i.e only for augmentations whose parameters are in distribution for the models trained as in the main text but out of distribution for the new models trained using weaker augmentations (right panel Fig. B.3). In the first panel, we can see that the best parameter decoding performance occurs when the pretraining distribution of transformations is matched to the evaluation transformations (i.e. weak-trained models slightly outperform the strong-trained models at decoding the parameters of weak transformations). In the right panel we see that models trained

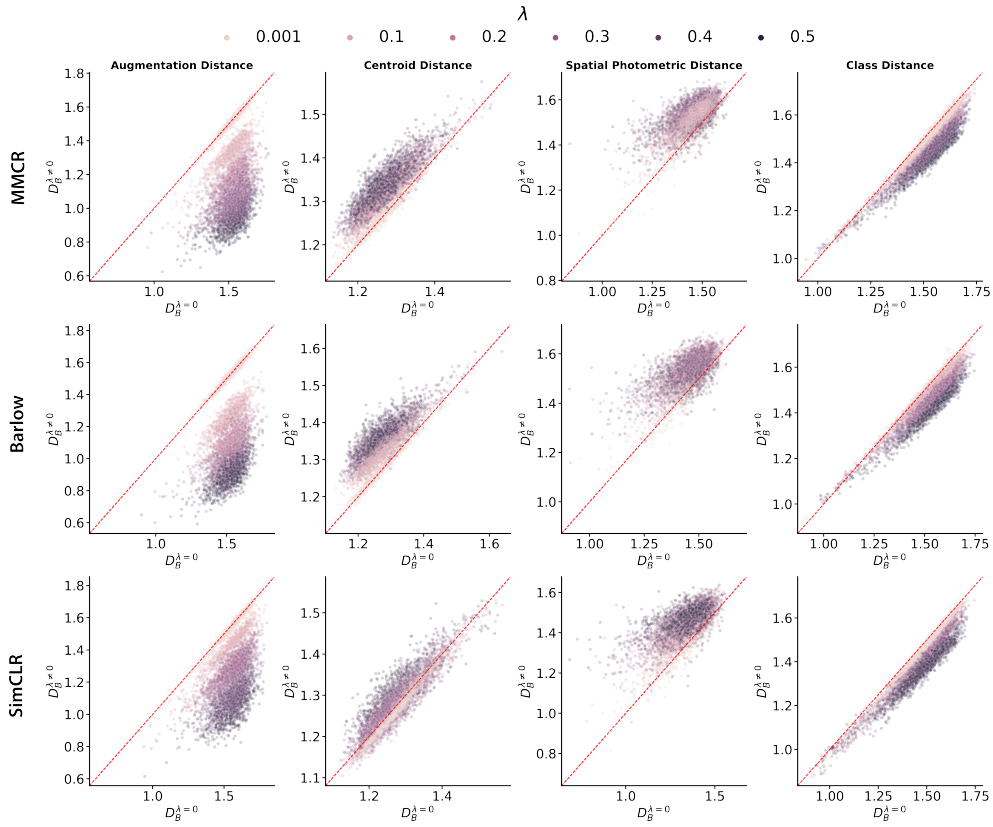

Figure B.2: Joint Distributions of Bures Metric Based Measurements. Joint distributions of invariant and equivariant networks (increasing λ increases the importance of the equivariant loss) for all of the Bures metric comparisons detailed in 3.4.2. The mean value curves in Fig. 3.2 are generated by taking the mean of $x - y$ for each of these plots, separately for each objective and value of λ. The confidence intervals are estimated from the distribution of $x - y$ values as well. Columns depict Bures distances between covariances estimated over different sources of variability, and columns index different base invariant objective functions.

Table B.3: Sources of Variability Summary. Table defining the sources of variability producing covariance matrices and the random variables that the expected Bures distance is computed over for the experiments in Fig. 3.2.

Column	**Source of Variability** C_1	**Source of Variability** C_2	**Ensemble for estimating expectation**
Augmentation-Augmentation Distance (A)	Augmentations of single image.	Augmentations of single image.	Random pairs of distinct images
Augmentation-Centroid Distance (B)	Augmentation manifold centroids over all images	Augmentations of single image.	Random images (those used to compute C_2)
Spatial-Photometric Distance (C)	Photometric augmentations (of a single image) after averaging over many random crops.	Random crops (of a single image) after averaging over many photometric augmentations.	Unique base images.
Class-Class Distance (D)	(Unaugmented) exemplars from one class	(Unaugmented) exemplars from one class	Random pairs of distinct classes

with strong augmentations have higher decoding performance, but models trained only on weak augmentations still demonstrate significantly increased ability to linearly decode strong augmentation parameters relative to the invariant trained models ($\lambda = 0$ models), indicating a degree of generalization in the learned equivariances.

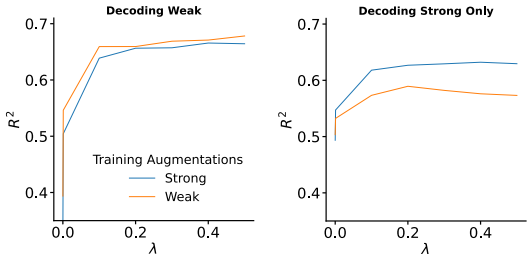

Figure B.3: Out of Distribution Parameter Decoding. Augmentation parameter decoding performance on held-out test images for networks trained using either strong or weak transformations. In the left panel we plot the decoding performance for weak transformations only, and in the right panel the performance for strong transformations only (which are not seen by the weak-trained networks during pretraining).

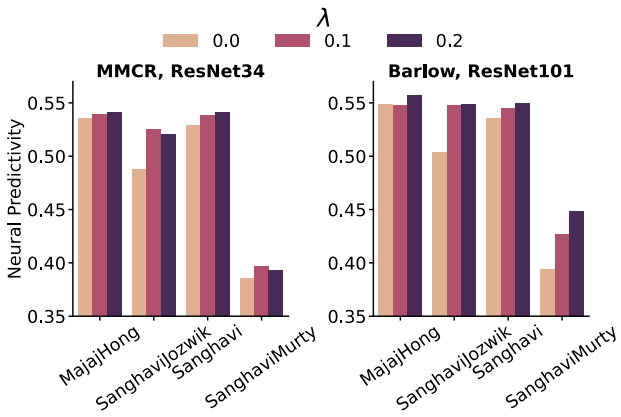

Figure B.4: CE-SSL Applied using Different Architectures. Neural predictivity results for models trained using the MMCR base loss and a ResNet-34 backbone (left panel), and the Barlow Twins base loss and a ResNet-101 backbone (left panel). We see similar trends in terms of increased neural predictivity for equivariant trained models as observed using the ResNet-50 backbone in the main text.

B.6 Different Backbone Architectures

To verify that the effect of equivariance neural predictivity observed in the main text is not limited to a specific choice of architecture, we trained invariant ($\lambda = 0.0$) and equivariant ($\lambda \in [0.1, 0.2]$) networks using $\mathcal{L}_{BT}$ with smaller (ResNet-34) and larger (ResNet-50) backbones. As shown in Fig. B.4, we observe the same trend across different architectures: contrastive-equivariant training increases alignment to area IT as measured by linear predictivity.

B.7 Compute Resources

All pretraining runs used 8 A100 Nvidia GPUs with 40GB of memory each. In our setting pretraining run times were around 15 hours, and we note that CE-SSL training generally increased training time by approximately 10% relative to standard self-supervised training. Subsequent evaluations ran on a single A100.

B.8 Limitations

We discuss some limitations not addressed in the discussion here. Our current training setup requires selection of the hyperparameter λ to balance between the equivariant and invariant loss functions. Future work could investigate methods to balance the two losses without explicitly training an individual network for each choice of λ. It is worth noting that some invariant SSL methods may be more or less sensitive to this additional hyperparameter. For example, some of the non-monotonicity of SimCLR curves in Fig. 3.2 may suggest that SimCLR representations are more difficult to smoothly shape within the CE-SSL paradigm.

Additionally, computational limitations prevent us from doing an extensive architecture search over the two projector networks employed in contrastive equivariant training. In addition to varying the depth and width of each projector, it would be of interest to "split" the representation space and have each projector operate on a subset of dimensions as input (as in SIE (Garrido et al., 2023b)). Additonally computational limitations prevented us from extensively evaluating the variability in neural predictiivity over independent runs of contrastive equivariant training (the BrainScore framework recently stopped providing estimates of the error of neural predicitivity, but in previous studies the reported error for IT predicitivity was $\approx$ 3e-3, which is small relative to the variability we observed across the parameter of interest λ). Pilot experiments indicated to us that the variability over training runs was small relative to the variability over values

of lambda (we trained two invariant models and two with $\lambda = 0.1$ to get a rough estimate of this, in both cases we kept the more predictive model for inclusion in all analyses that appear in this paper).

B.9 Broader Impacts

In this work we propose one strategy for inducing increased alignment between artificial and biological visual representations. Better understanding the computational principles underlying visual representations has the potential to benefit the quality of computer vision applications, to offer insights into the structure of the primate visual system, and to improve clinical treatment of disorders related to visual perception.

C | Spatiotemporal Maximum Manifold Capacity Representations

C.1 Neural Dataset and Evaluation Details

We used the neural dataset from Freeman et al. (2013); Ziemba et al. (2016) for evaluating alignment to areas V1 and V2, and data from Lieber et al. (2024) for evaluating alignment to V4. Below we describe the relevant aspects of the stimuli used in these experiments, and provide further details regarding the evaluation protocol.

Stimulus generation. Stimulus generation for the V1 and V2 datasets begins with a set of 15 photographs of natural visual textures (i.e. tree bark). For each texture image, a set of synthetic images is generated by initializing to samples of white noise and and updating pixel values via gradient descent to match a set of statistics defined by the parametric texture model of Portilla and Simoncelli (2000). Matched noise stimuli are generated by taking the Fourier transform of each texture image and scrambling the resulting phases while preserving the amplitudes, producing images with matched spectral energy but lacking the higher order structures characteristic of the texture images. Example synthetic textures and their spectrally-matched counterparts are shown in Fig. C.1.

The images used in the V4 dataset are derived from the UPenn Natural Image Database (Tkačik

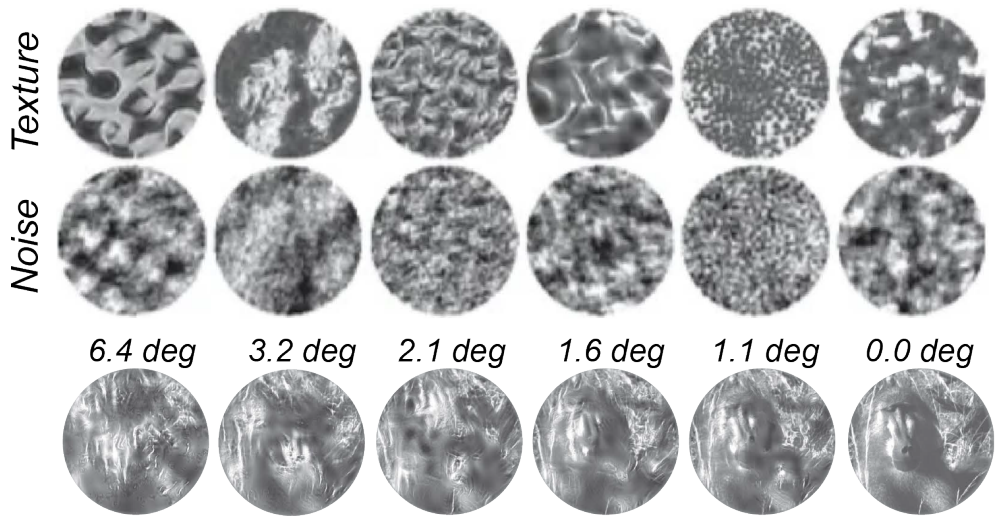

Figure C.1: Neural Dataset Stimuli. Example stimuli used for each of the three neural datasets. The top two rows show images used in the V1/V2 experiments of Freeman et al. (2013) and the last row shows images used in the V4 experiment of Lieber et al. (2024). These figures are reproduced from the original works with the permission of the authors.

et al., 2011). These base images are then "texturized," using an analogous statistical matching procedure to the one described above, but modified so that the texture statistics are matched within overlapping pooling regions of varying sizes. Because the images were presented at 6.4° of visual angle, when the pooling region is this size, this corresponds to the texture synthesis procedure used for V1/V2. Conversely a pooling region size of 0.0° corresponds to matching each individual pixel value, producing an image identical to the original base image. Example stimuli for each pooling region size are shown in the bottom row of Fig C.1.

Evaluation protocol. We follow the procedure described in Section 4.3.4, using 10 cross validation folds for the V1/V2 datasets and 5 for the V4 dataset. Here we describe the preprocessing steps applied to the raw stimuli used in each set of stimuli used to obtain the model responses. One key parameter is the assumed field of view input resolution associated with models trained on images with arbitrary resolution. For the V1/V2 stimuli and each trained baseline model (the standard and adversarially trained AlexNets) as well as the vast majority of models trained on ImageNet, an input size of 224 pixels and 8° field of view are known to be optimal in terms of pre-

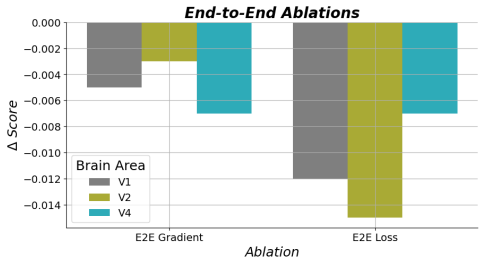

Figure C.2: Ablating Layerwise Local Learning. We train two models without layerwise gradient computations (left group of columns), and additionally without the application of losses at intermediate stages (right group of columns), and evaluate each for neural predictivity. We report the change in score relative to the default settings (the ST-MCMCR columns in Fig. 4.3.

dictivity (Schrimpf et al., 2018), and so we simply adopt this convention for our model. Because in the V1/V2 experiment stimuli were displayed at 4° of visual angle, the stimuli are resized to the appropriate resolution, and placed in the central 4° of the input image using a circular aperture with a 112 pixel diameter. For the V4 dataset the same procedure is applied, with a modification made to account for the fact that these stimuli were presented at a 6.4° field of view. Additionally we apply the standard normalization transform that was employed while models are training on ImageNet. Finally it is worth noting that, following Parthasarathy et al. (2024b) we skip all preprocessing steps when extracting responses from pyramid-based models.

End-to-End ablation. We ablated the "layerwise" aspect of our training procedure in two steps by training networks with (1) identical loss functions and architecture, but without "gradient cutting," and (2) full end-to-end gradients, but and without any internal losses. The results are shown below in Fig. C.2.

C.2 Input Transformation Details

The complete set of image transformations used to obtain frames for training the ST-MMCR representation is given in Table C.1. To obtain the first and final frames, we apply two random resized cropping operations. Intermediate frames are then obtained by linearly interpolating the

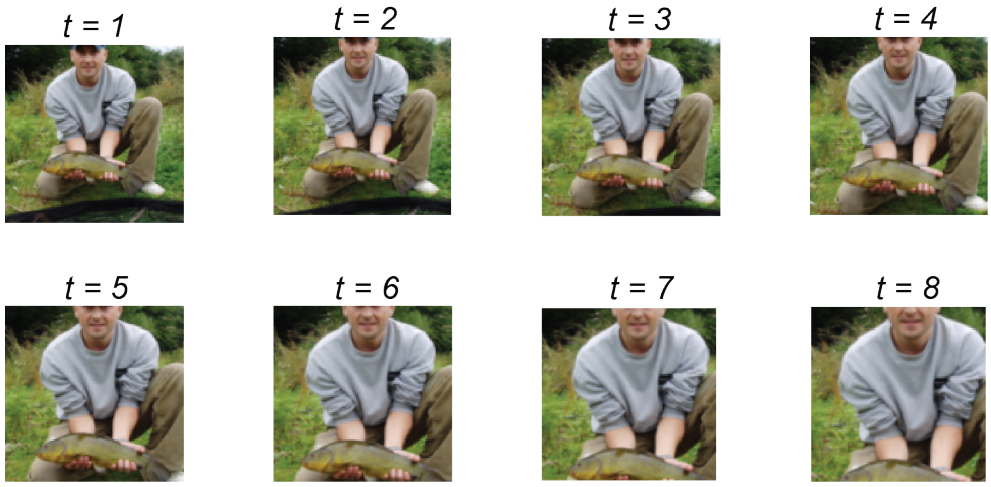

Figure C.3: Synthetic Video Frames . The first and final frames are obtained by taking two random crops of a single image, and intermediate frames are obtained by linearly interpolating crop parameters. Finally each frame is resized. This transformation is a crude way of simulating the smooth motion that can occur as the relative position of the observer and objects in the scene change over time.

coordinates of the anchor frames, an example sequence of frames is shown in Fig. C.3. Each frame is individually subjected to a series of photometric distortions: modest contrast and brightness modulation, a stochastic between color and grayscale, and a chance for the addition of gaussian noise with standard deviation randomly selected between 0.04 and 0.1.

For the lower complexity transformations of Fig. 4.2, we only change the spatial transformation step. To reduce the input feature complexity, we first resize images so the shortest edge is 224 pixels, then take center crops of sizes $[33, 56, 112]$ for complexities 1, 2, and 3, respectively. To modulate the transformation complexity we then set the minimum scale of the random resized cropping operations for each complexity to 0.9, 0.6, and 0.3, which are applied to the center cropped image patches. The fourth complexity transformation is the default for ST-MMCR described above. Thus across the x-axis of Fig. 4.2, inputs have more complicated visual features as the fraction of pixels included in the synthetic videos grows, and the extent of the transformations grows as the anchor crops are allowed to be separated by larger distances in pixel space.

Parameter	$t = 0$	All t	$t = T$
Minimum random crop scale	0.08		0.08
Random crop resize output size	224		224
Color jittering probability		0.8	
Color dropping probability		0.2	
Brightness adjustment max		0.2	
Contrast adjustment max		0.2	
Gaussian noise probability		0.5	

Table C.1: ST-MMCR Default Transformations. Parameters for the default transformation scheme used to train the ST-MMCR model.

C.3 Architecture And Optimization Details

As described in Section 4.3.2, we use the first and second convolutional layers of the AlexNet architecture to parameterize the first and second stages of the ST-MMCR representation, and the third and fourth convolutional layers for the third stage. For completeness we give a full description of each operation in Table C.2. Because computing our objective involves taking small crops of internal feature maps, we precede MaxPooling operations with a fixed (not trained) blurring operation to mitigate the impact of aliasing (using the Blurred max pooling operation from Zhang (2019)). Finally, we note that our implementation includes a batch normalization that precedes each rectifier, and uses a third pooling operation that appears "one stage early" relative to the original AlexNet architecture. While these represent a small departure from the more standard choices present in the supervised and robust baseline models, we observed that the absence of the batch normalization in these models did not significantly impact their neural predictivity or behavioral alignment scores (a result previously reported in Parthasarathy et al. (2024b)). We additionally provide exact details for each stages associated projector network in Table C.3.

The parameter optimization was performed using the Adam optimizer (Deng et al., 2009; Kingma and Ba, 2014), and iterating for 10 epochs of the ImageNet-1k dataset with a batch size

f_{θ_1}	**Conv2d**(in_channels=3, out_channels=64, ks=11, stride=4, padding=5, padding_mode='reflect') **BatchNorm2d**(64) **ReLU**() **BlurMaxPool2d**(ks=3, stride=2, padding=1)
f_{θ_1}	**Conv2d**(in_channels=64, out_channels=192, ks=5, stride=1, padding=2, padding_mode='reflect') **BatchNorm2d**(192) **ReLU**() **BlurMaxPool2d**(ks=3, stride=2, padding=1)
f_{θ_3}	**Conv2d**(in_channels=192, out_channels=384, ks=3, stride=1, padding=1, padding_mode='reflect') **BatchNorm2d**(384) **ReLU**() **Conv2d**(in_channels=384, out_channels=256, ks=3, stride=1, padding=1, padding_mode='reflect') **BatchNorm2d**(256) **ReLU**() **BlurMaxPool2d**(ks=3, stride=2, padding=1)

Table C.2: ST-MMCR Architecture. Detailed description of the architecture for each stage of the ST-MMCR representation.

of 32 and a constant learning rate of $1e-3$.

C.4 Behavioral Evaluation Details

Front-End depth experiment. As a step towards determining to which brain area alignment is most critical for inducing human-like behavior on classification tasks we trained ImageNet-1k classifiers on top of V1/V2/V4 front-ends. Specifically, we froze our pretrained model at the first, second or third stage, and appended the remaining stages of the standard AlexNet architecture to each front end (see below for details on the classifier training procedure). In this experimental design, the V2 based model is handicapped relative to the V1 (and the V4 relative to the V2) model as it has fewer trainable parameters dedicated to the classification task. To control for this we additionally trained classifiers on top of randomly-initialized front-ends that were frozen at

g_{ϕ_1}	**Conv2d**(in_channels=64, out_channels=64, ks=1, stride=1, padding=0) **BatchNorm2d**(64) **ReLU**() **Conv2d**(in_channels=64, out_channels=2048, ks=1, stride=1, padding=0)
g_{ϕ_2}	**Conv2d**(in_channels=192, out_channels=192, ks=1, stride=1, padding=0) **BatchNorm2d**(192) **ReLU**() **Conv2d**(in_channels=192, out_channels=2048, ks=1, stride=1, padding=0)
g_{ϕ_3}	**Conv2d**(in_channels=256, out_channels=256, ks=1, stride=1, padding=0) **BatchNorm2d**(256) **ReLU**() **Conv2d**(in_channels=256, out_channels=2048, ks=1, stride=1, padding=0)

Table C.3: ST-MMCR Projector Architecture. Detailed description of the architecture of the projector network associated with each stage of the ST-MMCR representation.

matched points along the hierarchy to isolate the impact of ST-MMCR pretraining.

We measure each models in-distribution generalization (performance on the Standard ImageNet-1k validation set), as well as out-of-distribution generalization and alignment with human behavioral choices on the suite of OOD datasets described in Geirhos et al. (2021). The results are summarized in Fig. 4.6. Despite marked decreases in in-distribution performance when freezing multiple stages (top row), OOD accuracy, observational consistency, and error consistency (as defined in Geirhos et al. (2021) and re-described in C.4) remain similar or superior to the standard supervised AlexNet.

Classifier training procedure. We trained classifiers on top of each of the three stages of the ST-MMCR architecture by "completing" the AlexNet architecture (with batch normalization) using randomly initialized modules. For example, when training on top of the outputs of f_{θ_1}, the

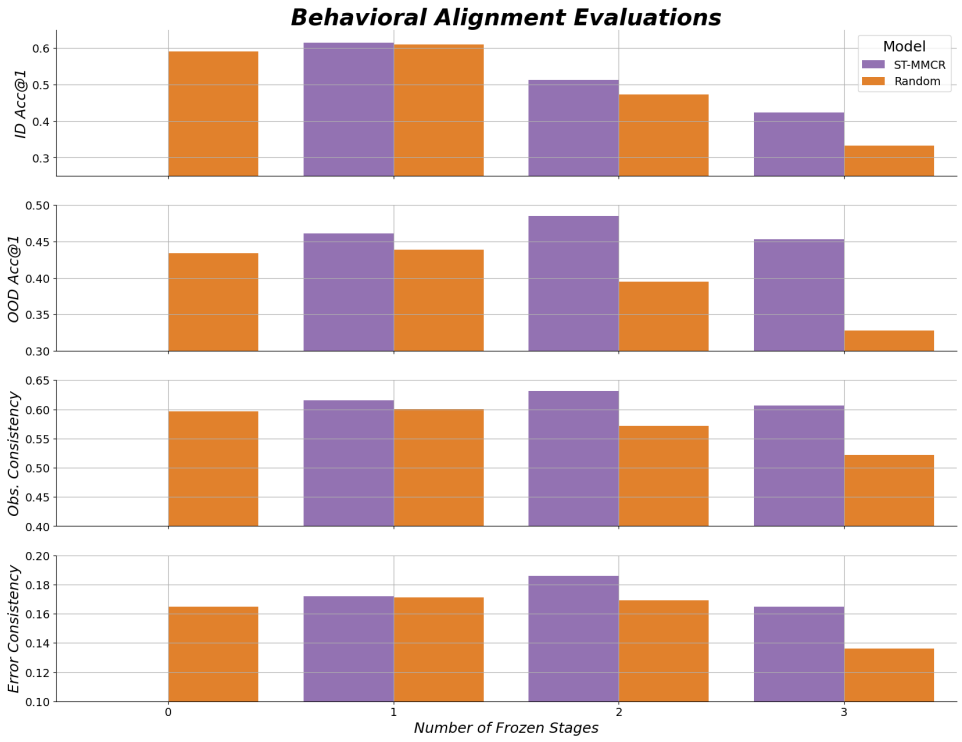

Figure C.4: Behavioral Evaluation of V1, V2, and V4 Front-ends. All networks are evaluated on the ImageNet-1k task (top row), OOD generalization on the 20 datasets from Geirhos et al. (2021) (second row), and the rate of agreement between model and human category choices on said datasets (bottom two rows).

second convolutional stage was freshly initialized to have random trainable parameters, while the parameters of the first convolution were frozen to the ST-MMCR values (i.e., this corresponds to 1 frozen stage, as indicated on the x-axis of Fig. 4.6). Note that for the outputs of the third stage of ST-MMCR we omitted the final downsampling operation to maintain consistency across classifier architectures (see Appendix C.3 for justificaiton). As a result of this strategy, the V2-frontend classifier has fewer trainable parameters than its V1-frontend counterpart. To better isolate the impact of ST-MMCR pretraining (rather than differences in model capacity), we compare each model to a matched version in which the frontend is randomly initialized and frozen. This pairing places the pretrained and randomly initialized models on more equal footing across conditions with varying numbers of frozen stages.

The trainable parameters of each classifier were trained on the ImageNet-1k dataset using the standard cross entropy loss and stochastic gradient descent with momentum and weight decay for 100 epochs. We used a batch size of 512, an base learning rate of 0.1 which decayed by a factor of 10 every 30 epochs, a momentum value of 0.9, and the weight decay penalty set to $1e-4$.

Evaluation metrics and additional results. In addition to reporting the standard accuracy of each model on the ImageNet-1k validation set, we evaluated the behavioral alignment of models using the datasets from Geirhos et al. (2021) (using the official implementation which can be found at `https://github.com/bethgelab/model-vs-human`). The data consists of human categorization responses to 17 out-of-distribution (from the model's perspective) tasks. These tasks are generated by distorting photographic images by applying various style-transfer techniques or through the application of parametric degradations such as the addition of noise or blurring and were originally described in Geirhos et al. (2019; 2018).

In addition to reporting the average accuracy on this suite of OOD tasks, we show the observation and error consistencies between each model and human responses. The observation consistency measures the rate at which the model and human both categorize a given sample ei-

ther correctly or incorrectly, and the error consistency measures whether the choice consistency is higher or lower than what would be obtained from a pair of random models with accuracies matched to the model and human respectively. For complete descriptions see Geirhos et al. (2021).

In addition to the results for the ST-MMCR and randomly initialized front-end models visualized in Fig. 4.6, we show results for each of these metrics in Tables C.4 and C.5.

model	accuracy diff. ↓	obs. consistency ↑	error consistency ↑	mean rank ↓
ST-MMCR-V2	**0.095**	**0.631**	**0.186**	**1.000**
ST-MMCR-V1	0.113	0.615	0.172	3.000
ST-MMCR-V4	0.112	0.607	0.165	3.667
RI-V1	0.123	0.601	0.173	4.000
Robust	0.145	0.573	0.176	4.667
Supervised	0.118	0.597	0.165	5.333
RI-V2	0.147	0.572	0.169	6.333
RI-V4	0.189	0.522	0.136	8.000

Table C.4: Behavioral Alignment of AlexNet Based Classifiers. The first column shows the difference in OOD accuracy between a model and human observer, and subsequent columns are described in the text. The -V1 indicates that one frozen stage served as a front-end for a classifier, and -V2 2 stages and -V4 three stages. Finally the RI- models correspond to randomly initialized frontends.

model	OOD accuracy ↑	rank ↓
ST-MMCR-V2	**0.485**	**1.000**
ST-MMCR-V1	0.461	2.000
ST-MMCR-V4	0.453	3.000
RI-V1	0.439	4.000
supervised	0.434	5.000
RI-V2	0.395	6.000
robust	0.391	7.000
RI-V4	0.328	8.000

Table C.5: Out of Distribution Accuracy of AlexNet Based Classifiers. Average accuracy on the OOD tasks from Geirhos et al. (2021). The model naming convention is the same as described in the caption of Table C.4.

C.5 Sparse Regression Details

To investigate how predictivity varies as a function of the number of model units used to interpolate a neural response, we used a variant of the relaxed LASSO procedure Meinshausen (2007). Concretely, for each neural unit and train test split we first compute the LASSO regularization path on the train data to obtain estimators with varying levels of sparsity (Friedman et al., 2010). We used `sklearn`'s implementation of lasso path, and find 200 different estimators using regularization coefficients that varied over 8 orders of magnitude. After this step, we selected the mappings that had nearest the desired levels of sparsity, k, which we set to be 10 logarithmically spaced values between 5 and 1000. Next we take the selected features (i.e. the approximately k features with non-zero coefficients), and refit the training data using only this reduced feature space, via ridge regression. We repeat this process for each neuron and each train-test split, and report the final score by taking the median over neurons and mean over splits, as described in Section 4.3.4.

C.6 Eigendistortion Details

Recall from Section 4.6, that we are interested in the extremal eigenvectors of Fisher Information matrices that are derived from functions that map from an input stimuli to a population of predicted mean firing rates. To calculate these we utilized the power iteration method implemented in the open source package plenoptic (Duong et al., 2023). We used a maximum of 1000 steps in the optimization and terminated the optimization early if successive iterations produced a change in the estimated eigenvalue below $1e-7$. Finally it is worth noting that the firing-rate predictors are undercomplete functions, as there are more input pixels than there are neurons to predict in all 3 datasets. As a result the FIM's have large null spaces, and as a result the lease noticable distortions we obtain are best interpreted as random samples from each prediciton

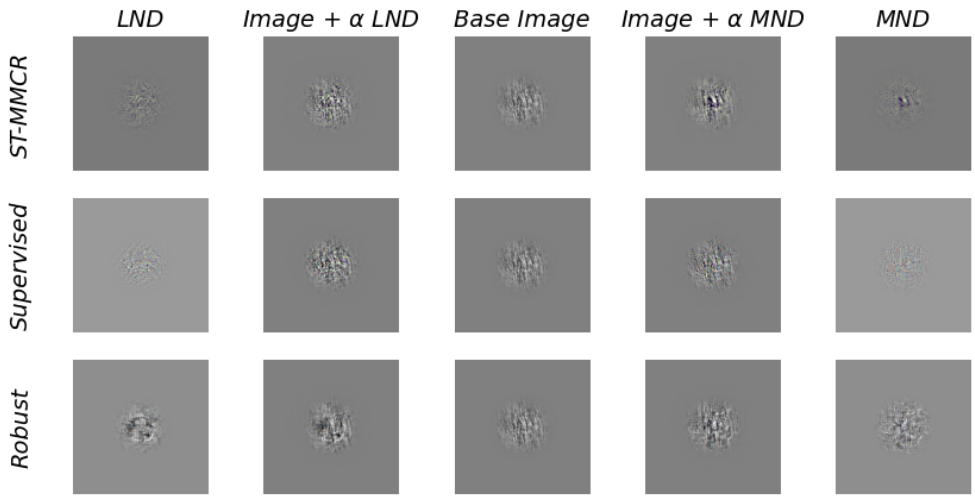

Figure C.5: Eigendistortions of Neural Firing Rate Predictors: V1. Formatting choices are matched to those of Fig. 4.7. Here we choose a "spectrally matched noise" stimulus from the V1/V2 dataset.

function's null space (specifically, the null space of a local linear approximation of said prediction function). Finally, we include example distortions for areas V1 and V2 in the same format as the V4 example in Fig. 4.7 in Figures C.5 and C.6 respectively.

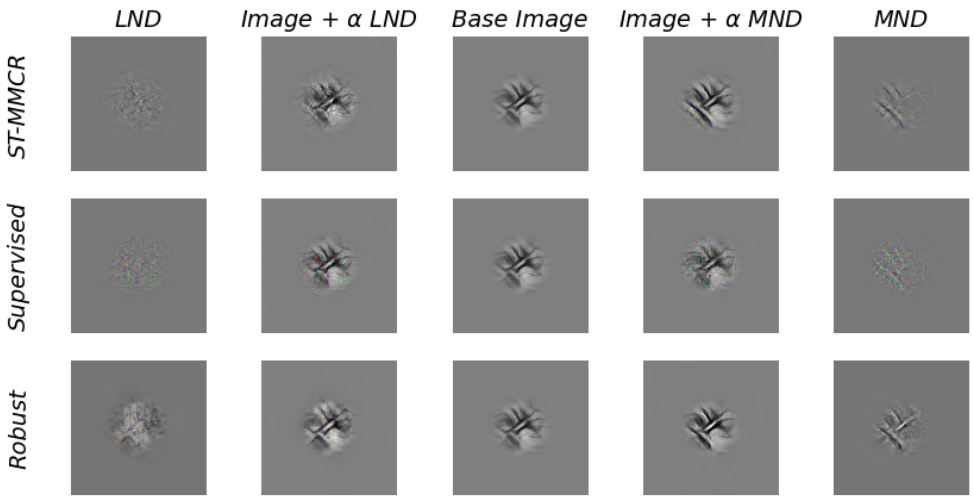

Figure C.6: Eigendistortions of Neural Firing Rate Predictors: V2. Formatting choices are matched to those of Fig. 4.7. Here we choose a "naturalistic texture" stimulus from the V1/V2 dataset.

Bibliography

Abbaras, A., Aubin, B., Krzakala, F., and Zdeborová, L. (2020). Rademacher complexity and spin glasses: A link between the replica and statistical theories of learning. In *Mathematical and Scientific Machine Learning*, pages 27–54. PMLR.

Agrawal, K. K., Mondal, A. K., Ghosh, A., and Richards, B. A. (2022). α-req : Assessing representation quality in self-supervised learning by measuring eigenspectrum decay. In Oh, A. H., Agarwal, A., Belgrave, D., and Cho, K., editors, *Advances in Neural Information Processing Systems*.

Alleman, M., Lindsey, J., and Fusi, S. (2024). Task structure and nonlinearity jointly determine learned representational geometry. In *International Conference on Learning Representations*, Vienna, Austria.

Atick, J. J. and Redlich, A. N. (1992). What does the retina know about natural scenes? *Neural Computation*, 4:196–210.

Attneave, F. (1954). Some informational aspects of visual perception. *Psychological Review*, 61(3):183–193.

Aubret, A., Ernst, M. R., Teulière, C., and Triesch, J. (2023). Time to augment self-supervised visual representation learning. In *International Conference on Learning Representations*, Kigali, Rwanda.

Bachman, P., Hjelm, R. D., and Buchwalter, W. (2019). Learning representations by maximizing mutual information across views. *Advances in neural information processing systems*, 32.

Bahri, Y., Kadmon, J., Pennington, J., Schoenholz, S. S., Sohl-Dickstein, J., and Ganguli, S. (2020). Statistical mechanics of deep learning. *Annual review of condensed matter physics*, 11(1):501–528.

Balestriero, R. and LeCun, Y. (2022). Contrastive and non-contrastive self-supervised learning recover global and local spectral embedding methods. *Advances in Neural Information Processing Systems*, 35:26671–26685.

Bardes, A., Ponce, J., and LeCun, Y. (2022). VICReg: Variance-invariance-covariance regularization for self-supervised learning. In *International Conference on Learning Representations*, Held Virtually.

Barlow, H. B. et al. (1961). Possible principles underlying the transformation of sensory messages. *Sensory communication*, 1(01):217–233.

Baylor, D. A. (1987). Photoreceptor signals and vision. proctor lecture. *Investigative ophthalmology & visual science*, 28(1):34–49.

Belghazi, M. I., Baratin, A., Rajeshwar, S., Ozair, S., Bengio, Y., Courville, A., and Hjelm, D. (2018). Mutual information neural estimation. In *International conference on machine learning*, pages 531–540. PMLR.

Belilovsky, E., Eickenberg, M., and Oyallon, E. (2019). Greedy layerwise learning can scale to imagenet. In *International conference on machine learning*, pages 583–593. PMLR.

Bell, A. J. and Sejnowski, T. J. (1996). The "independent components" of natural scenes are edge filters. *Vision Research*, 37:3327–3338.

Bengio, Y., Lamblin, P., Popovici, D., and Larochelle, H. (2006). Greedy layer-wise training of deep networks. *Advances in neural information processing systems*, 19.

Berardino, A., Laparra, V., Ballé, J., and Simoncelli, E. (2017). Eigen-distortions of hierarchical representations. *Advances in neural information processing systems*, 30.

Bernardi, S., Benna, M. K., Rigotti, M., Munuera, J., Fusi, S., and Salzman, C. D. (2020). The geometry of abstraction in the hippocampus and prefrontal cortex. *Cell*, 183(4):954–967.

Bordes, F., Balestriero, R., Garrido, Q., Bardes, A., and Vincent, P. (2023). Guillotine regularization: Why removing layers is needed to improve generalization in self-supervised learning. *Transactions on Machine Learning Research*.

Bossard, L., Guillaumin, M., and Van Gool, L. (2014). Food-101–mining discriminative components with random forests. In *European conference on computer vision*, pages 446–461. Springer.

Bruna, J. and Mallat, S. (2013). Invariant scattering convolution networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1872–1886.

Brunel, N. and Nadal, J.-P. (1998). Mutual information, fisher information, and population coding. *Neural computation*, 10(7):1731–1757.

Cadena, S. A., Willeke, K. F., Restivo, K., Denfield, G., Sinz, F. H., Bethge, M., Tolias, A. S., and Ecker, A. S. (2024). Diverse task-driven modeling of macaque v4 reveals functional specialization towards semantic tasks. *PLOS Computational Biology*, 20(5):e1012056.

Canatar, A., Feather, J., Wakhloo, A., and Chung, S. (2023). A spectral theory of neural prediction and alignment. *Advances in Neural Information Processing Systems*, 36:47052–47080.

Carandini, M. and Heeger, D. J. (2012). Normalization as a canonical neural computation. *Nature Reviews Neuroscience*, 13(1):51–62.

Caron, M., Misra, I., Mairal, J., Goyal, P., Bojanowski, P., and Joulin, A. (2020). Unsupervised learning of visual features by contrasting cluster assignments. *Advances in Neural Information Processing Systems*, 33:9912–9924.

Chavhan, R., Stuehmer, J., Heggan, C., Yaghoobi, M., and Hospedales, T. (2023). Amortised invariance learning for contrastive self-supervision. In *International Conference on Learning Representations*, Kigali, Rwanda.

Chen, P., Agarwal, C., and Nguyen, A. (2020a). The shape and simplicity biases of adversarially robust imagenet-trained cnns. *arXiv preprint arXiv:2006.09373*.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. (2020b). A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR.

Chen, X., Fan, H., Girshick, R., and He, K. (2020c). Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*.

Chen, X. and He, K. (2021). Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15750–15758.

Chklovskii, D. B., Schikorski, T., and Stevens, C. F. (2002). Wiring optimization in cortical circuits. *Neuron*, 34(3):341–347.

Chou, C.-N., Kim, R., Arend, L. A., Yang, Y.-Y., Mensh, B. D., Shim, W. M., Perich, M. G., and Chung, S. (2024). Geometry linked to untangling efficiency reveals structure and computation in neural populations. *bioRxiv*, pages 2024–02.

Chung, S. and Abbott, L. (2021). Neural population geometry: An approach for understanding biological and artificial neural networks. *Current opinion in neurobiology*, 70:137–144.

Chung, S., Lee, D. D., and Sompolinsky, H. (2018). Classification and geometry of general perceptual manifolds. *Physical Review X*, 8(3):031003.

Cimpoi, M., Maji, S., Kokkinos, I., Mohamed, S., and Vedaldi, A. (2014). Describing textures in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3606–3613.

Cohen, U., Chung, S., Lee, D. D., and Sompolinsky, H. (2020). Separability and geometry of object manifolds in deep neural networks. *Nature communications*, 11(1):1–13.

Conwell, C., Prince, J. S., Kay, K. N., Alvarez, G. A., and Konkle, T. (2022). What can 1.8 billion regressions tell us about the pressures shaping high-level visual representation in brains and machines? *BioRxiv*, pages 2022–03.

Cover, T. M. (1965). Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition. *IEEE transactions on electronic computers*, (3):326–334.

Da Costa, V. G. T., Fini, E., Nabi, M., Sebe, N., and Ricci, E. (2022). solo-learn: A library of self-supervised methods for visual representation learning. *Journal of Machine Learning Research*, 23(56):1–6.

Dangovski, R., Jing, L., Loh, C., Han, S., Srivastava, A., Cheung, B., Agrawal, P., and Soljacic, M. (2022). Equivariant self-supervised learning: Encouraging equivariance in representations. In *International Conference on Learning Representations*, Held Virtually.

Dapello, J., Feather, J., Le, H., Marques, T., Cox, D., McDermott, J., DiCarlo, J. J., and Chung, S. (2021). Neural population geometry reveals the role of stochasticity in robust perception. *Advances in Neural Information Processing Systems*, 34:15595–15607.

Dapello, J., Marques, T., Schrimpf, M., Geiger, F., Cox, D., and DiCarlo, J. J. (2020). Simulating a pri-

mary visual cortex at the front of cnns improves robustness to image perturbations. *Advances in Neural Information Processing Systems*, 33:13073–13087.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. IEEE.

Devillers, A. and Lefort, M. (2023). Equimod: An equivariance module to improve visual instance discrimination. In *International Conference on Learning Representations*, Kigali, Rwanda.

DiCarlo, J. J. and Cox, D. D. (2007). Untangling invariant object recognition. *Trends in cognitive sciences*, 11(8):333–341.

Doi, E., Gauthier, J. L., Field, G. D., Shlens, J., Sher, A., Greschner, M., Machado, T. A., Jepson, L. H., Mathieson, K., Gunning, D. E., et al. (2012). Efficient coding of spatial information in the primate retina. *Journal of Neuroscience*, 32(46):16256–16264.

Douglas, R. J. and Martin, K. A. (2004). Neuronal circuits of the neocortex. *Annu. Rev. Neurosci.*, 27(1):419–451.

Duong, L., Bonnen, K., Broderick, W., Fiquet, P.-É., Parthasarathy, N., Yerxa, T., Zhao, X., and Simoncelli, E. (2023). Plenoptic: A platform for synthesizing model-optimized visual stimuli. *Journal of Vision*, 23(9):5822–5822.

Dwibedi, D., Aytar, Y., Tompson, J., Sermanet, P., and Zisserman, A. (2021). With a little help from my friends: Nearest-neighbor contrastive learning of visual representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9588–9597.

El-Shamayleh, Y., Kumbhani, R. D., Dhruv, N. T., and Movshon, J. A. (2013). Visual response properties of v1 neurons projecting to v2 in macaque. *Journal of Neuroscience*, 33(42):16594–16605.

Elmoznino, E. and Bonner, M. F. (2024). High-performing neural network models of visual cortex benefit from high latent dimensionality. *PLoS computational biology*, 20(1):e1011792.

Ermolov, A., Siarohin, A., Sangineto, E., and Sebe, N. (2021). Whitening for self-supervised representation learning. In *International Conference on Machine Learning*, pages 3015–3024. PMLR.

Fairhall, A. L., Lewen, G. D., Bialek, W., and de Ruyter van Steveninck, R. R. (2001). Efficiency and ambiguity in an adaptive neural code. *Nature*, 412(6849):787–792.

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

Felsen, G., Touryan, J., and Dan, Y. (2005). Contextual modulation of orientation tuning contributes to efficient processing of natural stimuli. *Network: Computation in Neural Systems*, 16(2-3):139–149.

Földiák, P. (1991). Learning invariance from transformation sequences. *Neural computation*, 3(2):194–200.

Freeman, J., Ziemba, C. M., Heeger, D. J., Simoncelli, E. P., and Movshon, J. A. (2013). A functional and perceptual signature of the second visual area in primates. *Nature neuroscience*, 16(7):974–981.

Friedman, J. H., Hastie, T., and Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of statistical software*, 33(1):1–22.

Froudarakis, E., Cohen, U., Diamantaki, M., Walker, E. Y., Reimer, J., Berens, P., Sompolinsky, H., and Tolias, A. S. (2020). Object manifold geometry across the mouse cortical visual hierarchy. *BioRxiv*, pages 2020–08.

Fusi, S., Miller, E. K., and Rigotti, M. (2016). Why neurons mix: high dimensionality for higher cognition. *Current opinion in neurobiology*, 37:66–74.

Gallego, J. A., Perich, M. G., Miller, L. E., and Solla, S. A. (2017). Neural manifolds for the control of movement. *Neuron*, 94(5):978–984.

Ganguli, D. (2012). *Efficient Coding and Bayesian Estimation with Neural Populations*. PhD thesis, New York University.

Gardner, E. (1988). The space of interactions in neural network models. *Journal of physics A: Mathematical and general*, 21(1):257.

Gardner, E. and Derrida, B. (1988). Optimal storage properties of neural network models. *Journal of Physics A: Mathematical and general*, 21(1):271.

Garrido, Q., Chen, Y., Bardes, A., Najman, L., and LeCun, Y. (2023a). On the duality between contrastive and non-contrastive self-supervised learning. In *International Conference on Learning Representations*, Kigali, Rwanda.

Garrido, Q., Najman, L., and Lecun, Y. (2023b). Self-supervised learning of split invariant equivariant representations. In *International Conference on Machine Learning*, pages 10975–10996. PMLR.

Geirhos, R., Narayanappa, K., Mitzkus, B., Thieringer, T., Bethge, M., Wichmann, F. A., and Brendel, W. (2021). Partial success in closing the gap between human and machine vision. *Advances in Neural Information Processing Systems*, 34:23885–23899.

Geirhos, R., Rubisch, P., Michaelis, C., Bethge, M., Wichmann, F. A., and Brendel, W. (2019). Imagenet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. In *International Conference on Learning Representations*, New Orleans, Louisiana.

Geirhos, R., Temme, C. R., Rauber, J., Schütt, H. H., Bethge, M., and Wichmann, F. A. (2018). Generalisation in humans and deep neural networks. *Advances in neural information processing systems*, 31.

Gerl, F. and Krey, U. (1994). Storage capacity and optimal learning of potts-model perceptrons by a cavity method. *Journal of Physics A: Mathematical and General*, 27(22):7353.

Goyal, P., Duval, Q., Reizenstein, J., Leavitt, M., Xu, M., Lefaudeux, B., Singh, M., Reis, V., Caron, M., Bojanowski, P., Joulin, A., and Misra, I. (2021). Vissl. https://github.com/facebookresearch/vissl.

Grill, J.-B., Strub, F., Altché, F., Tallec, C., Richemond, P., Buchatskaya, E., Doersch, C., Avila Pires, B., Guo, Z., Gheshlaghi Azar, M., et al. (2020). Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, 33:21271–21284.

Gupta, S., Robinson, J., Lim, D., Villar, S., and Jegelka, S. (2024). Structuring representation geometry with rotationally equivariant contrastive learning. In *International Conference on Learning Representations*, Vienna, Austria.

Gutmann, M. and Hyvärinen, A. (2010). Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 297–304. JMLR Workshop and Conference Proceedings.

Halvagal, M. S. and Zenke, F. (2023). The combination of hebbian and predictive plasticity learns invariant object representations in deep sensory networks. *Nature neuroscience*, 26(11):1906–1915.

HaoChen, J. Z., Wei, C., Gaidon, A., and Ma, T. (2021). Provable guarantees for self-supervised deep learning with spectral contrastive loss. *Advances in Neural Information Processing Systems*, 34:5000–5011.

Harrington, A., DuTell, V., Tewari, A., Hamilton, M., Stent, S., Rosenholtz, R., and Freeman, W. T. (2023). Exploring perceptual straightness in learned visual representations. In *International Conference on Learning Representations*, Kigali, Rwanda.

He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. (2020). Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Hebb, D. O. (1949). *The organization of behavior: A neuropsychological theory*. John Wiley and Sons.

Heeger, D. J. (1992). Normalization of cell responses in cat striate cortex. *Visual neuroscience*, 9(2):181–197.

Hénaff, O. J., Bai, Y., Charlton, J. A., Nauhaus, I., Simoncelli, E. P., and Goris, R. L. (2021). Primary visual cortex straightens natural video trajectories. *Nature communications*, 12(1):1–12.

Hénaff, O. J., Goris, R. L., and Simoncelli, E. P. (2019). Perceptual straightening of natural videos. *Nature neuroscience*, 22(6):984–991.

Hénaff, O. J., Rabinowitz, N., Ballé, J., and Simoncelli, E. P. (2015). The local low-dimensionality of natural images. In *International Conference on Learning Representations*, San Diego, CA.

Hinton, G. E., Osindero, S., and Teh, Y.-W. (2006). A fast learning algorithm for deep belief nets. *Neural computation*, 18(7):1527–1554.

Hong, H., Yamins, D. L., Majaj, N. J., and DiCarlo, J. J. (2016). Explicit information for category-orthogonal object properties increases along the ventral stream. *Nature neuroscience*, 19(4):613–622.

Illing, B., Ventura, J., Bellec, G., and Gerstner, W. (2021). Local plasticity rules can learn deep representations using self-supervised contrastive predictions. *Advances in Neural Information Processing Systems*, 34:30365–30379.

Ioffe, S. and Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. pmlr.

Jaynes, E. T. (1957). Information theory and statistical mechanics. *Physical review*, 106(4):620.

Jing, L., Vincent, P., LeCun, Y., and Tian, Y. (2022). Understanding dimensional collapse in contrastive self-supervised learning. In *International Conference on Learning Representations*, Held Virtually.

Karklin, Y. and Simoncelli, E. (2011). Efficient coding of natural images with a population of noisy linear-nonlinear neurons. *Advances in neural information processing systems*, 24.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Knill, D. C. and Pouget, A. (2004). The bayesian brain: the role of uncertainty in neural coding and computation. *TRENDS in Neurosciences*, 27(12):712–719.

Knyazev, A. V. and Argentati, M. E. (2002). Principal angles between subspaces in an a-based scalar product: algorithms and perturbation estimates. *SIAM Journal on Scientific Computing*, 23(6):2008–2040.

Konkle, T. and Alvarez, G. A. (2022). A self-supervised domain-general learning framework for human ventral stream representation. *Nature communications*, 13(1):491.

Kriegeskorte, N. and Kievit, R. A. (2013). Representational geometry: integrating cognition, computation, and the brain. *Trends in cognitive sciences*, 17(8):401–412.

Krizhevsky, A., Hinton, G., et al. (2009). Learning multiple layers of features from tiny images. Technical report, University of Toronto.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25.

Kuoch, M., Chou, C.-N., Parthasarathy, N., Dapello, J., DiCarlo, J. J., Sompolinsky, H., and Chung, S. (2024). Probing biological and artificial neural networks with task-dependent neural manifolds. In *Conference on Parsimony and Learning*, pages 395–418. PMLR.

Laughlin, S. (1981). A simple coding procedure enhances a neuron's information capacity. *Zeitschrift für Naturforschung c*, 36(9-10):910–912.

Lee, H., Lee, K., Lee, K., Lee, H., and Shin, J. (2021). Improving transferability of representations via augmentation-aware self-supervision. *Advances in Neural Information Processing Systems*, 34:17710–17722.

Lennie, P. (2003). The cost of cortical computation. *Current biology*, 13(6):493–497.

Lezama, J., Qiu, Q., Musé, P., and Sapiro, G. (2018). Ole: Orthogonal low-rank embedding-a plug and play geometric loss for deep learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8109–8118.

Lieber, J. D., Oleskiw, T. D., Simoncelli, E. P., and Movshon, J. A. (2024). Responses of neurons in macaque v4 to object and texture images. *BioRxiv*, pages 2024–02.

Lindsay, G. W. (2021). Convolutional neural networks as a model of the visual system: Past, present, and future. *Journal of cognitive neuroscience*, 33(10):2017–2031.

Lindsey, J. W. and Issa, E. B. (2024). Factorized visual representations in the primate visual system and deep neural networks. *Elife*, 13:RP91685.

Linsley, D., Rodriguez Rodriguez, I. F., Fel, T., Arcaro, M., Sharma, S., Livingstone, M., and Serre, T. (2023). Performance-optimized deep neural networks are evolving into worse models of inferotemporal visual cortex. *Advances in Neural Information Processing Systems*, 36:28873–28891.

Liu, L., She, L., Chen, M., Liu, T., Lu, H. D., Dan, Y., and Poo, M.-m. (2016). Spatial structure of neuronal receptive field in awake monkey secondary visual cortex (V2). *Proceedings of the National Academy of Sciences*, 113(7):1913–1918.

Löwe, S., O'Connor, P., and Veeling, B. (2019). Putting an end to end-to-end: Gradient-isolated learning of representations. *Advances in neural information processing systems*, 32.

Ma, Y., Derksen, H., Hong, W., and Wright, J. (2007). Segmentation of multivariate mixed data via lossy data coding and compression. *IEEE transactions on pattern analysis and machine intelligence*, 29(9):1546–1562.

Madan, S., Xiao, W., Cao, M., Pfister, H., Livingstone, M., and Kreiman, G. (2024). Benchmarking out-of-distribution generalization capabilities of DNN-based encoding models for the ventral visual cortex. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. (2018). Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, Vancouver, Canada.

Majaj, N. J., Hong, H., Solomon, E. A., and DiCarlo, J. J. (2015). Simple learned weighted sums of inferior temporal neuronal firing rates accurately predict human core object recognition performance. *Journal of Neuroscience*, 35(39):13402–13418.

Mamou, J., Le, H., Del Rio, M., Stephenson, C., Tang, H., Kim, Y., and Chung, S. (2020). Emer-

gence of separable manifolds in deep language representations. In *International Conference on Machine Learning*, pages 6713–6723. PMLR.

Marques, T., Schrimpf, M., and DiCarlo, J. J. (2021). Multi-scale hierarchical neural network models that bridge from single neurons in the primate primary visual cortex to object recognition behavior. *bioRxiv*, pages 2021–03.

Marshall, A. W., Olkin, I., and Arnold, B. C. (1979). *Inequalities: theory of majorization and its applications*, volume 143. Springer.

Meinshausen, N. (2007). Relaxed lasso. *Computational Statistics & Data Analysis*, 52(1):374–393.

Mises, R. and Pollaczek-Geiringer, H. (1929). Praktische verfahren der gleichungsauflösung. *ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik*, 9(1):58–77.

Nassar, J., Sokol, P., Chung, S., Harris, K. D., and Park, I. M. (2020). On 1/n neural representation and robustness. *Advances in Neural Information Processing Systems*, 33:6211–6222.

Nieh, E. H., Schottdorf, M., Freeman, N. W., Low, R. J., Lewallen, S., Koay, S. A., Pinto, L., Gauthier, J. L., Brody, C. D., and Tank, D. W. (2021). Geometry of abstract learned knowledge in the hippocampus. *Nature*, 595(7865):80–84.

Nilsback, M.-E. and Zisserman, A. (2008). Automated flower classification over a large number of classes. In *2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing*, pages 722–729. IEEE.

Niu, X., Savin, C., and Simoncelli, E. P. (2024). Learning predictable and robust neural representations by straightening image sequences. In *Adv. Neural Information Processing (NeurIPS)*, volume 37, Vancouver.

Olshausen, B. and Field, D. (1996). Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381:607–609.

Oord, A. v. d., Li, Y., and Vinyals, O. (2018). Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

Ozsoy, S., Hamdan, S., Arik, S., Yuret, D., and Erdogan, A. (2022). Self-supervised learning with an information maximization criterion. *Advances in Neural Information Processing Systems*, 35:35240–35253.

Padamsey, Z., Katsanevaki, D., Dupuy, N., and Rochefort, N. L. (2022). Neocortex saves energy by reducing coding precision during food scarcity. *Neuron*, 110(2):280–296.

Padamsey, Z. and Rochefort, N. L. (2023). Paying the brain's energy bill. *Current opinion in neurobiology*, 78:102668.

Paraouty, N., Yao, J. D., Varnet, L., Chou, C.-N., Chung, S., and Sanes, D. H. (2023). Sensory cortex plasticity supports auditory social learning. *Nature Communications*, 14(1):5828.

Parkhi, O. M., Vedaldi, A., Zisserman, A., and Jawahar, C. (2012). Cats and dogs. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3498–3505. IEEE.

Parthasarathy, N., Eslami, S., Carreira, J., and Henaff, O. (2023). Self-supervised video pretraining yields robust and more human-aligned visual representations. *Advances in Neural Information Processing Systems*, 36:65743–65765.

Parthasarathy, N., Henaff, O. J., and Simoncelli, E. P. (2024a). Layerwise complexity-matched learning yields an improved model of cortical area v2. *Transactions on Machine Learning Research*. Featured Certification.

Parthasarathy, N., Henaff, O. J., and Simoncelli, E. P. (2024b). Layerwise complexity-matched learning yields an improved model of cortical area v2. *Transactions on Machine Learning Research*.

Pehlevan, C., Sengupta, A. M., and Chklovskii, D. B. (2017). Why do similarity matching objectives lead to hebbian/anti-hebbian networks? *Neural computation*, 30(1):84–124.

Portilla, J. and Simoncelli, E. P. (2000). A parametric texture model based on joint statistics of complex wavelet coefficients. *International journal of computer vision*, 40:49–70.

Rao, R. P. and Ballard, D. H. (1999). Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects. *Nature neuroscience*, 2(1):79–87.

Ringach, D. L. (2002). Spatial structure and symmetry of simple-cell receptive fields in macaque primary visual cortex. *Journal of neurophysiology*, 88(1):455–463.

Saxe, A., Nelli, S., and Summerfield, C. (2021). If deep learning is the answer, what is the question? *Nature Reviews Neuroscience*, 22(1):55–67.

Schaeffer, R., Khona, M., and Fiete, I. (2022). No free lunch from deep learning in neuroscience: A case study through models of the entorhinal-hippocampal circuit. *Advances in Neural Information Processing Systems*, 35:16052–16067.

Schrimpf, M., Kubilius, J., Hong, H., Majaj, N. J., Rajalingham, R., Issa, E. B., Kar, K., Bashivan, P., Prescott-Roy, J., Geiger, F., et al. (2018). Brain-score: Which artificial neural network for object recognition is most brain-like? *BioRxiv*, page 407007.

Schrimpf, M., Kubilius, J., Lee, M. J., Murty, N. A. R., Ajemian, R., and DiCarlo, J. J. (2020). Integrative benchmarking to advance neurally mechanistic models of human intelligence. *Neuron*.

Seriès, P., Stocker, A. A., and Simoncelli, E. P. (2009). Is the homunculus "aware" of sensory adaptation? *Neural computation*, 21(12):3271–3304.

Shannon, C. E. (1948). A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423.

Siddiqui, S., Krueger, D., LeCun, Y., and Deny, S. (2024). Blockwise self-supervised learning at scale. *Transactions on Machine Learning Research*.

Simoncelli, E. P. and Freeman, W. T. (1995). The steerable pyramid: A flexible architecture for multi-scale derivative computation. In *Proceedings., international conference on image processing*, volume 3, pages 444–447. IEEE.

Simoncelli, E. P. and Olshausen, B. A. (2001). Natural image statistics and neural representation. *Annual Review of Neuroscience*, 24(1):1193–1216.

Stephenson, C., Feather, J., Padhy, S., Elibol, O., Tang, H., McDermott, J., and Chung, S. (2019). Untangling in invariant speech recognition. *Advances in neural information processing systems*, 32.

Stork (1989). Is backpropagation biologically plausible? In *International 1989 Joint Conference on Neural Networks*, pages 241–246. IEEE.

Stringer, C., Pachitariu, M., Steinmetz, N., Carandini, M., and Harris, K. D. (2019). High-dimensional geometry of population responses in visual cortex. *Nature*, 571(7765):361–365.

Suau, X., Danieli, F., Keller, T. A., Blaas, A., Huang, C., Ramapuram, J., Busbridge, D., and Zappella, L. (2023). Duet: 2d structured and approximately equivariant representations. In *Proceedings of the 40th International Conference on Machine Learning*, pages 32749–32769.

Svaetichin, G. (1956). Spectral response curves from single cones. *Acta Physiol Scand*, 39:17–46.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. (2013). Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*.

Tian, Y., Krishnan, D., and Isola, P. (2020). Contrastive multiview coding. In *European conference on computer vision*, pages 776–794. Springer.

Tkačik, G., Garrigan, P., Ratliff, C., Milčinski, G., Klein, J. M., Seyfarth, L. H., Sterling, P., Brainard, D. H., and Balasubramanian, V. (2011). Natural images from the birthplace of the human eye. *PLoS one*, 6(6):e20409.

Tomasev, N., Bica, I., McWilliams, B., Buesing, L., Pascanu, R., Blundell, C., and Mitrovic, J. (2022). Pushing the limits of self-supervised resnets: Can we outperform supervised learning without labels on imagenet? *arXiv preprint arXiv:2201.05119*.

Tuckute, G., Feather, J., Boebinger, D., and McDermott, J. H. (2022). Many but not all deep neural network audio models capture brain responses and exhibit correspondence between model stages and brain regions. *bioRxiv*, pages 2022–09.

van Hateren, J. H. (1992). Theoretical predictions of spatiotemporal receptive fields of fly lmcs, and experimental validation. *Journal of Comparative Physiology A*, 171(2):157–170.

Venkataramanan, S., Rizve, M. N., Carreira, J., Asano, Y. M., and Avrithis, Y. (2024). Is imagenet worth 1 video? learning strong image encoders from 1 long unlabelled video. In *International Conference on Learning Representations*, Vienna, Austria.

Vintch, B., Movshon, J. A., and Simoncelli, E. P. (2015). A convolutional subunit model for neuronal responses in macaque v1. *Journal of Neuroscience*, 35(44):14829–14841.

Von Helmholtz, H. (1867). *Handbuch der physiologischen Optik*, volume 9. L. Voss.

Wakhloo, A. J., Slatton, W., and Chung, S. (2024). Neural population geometry and optimal coding of tasks with shared latent structure. *arXiv preprint arXiv:2402.16770*.

Wakhloo, A. J., Sussman, T. J., and Chung, S. (2023). Linear classification of neural manifolds with correlated variability. *Physical Review Letters*, 131(2):027301.

Wang, T. and Isola, P. (2020). Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International Conference on Machine Learning*, pages 9929–9939. PMLR.

Wang, Y., Lin, J., Cai, Q., Pan, Y., Yao, T., Chao, H., and Mei, T. (2022). A low rank promoting prior for unsupervised contrastive learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Wei, X.-X. and Stocker, A. A. (2012). Efficient coding provides a direct link between prior and likelihood in perceptual bayesian inference. *Advances in neural information processing systems*, 25.

Willeke, K. F., Restivo, K., Franke, K., Nix, A. F., Cadena, S. A., Shinn, T., Nealley, C., Rodriguez, G., Patel, S., Ecker, A. S., Sinz, F. H., and Tolias, A. S. (2023). Deep learning-driven characterization of single cell tuning in primate visual area V4 unveils topological organization. Technical Report 2023.05.12.540591, bioRxiv.

Willmore, B. D. B., Prenger, R. J., and Gallant, J. L. (2010). Neural representation of natural images in visual area V2. *J Neurosci*, 30(6):2102–14.

Wiskott, L. and Sejnowski, T. J. (2002). Slow feature analysis: Unsupervised learning of invariances. *Neural Computation*, 14(4):715–770.

Wu, Z., Xiong, Y., Yu, S. X., and Lin, D. (2018). Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3733–3742.

Xiao, T., Wang, X., Efros, A. A., and Darrell, T. (2021). What should not be contrastive in contrastive learning. In *International Conference on Learning Representations*, Held Virtually.

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

Yamins, D. L., Hong, H., Cadieu, C. F., Solomon, E. A., Seibert, D., and DiCarlo, J. J. (2014). Performance-optimized hierarchical models predict neural responses in higher visual cortex. *Proceedings of the National Academy of Sciences*, 111(23):8619–8624.

Yao, J. D., Zemlianova, K. O., Hocker, D. L., Savin, C., Constantinople, C. M., Chung, S., and Sanes, D. H. (2023). Transformation of acoustic information to sensory decision variables in the parietal cortex. *Proceedings of the National Academy of Sciences*, 120(2):e2212120120.

Yerxa, T., Feather, J., Simoncelli, E., and Chung, S. (2024a). Contrastive-equivariant self-supervised learning improves alignment with primate visual area it. *Advances in neural information processing systems*, 37:96045–96070.

Yerxa, T., Kuang, Y., Simoncelli, E., and Chung, S. (2023). Learning efficient coding of natural images with maximum manifold capacity representations. *Advances in Neural Information Processing Systems*, 36:24103–24128.

Yerxa, T., Kuang, Y., Simoncelli, E., and Chung, S. (2024b). Learning efficient coding of natural images with maximum manifold capacity representations. *Advances in Neural Information Processing Systems*, 36.

Yerxa, T. E., Kee, E., DeWeese, M. R., and Cooper, E. A. (2020). Efficient sensory coding of multidimensional stimuli. *PLoS computational biology*, 16(9):e1008146.

You, Y., Gitman, I., and Ginsburg, B. (2017). Large batch training of convolutional networks. *arXiv preprint arXiv:1708.03888*.

Young, T. (1802). Ii. the bakerian lecture. on the theory of light and colours. *Philosophical transactions of the Royal Society of London*, (92):12–48.

Yu, Y., Chan, K. H. R., You, C., Song, C., and Ma, Y. (2020). Learning diverse and discriminative representations via the principle of maximal coding rate reduction. *Advances in Neural Information Processing Systems*, 33:9422–9434.

Zbontar, J., Jing, L., Misra, I., LeCun, Y., and Deny, S. (2021). Barlow twins: Self-supervised learning via redundancy reduction. In *International Conference on Machine Learning*, pages 12310–12320. PMLR.

Zhang, R. (2019). Making convolutional networks shift-invariant again. In *International conference on machine learning*, pages 7324–7334. PMLR.

Zhuang, C., Yan, S., Nayebi, A., Schrimpf, M., Frank, M. C., DiCarlo, J. J., and Yamins, D. L. (2021). Unsupervised neural network models of the ventral visual stream. *Proceedings of the National Academy of Sciences*, 118(3):e2014196118.

Ziemba, C. M., Freeman, J., Movshon, J. A., and Simoncelli, E. P. (2016). Selectivity and tolerance for visual texture in macaque v2. *Proceedings of the National Academy of Sciences*, 113(22):E3140–E3149.