

TOWARDS ALIGNING ARTIFICIAL AND BIOLOGICAL VISION SYSTEMS
WITH SELF-SUPERVISED REPRESENTATION LEARNING

by

Nikhil Parthasarathy

A DISSERTATION SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY
CENTER FOR NEURAL SCIENCE
NEW YORK UNIVERSITY
JANUARY, 2024

Dr. Eero P. Simoncelli

© NIKHIL PARTHASARATHY

ALL RIGHTS RESERVED, 2024

ACKNOWLEDGMENTS

First and foremost, I'd like to thank my advisor, Eero Simoncelli, for his support, kindness, and wisdom throughout my PhD. His ability to distill a complex scientific problem to its essence, while being able to communicate these ideas so effectively is something I will strive to achieve in the rest of my career.

While many students also have a co-advisor to help them, I must extend my gratitude to my 'unofficial' co-advisor (and also friend), Olivier Hénaff, whose passion for science and positive attitude is infectious. His help during the final years of my PhD leading to my graduation and future job offer has been invaluable greatly appreciated.

From an advisory perspective I'd also like to thank my committee members (Mike Landy, Tony Movshon, and SueYeon Chung) for their help refining my thinking and ideas throughout the development of my thesis.

To my friends; Thank you to my first-year cohort for being a bundle of craziness, joy, friendship, and support that got me through so many of the trials and tribulations of a PhD. From my lab, special shout-out to Pierre-Étienne Fiquet, Lyndon Duong, Teddy Yerxa, and Zahra Kadhkodaie who provided invaluable help by being there to discuss science and commiserate about life every day. From high-school and college, thank you to my friends for always being there to take my mind off work and enjoy New York for the incredible city that it is.

To my family; thank you to all of my family in New York for the support and help whenever I needed it. Thank you to my parents, who despite both having PhDs, never forced anything on

me and only encouraged and let me discover my own passions. I'll never be able to truly repay them for the love, privileges, and support given to me through my whole life, but I appreciate it every day. Thank you to my brother, who has always been a best friend, sending me the most interesting recipes to cook or songs to listen to, not to mention being one of the smartest, kindest people I know.

Finally, I can't really put into words the amount of gratitude and love I have for my now fiancée, Geraldine, who truly got me through the lows of the PhD while being my biggest cheerleader during the highs. I could not have done this without you.

ABSTRACT

This dissertation investigates self-supervised learning methods for training deep neural network (DNN) models that better align with both human behavior and neural responses in early visual areas. We first propose VITO, an attention-guided contrastive video-pretraining method, that improves dramatically on prior work to learn general, robust, and more human-aligned representations from natural video data. We specifically demonstrate that dynamic temporal content is required for the improved robustness and human-alignment. We next explore a complementary line of work focused on improving the alignment of intermediate DNN representations with early visual areas. We first provide a simple demonstration that selectivity for visual texture can be learned via optimizing a single-layer objective in a biologically-inspired architecture modeled off of areas V1 and V2. We then refine and extend this study to a more general layerwise learning paradigm, capable of learning features simultaneously in a two-layer network. We do so by leveraging a novel self-supervised layerwise complexity-matched learning paradigm. Our trained model provides better predictions of neural responses in early visual areas and particularly achieves state-of-the-art predictions for cortical area V2. Finally, we provide some preliminary analyses probing the limitations of current regression-based evaluations for measuring alignment with neural responses. Taken together, this thesis lays the foundation for future research in using learned DNNs to reveal new organizing principles for how selectivities are formed in visual hierarchies, with potential implications for both neuroscience and machine learning.

Contents

Acknowledgments	iii
Abstract	v
List of Figures	ix
List of Tables	xi
List of Appendices	xii
1 Introduction	1
1.1 Normative models of vision	1
1.2 Evaluating alignment between brains and machines	7
1.3 Thesis Organization	11
2 Self-supervised video pretraining yields robust and more human-aligned representations	12
2.1 Overview	12
2.2 Introduction	13
2.3 Method	15
2.4 Results	19
2.5 Related work	30
2.6 Discussion	32

2.7	From behavior to neural alignment?	33
3	Self-supervised learning of a biologically-inspired visual texture model	35
3.1	Overview	35
3.2	Introduction	36
3.3	Methods	38
3.4	Related Work	45
3.5	Results	46
3.6	Discussion	52
4	Layerwise complexity-matched self-supervised learning yields improved models of cortical area V2	55
4.1	Overview	55
4.2	Introduction	56
4.3	Methods	60
4.4	Results	68
4.5	Discussion	76
4.6	Limitations and Future Work	79
5	On the limitations of current neural benchmarking	81
5.1	Overview	81
5.2	Spatially-resolved neural datasets	81
5.3	A sparse regression approach to measuring neural alignment	83
6	Discussion	88
6.1	Temporally-informed models of image perception	88
6.2	Layerwise complexity-matched learning	89
6.3	Evaluating DNN alignment with neurons and behavior	91

6.4 Concluding remarks	92
Appendices	93
Bibliography	127

List of Figures

1.1	Normative description of the computation of V1 simple and complex cells	2
1.2	DNNs as models of the ventral stream	4
1.3	Conceptual self-supervised contrastive learning depiction.	5
1.4	Example augmentations from the SimCLR method.	6
1.5	Example images from the Model-vs-human benchmark	8
1.6	Current paradigm in using DNNs as models of visual perception	10
2.1	Learning to attend to related video content	15
2.2	VITO is robust to natural, real-world corruptions.	22
2.3	VITO attention maps capture human-defined object saliency.	24
2.4	VideoNet dataset improves transfer performance to image tasks.	28
3.1	Biologically-inspired texture model architecture.	39
3.2	Texture classification evaluation methods	43
3.3	Self-supervised V2Net classifies textures most efficiently.	47
3.4	V2Net requires both simple and complex cells for optimal performance.	48
3.5	V2Net outperforms pre-trained VGG layers in texture family representational similarity with V2 neurons.	51
4.1	DNN object recognition performance predicts human recognition behavior, but not primate early visual responses.	58
4.2	Layerwise complexity-matched learning.	60

4.3	Layerwise complexity-matched objective.	63
4.4	LCL-V2 model outperforms other models in accounting for V2 responses.	68
4.5	LCL-V2 model outperforms other models in captureing texture modulation prop- erties of V2 neurons.	70
4.6	LCL-V1 outperforms learned models in V1 predictivity.	72
4.7	Substituting complexity-mismatched or non-contrastive objectives decreases neu- ral alignment.	73
4.8	LCL-V2Net improves OOD generalization and human behavior error consistency.	75
5.1	Method for recording spatially-resolved IT response maps	82
5.2	V1 PLS regression for random and trained ResNet-50	84
5.3	Sparse regression demonstration for explaining V1 data.	86
A.1	Example augmented frames and learned attention masks.	100
A.2	Key ablations of VITO model on PASCAL segmentation.	101
A.3	Training VITO with different temporal frame sampling schemes.	102
B.1	Example curated texture images	107
B.2	V2Net vs. DNN texture classification efficiency	108
B.3	V2Net discriminates natural texture from spectrally matched noise.	109
B.4	Model vs. V2 Neural T-SNE representation of texture images.	111
C.1	Pictorial diagram of the original Barlow Twins method.	117
C.2	Example texture family and spectrally-matched noise images.	121
C.3	Comparing Learned Receptive Fields.	124
C.4	Comparing spatial phase selectivity between models and macaque V1 neurons	125
C.5	Ny vs. Nx V1 receptive field structure.	125

List of Tables

2.1	VITO representations generalizes strongly to both image and video-based tasks	20
2.2	VITO attention maps correlate with human saliency maps.	24
2.3	VITO is more consistent with human judgments on shape-biased stimuli	25
2.4	All components of VITO pretraining matter for downstream performance.	29
A.1	VITO further dataset and method ablations.	103
A.2	Scaling VITO to transformer architectures performs well	103
A.3	VITO achieves strong transfer performance on downstream image-based tasks	104
A.4	VITO significantly outperforms all image-pretraining baselines on DAVIS 2017 video segmentation.	105
A.5	Additional evaluations of VITO on action-recognition tasks	106
C.1	LCL-V2 Two-layer Architecture	113
C.2	LCL-V2 projector network architectures.	113
C.3	LCL-V2Net downstream architecture.	114
C.4	LCL-V2Net generalizes to distribution shifts and correlates with human behavior.	126

List of Appendices

A Self-supervised video pretraining yields robust and more human-aligned representations	93
B Self-supervised learning of a biologically-inspired visual texture model	107
C Layerwise complexity-matched self-supervised learning yields improved models of cortical area V2	112

1 | INTRODUCTION

1.1 NORMATIVE MODELS OF VISION

Humans possess astounding visual systems that allow us to flexibly and robustly solve complex visual tasks. However, even after decades of research, how such a complex cascade of brain areas coordinates to produce coherent percepts remains a question.

The history of building normative models and theories to understand vision goes back to seminal work by [Hubel and Wiesel \(1959; 1962; 1968\)](#), who found that single neurons in primary visual cortex (V1) are selective for orientation of luminance edges. They further developed a categorization of these cells into ‘simple cells’ (selective for edges with a specific polarity and location) or ‘complex cells’ (selective for orientation regardless of polarity and location). These experiments allowed for proposals for the computations implemented by simple cells (pooling from LGN cells) and complex cells (pooling from V1 simple cells). Eventually, these results led to “normative theories” of the computations performed by area V1: simple cells described by a linear filter and rectifying nonlinearity ([Movshon et al., 1978b](#)) and complex cells described by sums of rectified simple cells ([Movshon et al., 1978a](#)). See [Fig. 1.1](#) for a visual depiction of this specifically using quadrature phase simple cells for constructing the complex cell. Eventually, these models were extended to include more complex nonlinearities (gain-control) ([Carandini et al., 1997](#); [Shapley and Victor, 1979](#)) and then to population models that could expand the single linear filter to multiple channels, tiling orientations and spatial frequencies ([Simoncelli and Freeman, 1995](#)).

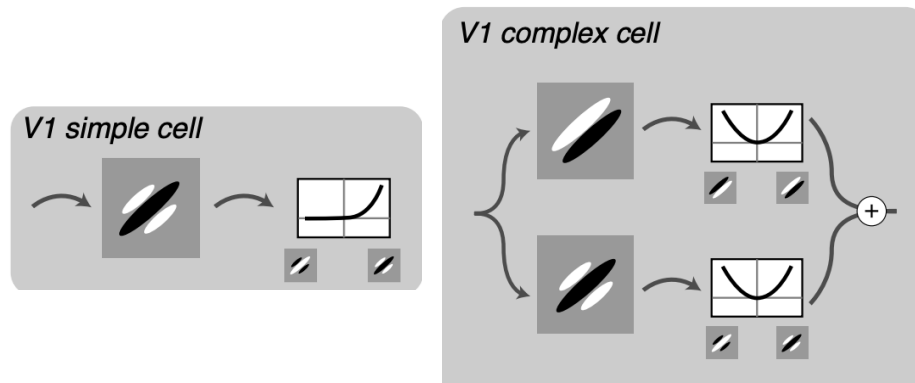


Figure 1.1: Left: V1 simple cells are captured by an oriented linear receptive field followed by rectifying nonlinearity. Right: V1 complex cells are captured by pooling quadrature phase subunit simple cells. Figure adapted from (Ziomba, 2016).

In a similar manner, experimental discoveries have slowly uncovered more complex single neuron selectivities in other areas of the visual hierarchy, for example texture in area V2 (Freeman et al., 2013; Ziomba, 2016) and objects in area IT (DiCarlo et al., 2012; Gross, 1973; Mishkin and Ungerleider, 1982). However, even with an understanding of the selectivities in these stages of the visual hierarchy, it has proven difficult to hand-craft normative models that can produce such selectivities (as done in V1). Therefore, while this approach was useful for characterizing and developing models of early stages of vision, we must take a different path to building models of the larger visual hierarchy.

1.1.1 BOTTOM-UP LEARNED MODELS

Complementary to the hand-engineered approaches to modeling V1, many have attempted to specify bottom-up normative principles that govern computations in the brain and use these principles to then learn models of early vision. A non-exhaustive list of such principles include sparsity, coding efficiency, or temporal prediction (Atick and Redlich, 1990; Bell and Sejnowski, 1997;

Cadieu and Olshausen, 2012; Hoyer and Hyvärinen, 2002; Karklin and Lewicki, 2009; Karklin and Simoncelli, 2011; Li, 1996; Olshausen and Field, 1996; Schwartz and Simoncelli, 2001; Van Hateren and van der Schaaf, 1998; Wiskott and Sejnowski, 2002). While these methods also could not scale far beyond descriptions of V1, they were critical in framing a new way of approaching modeling of vision. Specifically, they were the first to use parameterized architectures (with basic computational units like linear filters, rectifiers, and divisive normalization) where the parameters are learned based on *optimizing an objective function*. The goal is to specify the objective function to satisfy one of the aforementioned normative principles.

1.1.2 THE DEEP LEARNING AGE

Extending this idea beyond early layers, the emergence of optimized deep neural networks (DNNs) provided new opportunities for developing models of previously unexplained parts of the visual hierarchy (Douglas et al., 1989; Fukushima, 1980; Heeger et al., 1996; LeCun et al., 1989; Riesenhuber and Poggio, 1999). Leveraging networks built from simple parameterized computational units (linear filters and rectifiers) and new optimization techniques, there has been an explosion of ‘task-driven’ DNNs optimized to perform specific visual tasks. The most powerful of these tasks has proven to be object recognition. Optimizing DNNs for this objective has led to the first models that begin to capture response properties of neurons deep in the visual hierarchy (Kubilius et al., 2019; Schrimpf et al., 2018; Yamins et al., 2014; Zhuang et al., 2021). Rather than comparing hand-crafted model filters to receptive fields of neurons, the current best models of many brain areas are now obtained by extracting responses from intermediate representations of these trained DNNs and comparing the responses directly to responses of neurons in a corresponding brain region. This framework is depicted in Fig. 1.2, and has become the dominant method for developing hierarchical models of cortical neurons.

Moreover, because these networks are image computable and capable of solving real-world visual tasks, they can also be compared with primate or human task performance and behavior.

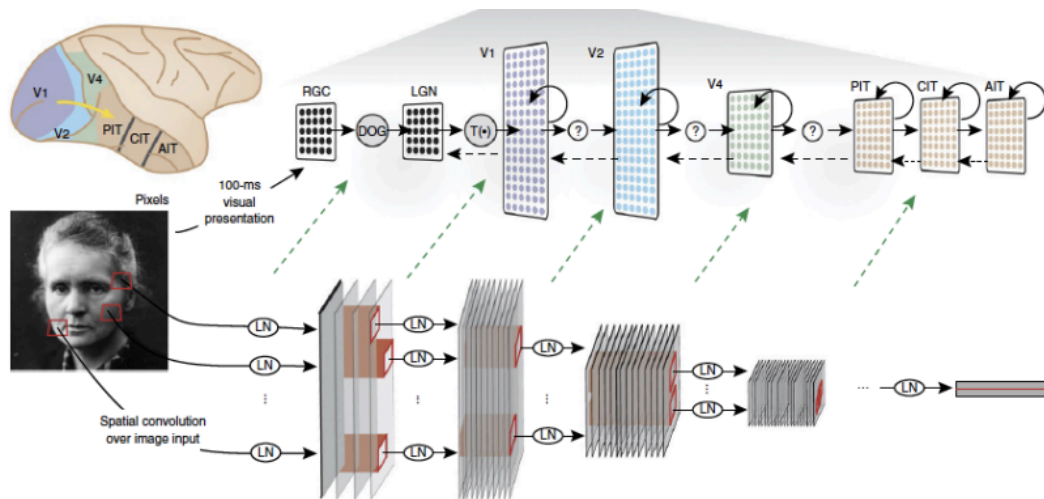


Figure 1.2: An image is presented to a DNN model (Bottom) and a primate brain (Top). Neural responses in the brain are recorded to the stimulus at a specific layer. Similarly, responses are mapped to a specific layer (green arrows) responses in the DNN. Figure adapted from [Yamins et al. \(2014\)](#)

Early results showed that these supervised DNNs are in fact also generally predictive of the overall category-level decisions of primates during object recognition tasks ([Ghodrati et al., 2014](#); [Jozwik et al., 2016](#); [Kheradpisheh et al., 2016](#)).

1.1.2.1 SELF-SUPERVISED LEARNING

However, while supervised object recognition has been a powerful objective function for optimizing DNNs, it is widely thought that receiving the scale of supervision these networks require (millions of labeled examples) is biologically-implausible. As a result, in recent years, there has been a revolution in *self-supervised learning*.

This dissertation focuses heavily on the use of self-supervised learning for training DNNs, so it is worth providing a brief conceptual understanding. The term self-supervision, comes from the intuition that instead of having labeled examples for classification, the supervisory signal is generated from the original data itself or via another *internal process*. Examples of early self-

supervised methods include tasks such as rotation prediction (predicting the rotation of an image) (Gidaris et al., 2018) and spatial jigsaws (ordering shuffled image patches) (Noroozi and Favaro, 2016). However, for this work we will focus on the class of ‘contrastive’ self-supervised methods. Contrastive self-supervised methods start with an ‘anchor’ image. A ‘positive’ view, is generated from this anchor via a deformation (aka augmentation). All other images in a batch or dataset are considered ‘negatives’. The goal of sample-contrastive methods (that we use in Chapter 2) is to optimize the weights of the encoder network such that anchor image representations are closer to the positive representations than to other negative image representations. This process is depicted in Fig. 1.3. It is important to note that unlike in supervised learning, negative images

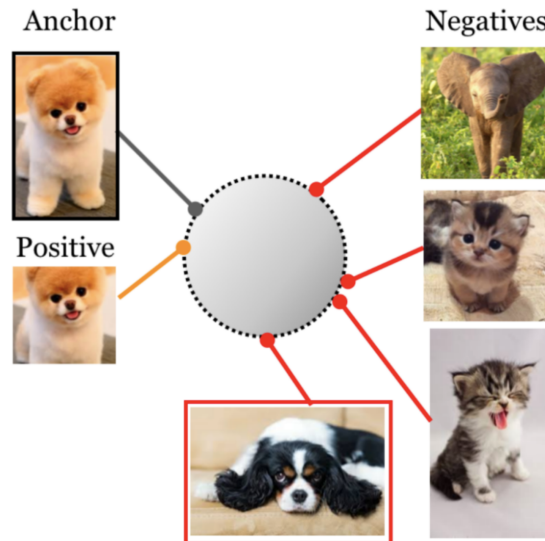


Figure 1.3: Anchor image and augmented positive images are brought closer in representation space (normalized hypersphere) than anchor images and negative images. Negative image (red outlined dog) may still come from the same class as the anchor. Figure adapted from Khosla et al. (2020).

may contain classes that are the same as the anchor class. The network is trained to discriminate individual images while remaining invariant to the augmentations applied to the anchor view. As an example, we show a wide range of synthetic augmentations that are normally applied to images for training of end-to-end self-supervised DNNs. The set of augmentations (ranging from spatial to photometric deformations) is shown in Fig. 1.4 (taken from Chen et al. (2020b)). The

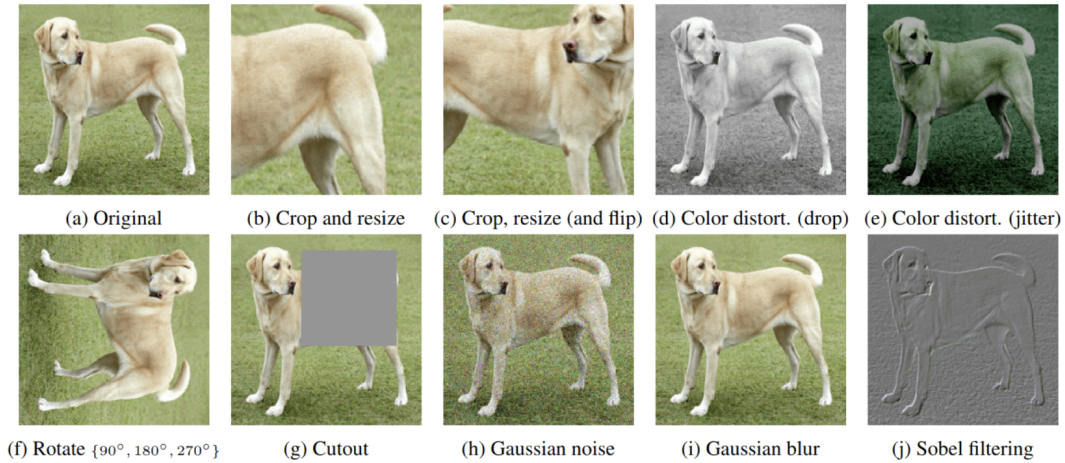


Figure 1.4: Example augmentations used in the SimCLR contrastive learning method. Figure is adapted from [Chen et al. \(2020b\)](#)

choice of these augmentations has empirically been made over time to maximize usefulness of the learned representations for object-recognition tasks. The magnitude of each augmentation and the number of composed augmentations, controls the complexity (or difficulty) of the invariance learning problem, a topic we revisit in [Chapter 4](#).

In the following chapters, we will demonstrate how standard augmentation schemes are far too aggressive and may lead to learning of invariances that do not align with those learned by primate visual systems.

Additionally, we note that as a normative learning principle, contrastive self-supervision seems to be promising, as recent work has shown these networks to be highly predictive of neural representations in later stages of visual cortex ([Zhuang et al., 2021](#)).

1.1.3 LEARNING SPATIAL REPRESENTATIONS FROM TEMPORAL EXPERIENCE

Till now we have described procedures for learning visual representations from static image inputs. However, as noted above, some of the early normative principles used to describe learning in early visual cortex centered on capturing image statistics that are predictable over

time (Wiskott and Sejnowski, 2002). Even more recent theories have been developed and verified along similar lines of temporal prediction (Hénaff et al., 2021a; 2019a).

From a behavioral point of view, the impact of temporal learning on static object perception in particular is even more striking. Work of Kellman and Spelke (1983); Spelke (1990); Spelke and Kinzler (2007) has shown that infants learn to recognize object structure from motion. Even in adults, it has been shown that altered spatiotemporal experience that changes learned temporal associations of object shape before an after saccades, can drastically affect position-invariant recognition (Cox et al., 2005).

We provide this context to note that visual perception and neural representations of static objects and scenes are shaped by learning within a temporally evolving world. However, from the perspective of DNN vision models, the impact of temporal learning (learning from videos) has largely been constrained to models that are then evaluated on video-level tasks. This motivates our work in Chapter 2, as we demonstrate how models of static vision can be trained effectively to leverage and learn from spatiotemporal content.

1.2 EVALUATING ALIGNMENT BETWEEN BRAINS AND MACHINES

The explosion of highly-performant supervised and self-supervised DNNs trained in recent years has led to a corresponding growth in attempts to take these networks seriously as models of both human behavior and neural responses. Here we briefly review these evaluation methodologies.

1.2.1 HUMAN-BEHAVIORAL COMPARISONS

Many psychophysical-based evaluations have been proposed in recent years to test DNN abilities to capture basic aspects of human visual perception. These include methods such as eigendis-tortions (Berardino et al., 2017), controversial stimuli (Golan et al., 2020) and metamers (Feather

et al., 2023). While these have been quite useful in demonstrating limitations and/or failings of current DNNs as models of human perception, they require human extensive evaluation to obtain quantitative metrics.

On the other hand, there has also been a long line of work attempting to quantitatively measure how well model decisions match human decisions on the same recognition tasks. Initial studies showed that DNNs are generally predictive of the overall category-level decisions of primates during object recognition tasks (Ghodrati et al., 2014; Jozwik et al., 2016; Kheradpisheh et al., 2016). However, they have not been predictive of more detailed behavior, as measured by model consistency with human recognition confusion matrices (Rajalingham et al., 2018). More recent benchmarks have taken these comparisons further by creating datasets to measure the per-trial consistency between models and humans on recognition tasks using a wide range of out-of-distribution (OOD) images. The specific benchmark that has obtained prevalence in recent years was proposed by Geirhos et al. (2021). The benchmark consists of 17 OOD image classes (examples shown in Fig. 1.5 based on applying both parametric and non-parametric distribution shifts to original images from the ImageNet-1K dataset (Krizhevsky et al., 2012).

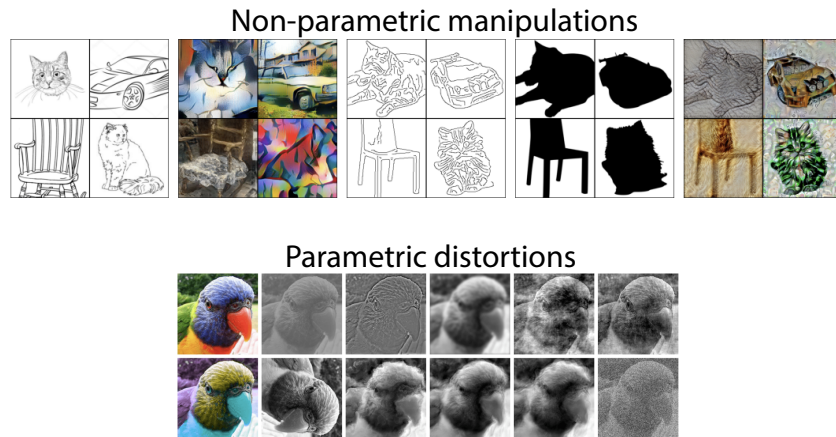


Figure 1.5: Examples of the 17 OOD distribution shifts applied to images in the model-vs-human benchmark. Non-parameteric shifts are shown in the top row, while parameteric (noise-based) shifts are shown in the bottom row. Figure adapted from Geirhos et al. (2021)

We use this benchmark in both Chapters 2 and 4 for its comprehensiveness. In Chapter 2, we explore an additional class of alignment evaluations specifically concerning human saliency data from [Linsley et al. \(2018\)](#).

While these benchmarks still cover a very limited set of the potential ways to evaluate model alignment with human behavior, we believe they provide a sufficient starting point to test our hypotheses.

1.2.2 NEURAL RESPONSE COMPARISONS

Unlike behavioral benchmarks, comparing DNN responses to neural responses in visual cortex is a far more ambiguous task. One line of work centers on understanding and comparing representational geometries or how responses are organized in a high-dimensional space ([Chung and Abbott, 2021](#); [Chung et al., 2018](#); [Kriegeskorte and Wei, 2021](#)). Quantitative methods for comparing representational geometries began with representational similarity analysis (RSA) ([Kriegeskorte et al., 2008](#)), and have continued to be developed into many variants ([Duong et al., 2022](#); [Schütt et al., 2023](#); [Williams et al., 2021](#)). These methods have many benefits, but are in some ways less direct than the second major set of comparison protocols that use regression-based metrics. Starting in [Yamins et al. \(2014\)](#) and being developed in [Schrimpf et al. \(2018\)](#), the BrainScore benchmark, has become a highly used evaluation protocol for comparing the direct ability for model responses to predict corresponding neural responses in the brain. Briefly, this method uses a form of linear regression (PLS) to linearly weight model responses to a set of visual stimuli, in order to best predict neural responses to the matched stimuli. We will primarily focus on this regression-based neural alignment in the following chapters.

1.2.3 SUMMARY AND MOTIVATION

Given this background, we provide a brief summary of the current paradigm of using DNNs as models of behavior and the brain. In Fig. 1.6, we highlight that the current workflow involves training DNNs based on normative principles (generally supervised or self-supervised learning objectives). These models are first evaluated on classical image task benchmarks. If a model doesn't generalize across basic visual tasks, then we disregard it for not capturing basic visual capabilities. However, if this threshold is passed, then this model is evaluated in terms of both human behavior alignment and neural alignment. We now highlight two problems that motivate

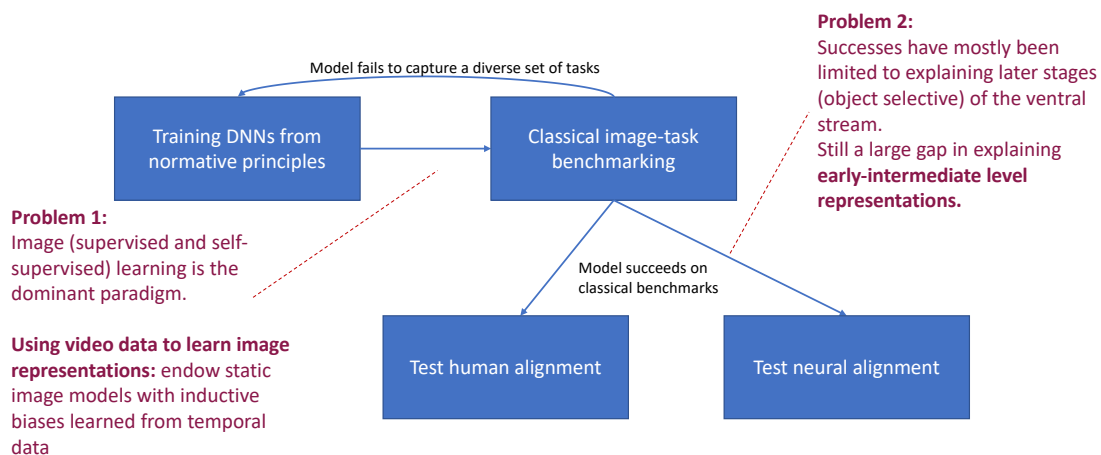


Figure 1.6: We depict the flowchart of how DNN models are currently designed and evaluated as models of visual perception. In red, we highlight two of the main current problems and how this motivates our work.

the primary work in this dissertation.

First, most current state-of-the-art DNNs are only trained on static image datasets. However, as described in Sec. 1.1.3, there is much psychological and neuroscientific evidence to support training models of visual perception that leverage learning from time.

Second, while this current paradigm has led to successes in finding models that explain later stages (IT) of the ventral stream, there is still a surprisingly large gap in explaining responses from neurons early and intermediate cortical areas.

We attempt to take steps towards more behaviorally- and biologically-aligned DNNs by specifically addressing these two issues.

1.3 THESIS ORGANIZATION

The remaining chapters of this thesis are organized as follows. Chapter 2 proposes a novel method for learning spatial representations from natural video data that improves the model’s task generality, robustness and alignment with human behavior. Chapters 3 and 4 develop layer-wise self-supervised learning methods for constraining intermediate ANN representations such that they better predict neural responses in early visual areas. In Chapter 5, we briefly explore ways in which current popular benchmarks for measuring alignment of models and brain responses may be limited. Finally, Chapter 6 provides concluding thoughts and remarks.

2 | SELF-SUPERVISED VIDEO PRETRAINING YIELDS ROBUST AND MORE HUMAN-ALIGNED REPRESENTATIONS

2.1 OVERVIEW

The main findings in this work are to be published in the Proceedings of the 37th Conference on Neural Information Processing Systems. (Parthasarathy et al., 2023a).

Humans learn powerful representations of objects and scenes by observing how they evolve over time. Yet, outside of specific tasks that require explicit temporal understanding, static image pretraining remains the dominant paradigm for learning visual foundation models. We question this mismatch, and ask whether video pretraining can yield visual representations that bear the hallmarks of human perception: generalisation across tasks, robustness to perturbations, and consistency with human judgements. To that end we propose a novel procedure for curating videos, and develop a contrastive framework which learns from the complex transformations therein. This simple paradigm for distilling knowledge from videos, called VITO, yields general representations that far outperform prior video pretraining methods on image understanding tasks, and image pretraining methods on video understanding tasks. Moreover, VITO representations are significantly more robust to natural and synthetic deformations than image-, video-, and

adversarially-trained ones. Finally, VITO’s predictions are strongly aligned with human judgements, surpassing models that were specifically trained for that purpose. Together, these results suggest that video pretraining could be a simple way of learning unified, robust, and human-aligned representations of the visual world.

2.2 INTRODUCTION

With the explosion of recent AI breakthroughs, humans now interact with and depend on the outputs of these models at an unprecedented rate. It is therefore increasingly important that these models be aligned with human abilities, judgements, and preferences. In the context of computer vision systems, human alignment can be quantified with accurate generalization across a wide range of tasks (Everingham et al., 2015; Soomro et al., 2012; Zhou et al., 2017), robustness to various input deformations (Taori et al., 2020), and consistency with human perceptual judgements (Geirhos et al., 2020b). While each of these challenges has been tackled separately, progress along one axis has often come at the expense of the others. For example, gains in robustness (Goodfellow et al., 2014) or temporal understanding (Gordon et al., 2020; Wu and Wang, 2021; Xu and Wang, 2021) have thus far come at the cost of spatial understanding, and scaling the model and dataset size, while improving task-generality and robustness (Dehghani et al., 2023; Oquab et al., 2023), can be detrimental for their consistency with human perception (Dehghani et al., 2023; Kumar et al., 2022).

In this work we question this trend, and ask whether improvements to all aspects of human alignment can be made with the appropriate pretraining methodology. Specifically, humans and animals have long been thought to learn from the dynamic evolution of natural scenes (Barlow et al., 1961; Palmer et al., 2015; Rao and Ballard, 1999) and we hypothesize that artificial visual systems will be more aligned by appropriately leveraging natural video pretraining. In particular, while many current self-supervised methods (Caron et al., 2021; Chen et al., 2020b; He et al., 2020;

Hénaff et al., 2019b) learn representations that are invariant to synthetic augmentations that capture important image priors such as scale-, color-, and translation-invariance, these represent a small part of the complex (and signal-rich) changes in pose, viewpoint, and motion that are captured from natural videos. Predicting the evolution of videos is also a natural means of learning intuitive physics and model-based reasoning (Battaglia et al., 2013; Hénaff et al., 2021a; 2019a).

Practically, we develop a self-supervised contrastive framework which learns to locate the most stable and distinctive elements in temporally displaced video frames, and maximizes their invariance. Secondly, we find the statistics of standard video datasets to have a detrimental effect on the quality of the resulting representations, as measured by their performance on canonical scene understanding tasks. We therefore introduce a simple, yet powerful video curation procedure—VideoNet—which aligns their class distribution with that of ImageNet, and which redresses the imbalance between image and video learning. In concert, this paradigm constitutes a new methodology for distilling the knowledge of **videos into** visual representations: VITO.

VITO yields task-general representations that perform well across both spatial and temporal understanding tasks. Particularly, VITO shows large gains over prior video pretraining efforts in scene understanding tasks, while achieving similarly large performance gains over image pretraining on video understanding tasks. Furthermore, VITO significantly outperforms the default ImageNet pretraining as well as adversarial pretraining on image classification tasks subject to natural distribution shifts. Finally, we find that even without a significant expansion in model size, VITO is not only task-general and robust in performance, but also quantitatively captures multiple aspects of human perceptual judgements, surpassing models specifically trained for that purpose.

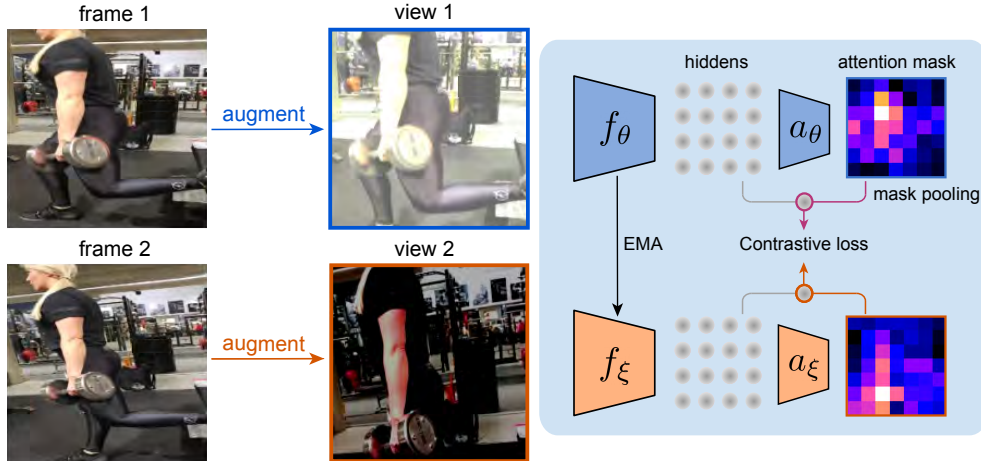


Figure 2.1: Learning to attend to related video content. Each augmented frame is encoded by the network f as a spatial array of hidden vectors. The attention module a takes as input features from one view and produces a mask that isolates features that are likely to be predictive of the other, temporally-displaced view. The attention-gated features are pooled accordingly, and both the feature extractor and attention module are trained to satisfy the contrastive objective. Subscripts θ and ξ refer to online and target (EMA) networks respectively.

2.3 METHOD

We pretrain image representations using video datasets, then transfer them to a range of downstream tasks that test image, video, and robust understanding. We adopt the ResNet-50 architecture for our initial exploration, then validate our results with Swin transformers (see Sec. A.2.4).

2.3.1 SELF-SUPERVISED PRETRAINING

Our method for distilling **videos into** image representations, **VITO**, builds robust visual representations by learning to track stable and distinctive content in videos while they evolve over time.

Natural video pipeline. The key to our method is to distill the natural transformations present in videos into image-based representations. Given a video-clip, we sample frames according to a

distribution \mathcal{T} and further transform each frame with image-based augmentations:

$$v^1 \sim \mathcal{A}_1(x_1) \quad v^2 \sim \mathcal{A}_2(x_2) \quad x_1, x_2 \sim \mathcal{T}(\{x_t\}_{t=1,\dots,T}) \quad (2.1)$$

where the distribution \mathcal{T} samples frames uniformly from a video clip of length $T = 2.56s$ and the image transformations \mathcal{A}_l include random cropping, flipping, blurring, and point-wise color transformations (Grill et al., 2020), see appendices A.1.1 and A.2.2, and Figure A.3 for an ablation.

We note that video frames (or even uncurated image data) typically differ from the statistics of (object centered) ImageNet images, with more variable viewpoints and a larger field-of-view that can cover multiple objects in complex scenes. As a result, the aggressive random cropping from Grill et al. (2020) (whose smallest crops cover only 8% of the original image) can result in “positive” pairs with very different semantic content (e.g. entirely different objects). We therefore suggest and empirically validate that larger crop sizes (e.g. increasing the minimum crop size to 40%) are beneficial when learning from real-world video frames (see Figure A.2).

Multi-scale contrastive attention pooling. Standard contrastive frameworks use global average pooling of hidden vectors to obtain a single representation of each view. It has been shown that using dense contrastive losses can lead to significant improvements (Bai et al., 2022; Hénaff et al., 2021b; Wang et al., 2021b; Xie et al., 2021c), but these methods require establishing correspondences across views. Whereas correspondences can easily be obtained from static images, when temporal deformations are introduced they require some form of object or point tracking (Sharma et al., 2022). Furthermore, with the larger field-of-view of video frames, correspondence learning becomes an increasingly difficult task. In this work, we propose a more general, adaptive method for learning correspondences at multiple scales. Our method learns what features should be attended to in order to solve the contrastive learning problem across temporally displaced views.

As shown in Figure 2.1, given a view v^l the feature extractor outputs a spatial map of feature

vectors $\mathbf{h}_\theta^{l,s} \in \mathcal{R}^{h \times w \times c}$ at a given scale s , where different scales correspond to the outputs of different blocks of a ResNet for example. At each scale, we introduce a 2-layer attention MLP a_θ^s which outputs a mask $\mathbf{m}^{l,s} = \text{softmax}(a_\theta(\mathbf{h}_\theta^{l,s}))$ that we use to spatially weight and pool hidden vectors:

$$\hat{\mathbf{h}}_\theta^{l,s} = \sum_{i,j} \mathbf{m}^{l,s}[i,j] \mathbf{h}_\theta^{l,s}[i,j] \quad (2.2)$$

which we concatenate and transform with the two-layer MLP projector: $\mathbf{z}_\theta^l = g_\theta(\hat{\mathbf{h}}_\theta^l)$ where $\hat{\mathbf{h}}_\theta^l = [\hat{\mathbf{h}}_\theta^{l,s}, s \in 1 \dots S]$. In our experiments, we find that for the canonical ResNet-50 architecture, attending over the outputs of the last two ResNet blocks (i.e. $S = 2$) is optimal given our evaluations. These hidden vectors are then transformed with a standard two-layer MLP g_θ , yielding projections $\mathbf{z}_\theta^l = g_\theta(\hat{\mathbf{h}}_\theta^l)$. We enforce invariance across views using the standard InfoNCE loss (Oord et al., 2018), encoding targets with slowly-varying *target* networks f_ξ and g_ξ that are exponential moving averages of the online network (Grill et al., 2020)

$$\mathcal{L}^{ij}(\theta; \xi) = -\log \frac{\exp(\mathbf{z}_\theta^i \cdot \mathbf{z}_\xi^j)}{\exp(\mathbf{z}_\theta^i \cdot \mathbf{z}_\xi^j) + \sum_n \exp(\mathbf{z}_\theta^i \cdot \mathbf{z}_\xi^n)}. \quad (2.3)$$

$\{\mathbf{z}_\xi^n\}_n$ are *negative* features computed from frames from other videos in the batch. The final, multi-view loss is evaluated for all pairs $\mathcal{L}(\theta; \xi) = \sum_{i \neq j} \mathcal{L}^{ij}(\theta; \xi)$.

2.3.2 ADDRESSING DATASET DOMAIN MISMATCH

We began investigating the potential for learning general representations from videos, using standard datasets including Kinetics, AudioSet, and YouTube-8M. However, Kinetics is quite small and is limited in scope to human actions. On the other-hand, AudioSet and YouTube-8M are noisy and have very imbalanced class distributions. Additionally, prior work has shown that even self-supervised methods are quite sensitive to the pretraining distribution (Tian et al., 2020). Yet over the last decade, it has been shown that ImageNet can be used for learning image representations

that transfer well to many downstream tasks. As a result, we hypothesized that collecting a minimally-curated video dataset matched to the rough properties of ImageNet would be beneficial for learning a more general visual model from videos.

To test of this hypothesis, we developed a data curation pipeline—*VideoNet*—to filter online videos such that our training data more closely matches the distribution of ImageNet categories. For each of the 1,000 ImageNet categories, we retrieved 5,000 video clips whose title included the category’s name or a synonym. We then filtered these videos by applying an image classifier (pretrained ResNet-50 on ImageNet) to verify that the videos contained the intended object category. We classified the first 100 frames of each video and discarded videos for which the query category was not equal to the ResNet’s top-1 prediction for any of the frames. We also discarded videos of less than 10s in length.

While the VideoNet procedure is close in conceptualization to the method used to create the R2V2 dataset proposed by [Gordon et al. \(2020\)](#), it differs in a few ways. First, we utilize full video clips that allow us to uniformly sample frames at any time point rather than the fixed sampling of frames that are 5s apart in R2V2. Second, by using the ImageNet classifier to filter videos, we can reduce mismatch with the ImageNet distribution that can arise from incorrect tagging and noisy labeling of online videos. This is verified by the fact that only 1.18M of the 5M retrieved videos met our filtering criteria. We also note that the use of classification-based filtering is just one method of curation. While we demonstrate in [Sec. 2.4.3](#), that this curation does provide large benefits in the context of video pre-training compared with existing datasets, there is still great potential to make improvements by utilizing larger target datasets (such as ImageNet-22K) and utilizing alternative curation strategies such as the nearest-neighbor retrieval proposed by [Oquab et al. \(2023\)](#) in creating the LVD-142M image dataset.

2.4 RESULTS

Humans are able to solve a range of visual tasks that require complex spatial and temporal reasoning, including generalizing to noisy or out-of-distribution (OOD) scenarios. Therefore, we first benchmark VITO against image and video pretrained models on a variety of tasks to demonstrate sufficient generality and robustness in task performance. We then assess whether VITO not only captures these task-based properties, but also displays strong quantitative alignment with human behavior.

2.4.1 VITO GENERALIZES ACROSS DIVERSE VISUAL TASKS

We present in Table 2.1 the transfer performance of VITO compared to strong supervised and self-supervised baselines on dense scene understanding (semantic segmentation and object detection), video understanding (video segmentation and action recognition), and out-of-distribution (OOD) object recognition. On every benchmark, VITO either outperforms or is competitive with the best baselines *for that specific task*.

Scene understanding. We first note that VITO provides large gains over all prior video pretraining methods on scene understanding and robust object recognition. We further validate these comparisons on three additional benchmarks and find that VITO strongly outperforms the prior work across all 5 datasets (PASCAL/ADE20K/COCO/LVIS/IN-1K, see Table A.3). For example, VITO improves over VIVI (Tschannen et al., 2020) by 2-10%, highlighting the importance of data curation and our contrastive formulation. VITO improves over VINCE (Gordon et al., 2020) by 1-12%, highlighting the importance of fine-grained temporal deformations. Finally, VITO improves even over MMV (Alayrac et al., 2020) by 2-15%, despite their use of large-scale text supervision, highlighting the relevance of video-only learning.

Compared with the best supervised and self-supervised image-pretraining methods VITO

Pretraining	Dataset	Scene Understanding		Video Understanding		OOD Recognition	
		ADE20K (mIoU)	COCO (mAP)	DAVIS ($\mathcal{J}\&\mathcal{F}$ mean)	UCF101 (top-1)	IN-A (top-1)	IN-Vid (pm0/ pm10)
Random	-	27.9	39.0	-	-	-	-
<i>Standard image pretraining</i>							
Supervised	IN-1K	33.5	44.2	66.1	83.4	2.2	67.7/52.4
BYOL (Grill et al., 2020)	IN-1K	38.8	43.7	66.6	85.6	-	-
MoCLR (Tian et al., 2021)	IN-1K	39.2	43.9	65.5	85.5	3.7	64.7/50.0
DINO (Caron et al., 2021)	IN-1K	39.0	44.3	65.3	85.4	5.0	65.2/52.0
<i>Robust image pretraining</i>							
Stylized-IN (?)	SIN+IN	-	-	-	83.3	2.0	68.4/51.7
L2-Robust (Madry et al., 2017)	IN-1K	-	-	-	83.7	2.1	65.2/51.6
<i>Video pretraining</i>							
VIVI (Tschannen et al., 2020)	YT8M	34.2	41.3	-	-	0.5	57.9/36.5
MMV-VA (Alayrac et al., 2020)	AS+HT	32.5	41.3	-	-	-	-
VINCE (Gordon et al., 2020)	R2V2	35.7	42.4	66.1	-	-	-
VFS (Xu and Wang, 2021)	K400	31.4	41.6	67.8	-	-	-
CycleCon (Wu and Wang, 2021)	R2V2	35.6	42.8	-	82.8	0.4	50.4/30.1
VITO	VidNet	39.4	44.0	68.2	87.4	5.4	70.6/57.2

Table 2.1: VITO representations generalize to a variety of tasks in both image and video modalities.. VITO surpasses models specialized for each task. For external models, we finetune publicly available checkpoints.

achieves competitive performance on these same benchmarks (Table 2.1 and Table A.3). To our knowledge, VITO is the first video pretrained method to close the gap with ImageNet pretraining on large-scale scene understanding benchmarks such as these.

Video understanding. We next ask whether this increased spatial understanding come at the cost of traditional benefits of video pretraining on video tasks. We find that this is not the case, evaluating on DAVIS segmentation and UCF-101 action recognition. On DAVIS, which tests the ability to segment an object over its dynamic temporal evolution, VITO features capture fine-grained temporal deformations of objects far better than ImageNet pretraining methods, as well as the best video pretraining methods (See Table A.4 for additional comparisons). On UCF-101, which tests the ability to classify global spatio-temporal features, we find that a simple average pooling of VITO frame representations again outperforms all image pretraining and prior frame-

based video pretraining significantly. VITO even outperforms a number of recent methods that use specialized video architectures (See Table A.5). While VITO under-performs relative to the best video models, we note that these methods either cannot be tested or under-perform on spatial understanding. Additionally, as shown in Table A.5 and Sec. A.1.5, simple learned temporal pooling strategies on top of VITO representations further close the gap with the best video architectures.

Object recognition under distribution shifts. A key feature of human perception is being able to generalize under distribution shifts away from the training data. The standard ImageNet benchmark does not test this, as the validation set is drawn from a similar distribution as the train set. We hypothesize that while ImageNet pretraining can lead to strong performance in-distribution, pretraining on videos can endow models with better generalization capabilities.

We thus evaluate on a suite of benchmarks designed to test distributional robustness (Taori et al., 2020). To test recognition under *natural* shifts we evaluate on the ImageNet-Vid-Robust and ImageNet-A benchmarks (Table 2.1). ImageNet-Vid-Robust tests generalization of image classifiers to natural deformations over time. The anchor frame is identified as the cleanest frame capturing the object, and as time evolves, recognition becomes more difficult. We see that VITO surpasses all models on the anchor frame accuracy (+3% relative to supervised ImageNet training for $pm0$), but more importantly, the accuracy gap grows for the largest temporal displacement (+5% for $pm10$). ImageNet-A on the other hand contains ImageNet-like images that systematically fool ImageNet classifiers (i.e. ‘natural adversarial examples’). On this dataset, while performance is very low across all models, VITO again shows more robustness. For additional comparison, we also evaluate two models (SIN-IN and L2-Robust ($\epsilon = 1$)) which are models trained specifically for robustness (to shape-bias and adversarial attacks respectively). While SIN-IN yields modest improvements on ImageNet-Vid-Robust, neither method approaches the gains in robustness afforded by VITO. Finally, we evaluate robustness on the ImageNet-3DCC dataset, which contains naturalistic and synthetic corruptions applied to clean images from the ImageNet validation set

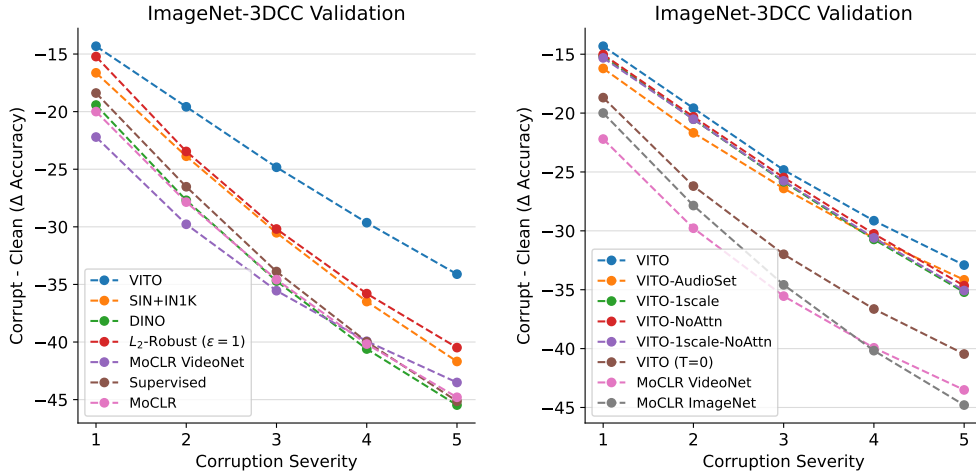


Figure 2.2: VITO is robust to natural, real-world corruptions. ImageNet-3DCC validation accuracy for different levels of corruption severity. (Left): Comparisons with prior work including methods specifically designed to enhance robustness (SIN+IN1K and L2-Robust). (Right): comparisons with ablations of the VITO method/model.

(Kar et al., 2022). To test robustness to conditions of real-world deployment, we choose the subset of corruptions designed with 3D models to be consistent with scene geometry. These include things like fog, near/far focus, motion blur, etc. and have 5 different severity levels per image. In Fig. 2.2 (Left), we plot the difference in accuracy between clean (ImageNet val) and corrupted accuracy across severity levels. This “ Δ -accuracy” provides a measure of how robust a model is as distortion levels increase. We see that across all corruption strengths, VITO shows increased robustness compared to supervised and self-supervised (MoCLR, DINO) ImageNet pre-trained models. The robustness gap grows significantly at the highest corruption levels, demonstrating the generality of this effect (+10% relative to supervised ImageNet training). While the robust training methods (SIN+IN1K and L2-Robust) outperform supervised and MoCLR models, VITO remains significantly more robust, demonstrating that learning from video deformations may endow a more general form of robustness than that provided by either style-transfer or adversarial images.

To quantify further the specific impact of individual components of VITO on robust recognition, we show the same plot (Fig. 2.2 (Right)), now with the ablations described in Sec 2.4.3.

We find that all components of our method and architecture are necessary for best robustness, but in particular there is a striking split between models trained with only spatial deformations (VITO (T=0), MoCLR ImageNet, MoCLR VideoNet) and those trained with video deformations. We find that the models that learn only from image-level spatial deformations suffer significantly in robustness against all of the models that learn from video deformations.

2.4.2 MEASURING EXPLICIT HUMAN-ALIGNMENT

Given that VITO representations display strong generalization across many tasks and robustness to distribution shifts, two signatures of human perceptual intelligence, we now directly ask whether they align with human perceptual representations.

Visual saliency via contrastive attention. We start by comparing VITO’s learned attention masks to human saliency data from the ClickMe dataset (Linsley et al., 2018), as well as saliency maps obtained from a collection of ResNet-50 models. For the supervised and MoCLR ResNets we use standard gradient-based saliency as in Fel et al. (2022). Since our model contains two attention maps at two scales of the ResNet, we upsample both maps to the image size and simply average them to obtain a single map. We compare our attention maps additionally to those obtained from the modified CLIP ResNet (Radford et al., 2021), which also utilizes attention-pooling in the final layer but is trained for image-language alignment (the canonical approach for training state-of-the-art visual language models). Because the CLIP pooling uses multi-head attention, we upsample these maps and average them across heads. Finally, we also compare to the gradient-based saliency maps from a “harmonized” model explicitly trained to align with human saliency (Fel et al., 2022).

Qualitatively, VITO saliency maps appear significantly more aligned with human perception maps than the supervised and CLIP ResNets (Figure 2.3). Surprisingly, VITO appears more aligned than the Harmonized saliency maps across the 4 examples. Quantitatively (using Spearman rank correlation) VITO outperforms the supervised, MoCLR, and CLIP models by a large margin, and

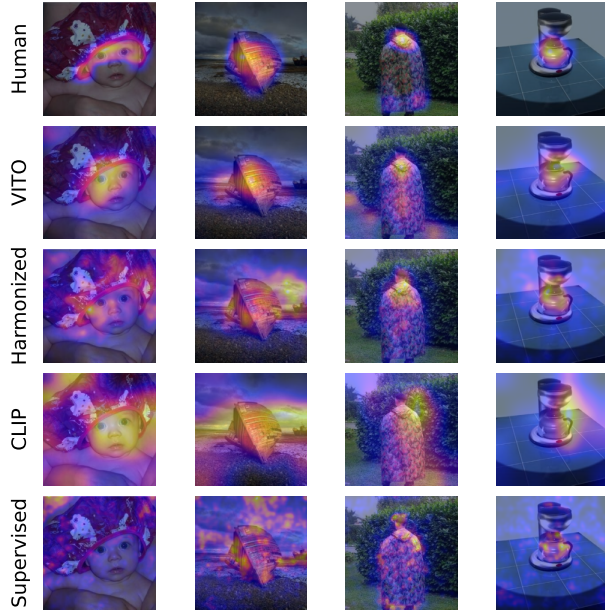


Figure 2.3: VITO attention maps capture human-defined object saliency. Example human saliency maps from the ClickMe dataset (Linsley et al., 2018) and ResNet-50 models. Gradient-based saliency is shown for Supervised and Harmonized (Fel et al., 2022). Attention maps are shown for CLIP and VITO model. We use multi-head attention pool weights for CLIP and average of weights from last 2 attention pooling scales in VITO.

Method	Trained for alignment	Human Alignment
MoCLR	✗	21.4
Supervised	✗	34.4
CLIP	✗	41.8
Harmonized	✓	45.5
VITO	✗	47.7

Table 2.2: Quantitative comparison between gradient-based saliency maps (from Supervised, MoCLR, CLIP-RN50 (attention-map), and Harmonized networks), VITO attention weights, with human saliency maps using a correlation based alignment score from Linsley et al. (2018)

even surpasses the Harmonized model which has been specifically trained for this purpose (Table 2.2).

This result suggests that as opposed to image-based objectives or image-language alignment, human perception of feature importance across the visual scene can be better explained as a consequence of learning what to attend to in the context of self-supervised video-based learning. We hypothesize that these attention masks could underlie the formation of high-level concepts via “semantic binding”, which we investigate in Figure A.1 and Section A.2.1.

Human error consistency in shape-biased tasks. Based on this result relating to object saliency, we hypothesize that VITO may be capturing global object shape features better than traditional deep networks which have been shown to heavily rely on textural cues for classifica-

tion (Geirhos et al., 2018).

Method	accuracy diff. ↓	obs. consistency ↑	ceiled error consistency ↑
<i>Image pretraining</i>			
DINO (Caron et al., 2020)	0.236	0.504	0.291
Supervised	0.215	0.511	0.329
SIN+IN1K (Geirhos et al., 2018)	0.203	0.527	0.330
MoCLR (Tian et al., 2021)	0.190	0.536	0.335
L2-Robust (Madry et al., 2017)	0.178	0.544	0.389
CLIP (Radford et al., 2021)	0.108	0.612	0.482
<i>Video pretraining</i>			
R3M (Nair et al., 2022)	0.392	0.359	0.054
CycleCon (Wu and Wang, 2021)	0.237	0.484	0.258
VINCE (Gordon et al., 2020)	0.210	0.501	0.269
VITO	0.157	0.564	0.422

Table 2.3: Accuracy difference and consistency with human judgments on stimuli that are biased to requiring global-shape understanding (instead of texture) for recognition/discrimination. VITO surpasses all comparable trained models (both image and video pretraining) in all benchmarks, including those that are trained specifically to be robust (SIN+IN1K, and L2-robust). We underperform the CLIP model; however, we note that CLIP is trained with an order of magnitude more images (400M) and explicit human-language supervision.

To evaluate this quantitatively, we used a subset of the dataset proposed in Geirhos et al. (2021) to test both the accuracy and consistency with human judgments of model classifications of stimuli that require shape-cues for effective discrimination (Table 2.3). Specifically, these stimuli are categorized into 4 groups: edge drawings, cue-conflict / stylized (mixing of shapes with contradictory textures through style-transfer), variable low-pass filtering (to remove high-frequency local content), uniform noise (corrupts local texture features). Based on the original methodology proposed in Geirhos et al. (2020b), we report the accuracy difference (from human accuracy), the raw consistency with human judgments, and ceiled error consistency (method from Geirhos et al. (2020b)).

We compare to supervised and MoCLR ResNets, the robust training methods cited earlier, as well as CLIP (Radford et al., 2021). We also compare to various video pre-training methods cited

earlier and another (R3M (Nair et al., 2022)), which has specifically shown to have human- and neurally-aligned representations of dynamic, object-centric scenes (Nayebi et al., 2023b). For all networks, we train linear classifiers on the ImageNet validation set and evaluate on the modified shape-biased stimuli. Compared with all other comparable image pretrained models, VITO achieves stronger robustness to shape-biasing transformations (lower accuracy difference relative to original images). Furthermore, VITO makes predictions more consistent with human judgements in terms of per-trial classification behavior. This is particularly surprising as VITO even outperforms the adversarially-trained robust model without requiring any explicit robust training procedure. Moreover, this improvement is not captured by prior video pretraining efforts (which are in fact far worse than the image pretraining methods). The R3M model, in particular, performs surprisingly poorly. Because the images used to collect the human judgments are modified versions of those from the ImageNet validation set, we hypothesize that this performance can be attributed to the poor transfer of the Ego4D datasets to the diverse classes present in ImageNet (contrarily to VideoNet). Indeed, the R3M model only achieves 13% accuracy on the clean ImageNet validation set (see Table A.3). Finally, we note that VITO does underperform CLIP on this benchmark; however, this comparison is not truly fair as CLIP is trained with explicit human supervision via large-scale image-language mappings. In fact, we believe that our method can be augmented with similar language supervision to improve human alignment even further.

In summary, VITO captures aspects of how humans process shape-information that cannot be captured by other strong visual models. Understanding more about this effect and what aspects of learning from videos lead to this remain interesting opportunities for future work.

2.4.3 ABLATIONS

To understand more about how the components of VITO training contribute to its performance, we vary the different aspects of our paradigm in isolation: our method for data curation (VideoNet), multi-scale attention pooling, and details of the input data (spatial crop size and the

temporal sampling scheme). We explore some ablations in detail on an example benchmark (PASCAL segmentation), but also evaluate ablations across many of the benchmarks used in this work. Finally, we provide a brief exploration demonstrating that our method scales well to larger architectures.

Effect of pretraining data. To demonstrate the effect of the pretraining data distribution on transfer performance, we pretrain a baseline MoCLR model (using 2 views) on a variety of image and video datasets, where we initially treat video datasets as collections of individual frames. We train each model for 300 ImageNet-equivalent epochs, referred to hereafter as “epochs” (i.e. 1 epoch = learning from 1.28M examples, irrespective of the dataset), such that each model benefits from the same amount of computation. Figure 2.4 (left) shows their transfer performance on PASCAL semantic segmentation. As expected, ImageNet pretraining works very well, but pretraining on standard video datasets results in a substantial drop in performance (e.g. -6.8% or -5% mIoU from pretraining on Kinetics700 or AudioSet). This performance gap between video and image pretraining can be attributed to a combination of increased complexity and field-of-view of video frames and domain mismatch between the dataset categories (Figure 2.4, right). Consistent with this, training on JFT (Sun et al., 2017), an uncurated dataset with a heavy-tailed class distribution, also results in a loss in performance. Notably, this is despite the much larger size of JFT (300M images). We find that applying the same baseline pretraining to frames from our curated video dataset performs better than existing large-scale video datasets like Audioset ($+1.6\%$ mIoU), but still underperforms image pretraining on JFT and ImageNet (Figure 2.4). This demonstrates the importance of aligning the distribution of video frames with that of common image datasets. We therefore use VideoNet as our primary pretraining dataset for the rest of the study. In Sec A.2.3 we disentangle the power of our method and dataset by confirming that each independently have strong effects: MoCLR trained on VideoNet, and VITO trained on standard datasets (Audioset or YT8M) also outperform all prior work (including models trained on much larger image datasets like JFT-300M).

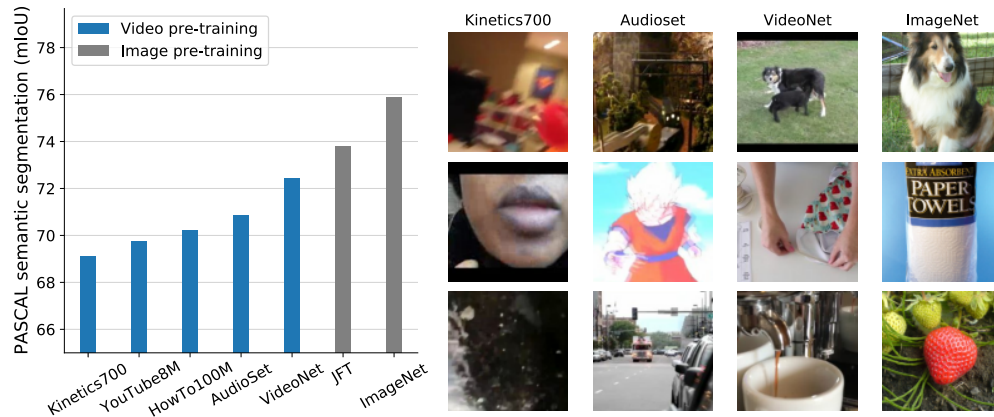


Figure 2.4: VideoNet dataset improves transfer performance to image tasks. Impact of pretraining data’s spatial content on representation quality. Left: transfer performance of models pre-trained on single frames from image datasets (grey bars) or individual videos (blue bars). Right: example frames from different video and image datasets.

Multi-scale attention pooling. We decompose the proposed multi-scale contrastive attention pooling to isolate the effects of multi-scale learning from those of attention pooling (Figure A.2, right). While we find only modest gains from adding attention pooling to a single-scale version of the model (+0.2% mIoU), we find that the 2-scale model (without attention pooling) improves over the single scale model more robustly (+0.6% mIoU). Interestingly, we find that the combination of the 2-scale model with attention pooling has a synergistic effect (+1% mIoU over the single-scale attention model), highlighting the importance of handling the variability in scales present in natural videos.

Spatial and temporal augmentation parameters. We first validate in Figure A.2 (left) our hypothesis that increasing the minimum crop-scale in the random-resized crop operation during training leads to models that generalize better to fine-grained tasks like semantic segmentation. Specifically, we find that a minimum crop scale of 0.4 (as opposed to the traditional 0.08) results in the best transfer performance (+1.7% mIoU). Note that this conclusion differs slightly from that of Feichtenhofer et al. (2021) who find more aggressive cropping to be beneficial for action recognition.

Next, to study the effect of different temporal sampling schemes, for each training example,

Pretraining	Dataset	PASCAL (mIoU)	UCF101 (top-1)	IN-A (top-1)	IN-Vid (pm0/pm10)	Human error con- sistency
MoCLR	VideoNet	72.8	83.0	2.3	55.5/40.5	0.224
VITO 1scale (w/o attn)	VideoNet	75.2	85.5	3.9	67.3/55.5	0.359
VITO 1scale (attn)	VideoNet	75.4	85.7	3.5	65.6/52.9	0.368
VITO 2scale (w/o attn)	VideoNet	75.8	86.2	4.2	67.4/54.9	0.390
VITO (T=0)	VideoNet	74.8	83.2	3.9	63.9/49.5	0.323
VITO	AudioSet	73.8	84.8	3.4	55.7/42.4	0.401
VITO	VideoNet	76.3	87.4	5.4	70.6/57.2	0.422

Table 2.4: All components of VITO pretraining matter for downstream performance. Summary of ablation models on key evaluations covering image understanding, video understanding, and human alignment on ood object recognition. In summary, it is clear that all components (pretraining data, temporal deformations, and the multi-scale attention pooling) are required for best performance across all tasks.

we sample 3 views using marginal sampling of each frame from the video clip of length $T = 2.56$ seconds. This length determines the distribution of time differences between any pair of frames, and thus the time-scale over which the contrastive model learns invariances. We verify our choice by varying the total length of clips. While going to longer time-scales $T = 3.2s$ does not hurt performance much, we find a significant improvement over using shorter clips (e.g. $T = 1.28s$, +1.0% mIoU; Figure A.2, center). This suggests that invariance to the rich temporal deformations present in video clips is indeed a beneficial criterion for learning fine-grained spatial representations.

Comprehensive ablation summary. In Table 2.4, we extend the above ablation studies to a more comprehensive benchmark set. In addition to the PASCAL segmentation task, we evaluate the key ablated models on video understanding (UCF101), OOD recognition (IN-A/IN-Vid) and human alignment on the shape-bias tasks specified in Sec 2.4.2. We confirm that all of the major methodological components (VideoNet dataset, multi-scale attention pooling, and using temporal deformations) work in concert, and are required for best performance across all tasks. Notably, we see a particularly striking dichotomy between models trained with and without temporal deformations on human error-consistency. Specifically, models trained without temporal defor-

mations (MoCLR and VITO (T=0)) have a significant drop in human error-consistency relative to all other models trained with temporal deformations, highlighting the importance of learning these kinds of invariances.

Scaling model architectures. We briefly demonstrate that VITO scales to more recent larger architectures. Specifically, we show preliminary results that VITO achieves highly competitive performance on four scene understanding benchmarks using the Swin-S transformer architecture (Liu et al., 2021). In Sec. A.2.4, we show that performance improves dramatically over the ResNet-50 architecture and is competitive with a strong, specialized ImageNet pretrained baseline for fine-grained scene understanding (DetCon (Hénaff et al., 2021b)).

2.5 RELATED WORK

Learning general visual representations from videos. Many prior works have considered self-supervised representation learning for capturing spatio-temporal invariances, beginning with methods that leveraged temporal coherence, optical flow, and object tracking (Agrawal et al., 2015; Goroshin et al., 2015; Hurri and Hyvärinen, 2003; Kulkarni et al., 2019; Misra et al., 2016; Pathak et al., 2017; Srivastava et al., 2015; Wang and Gupta, 2015; Wiskott and Sejnowski, 2002). More recently, many successful approaches have leveraged contrastive learning, masked autoencoding, and other self-supervised pretext tasks to learn strong video representations (Dave et al., 2022; Dorkenwald et al., 2022; Feichtenhofer et al., 2022; 2021; Qian et al., 2021; Recasens et al., 2021; Sermanet et al., 2018). However, most of these methods employ specialized video architectures and only transfer to video-based tasks such as action recognition and motion segmentation.

Yet natural motion-induced deformations are powerful learning signals that should allow for learning better *image* representations as well. Indeed, human infants can form complex understanding of objects and shape within months, specifically driven by their observations of how they move (Spelke, 1990; Spelke and Kinzler, 2007). Given this inspiration, some works have

demonstrated that self-supervised contrastive learning in videos can lead to aspects of efficient human learning and robust recognition (Kong and Norcia, 2021; Orhan et al., 2020; Zhuang et al., 2022). In computer vision, cycle-consistency (Bian et al., 2022; Jabri et al., 2020) and optical flow (Sharma et al., 2022; Xiong et al., 2021) have been used to learn correspondences between temporally ordered image patches. The most similar works to ours utilize video-based contrastive learning (Gordon et al., 2020; Wu and Wang, 2021; Xu and Wang, 2021) to improve performance on temporal understanding tasks, however they do so at the cost of spatial scene understanding.

Robustness to distribution shifts. As standard benchmarks have been progressively saturated (Beyer et al., 2020), the community has turned to measuring robustness to adversarial attacks (Carlini et al., 2019), corruptions (Hendrycks and Dietterich, 2019), and out-of-distribution datasets (Hendrycks et al., 2021b; Kar et al., 2022; Shankar et al., 2021; Taori et al., 2020). We focus on a subset of these benchmarks that are as “natural” as possible, to evaluate generalization with respect to shifts that are most likely to appear in the real world. While there have been many efforts to specifically encourage regularize models for these kinds of robustness (Geirhos et al., 2018; Madry et al., 2017; Rusak et al., 2020; Xie et al., 2020), we instead investigate the complementary question of whether image and video pretraining differ in this respect.

Human-aligned representations. Most recent progress in achieving more behaviorally-matched representations has been by scaling existing approaches. Indeed, recent examples (Dehghani et al., 2023; Oquab et al., 2023; Radford et al., 2021) show that as data and model sizes grow by orders of magnitude, generality and robustness of representations tend to emerge. Moreover some aspects of human perception such as an increased shape-bias and consistency with human perceptual behavior (Dehghani et al., 2023; Geirhos et al., 2021) can be captured reasonably well by certain large models. However this scaling property tends to be brittle, with some large-scale models displaying significantly worse consistency with human perception (Dehghani et al., 2023; Kumar et al., 2022). Additionally, more recent work on alignment has found that scaling and archi-

ecture are not as important for alignment on specific benchmarks, in comparison to the training dataset and objective function (Muttenthaler et al., 2022). Therefore, while scaling may continue to lead to task-performance gains, it is unclear whether only scaling image-based pretraining will close the gap with general human behavior. We therefore explore the complementary and potentially synergistic question of whether video pretraining can improve the task-generality, robustness, and behavioral similarity of learned visual representations.

2.6 DISCUSSION

Summary. We propose VITO, a simple method for distilling videos into visual representations. The key features of our method include improved dataset curation, adapting augmentation pipelines to appropriately handle video frames, and using attention-guided contrastive learning. With these components, VITO surpasses both prior video pretraining in spatial understanding, and image pretraining on temporal understanding and robustness. In addition to these hallmarks of human perception, VITO explicitly aligns with aspects of human saliency and image recognition behavior that are not captured by other high-performance representation learning techniques. In sum, despite the many successes in video representation learning, our results suggest that there is a great untapped potential in video pretraining as a paradigm for learning general, human-aligned visual representations.

Limitations and Future Work. We believe this work can be a foundation for future video pretraining efforts, as our approach is powerful, yet simple and extensible. However, we recognize that this demonstration is mostly limited to a single contrastive learning framework and ResNet-50 architecture. We leave for future work, the validation and exploration of similar analyses with larger models and other self-supervised training objectives (such as MAEs and self-distillations methods like DINO). Additionally, while we have shown the benefits of a surprisingly simple attention module for learning correspondences in video data, there are more powerful attentional

architectures we can leverage along with scaling dataset size as in [Oquab et al. \(2023\)](#). We have started these experiments with our exploration of Swin transformer architectures.

2.7 FROM BEHAVIOR TO NEURAL ALIGNMENT?

This work takes steps towards learning more human-like visual representations by leveraging a more naturalistic video pretraining paradigm. We provide extensive evaluations demonstrating that VITO, in comparison to standard image pretrained networks, displays more of the hallmarks of human visual capabilities (specifically task-generalty and robustness to out-of-distribution shifts), in addition to being more explicitly aligned with aspects of human perception. It is natural to then hypothesize that the internal representations in our model might also be better aligned with neural representations along the ventral stream. To assess this, we use the BrainScore benchmark (details in ([Schrimpf et al., 2018](#))), to assess how well a linear weighting of model neurons (fit on a subset of images) predicts responses to biological neurons on held-out images. Surprisingly, we find that both VITO and the baseline ImageNet pretrained ResNet-50 model capture approximately the same amount of variance in all cortical areas. In particular, across 4 different IT cortex datasets, both models explain on average approximately 48 % of the neural response variance.

How can our model provide a significantly different (and better) model of human behavior, yet not produce better models of cortical responses? We believe there are three potential strong hypotheses:

1. End-to-end, objective driven learning can constrain behavior or task performance of a deep network, but does not provide strong enough constraints on intermediate layers to learn biologically-aligned representations.
2. Current neural benchmarks are not sufficiently able to distinguish differences between model representations.

3. The recorded ventral stream responses are not close enough to the ‘behavioral readout’ (requiring many more layers of unknown transformations) and thus the two metrics are not correlated.

The last hypothesis is hard to evaluate; however, in the following two chapters, we take seriously the first hypothesis and evaluate whether layerwise constrained models do indeed provide better accounts of neural responses. We also very briefly explore the second hypothesis in preliminary experiments in [Chapter 5](#).

3 | SELF-SUPERVISED LEARNING OF A BIOLOGICALLY-INSPIRED VISUAL TEXTURE MODEL

3.1 OVERVIEW

Versions of the work in this chapter were presented at Computational and Systems Neuroscience (2020), and published in preprint form (Parthasarathy and Simoncelli, 2020).

As described in Sec. 2.7, the goal of the following two chapters is to explore the following hypothesis: that layerwise constrained network representations better align with primate neural representations. This chapter provides a practical, small step in this direction. Specifically, given a reasonable hand-crafted model for cortical area V1, we ask whether a single layer transformation can be learned that produces neurons with complex feature selectivity resembling selectivities found in area V2.

We develop a model for representing visual texture in a low-dimensional feature space, along with a novel self-supervised learning objective that is used to train it on an unlabeled database of texture images. Inspired by the architecture of primate visual cortex, the model uses a first stage of oriented linear filters (corresponding to cortical area V1), consisting of both rectified units (simple cells) and pooled phase-invariant units (complex cells). These responses are processed by

a second stage (analogous to cortical area V2) consisting of convolutional filters followed by half-wave rectification and pooling to generate V2 ‘complex cell’ responses. The second stage filters are trained on a set of unlabeled homogeneous texture images, using a novel contrastive objective that maximizes the distance between the distribution of V2 responses to individual images and the distribution of responses across all images. When evaluated on texture classification, the trained model achieves substantially greater data-efficiency than a variety of deep hierarchical model architectures. Moreover, we show that the learned model exhibits stronger texture category representational similarity to responses of neural populations recorded in primate V2 than an end-to-end supervised pre-trained deep CNN.

3.2 INTRODUCTION

Most images contain regions of "visual texture" - comprised of repeated elements, subject to some randomization in their location, size, color, orientation, etc. Humans are adept at recognizing and differentiating materials and objects based on their texture appearance, as well as using systematic variation in texture properties to recover surface shape and depth. At the same time, we are insensitive to the details of any particular texture example - to first approximation, different instances of any given class of texture are perceived as the same, as if they were "cut from the same cloth". This invariance is usually captured through the use of statistical models. Bela Julesz initiated the endeavor to build a statistical characterization of texture, hypothesizing that a texture could be modeled using n -th order joint co-occurrence statistics of image pixels (Julesz, 1962). Subsequent models can be partitioned into three broad categories: 1) orderless pooling of handcrafted raw-pixel features such as local binary patterns (Liu et al., 2016; Ojala et al., 2002), 2) local statistical models using Markov random fields (Chellappa and Chatterjee, 1985; Cross and Jain, 1983; Derin and Elliott, 1987; Portilla and Simoncelli, 2000), and 3) statistical characterization of fixed convolutional decompositions (i.e. wavelets, Gabor filters, multi-scale pyramids)

(Bergen and Adelson, 1986; Bovik et al., 1990; Bruna and Mallat, 2013; Heeger and Bergen, 1995; Portilla and Simoncelli, 2000; Sifre and Mallat, 2013). More recent models are based on statistics of nonlinear features extracted from pre-trained deep convolutional neural networks (CNN's) (Cimpoi et al., 2015; Gatys et al., 2015; Song et al., 2017; Ulyanov et al., 2017; Xue et al., 2017). A comprehensive review of these is available in Liu et al. (2019).

The fixed-filter methods are generally chosen to capture features considered fundamental for early visual processing, such as local orientation and scale. Similar filters can be learned using methods such as sparse coding (Olshausen and Field, 1996) or independent components analysis (Bell and Sejnowski, 1997). On the other hand, deep learned methods provide great benefits in terms of extracting relevant complex features that are not so easily specified or even described.

However, recent work in understanding the representation of texture in the primate brain has shown that texture selectivity arises in Area V2 of visual cortex (Freeman et al., 2013; Ziemba et al., 2016), which receives primary input from Area V1. Therefore, it seems that the brain can achieve selectivity for texture in far fewer stages than are commonly used in the deep CNNs. Motivated by this fact, we construct a simple, hybrid texture model that blends the benefits of the aforementioned fixed-filter image decompositions with the power of learned representations. There are two main contributions of our work. First, the model represents textures in a relatively low-dimensional feature space (in contrast to the extremely high-dimensional representations found in CNN models). We propose that this low-dimensional representation can be used to perform texture family discrimination with small amounts of training data when it is coupled with an interpretable non-linear decoder. Moreover, we show that a novel self-supervised learning objective plays an important role in achieving this result. Finally, while pre-trained deep CNNs can achieve better texture classification accuracy, we show that our learned model exhibits much stronger representational similarity to texture responses of real neural populations recorded in primate V2.

3.3 METHODS

3.3.1 V2NET MODEL ARCHITECTURE

It is well-known that the primary inputs to V2 are feed-forward outputs from area V1 (Girard and Bullier, 1989; Schiller and Malpeli, 1977; Sincich and Horton, 2005). Inspired by these physiological results, we propose a computational texture model as a two-stage network that functionally mimics the processing in these two early visual areas.

The V1 stage is implemented using a set of fixed convolutional basis filters that serve as a functional model for V1 receptive fields (Ringach, 2002). The filters are localized in orientation and scale, specifically utilizing a complex-steerable derivative basis (Jacobsen et al., 2016; Simoncelli and Freeman, 1995). We choose a specific set of 4 orientations and 5 scales (octave-spaced) with two phases (even and odd), for a total of 40 filters. The full set of V1 responses are a combination of both half-wave rectified simple cells and L_2 -pooled (square root of the sum of squares) complex cells, yielding a total of 60 feature maps.

The V1 responses provide input to a V2 stage that consists of a set of D learned convolutional filters. In the macaque, V1 and V2 are known to have similar cortical surface area and output fibers (Wallisch and Movshon, 2008), so in our experiments we set $D = 60$ to match the dimensionality of the V1 and V2 stages of our model. The convolutional layer is then followed by half-wave rectification, spatial L_2 -pooling and downsampling to produce V2 ‘complex cell’ responses (Fig. 3.1). Unlike standard max-pooling, L_2 -pooling is used in both stages of our model because it is more effective at capturing local energy of responses without introducing aliasing artifacts (Bruna and Mallat, 2013; Hénaff and Simoncelli, 2015).

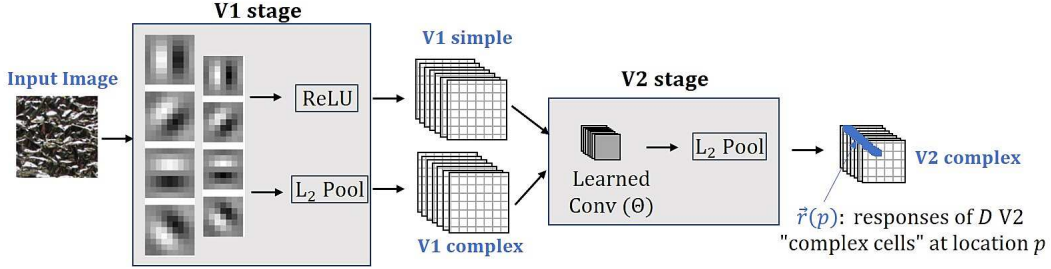


Figure 3.1: Biologically-inspired texture model architecture. The V1 stage is built using a 5-scale 4-orientation complex steerable pyramid (Portilla and Simoncelli, 2000), followed by two nonlinearities to generate simple and complex cell responses. The latter uses specialized L_2 pooling. The V2 stage consists of convolution with D filters followed by spatial L_2 pooling.

3.3.2 LEARNING OBJECTIVE

Consider the model in Fig. 3.1 as a function $f(T; \Theta)$ that takes as input a texture image T , and computes responses based on parameter vector $\Theta = [\Theta_1, \dots, \Theta_D]$, which contains the D V2 filters¹. Given a dataset of N texture images (T_n) and their corresponding model responses $\mathbf{r}_n = f(T_n; \Theta)$, we seek an objective function, $L(\cdot)$, for optimizing the V2 filter weights: $\Theta_{opt} = \arg \min_{\Theta} L(\{f(T_n; \Theta)\})$. We assume a curated image dataset with two properties that underlie the formulation of the objective: 1) individual images contain a single texture type (homogeneous across their spatial extent) and 2) the N images in the dataset represent a diverse set of texture types.

Our learning objective is motivated by the experimental observations in Ziemba et al. (2016) suggesting that V2 represents textures such that responses within texture families (i.e. classes) are largely invariant to variability within the texture families- the responses are less variable within texture families than across families. To learn such a representation, one could simply utilize an objective function that reduces variability of responses to each family while maintaining variability across all families. This can usually be achieved by supervised methods that optimize responses to predict the class identity for an image. However, we desire an objective that has

¹each Θ_d is a $60 \times 7 \times 7$ set of weights, as each V2 filter operates over the full set of 60 V1 channels

no supervisory knowledge of which images correspond to which texture families. As a result, we propose a contrastive objective that seeks to 1) Minimize the variability of model responses ($\mathbf{r}_n(p)$) across locations p within each individual texture image and 2) Maximize variability of these responses across neighborhoods sampled from the entire set of N images. Therefore, rather than using labels to enforce grouping of similar texture families, we utilize the natural spatial homogeneity of *individual texture images* as a form of ‘self-supervision’.

To formulate this mathematically, we first model the distribution of V2 responses over positions p within each image ($\mathbf{r}_n(p) \in \mathbb{R}^D$) as multivariate Gaussian, parameterized by the sample mean and covariance: $\boldsymbol{\mu}_n \in \mathbb{R}^D$ and $C_n \in \mathbb{R}^{D \times D}$. The global distribution of responses across all images is then a Gaussian mixture with mean and covariance: $\boldsymbol{\mu}_g = \frac{1}{N} \sum_{n=1}^N \boldsymbol{\mu}_n$; $C_g = \frac{1}{N} \sum_{n=1}^N C_n + (\boldsymbol{\mu}_n - \boldsymbol{\mu}_g)(\boldsymbol{\mu}_n - \boldsymbol{\mu}_g)^\top$. Under this parameterization, the two goals for the objective can be achieved by maximizing the ‘discriminability’ between the individual and global response distributions based on their covariances. A suitable measure of discriminability must capture the differences in both size (total variance) and shape of the distributions.

There has been extensive work on developing measures that approximate the discriminability between Gaussian distributions based on their mean and/or covariance statistics (Abou-Moustafa et al., 2010; Bhattacharyya, 1946; Bures, 1969; De la Torre and Kanade, 2005; Dryden et al., 2009; Huang et al., 2015; Nenadic, 2007). In order to choose a distance for this problem we define a set of criteria the distance must satisfy. First, the distance must be *scale invariant*: global rescaling of the image data should not change the value of the distance measure, which is meant to capture *relative* differences in variability. This is especially important for an objective function, as the responses can be arbitrarily scaled by the learned weights. Second, for maximization it is preferable that a distance *have an upper bound* as this can stabilize optimization and avoid degenerate solutions where the distance can take on extremely large, unbounded values. Third, for any given texture image, not all of the V2 dimensions may be important (i.e. the covariance is low-rank), so the distance must be stable in this regime.

Given these criteria, it is clear that many of the statistical distances and manifold-based log-Euclidean distances are problematic because the log transformation is unstable when covariances are low-rank. The work of Faraki et al. (2016) has shown that regularizing the log-Euclidean approach with standard covariance shrinkage can lead to large errors, and we have observed this in our experiments as well. A novel attempt to resolve this issue was proposed using a Riemannian optimization method (Faraki et al., 2016), but this method only works for fixed low-rank matrices. As a result, we construct our distance on the form $\|C_1^{1/2} - C_2^{1/2}\|_F$ corresponding to the Bures metric ² (Bures, 1969; Muzellec and Cuturi, 2018). We modify this to make it bounded and scale-invariant, arriving at a novel measure of distance between the global response covariance and that of image T_n :

$$d_n = \frac{\|C_g^{1/2} - C_n^{1/2}\|_F}{\|C_g^{1/2}\|_F}, \quad (3.1)$$

where $(\cdot)^{1/2}$ indicates matrix square-root and $\|\cdot\|_F$ is the Frobenius norm. This may be seen as a normalized variant of the log-Euclidean distances (Huang et al., 2015), in which replacement of $\log(\cdot)$ by $(\cdot)^{1/2}$ retains the primary benefit of the log-Euclidean framework (transforming the covariance eigenvalues with a compressive nonlinearity), while remaining stable and well-defined in low-rank conditions.

After calculating the distance in Eqn. (3.1) for each individual image, we then combine over all images to obtain a single scalar objective. To force all distances to be as large as possible, we maximize the minimum of these distances. For stable optimization, we use a soft-minimum function, which yields our variability-based objective:

$$\mathbf{L}_{\text{var}} = \text{softmin}(d_1, d_2, \dots, d_N) = \frac{\sum_n d_n e^{-d_n}}{\sum_n e^{-d_n}}. \quad (3.2)$$

To allow for robust estimation of the covariance, we make a diagonal approximation where

²Equivalent to the covariance term of the 2-Wasserstein distance between multivariate Gaussian distributions in the special case when the two covariance matrices commute

C_n and C_g are each taken to be diagonal. Therefore, the matrix square-roots can be implemented as element-wise square roots of the individual response variances along the diagonal and the Frobenius norm becomes the standard vector L_2 norm. However, because a diagonal approximation can be poor if the covariances have strong co-variability, we use an additional orthogonal regularization term to encourage orthogonalization of the V2 filters (Bansal et al., 2018):

$$\mathbf{L}_{\text{orth}} = || \Theta \Theta^T - I ||_F. \quad (3.3)$$

Minimizing this loss forces the responses of each channel to be roughly independent and thus more amenable to the diagonal approximation. The final objective is a weighted combination of the two terms:

$$\max_{\Theta} [\mathbf{L}_{\text{var}} - \lambda \mathbf{L}_{\text{orth}}]. \quad (3.4)$$

3.3.3 EVALUATION METHODOLOGY

After training the model with the self-supervised objective in Eqn. (3.4), we use a separate labeled dataset to train and test a texture family classifier. We first compute the spatially global-average pooled (GAP) responses for each image in the new dataset, such that each image T_n is represented by a single D -dimensional vector, μ_n . We again make a Gaussian assumption on the distribution of these mean response vectors for each texture family and fit and test a quadratic discriminant classifier (QDA) to predict the texture class labels. This process is shown in Fig. 3.2 (a). Although the choice of a QDA classifier is not common, state-of-the-art texture classification methods generally use some form of quadratic feature encoding (Fisher vectors, bilinear layers, etc.) before applying a trained linear classifier (i.e. SVM) (Cimpoi et al., 2015; Lin et al., 2015; Song et al., 2017). Rather than compute all pairwise products, which can be prohibitively expensive in terms of number of parameters, we use mean pooling to produce a low-dimensional representation, followed by a bilinear readout. In our context, a quadratic discriminant is the optimal

bilinear method for discrimination under the Gaussian assumption.

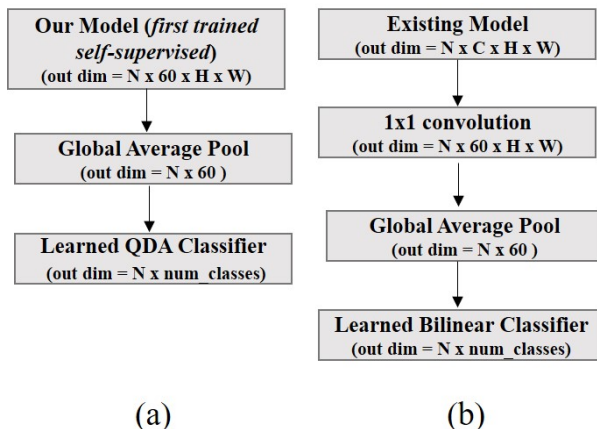


Figure 3.2: (a) Evaluation method for our self-supervised model. (b) Evaluation method for the supervised networks

One issue with QDA classification is that it requires the estimation of class-covariance matrices. These matrices can only be reasonably estimated when the number of samples per class is much larger than the dimensionality of the features, so QDA is only amenable to low-dimensional feature representations. In our experiments, we compare our model to supervised methods that use popular network architectures as the base feature extractor. However, most of these networks produce very high-dimensional output feature spaces that are usually evaluated with linear classifiers. As a result, we devise an evaluation protocol for these methods such that the dimensionality of the feature representation *and* the expressivity of the classifiers is matched to that of our model (Fig. 3.2(b)). Specifically, we first reduce the dimensionality of the feature representation to match that of our V2Net model ($D = 60$) using a trained 1x1 convolutional layer (as is common in the literature (Howard et al., 2017; Xue et al., 2018)). This is followed by the same classification procedure as ours: spatial GAP followed by a bilinear classifier. However, because QDA cannot be implemented for supervised, end-to-end trained networks, we use a parameterizable bilinear layer of the form: $x^T Ax + Bx + c$. The model parameters, 1x1 conv layer, and bilinear are all trained end-to-end, in contrast with our model which is first trained separately with the self-supervised objective.

One might ask if the dimensionality reduction of the existing network architectures is too restrictive and if our comparisons will be biased because of this. In fact, a similar methodology has shown minimal loss in performance for texture retrieval with PCA reduction down to 64 dimensions (Valente et al., 2019). Therefore, it is unlikely that we are biasing our comparisons by stifling the capacity of the network. Moreover, the 1x1 convolution approach is arguably more effective than PCA because it allows this dimensionality reduction to be optimized in the context of the classification task. Nevertheless, we additionally verified that results for all tested networks were close to those achieved using a linear classifier on the full-dimensional feature space.

The specific models we compare to are chosen to span a diverse set of methods from the literature:

ScatNet: We implement the front-end two-stage scattering model as described in Bruna and Mallat (2013); Sifre and Mallat (2013) that has 5 scales and 8 angles. The scattering model is then fixed and the 1x1 convolution layer and the bilinear classifier are learned. The number of channels before dimensionality reduction is 681.

DAWN(16-init): Recent work has performed a similar experiment using a hybrid deep adaptive wavelet network that is found to be more data-efficient than previous methods (Rodriguez et al., 2020). We implemented the same model and regularization, with 16 initial convolutional layers, followed by the multi-scale representation. The number of channels before dimensionality reduction is 256.

ResNet-18: Based on recent success as a feature extractor for texture recognition (Xue et al., 2018) we also included an 18-layer ResNet model. We extract features from the *layer4* level of the network, as these have been deemed as the most powerful features for texture classification in previous work (Xue et al., 2018; Zhai et al., 2019). The number of channels before dimensionality reduction is 512.

VGG-16: VGG networks and their variants have been the most common network architectures used for feature extraction in the literature. The work of Cimpoi et al. (2014; 2015) demon-

strated that a Fisher vector decoder, and even linear classification from pooled features of the last convolutional layer, can be effective for texture classification. Based on this work, we used features from the *conv5* layer of a VGG-16 network. The number of channels before dimensionality reduction layer is 512.

3.4 RELATED WORK

Model Architecture. Many fixed-filter, hierarchical image decompositions have been used in the construction of texture representations that are similar to our V1 stage (Bruna and Mallat, 2013; Simoncelli and Freeman, 1995). However, we note that our V1 responses include both rectified simple cells *and* L_2 -pooled complex cells. This formulation is motivated by physiological experiments studying the projections of V1 to V2 neurons (El-Shamayleh et al., 2013), and represents a departure from the classical view of hierarchical visual modeling that assumes only pooled responses are transmitted to the downstream layers (Bruna and Mallat, 2013; Fukushima, 1980; Riesenhuber and Poggio, 1999).

Recent deep learning approaches to representing texture have been heavily optimized and hand-crafted for specific tasks such as texture classification (Cimpoi et al., 2015; Xue et al., 2018), synthesis (Gatys et al., 2015; Ulyanov et al., 2017), and retrieval (Qian et al., 2017; Valente et al., 2019). However, there are a few common themes in these methods that we highlight for their relevance to our model and the models we use for comparison. First, all SoA methods, regardless of task, rely on extraction of features or statistics from deep CNNs trained for object recognition, primarily the VGG and ResNet architectures (He et al., 2016; Simonyan and Zisserman, 2015). With the exception of a few studies (Fujieda et al., 2018; Rodriguez et al., 2020), performing texture classification with networks trained from scratch has been relatively understudied. Second, it has been consistently shown that "orderless" pooling of the features before classification layers results in a far better texture representation. Simple global average pooling (GAP) has been shown to

be quite effective (Dumoulin et al., 2016; Valente et al., 2019; Xue et al., 2018; Zhang et al., 2020b) as well as methods that pool based on 2nd-order statistics (Cimpoi et al., 2015; Gatys et al., 2015; Lin et al., 2015).

Objective functions. In the context of texture classification, current human-labeled homogeneous texture databases are few and small, so most deep learning methods transfer features from networks trained with full supervision on an alternative task (typically, object recognition). Some authors have developed limited unsupervised methods based on vector quantization (Greenspan et al., 1991; Raghu et al., 1997), and non-negative matrix factorization (Qin et al., 2008). Nevertheless, in concert with CNN models, we believe ours is the first competitive self-supervised learning objective for this problem.

Conceptually, our objective is inspired by principles of *contrastive learning* that have recently seen much success in competing with more traditional supervised methods (Hénaff et al., 2014; 2019b; Oord et al., 2018; Wu et al., 2018; Zhuang et al., 2019). However, the specific construction of our learning objective differs substantially from these methods as it relies on a diagonal Gaussian parameterization of sample distributions that provides many computational benefits such as easy generalization to incremental learning where the sufficient statistics are updated online without use of large in-memory batches.

3.5 RESULTS

3.5.1 DATA-EFFICIENT TEXTURE CLASSIFICATION

We hypothesize that our objective function enables the learning of a more powerful texture representation from small data. To test this, we use an experimental paradigm similar to Hénaff et al. (2019b). We train and test all models on varying amounts of data from a texture dataset. We use a modified version of the challenging KTH-TIPS2-b dataset (Caputo et al., 2005) for both train-

ing and evaluation. The original dataset includes 11 families of textured materials photographed with different viewpoints, illumination levels, and scales. The total dataset is relatively small (4752 images), so we augment it with 3 rotated versions of each image (90, 180, and 270 degrees) to obtain a total of 19008 samples. As texture representations should be invariant to rotation, this is a sensible augmentation that increases the difficulty of the task. We use the original 4 splits of the KTH-TIPS2-b data (training on 3 splits and testing on the 4th). For all experiments we use a fixed validation set of 3256 images and each test set contained 4752 images. We then conduct three experiments varying the amount of training data (reducing evenly the number of images per texture family). We report results for the full training data (1000 images per family), 50 percent training (500 images per family), and 25 percent (250 images per family).

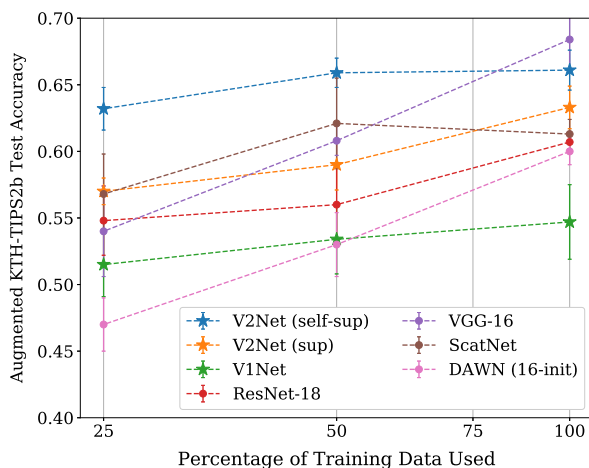


Figure 3.3: Self-supervised V2Net classifies textures most efficiently. We plot the mean and standard error across the 4 train/test splits as a function of the percentage of training data used for all models.

All models (ours and those listed in Sec. 3.3.3) are trained from scratch without any pre-trained information. For the supervised networks we vary learning rates (from 0.0001 to 0.01) and batch sizes from (50 to 200) and choose the best model for each train/test split. For our model (V2Net (self-sup)), the objective function relies on calculating the global mean and variances over the entire dataset. However, because our training is done through stochastic gradient descent, we approximated these global statistics by the global statistics over batches of 275 images. We

choose the batch size heuristically so that individual batch statistics do not deviate significantly from the statistics over the whole dataset. Interestingly, the batch size does not need to be as large as is necessary in most other contrastive learning approaches (Chen et al., 2020b; Hénaff et al., 2019b). We use a learning rate of 0.001 and additionally included a BatchNorm layer at the output of the network to stabilize the global statistics across batches.

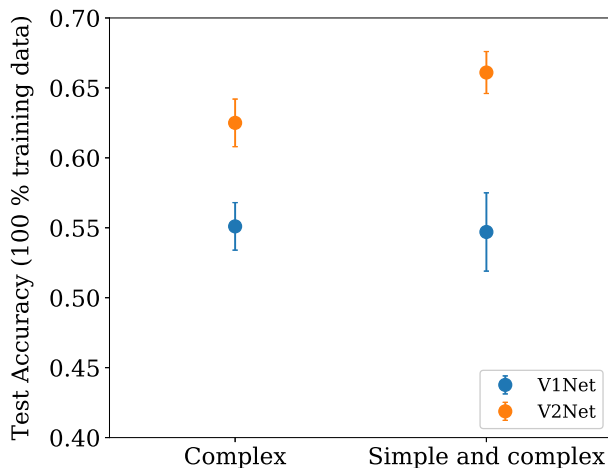


Figure 3.4: V2Net requires both simple and complex cells for optimal performance.. We ablate the V1 simple cell contribution in both the V1Net and V2Net models. We see that the V2Net optimal texture recognition performance requires V1 simple cell inputs.

The results for the 3 training experiments are shown in Fig. 3.3. We report the mean and standard error for the 4 train/test splits within each experiment. First, we can see that just using the fixed V1 stage (V1Net) followed by QDA provides a reasonable baseline. This model has marginal performance difference across differing amounts of training data, which can be solely attributed to the estimation error of the class covariances when training the QDA classifier. Second, we find that the two-stage V2 model performs similarly to the VGG-16 network with full training data, but significantly outperforms all networks when using 50% or 25% of the training data, indicating much greater data-efficiency. To better understand the impact of our objective function, we also report results for a network with the same architecture as V2Net, but trained with a supervised cross-entropy loss (V2Net (sup))³. As seen in Fig. 3.3, this network performs

³We use the same bilinear classifier model as was used for the other networks

comparably to the other supervised networks but still seems to overfit in the small-data regimes. This suggests that even small networks can overfit with small amounts of training data, implying that it is the design of our objective function that allows our network to remain data-efficient in these cases.

To assess the impact of the inclusion of V1 simple cells in our network, we next compare the V2Net classification accuracy to a model trained with V1 simple cells removed. The results are shown in Fig. 3.4. The performance of both V1 models is roughly the same, and in both cases the V2 model improves on the V1 model. However, the gap between the V1 and V2 performance is noticeably larger when the V1 layer contains both simple and complex cells. This result suggests that a more effective V2 representation can be learned when the inputs come from both simple and complex cells.

3.5.2 TRANSFER LEARNING

To verify the generalization of our learning objective, we collected an unlabeled dataset of texture photographs. Original images were manually cropped to be globally homogeneous (by eye) over their entire spatial extent. The scale, viewpoint etc. were not controlled in any particular way, although most textures are on approximately front-parallel surfaces. The types of texture in the dataset span a wide range (including leaves, grass, wood bark, brick, ceramic tile mosaics, etc) that is far more diverse than the KTH-TIPS2-b dataset. We train our model on 11000 of these images and re-evaluate the performance on the four KTH train/test splits by retraining the QDA classifier. Performance of this pre-trained model slightly improves on the performance of the models trained from scratch (average gain of 1.4 % mean accuracy across the three experiments) and displays the same level of robustness to the reduction of training data. This demonstrates that our results are not specific to the training dataset and that our learning objective in fact generalizes across texture datasets with very different distributions of images. We additionally compared the performance of our pre-trained (but still self-supervised) network against the ResNet-18 and

VGG-16 architectures pre-trained on ImageNet classification. The results of this experiment are given in Sec. B.1. Our network does not achieve the performance of these pre-trained networks, but the performance gap (5-10%) is surprisingly small given that our model is pre-trained without supervision, using two orders of magnitude fewer images (11k vs. 1M).

3.5.3 SELECTIVITY FOR NATURAL TEXTURE VS. SPECTRALLY-SHAPED NOISE

Physiological results in Freeman et al. (2013); Ziemba et al. (2016) suggest that texture selectivity in the brain not only manifests as an ability to separate texture families, but also can also be used to distinguish natural textures from their phase-scrambled counterparts. We construct a test along these lines to gain a deeper understanding of our learned model and its selectivities. We retrain our V2Net model using phase-scrambled versions of the images from our unlabeled texture dataset from Sec. 3.5.2. By training on phase-scrambled images, the model no longer has access to the natural statistics that define textures beyond their spectral power. As a result, if our model is truly capturing higher-order texture statistics, its performance on natural images will drop significantly when trained on the phase-scrambled images. In fact, we find that the average test accuracy of the model trained on phase-scrambled images (V2Net (PS)) is 51.5% vs. 67.4% for the model trained on natural images (V2Net (Natural)). Upon further inspection, there are certain texture classes that have high accuracy for the V2Net (PS) model, indicating that these families are readily distinguished using spectral power statistics. We verify that this is also true perceptually: phase-scrambled versions of these classes are visually similar to the original images. However, the classes where there is a large deviation between V2Net (Natural) and V2Net (PS) are those where the phase-scrambled images carry little information about the original texture. For more details, see Sec. B.2.

3.5.4 TEXTURE REPRESENTATIONAL SIMILARITY

Having established that our learned texture model reproduces qualitative texture selectivities, we next explore the relationship with the physiology by comparing the representational similarity between our model and recorded responses of V2 neurons to texture images. We use the dataset described in [Freeman et al. \(2013\)](#); [Ziemba et al. \(2016\)](#), which provides electrophysiological recordings of 103 V2 neurons responding to 15 samples of textures from 15 different texture families. As was done in [Ziemba et al. \(2016\)](#), we first use the standard low-dimensional embedding technique T-SNE ([Van der Maaten and Hinton, 2008](#)) to understand qualitatively how our model represents the 225 texture images compared with the actual neural representation. We see in Fig. B.4 that our model seems to capture the relative relationships between texture family centroids, but over-compresses within-family variability. To quantitatively understand the

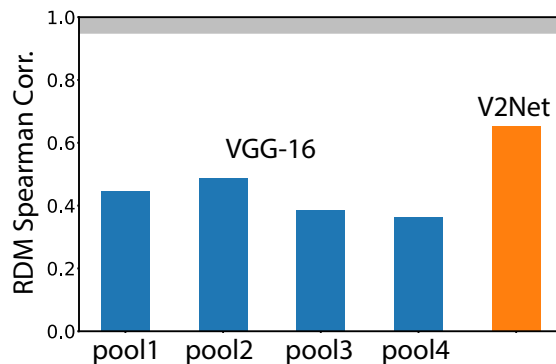


Figure 3.5: V2Net outperforms pre-trained VGG layers in texture family representational similarity with V2 neurons. Spearman rank correlation is plotted for each major layer of the VGG-16 network, as well as V2Net. We see that V2Net better captures the relative centroid positions of the texture families (averaged over samples) than any of the VGG layers, even though the VGG network has been trained with an order of magnitude more data. We show the noise ceiling (estimated from internal splits of the neural data) in gray.

representational similarity between our model and the neural data at the level of texture families (categories), we first compute the averaged response (across samples) of both the model and neural responses to each of the 15 texture families. Next, for each representation, we construct

a dissimilarity matrix based on the pairwise correlation distance. There are many distances one could choose but the correlation distance is one of the most common and performs fairly robustly in comparison with distances such as euclidean distances (Kriegeskorte et al., 2008; Mehrer et al., 2020). As has been noted in the literature (Nili et al., 2014), it is not common to assume a linear relationship between dissimilarity matrices, but it is rather more appropriate to assume the model RDM predicts the rank order of the dissimilarities (Nili et al., 2014). Therefore, we compute the Spearman rank correlation between the dissimilarity matrices of our model and the V2 neural data. Finally, we perform the same analysis for the main blocks of a pre-trained VGG-16 network (as this network has been used heavily as a model of texture (Gatys et al., 2015)). For more details on the physiology data and image presentation see Sec. B.3.

We find that the V2Net representation is more correlated with the V2 population representation than any of the VGG layers. Additionally, we note that the best VGG layer is the block2pool layer, which is in fact the output of many more nonlinear layers than our model. As a result, not only does our model better capture the data, but it also does so with far more limited model capacity. This suggests that stacking a hierarchical model on top of our learned network may lead to an improvement in SoA classification performance while maintaining consistency with biological architectures.

3.6 DISCUSSION

In this work, we demonstrate successful data-efficient self-supervised learning of a simple, yet powerful computational model for representing texture. Rather than learn a very high-dimensional representation followed by linear classification, we use a simpler two-stage model whose responses are then decoded with an *interpretable* non-linear decoder (QDA). This provides the benefit that moving forward we can more easily probe the underlying learned feature space and understand explicitly how those features impact decoding of texture families (through their

covariance structure). In fact, we are not the first to propose such a scheme in the context of neural decoding as QDA has been shown to provide a possible basis for a biologically-plausible non-linear decoding method that can explain quadratic transformations that have been observed between layers of processing in the visual system (Pagan et al., 2016; Yang et al., 2020b). Within this framework, we show that a modification of the common view of hierarchical visual processing (reminiscent of skip-connections (He et al., 2016)), that includes both V1 simple and complex cells as input to a second V2-like processing stage can provide functional benefits in the learning of the texture representation both in terms of classification accuracy and representation similarity with recording neurons in primate area V2. More importantly, we demonstrate that smaller networks do not necessarily perform much better with small training data, but that learning robustly from small numbers of training examples required the development of a novel self-supervised learning objective.

Our learning objective is inspired by recent unsupervised contrastive objectives (separating positive examples from a collection of negatives) (Hénaff et al., 2019b; Oord et al., 2018; Wu et al., 2018; Zhuang et al., 2019). While these methods are general, in they are non-parametric with respect to the distribution of the data, we believe that our parameterization in terms of mean and covariance allows our method to 1) constrain learning in small data regimes and 2) provide opportunities to explore more biologically plausible on-line learning implementations. In particular, it is implausible that the brain can store all samples of the global distribution, and our parameterization allows for on-line sequential update of the mean and covariance statistics for each observed image.

Our method currently assumes a dataset of homogeneous textures as input, as this enables a simple form of objective that minimizes spatial variability of the responses across each image. While this is useful, we believe it to ultimately be a limitation of our method, especially when attempting to learn a model that is aligned with cortical area V2. As seen in the T-SNE visualization (Fig. B.4), our model heavily compresses variability within textures to a point that seems

both biologically inaccurate (as compared with the V2 responses), and potentially problematic for learning visual representations that generalize beyond texture. As a result, we hope in future work to extend our method to allow learning from whole natural scenes, by minimizing variability of responses within *local* spatial neighborhoods, while maximizing global variability. This is motivated by the local consistency of natural images - nearby spatial regions are more likely to be similar than distant ones. In fact, there have been some efforts to use spatial coherence as a learning signal (Becker and Hinton, 1995; Danon et al., 2019; Jean et al., 2019; Ji et al., 2018), splitting the image into independent patches that are processed as inputs to the model during learning. Our objective offers an alternate methodology that can process full images while imposing the locality constraint in the response space. Because of the layer-wise nature of our objective, there is also the potential to extend the method to learn filters in multiple stages of a hierarchical model.

4 | LAYERWISE COMPLEXITY-MATCHED SELF-SUPERVISED LEARNING YIELDS IMPROVED MODELS OF CORTICAL AREA V2

4.1 OVERVIEW

This work is in submission for publication in the proceedings of the Transactions on Machine Learning Research (Parthasarathy et al., 2023b).

In this chapter, we directly extend the work in Chapter 3, in an attempt to overcome many of the limitations of the learned texture model. We will describe a novel canonical layerwise learning method that avoids the prior dataset limitations (being only able to train on textures), and generalizes to learning multiple stages of a visual hierarchy.

Human abilities to recognize complex visual patterns arise through successive transformations in a sequence of areas in the ventral visual cortex. Deep neural networks trained end-to-end for object recognition approach human capabilities, and offer the best descriptions to date of neural responses in the late stages of the hierarchy. But these networks provide a poor account of the early stages, compared to traditional hand-engineered models, or models optimized for coding efficiency or prediction. Furthermore, the gradient backpropagation required for end-to-end learning is widely considered to be a biologically implausible mechanism. Here, we overcome both of

these limitations by developing a bottom-up self-supervised training methodology that operates independently on successive layers. Specifically, we maximize feature similarity between pairs of locally-deformed natural image patches, while decorrelating features across patches sampled from other images. Crucially, the deformation amplitudes are adjusted proportionally to receptive field sizes in each layer, thus matching the task complexity to the capacity at each stage of processing. In comparison with architecture-matched versions of previous models, we demonstrate that our layerwise complexity-matched learning (LCL) formulation produces a two-stage model (LCL-V2) that is better aligned with selectivity properties and neural activity in primate area V2. We demonstrate that the complexity-matched learning paradigm is critical for the emergence of the improved biological alignment. Finally, when the two-stage model is used as a fixed front-end for a deep network trained to perform object recognition, the resultant model (LCL-V2Net) is significantly better than standard end-to-end self-supervised, supervised, and adversarially-trained models in terms of generalization to out-of-distribution tasks and alignment with human behavior.

4.2 INTRODUCTION

Perception and recognition of spatial visual patterns, scenes and objects in primates arises through transformations performed in a cascade of areas in the ventral visual cortex (Ungerleider and Haxby, 1994). The early stages of visual processing (in particular, the retina, lateral geniculate nucleus, and cortical area V1), have been studied for many decades, and hand-crafted models based on linear filters, rectifying nonlinearities, and local gain control provide a reasonable account of their responses properties (Adelson and Bergen, 1985; Carandini et al., 1997; McLean and Palmer, 1989; Shapley and Victor, 1979) Complementary attempts to use bottom-up normative principles such as sparsity, coding efficiency, or temporal prediction have provided successful accounts of various early visual properties (Atick and Redlich, 1990; Bell and Sejnowski, 1997;

Cadiou and Olshausen, 2012; Hoyer and Hyvärinen, 2002; Karklin and Lewicki, 2009; Karklin and Simoncelli, 2011; Li, 1996; Olshausen and Field, 1996; Schwartz and Simoncelli, 2001; Van Hateren and van der Schaaf, 1998; Wiskott and Sejnowski, 2002). But these also have been limited to early stages up to area V1, and have thus far not succeeded in going beyond.

Deep neural networks (DNNs), whose architecture and functionality were inspired by those of the primate visual system (Douglas et al., 1989; Fukushima, 1980; Heeger et al., 1996; Riesenhuber and Poggio, 1999), have offered a new opportunity. When trained with supervised and self-supervised end-to-end backpropagation, DNNs have provided the first models that begin to capture response properties of neurons deep in the visual hierarchy (Kubilius et al., 2019; Schrimpf et al., 2018; Yamins et al., 2014; Zhuang et al., 2021). Early results showed that these DNNs are also generally predictive of the overall category-level decisions of primates during object recognition tasks (Ghodrati et al., 2014; Jozwik et al., 2016; Kheradpisheh et al., 2016); however, they have not been predictive of more detailed behavior, as measured by alignment with individual image confusion matrices (Rajalingham et al., 2018). Nevertheless, as the field has rapidly progressed, more recent results demonstrate that scaling end-to-end task-optimization (both in training data and model size) leads to significant improvements in predicting this trial-by-trial human behavior in matched visual tasks (Geirhos et al., 2021; Sucholutsky et al., 2023).

Ironically, despite their historical roots, these same networks have not provided convincing models of early visual areas such as V1 and V2, and do not account for other perceptual capabilities (Berardino et al., 2017; Bowers et al., 2022; Feather et al., 2023; Fel et al., 2022; Hénaff et al., 2019a; Subramanian et al., 2023). Figure 4.1 summarizes these observations for a set of models with a wide variety of architectures and training paradigms, drawn from the BrainScore platform (Schrimpf et al., 2018). The left panel shows that improvements in object recognition performance are strongly correlated ($r = 0.57$) with improvements in accounting for human recognition capabilities. This is encouraging, but perhaps expected, since the recognition databases used for training represent human-assigned labels. The right panel shows that there is also a positive cor-

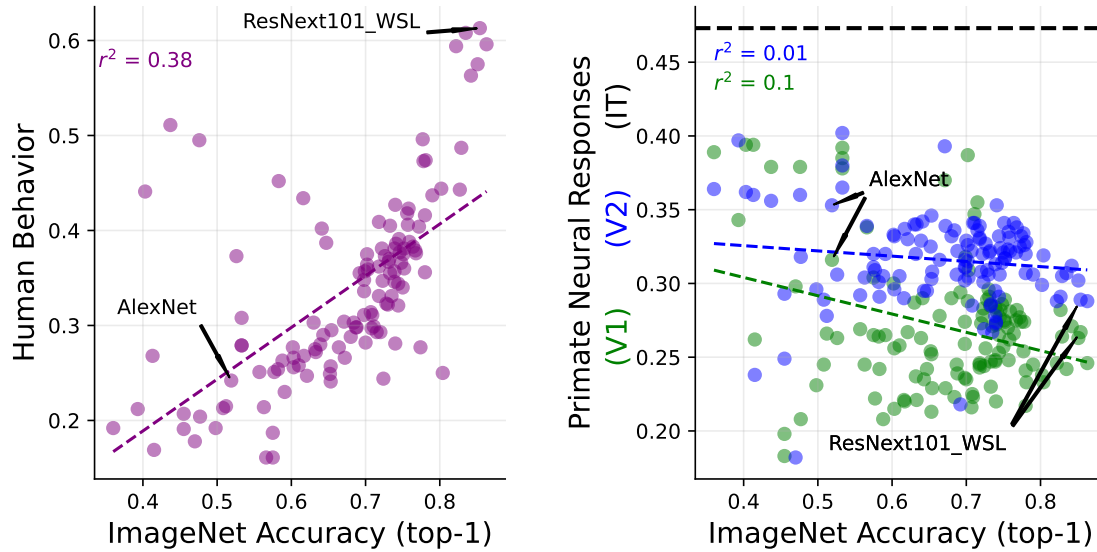


Figure 4.1: DNN object recognition performance predicts human recognition behavior, but not primate early visual responses. Each plotted point corresponds to a DNN model from the BrainScore database (Schrimpf et al., 2018). Horizontal axis of both panels indicates recognition accuracy (top-1) on the ImageNet dataset (Krizhevsky et al., 2012). **Left:** Comparison to alignment with human visual recognition performance (combination of benchmarks taken from (Geirhos et al., 2021) and (Rajalingham et al., 2018)). **Right:** Comparison to neural variance explained by regressing the best-fitting DNN layer to neural responses measured in macaque V1 (green), V2 (blue) (Freeman et al., 2013; Ziemba et al., 2016) and IT (black) (Majaj et al., 2015; Sanghavi and DiCarlo, 2021; Sanghavi et al., 2021a;b)

relation (albeit weaker) between recognition performance and ability to explain responses of IT neurons recorded in macaque monkeys. Again, this is perhaps not surprising, given that object-recognition behavior can be to some extent explained by linear weightings of IT responses (Majaj et al., 2015). (It is worth noting, though, that for models with very high recognition performance (>70%), ability to explain IT neurons in fact has been getting worse, an observation that has recently been explored in greater detail in (Linsley et al., 2023)). However, surprisingly, recognition performance is uncorrelated (or even slightly anti-correlated) with the ability to explain responses of early visual neurons in cortical areas V1 and V2.

Why do these networks, which offer human-like performance in complex recognition tasks, and which provide a reasonable account of neural responses in deep stages of the visual hierarchy, fail to capture earlier stages? We interpret this as an indication that intermediate DNN layers are

insufficiently constrained by end-to-end training on recognition tasks. More specifically, the extremely high model capacity of these networks allows the training procedure to find “shortcuts” that satisfy single end-to-end objectives (both supervised and self-supervised) (Geirhos et al., 2020a; Robinson et al., 2021). As a result, it is common for networks to utilize unreliable feature representations that do not generalize well (Hermann and Lampinen, 2020). This is further evidenced by the fact that standard trained networks can be fooled by ‘adversarial examples’ (small pixel perturbations that can large shifts in internal classification decision boundaries) (Goodfellow et al., 2014; Szegedy et al., 2013; Tramèr et al., 2017). With this in context, it makes sense that the best models for V1/V2 (just under 40% explained variance) seem to be those that are trained to increase robustness to adversarial attacks (Madry et al., 2017). However, the specific solution of adversarial training comes at a significant cost in standard image recognition performance, as well as being both computationally expensive and biologically-implausible.

In this work, we hypothesize that representations throughout a DNN can be constrained in a more biologically-plausible manner through the use of *layerwise* self-supervised learning objectives. We propose a natural method for matching the complexity (or difficulty) of these objective functions with the computational capacity at each stage of processing. When used to train a two-stage model, the resulting network achieves state-of-the-art predictions of neural responses in cortical area V2. Furthermore, when using this learned model as a front-end for supervised training with deeper networks, we show that (in contrast with adversarial training) the increased neural alignment does not come at the cost of object recognition performance, and in fact results in significant improvements in out-of-distribution recognition performance and alignment with human behavior.

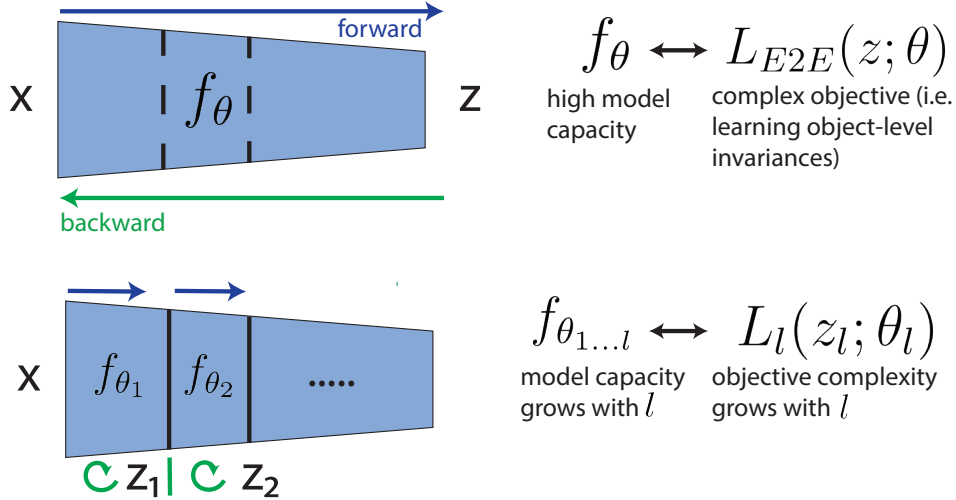


Figure 4.2: Layerwise complexity-matched learning. Top: The standard end-to-end (E2E) learning paradigm used with DNNs. The loss function (L_{E2E}) operates on the network output and is typically chosen to favor object-level invariances, through supervised training on labelled data or self-supervised training on augmented examples. To solve these E2E objectives, the network $f(\theta)$, must have a high model capacity (sufficiently large number of parameters and non-linearities). **Bottom:** In a layerwise training system, the loss is a function of all intermediate outputs (z_1, z_2, \dots). Losses at each layer L_l are used to train each encoder stage f_{θ_l} independently, with gradients operating only within stages. For effective training, we hypothesize that the loss at each stage, L_l , should be matched in complexity to the model capacity defined by the network up to layer l .

4.3 METHODS

Our layerwise training approach is illustrated in Fig. 4.3. We first describe the key conceptual underpinnings of the method, and then provide the experimental training and evaluation details.

4.3.1 LAYERWISE COMPLEXITY-MATCHED LEARNING

Layerwise (more generally, blockwise) methods for DNN training have been previously developed to alleviate the global propagation of gradients required in end-to-end (E2E) training training (Belilovsky et al., 2019; Bengio et al., 2006; Halvagal and Zenke, 2023; Hinton et al., 2006; Illing et al., 2021; Siddiqui et al., 2023). Figure 4.2 illustrates the relationship between the two approaches. Given a set of inputs x and corresponding output labels z , the E2E approach op-

timizes all network parameters θ to minimize the loss function L_{E2E} via full backpropagation. Successful training of high-capacity networks has generally been achieved with large amounts of training data and complex objectives: (1) supervised data that encourages object-level semantic invariances (Krizhevsky et al., 2012), (2) self-supervised data generated using a combination of spatial and photometric augmentations (Chen et al., 2020b; Grill et al., 2020; Zbontar et al., 2021), or self-supervised masked autoencoding with substantial levels of masking (He et al., 2021). In general, the quality of learned features and the success in recognition depends on the complexity (or difficulty) of the learning problem. For example, if we consider a supervised classification objective, the difficulty of this problem will depend on factors such as the number of classes, complexity of the image content (simple shapes vs. real-world objects), or the magnitude of the within-class variability (deformations under which each object is seen). Similarly, these training set properties control the complexity of the self-supervised problem (Jing et al., 2021; Robinson et al., 2021). In contrast, in the layerwise approach, the objective is partitioned into sub-objectives that operate separately on the output of each layer, and the optimization thus relies on gradients that propagate within (but not between) layers. The model at a given layer l is composed of all stages up to that layer: $f_{\theta_{1..l}}$. Thus, the computational capacity is low in the early layers (only a few non-linearities and small receptive fields) and increases gradually with each successive layer. To achieve successful training in this scheme, we propose to *match the complexity of the data diversity and objective function with the effective model capacity at a given layer*.

4.3.2 SELF-SUPERVISED CONTRASTIVE OBJECTIVE

We construct a layerwise objective based on the “Barlow Twins” self-supervised loss (Zbontar et al., 2021), a feature-contrastive loss that is robust to hyperparameter choices and has recently shown success in blockwise learning (Siddiqui et al., 2023). Briefly, each image x in a batch is transformed into two views, x^A and x^B , via randomly selected spatial and photometric deformations. Both views are propagated through an encoder network f_θ and a projection head g_θ

to produce embeddings $z = g_\theta \circ f_\theta(x)$. We define a cross-correlation matrix over each batch of images and corresponding view embeddings:

$$c_{ij} = \frac{\sum_b z_{b,i}^A z_{b,j}^B}{\sqrt{\sum_b (z_{b,i}^A)^2} \sqrt{\sum_b (z_{b,j}^B)^2}} \quad (4.1)$$

where b indexes the batch and i and j index the components of the projection head response. The Barlow Twins objective function is then:

$$L_{BT} = \sum_i (1 - c_{ii})^2 + \lambda \sum_i \sum_{j \neq i} c_{ij}^2 \quad (4.2)$$

This loss encourages formation of invariant projection-features (or equivariant encoder features) across the two views (maximizing the diagonal terms of the correlation) and decorrelated features across the different images in a batch (minimizing the off-diagonal terms). This objective can thus be thought of as a “feature-contrastive” method. As noted in (Garrido et al., 2022), there is a strong duality between this loss and with sample-contrastive losses (such as SimCLR (Chen et al., 2020b)). Accordingly, we achieve similar results in our framework using sample-contrastive losses (more in Sec. 4.4.4), but find slight improvements in performance and stability with the Barlow Twins objective.

4.3.3 TRAINING METHODOLOGY

We applied our Layerwise Complexity-matched Learning paradigm (LCL) to a two-stage model, denoted **LCL-V2**. The training methodology is depicted in Fig. 4.3. The loss function aims to optimize feature invariance across augmented views of an image, while decorrelating features across different images. We control the complexity (difficulty) of each of these learning problems by changing both the size of input images and the strength of augmentation deformation that are used to compute the per-layer loss functions. Fig. 4.3 (a) depicts this input processing for an

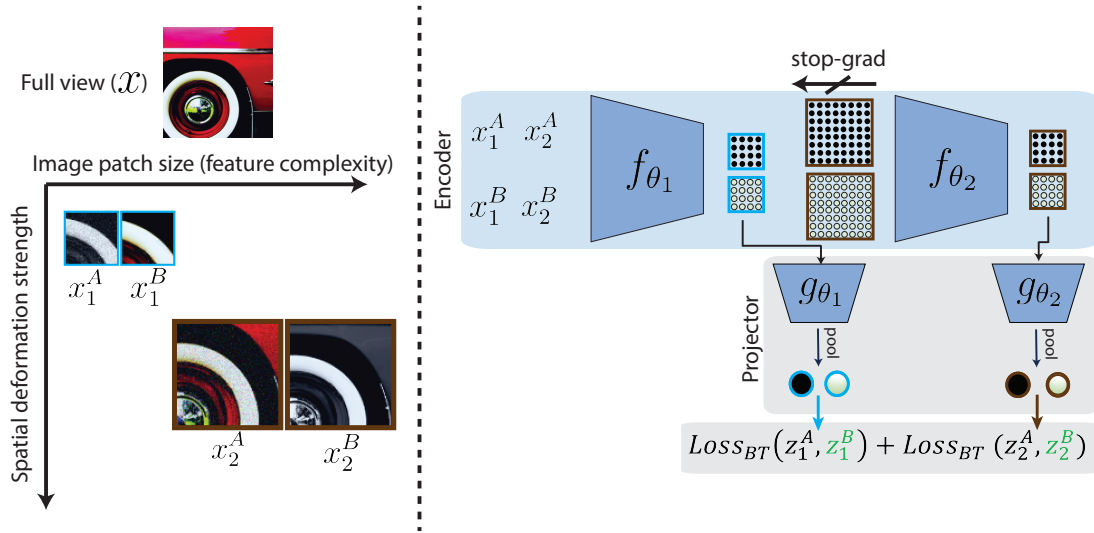


Figure 4.3: Layerwise complexity-matched objective. Layerwise complexity-matched objective. **Left:** For each layer, the objective encourages invariance to feature perturbations by comparing the representation of two augmented views of the same image. For layer l , the feature complexity of generated image pair (x_l^A, x_l^B) is controlled through choice of patch size, and the magnitude of spatial deformations (translation, dilation). **Right:** The parameters θ_1 of the first layer encoder f_{θ_1} are updated using the Barlow Twins feature-contrastive loss (Zbontar et al., 2021) operating on the two views of the smallest patch size (x_1^A, x_1^B) . This set of views is only propagated to this layer output. The parameters θ_2 of the second layer encoder f_{θ_2} are updated with the same loss, but using the views that cover a larger spatial region, and include larger spatial deformations.

example image, considering our two-layer network. Given the full view x , a patch is cropped for layer 1 (x_1^A) and layer 2 (x_2^A). We choose an initial patch size for x_1^A and note that the patch size for layer 2 is simply scaled by a factor of 2, roughly matched to the scaling of biological receptive field sizes between areas V1 and V2 (Freeman and Simoncelli, 2011). For the selected patches, we then generate augmented versions (x_2^A, x_2^B) using photometric and spatial deformations. For simplicity, we maintain the same photometric deformations and scale the problem complexity by proportionally adjusting the strength of the spatial deformation by a factor of 2 between the two layers. Visually, we see that this procedure results in paired images for layer 1 that have low feature complexity and small translation and scale differences while the images for layer 2 have higher feature complexity and larger deformations.

Given these inputs, Fig. 4.3(b) shows the procedure for computing the per-layer loss functions. We generate projection embeddings for layer 1 and layer 2 by propagating the corresponding input patches to the corresponding model blocks:

$$z_1^A = \text{Pool} \circ g_{\theta_1} \circ f_{\theta_1}(x_1^A)$$

$$z_2^A = \text{Pool} \circ g_{\theta_2} \circ f_{\theta_2} \circ f_{\theta_2}(x_2^A)$$

f_{θ_i} refers to the encoder blocks and g_{θ_i} corresponds to the projection heads for each layer. z_1^B and z_2^B are computed analogously from x_1^B and x_2^B . The loss is then computed as the sum of losses for each layer: $Loss = L_{BT}(z_1^A, z_1^B) + L_{BT}(z_2^A, z_2^B)$. As in (Siddiqui et al., 2023), the loss computation only requires backpropagation within each layer, and gradients from the layer 2 loss do not affect parameters in f_{θ_1} .

In summary, we implement a complexity-matched layerwise learning formulation where the difficulty of the learning problem at layer 2 is scaled in comparison with that at layer 1. The model must learn invariant features across images that have more complex content (larger patch size) that are also more strongly deformed (in scale and translation). This increase in objective complexity accompanies a corresponding increase in model capacity in the second layer (due to growth in receptive field size and number of nonlinearities).

4.3.4 IMPLEMENTATION DETAILS

Architecture. As in many previously published results (Caron et al., 2018; Gidaris et al., 2018), we chose to use the AlexNet architecture (Krizhevsky et al., 2012) with batch normalization, (Ioffe and Szegedy, 2015). While many recent results make use of more complex architectures (eg, ResNets (He et al., 2016), Vision Transformers (Dosovitskiy et al., 2020) etc.) our method can be more effectively evaluated with a very shallow network, as we can severely restrict model capacity in training these early layers without confounding architectural features such as skip con-

nections, attention blocks etc. Additionally, as mentioned earlier, much of the biological anatomy and computational theories suggest that the feed-forward aspect of areas V1 and V2 should be explainable by networks with few computational stages. As a result, we hypothesize that the AlexNet architecture can provide a more parsimonious and interpretable model of these areas.

For LCL-V2 we train the first two convolutional stages of the AlexNet architecture and utilize a standard multi-layer perceptron (MLP) with a single hidden layer for the projector networks at each layer. The computational capacity is increased between the two stages, with each stage incorporating two non-linearities (ReLU activation and MaxPooling). In addition, capacity is scaled by increasing the number of channels (64 to 192) and receptive field size (via (2x) subsampled pooling). In Sec. 4.4.5, we additionally evaluate the effectiveness of LCL-V2 as a fixed front-end model (similar to (Dapello et al., 2020)). We train the remaining AlexNet layers (with batch normalization) on top of the fixed LCL-V2 front-end and refer to this full network as **LCL-V2Net**. For more specific architecture details, see appendix Sec C.1.

Data and Optimization. We trained LCL-V2 and its ablations (see Sec. 4.4.4) on the ImageNet-1k dataset (Krizhevsky et al., 2012). We resized the original images to minimum size 224x224. For layer 1 we centrally cropped a 56x56 patch and generated spatially augmented views of size 48x48 via the RandomResizedCrop (RRC) operator with scale = (0.6, 0.9). For layer 2, we central cropped a 112x112 patch and generate views of size 96x96 with RRC crops (scale = (0.3, 0.9)). As a result, both the final patch size and crop scale range are doubled between layer 1 and layer 2. For each set of patches, we also applied a fixed set of photometric distortions by weakly varying contrast, luminance, and adding random Gaussian noise with variable standard deviation (details in Sec. C.3). Unlike standard E2E self supervised approaches, we do not use the more aggressive augmentations (large color jitter, flipping etc), which seem less perceptually relevant.

We use the Adam optimizer (Kingma and Ba, 2014) without weight decay and $lr = 0.001$. We train the model until the summed validation loss (evaluated on a held-out set of images) does not improve above beyond a fixed threshold. While recent work in self-supervised learning has found

benefits from using more complex optimizers and learning rate schedules, we find no significant benefits in our two-layer setting. To train the full LCL-V2Net, we fix the pretrained LCL-V2 as a front-end and use a standard supervised cross-entropy loss to train the subsequent stages. We train for 90 epochs using the SGD optimizer ($lr = 0.1$) with a step-wise learning rate scheduler that reduces the learning rate every 30 epochs.

4.3.5 EXPERIMENTAL SETUP

Model comparisons. Throughout this work we compare to a variety of previous models of three types (see Sec. C.1 for details):

- **E2E (standard):** End-to-end AlexNet models trained with standard supervised or self-supervised objective functions on the ImageNet-1K dataset: Supervised (Krizhevsky et al., 2012), Barlow Twins (Zbontar et al., 2021), and VOneNet (fixed-V1 stage and supervised learning for downstream stages) (Dapello et al., 2020).
- **E2E (robust):** End-to-end AlexNet models trained with state-of-the-art robustification methods specifically to maintain robustness to adversarial pixel perturbations: standard adversarial training (L2-AT ($\epsilon = 3.0$)) (Madry et al., 2017), adversarial noise training with a parameterized noise distribution (ANT) (Rusak et al., 2020).
- **Layerwise training:** AlexNet model trained with the Barlow Twins objective using the standard image augmentation scheme applied layerwise (Siddiqui et al., 2023), Latent predictive learning (LPL) (Halvagal and Zenke, 2023).
- **Hand-crafted:** Steerable pyramid layer (Simoncelli and Freeman, 1995) (with simple and complex cell nonlinearities), followed by a layer of spatial L_2 (energy) pooling.

Neural alignment evaluations. We compare all models quantitatively in their ability to predict aspects of V2 neurons from the dataset used in BrainScore (Schrimpf et al., 2018). This dataset,

described in (Freeman et al., 2013; Ziemba et al., 2016), provides electrophysiological recordings of 103 V2 neurons responding to texture images synthesized with the Portilla-Simoncelli texture model (Portilla and Simoncelli, 2000). The data include responses to 15 texture samples, in addition to 15 samples of spectrally-matched noise images, for 15 different texture families (a total of 450 images). To measure model predictivity of this neural data, we fit models with the data splits and implementation of the partial least squares (PLS) regression method proposed in (Schrimpf et al., 2018), and then compute explained-variance scores for each fitted model. Details are provided in Sec. C.4.

To better understand the ability of models to capture selectivities of V2 neurons, we provide additional evaluations (Sec. 4.4.2) that use the texture modulation ratio statistic introduced in (Freeman et al., 2013). Specifically, we define: $R_{mod_{n,i}} = \frac{tex_{n,i} - noise_{n,i}}{tex_{n,i} + noise_{n,i}}$, where $tex_{n,i}$ is the response of neuron n (averaged across 15 image samples) to texture family i and $noise_{n,i}$ is the corresponding response to the spectrally-matched noise for family i .

LCL-V2Net recognition and human behavior evaluations. We primarily use the out-of-distribution (OOD) generalization benchmark of (Geirhos et al., 2021) to test the performance of LCL-V2Net. This dataset consists of 17 OOD classification tasks based on adding various kinds of noise, distortions, and shape-biasing transformations to ImageNet images. We evaluate both OOD accuracy and consistency with human behavior. For more information on the benchmark, specific list of distortions, and evaluation metrics see Sec. C.5.

We additionally report performance on the original ImageNet-1K (Krizhevsky et al., 2012) validation set as well as more recent large-scale validation sets (ImageNet-R (Hendrycks et al., 2021a) and ImageNet-vid-robust (Shankar et al., 2021)) for testing generalization.

4.4 RESULTS

4.4.1 POPULATION FITS TO NEURAL DATA

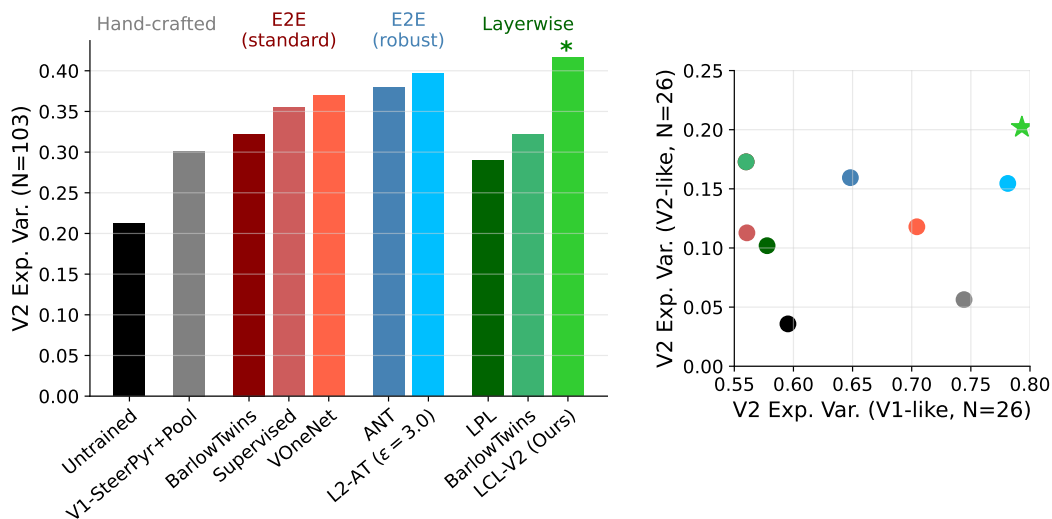


Figure 4.4: Left: Median explained variance of models fitted with PLS regression to 103 primate V2 neural responses. For models with more than two layers, all layers are evaluated and the performance of the best layer is provided. **Right:** Comparison of median explained variance for “V1-like” and “V2-like” V2 cells. These categories correspond to the top and bottom quartiles (N=26) of V2 cells sorted by how well they are fit by a canonical hand-constructed V1 model (V1-SteerPyr+Pool). The minimum explained variance of the V1 model over the set of “V1-like” neurons is 57 %. The maximum explained variance over the set of “V2-like” neurons is 14 %. The LCL-V2 and L2-AT models significantly outperform all other models on the V1-like subset, even surpassing the baseline V1 model. The LCL-V2 model also significantly outperforms the L2-AT model on the least V1-like subset.

Overall V2 Predictivity. The first panel of Fig. 4.4 shows the overall BrainScore explained variance of the models outlined in Sec. 4.3.5. For all models, the best layer was chosen by evaluating predictions on a validation set prior to fitting the final PLS regression on the held-out test set. We see that LCL-V2 outperforms all architecture-matched models, including the L2-AT trained network. In fact, although we only show architecture-matched results here, our model provides the best account of the V2 data across all architectures currently on the BrainScore leaderboard (Schrimpf et al., 2018; 2020). Interestingly, previous layerwise training methods (Barlow (layer-

wise) from (Siddiqui et al., 2023) and LPL (Halvagal and Zenke, 2023)) exhibit significantly worse performance than the standard end-to-end training. This suggests that the benefits of our method specifically arise from complexity-matching, something we quantify further in Sec. 2.4.3.

Partitioning V2 with a V1-baseline model. The V1-SteerPyr+Pool model provides a baseline measure of how well V2 neurons can be predicted simply by combining rectified and L_2 -energy pooled oriented filter responses (as are commonly used to account for V1 responses). Nearly 30 % of the variance across all 103 V2 neurons can be explained given this model, suggesting that there are a number of V2 neurons that are selective for orientation and spatial frequency selectivity. In fact, this aligns with prior studies that have found subsets of V2 neurons with tuning similar to V1 neurons (but with larger spatial receptive fields) (Foster et al., 1985; Lennie, 1998; Levitt et al., 1994; Willmore et al., 2010).

Given this baseline model, we partition the V2 neural datasets into neurons that are ‘V1-like’ (top quartile, in terms of how well they are explained by the V1-SteerPyr+Pool model) and those that are ‘not-V1-like’ (bottom quartile). In the right panel of Fig. 4.4, we compare the median performance of each of the models on each subset. We see that all other non-adversarially trained models (both layerwise and end-to-end) are significantly worse at predicting the ‘V1-like’ subset than the baseline V1 model. Surprisingly, both LCL-V2 (ours) and L2-AT models outperform the V1 baseline on this subset, suggesting that although these neurons are most-likely orientation and spatial frequency tuned, they also have some selectivity that is not captured by the simple V1 model. On the ‘not-V1-like’ subset, the performance of all models is significantly worse; however, there is now an even larger gap (approx 5%) between LCL-V2 and the L2-AT model. Thus, the adversarial training achieves better predictions of area V2, primarily by better explaining the neurons that have ‘V1-like’ properties. LCL-V2 maintains this improvement, but also provides better fits to neurons whose complex feature selectivity is not well described by the baseline V1 model. In the following section, we examine whether the models exhibit known feature selectivities found in the V2 data.

4.4.2 MODEL COMPARISONS VIA TEXTURE MODULATION

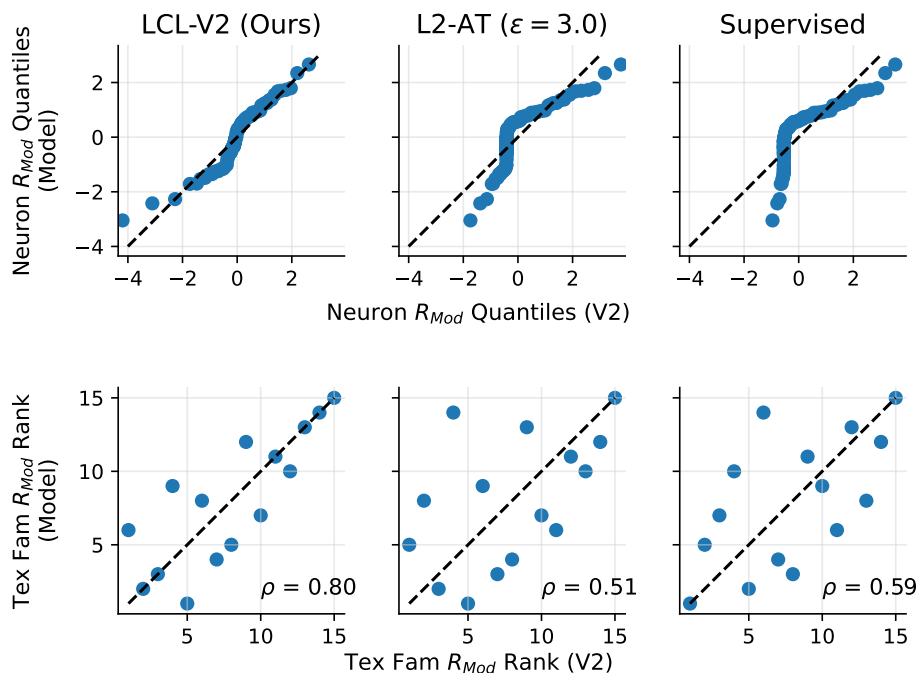


Figure 4.5: The LCL-V2 model outperforms other models in capturing texture modulation properties of V2 neurons. Here, we compare the top 3 (in terms of overall V2 predictivity) fully-learned models: LCL-V2 (Ours), L2-AT, and Supervised. **Top:** Quantile-quantile (Q-Q) comparison of the distribution of texture modulation index values (R_{mod} , averaged over texture families) for real and model neurons. The LCL-V2 model shows significantly better alignment with the physiological distribution (closer to the identity line (dashed)) than the other two models. **Bottom:** Comparison of texture modulation indices for each of 15 texture families (averaged over neurons). The texture modulation indices for both model and real neurons are ranked (1 = lowest modulation family, 15 = highest modulation family), and scatter-plotted against each other. Our model provides significantly better alignment with the V2 data, achieving a Spearman rank correlation of $\rho = 0.80$.

Cortical area V2 receives most of its input from V1. A fundamental property of V2 neural responses that is not present in V1 responses is that of *texture modulation* (Freeman et al., 2013; Ziemba et al., 2016), in which responses to homogeneous visual texture images are enhanced relative to responses to spectrally-matched noise. As described in Sec. 4.3.5, we compute a texture modulation index $R_{mod_{n,i}}$ for each of the 103 neurons, for each of the 15 texture families. We computed the same modulation index for each neuron in the selected V2-layer from each

computational model. In Fig. 4.5 we compare LCL-V2 against the top two other *fully learned* models in terms of overall V2 explained variance (L2-AT and standard ImageNet1K-Supervised). We exclude the VOneNet model here as it uses a fixed front-end with a different architecture.

We first compare the three models in terms of their ability to capture the full distribution of texture modulation ratios in the V2 dataset (Fig. 4.5(a)). We compute a modulation ratio for each neuron by averaging over texture families: $R_{mod_n} = \frac{1}{T} \sum_{i=1}^T R_{mod_{n,i}}$. We use a quantile-quantile (Q-Q) plot to compare the quantiles of the distribution of these values to those arising from the modulation ratios of each fitted model neuron. It is visually clear that while none of the models perfectly match the V2 neural distribution, LCL-V2 is significantly closer than the other two.

Next, we compute texture modulation ratios for each texture family by averaging over neurons $R_{mod_i} = \frac{1}{N} \sum_{n=1}^N R_{mod_{n,i}}$. Because different texture classes have different types of feature content, they stimulate V2 neurons differently, relative to their spectrally-matched counterparts. We compare the rank-ordering of modulations ratios over the texture families for the model and real neurons and scatter-plot the ranks against each other (Fig. 4.5(b)). The texture family ranks of the LCL-V2 model are well-aligned with those of the actual V2 neurons, whereas both L2-AT and Supervised models yield ranks with many more outliers. This is quantified by the Spearman rank correlation for LCL-V2 ($\rho = 0.8$), which is significantly higher than that of the other two models ($\rho = 0.51$, $\rho = 0.59$). It is worth noting that (Laskar et al., 2020) find that this rank correlation can be improved for models by incorporating a subset selection procedure to restrict the specific model neurons used in the comparison.

In summary, although the L2-AT and Supervised models provide competitive predictivity of the V2 neural responses (Fig. 4.4), the LCL-V2 model provides a better account of the texture selectivity properties of these neurons.

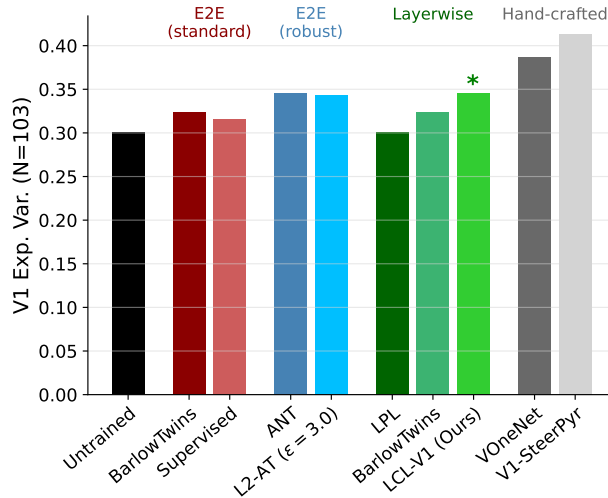


Figure 4.6: LCL-V1 outperforms learned models in V1 predictivity and approaches the performance of hand-tailored V1 models. Analogous to the V2 comparisons (Fig. 4.4), we also evaluate the best model layers for explaining the V1 neural responses from the same dataset. The highest explained variance is obtained by the hand-designed V1-SteerPyr and VOneNet models. The LCL-V1 model performs similarly to the adversarially robust models, and outperforms all other trained models.

4.4.3 V1 LAYER ANALYSIS

To demonstrate the generality of our LCL approach in learning feature hierarchies, we evaluate the first-stage (LCL-V1) in terms of alignment with V1 responses and selectivities. In Fig. 4.6, we find that the LCL-V1 model outperforms all non-adversarially trained models in terms of V1 explained variance (approx 2-4 % improvement on average), and is on par with both adversarially-trained models (ANT and L2-AT). Furthermore, when visualizing and characterizing the learned receptive fields, we find reasonable qualitative similarity with receptive field properties extracted from the V1 data in (Ringach, 2002) (for details, see Sec. C.6). Note, however, that the hand-crafted models (VOneNet and V1-SteerPyr) still provide better accounts for the V1 responses than all learned models. We suspect this is largely due to the limited receptive field sizes of the learned models, all of which use a single convolutional layer with 11x11 kernels.

4.4.4 ABLATIONS

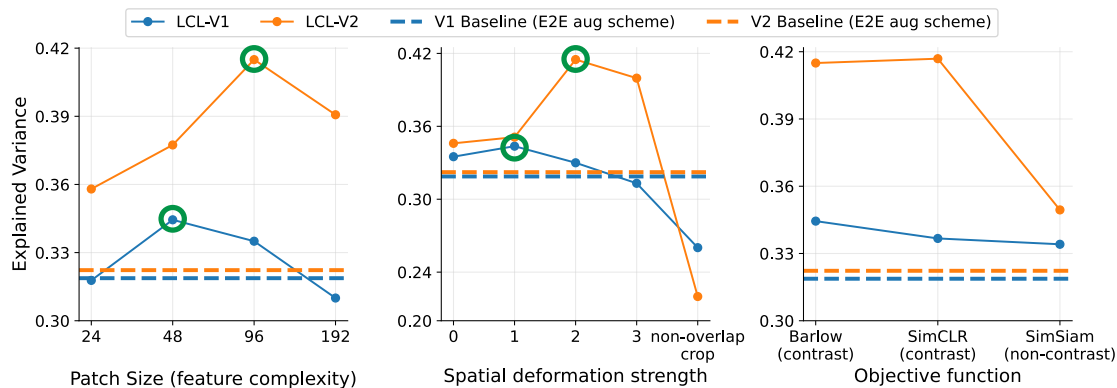


Figure 4.7: Substituting complexity-mismatched or non-contrastive objectives decreases neural alignment. For complexity ablations (left and middle panels), we vary the patch size or spatial deformation strength for the V1 layer (LCL-V1). We then hold these parameters at the optimal values for the V1 layer fixed and again vary the parameters used for training the V2 layer. **Left:** Indicated by the green circle, we see that there is an optimal patch size (feature complexity) for best V1 prediction at 48px and the optimal patch size for V2 is then scaled accordingly (factor of 2 larger). **Middle:** We see that there is also an optimal spatial deformation strength for each layer that is also scaled by a factor of 2 (in minimum crop scale) between each layer. Spatial deformation strength 0 refers to no spatial deformation. Deformations (1-3) refer to the minimum random resized crop scale of (0.6, 0.3, 0.08). ‘Non-overlap crop’ refers to only using non-overlapping crops. **Right:** We ablate the loss function used to train each layer. We find that performance is very similar with SimCLR but gets significantly worse (especially for the V2 stage) when using a non-contrastive method like SimSiam. Baseline comparisons (dashed lines) indicate performance of the layerwise Barlow method proposed in (Siddiqui et al., 2023), which uses the end-to-end training augmentation scheme (details in Sec. C.3) from (Zbontar et al., 2021) for each layer.

We examined the effect of ablations of our architectural and training choices on physiological alignment.

Complexity-mismatch. Fig. 4.4 shows a significant improvement in V2 predictivity over all non-adversarially trained models and in particular improves dramatically over the application of the layerwise training approach outlined in (Siddiqui et al., 2023). Since that model was also trained with the Barlow Twins objective, the primary difference with our model is in the complexity-matching of our objective. Specifically, while they use at each layer a set of augmentations generally used to train large object recognition networks, we approximately complexity-match the objective with capacity at each stage of our model. To better understand the quanti-

tative impact of this, we evaluate the neural predictivity of our learned model when introducing complexity-mismatch via changes in the patch size (feature complexity) or random crop scale (spatial deformation strength). Fig. 4.7 shows that the optimal performance (in terms of neural predictivity) is achieved only when the relative complexity is matched between the two layers. Specifically, we first see that there is an optimal patch size (48px) and deformation strength ($s=1$) which produces the most aligned V1 layer. This is surprising, as we chose these parameters initially based on the hypothesis that the the layer 1 views should contain simple edge-like content with small scale deformations. More importantly, once the optimal parameters for the V1 layer training are fixed, we find that both patch size and spatial deformation strength must be scaled accordingly to achieve the optimal V2 model (highlighted in green). This again justifies the initial choice of these scaling parameters based on the natural approximate doubling of receptive field size between V1 and V2.

Contrastive vs non-contrastive losses. While numerous self-supervised learning objectives have been proposed over the years, they generally can be classified as contrastive or non-contrastive. While the Barlow Twins loss is ‘feature-contrastive’, there have been studies demonstrating a duality with ‘sample-contrastive’ approaches (Balestriero and LeCun, 2022; Garrido et al., 2022). Non-contrastive losses; however, resort to very different mechanisms for avoiding collapsed solutions, with most using some form of ‘stop-gradient’ based method with asymmetric encoder networks (Chen et al., 2020b; Grill et al., 2020). In Fig. 4.7(c) we show that some form of a contrastive term (either Barlow Twins or SimCLR) is necessary for achieving optimal neural alignment (especially in the second stage). For example, using SimSiam (Chen et al., 2020b) as a representative non-contrastive loss greatly hurts the V2 predictivity from the second stage. Therefore, in this context, it seems that in addition to the problem of learning invariances, it is also necessary to include feature decorrelation or sample discrimination as part of the final objective.

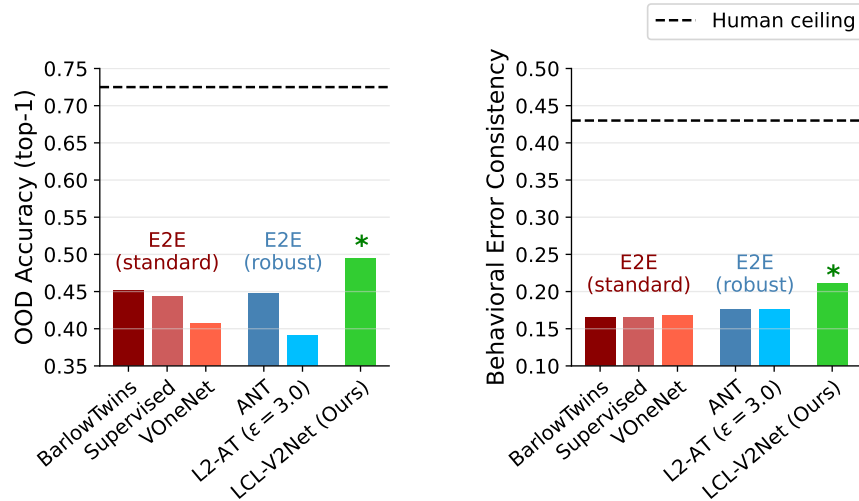


Figure 4.8: LCL-V2Net improves OOD generalization and human behavior error consistency. **Left:** Supervised training of the later layers of an AlexNet model on top of LCL-V2 leads to significantly increased OOD accuracy compared to all other architecture-matched models including those trained with standard self-supervised and supervised objectives as well as those trained for robustness. **Right:** Compared with the same set of architecturally-matched models, LCL-V2Net also shows significantly increased human alignment, as measured by behavioral error consistency on the OOD recognition task. Human-level accuracy and human to human error consistency are indicated by dashed lines.

4.4.5 OOD OBJECT RECOGNITION AND HUMAN-ALIGNMENT

In the spirit of (Dapello et al., 2020), we hypothesize that a more biologically-aligned model of early visual areas (specifically area V2) may provide additional benefits for both recognition performance and alignment with human behavior on visual tasks. We therefore train a cascade of additional AlexNet stages appended to the fixed LCL-V2 front-end model on supervised object recognition. We then evaluate the full network on the benchmark proposed by (Geirhos et al., 2021) which tests both out-of-distribution (OOD) generalization and prediction of human behavior on this task. Again, we refer to this trained recognition network as LCL-V2Net.

Object recognition accuracy. We first evaluate the accuracy of our trained network on the ImageNet-1K (Krizhevsky et al., 2012) validation set as well as OOD image set proposed in (Geirhos et al., 2021). Fig. 4.8 (Left) shows that LCL-V2Net significantly outperforms all architecturally-matched (AlexNet-based) models in OOD accuracy by a large margin (4-10 %

improvement - see Table C.4). This is particularly striking because the other robust models (VOneNet and L2-AT) do not exhibit a similar improvement, suggesting a potential link between the improved V2 predictions of the LCL-V2 front-end and the generalization of recognition over shifts in the data distribution.

Human behavioral consistency. In addition to absolute recognition performance, we also evaluate the the ability of the same models to capture human behavioral performance on the same recognition task. The right panel of Fig. 4.8 shows that LCL-V2Net has significantly better error consistency with the per-trial human recognition decisions. Compared with standard supervised training (consistency=0.165), the only models to show significant improvement are those trained for adversarial robustness (ANT and L2-AT). These models each achieve a consistency of 0.176 (a 6.6 % relative improvement). Without the computational overhead of adversarial training procedures, LCL-V2Net achieves a consistency of 0.211 (a 28 % relative improvement).

4.5 DISCUSSION

We have developed a novel normative theory for learning early visual representations without end-to-end backpropagation or label supervision. We hypothesize that the reason why state-of-the-art DNNs have failed to predict responses of neurons in early visual areas is because they are insufficiently constrained. We then proposed a solution that imposes these constraints through layerwise complexity-matched learning (LCL) that leverages a canonical self-supervised objective at each layer. When applied in a two-stage architecture (LCL-V2), we showed that our trained model is more effective in predicting V1 neural responses in the first layer than other architecturally-matched models, and achieves state-of-the-art quantitative predictions of V2 neural responses. Furthermore, we demonstrated that the LCL-V2 model can be used as a fixed front-end to train a supervised object recognition network (LCL-V2Net) that is significantly more robust to distribution shifts and aligned with human task behavior.

As described in the introduction, there is a substantial literature on using normative principles such as sparsity, coding efficiency, or temporal prediction to explain early visual properties. These have been mostly limited to early stages (up to and including cortical area V1), and those that have shown some qualitative success in reproducing V2-like selectivities, have again not scaled well beyond small image patches (Bányai et al., 2019; Hosoya and Hyvärinen, 2015; Rowekamp and Sharpee, 2017; Willmore et al., 2010) or have been restricted to texture images (Parthasarathy and Simoncelli, 2020). These are significant limitations, given recent work showing the importance of training on diverse natural image datasets for achieving strong biological alignment (Conwell et al., 2022).

End-to-end trained networks (both supervised and self-supervised) have provided strong accounts of neural responses in late stages of primate cortex as well as recognition behavior (Geirhos et al., 2021; Schrimpf et al., 2018; Zhuang et al., 2021). While unconstrained task-optimization of these models has been the standard for many years, recent efforts demonstrate that constraints on model capacity can lead to better alignment with aspects of biological representation. For example, (Nayebi et al., 2023a) show that self-supervised (contrastive) E2E training of shallow-networks account well for neurons in mouse visual cortex (due to the limited capacity nature of mouse visual cortex). In primate visual cortex, (Margalit et al., 2023) have shown that self-E2E objectives coupled with a layerwise spatial-smoothness regularizer over neural responses produce topographically-aligned models of both primary visual cortex and IT cortex. In contrast with these studies, we hypothesize that E2E objectives do not appropriately constrain intermediate representations, and that such constraints are better imposed locally, via per-layer objective functions that do not propagate gradients between layers.

In the machine learning literature, there have been multiple studies that use layerwise learning to train DNNs (Belilovsky et al., 2019; Löwe et al., 2019; Siddiqui et al., 2023; Xiong et al., 2020). These efforts have been primarily focused on demonstrating that layerwise objectives can approximate the performance of corresponding end-to-end backpropagation when evaluating

on downstream visual tasks. A few studies (Halvagal and Zenke, 2023; Illing et al., 2021) emphasize the biological plausibility of layerwise learning from a theoretical perspective, but were not scaled to large-scale training datasets. More importantly, previous studies have not assessed whether these biologically-plausible layerwise learning objectives result in more biologically-aligned networks. Here, we’ve shown that previous layerwise learning approaches (layerwise Barlow (Siddiqui et al., 2023) and LPL (Halvagal and Zenke, 2023)), do not offer the same benefits as our framework. We further demonstrate that a canonical feature-contrastive objective (same as (Siddiqui et al., 2023)) only leads to improved biological alignment *when the objective is complexity-matched with the corresponding computational capacity of the model stage* (See Sec. 2.4.3).

(Dapello et al., 2020) have shown that a biologically-inspired V1 stage greatly improves the adversarial robustness of trained networks. But, a closer analysis of their results (along with the evaluations presented here) demonstrate that a V1-like front end does not in fact provide significantly better robustness to image distribution shifts or alignment with human object recognition behavior. On the other hand, there have been a number of published networks that demonstrate greatly improved behavioral error consistency (Sec. 4.4.5) at the expense of worse alignment with biological neural responses. These include model scaling (Dehghani et al., 2023), training with natural video datasets, (Parthasarathy et al., 2023a), and use of alternative training paradigms (Jaini et al., 2023; Radford et al., 2021; Xie et al., 2021b). Our work provides a step towards resolving this discrepancy, by providing an improved model of early visual areas (specifically area V2) that is accompanied by a corresponding improvement in model generalization and behavioral alignment.

4.6 LIMITATIONS AND FUTURE WORK

We briefly describe some of the limitations of this work and opportunities for future work. First, while we have explored the benefits of layerwise training in a two-stage model, there is opportunity to explore extensions to learning of stages deeper in the visual hierarchy. In order to appropriately scale both the feature complexity (image field-of-view) and spatial deformation strength effectively in more layers, we will need to leverage either larger, scene-level images (Xie et al., 2021a) or natural video datasets (Gordon et al., 2020; Parthasarathy et al., 2023a). For many years, improvements in task performance of deep networks was correlated with improved predictions of neurons in late visual areas, but the most recent task-optimized networks have shown a degradation of neural predictivity (Linsley et al., 2023). As a result, there is potential for the extension of our work in deeper layers to address these inconsistencies.

Second, the examples and comparisons in this article focused on a single network architecture (AlexNet), but we believe the complexity-matching property is of broader applicability. Extending it, however, will require development of more quantitative measures of 1) image content complexity that can be used in place of the current ‘patch-size’ proxy, and 2) computational capacity of a neural network stage, depending on the specific computations (e.g., number and size of filter kernels, choice of nonlinearity, etc). This is especially important for extending to recent alternative architectures such as residual networks (He et al., 2016) or transformers (Dosovitskiy et al., 2020). These networks contain additional computational elements such as “skip connections” and spatially-global computations, making it difficult to appropriately complexity-match an objective with a given layer in these architectures.

Finally, from a neuroscience perspective, we see a number of opportunities for enhancing and extending the current framework. On the theoretical side, although our layerwise learning is arguably more biologically-plausible than standard supervised, self-supervised, or adversarial training, it still relies on implausible within-layer dependencies. We hope to leverage recently

developed methods (e.g., (Illing et al., 2021)) to bridge this gap in biological plausibility. Experimentally, the current results are also limited to a few evaluations on a dataset of about 100 neurons and their responses to naturalistic texture and spectrally matched noise images (Freeman et al., 2013; Ziemba et al., 2016). While this dataset is informative, it will be important to compare the LCL-V2 model to V2 responses on a wider selection of stimuli. Perhaps more exciting is the possibility that studying the structure and response selectivities of the learned LCL-V2 model will reveal new organizing principles for understanding the mysteries of primate visual area V2 and beyond.

5 | ON THE LIMITATIONS OF CURRENT NEURAL BENCHMARKING

5.1 OVERVIEW

Until now, we have focused primarily on normative theories and methods for learning behaviorally and neurally-aligned visual representations. In this brief chapter, we shift our focus to understanding whether current neural datasets and evaluation protocols are limiting our ability to thoroughly assess neural alignment of computational models.

5.2 SPATIALLY-RESOLVED NEURAL DATASETS

In Sec. 2.7, we highlighted a potential inconsistency: that our video pretrained model, VITO improves predictions of human visual perception (or behavior), but does not have internal representations that align better with ventral stream responses (even in late stages like IT cortex). Yet, the current datasets in BrainScore, primarily focus on single neuron responses to centered object images, similar to the training data of these networks. However, an important feature of VITO (as compared with standard supervised networks) is the learned gain-modulation (attention) network, which modulates responses across space. As a result, it is natural to ask whether the VITO responses are more aligned with ‘spatially-resolved’ neural responses across images.

In fact, this kind of neural data has been recently collected for IT neurons (Arcaro et al., 2020), and has been used in recent deep network model benchmarking Linsley et al. (2023). The basic

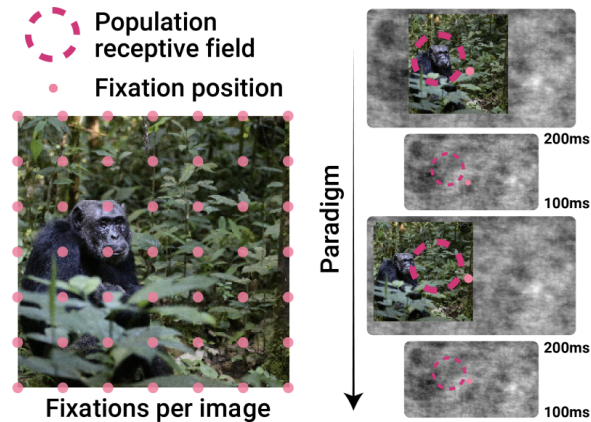


Figure 5.1: Depiction of the experimental protocol from Arcaro et al. (2020). We focus on a subset of the data, recording neurons in medial-lateral (ML) IT (in one monkey) in a spatially-resolved manner. The monkey was presented each image multiple times for 200ms each presentation. Images were positioned differently each time (in a 17x17 grid) to measure neural responses to every part of the image. Adapted from Linsley et al. (2023).

experiment is depicted in Fig. 5.1. We use a subset of the full dataset (neural responses from primate medial-lateral (ML) IT of one monkey). These responses were recorded to flashed images positioned at different locations (relative to monkey fixation), creating a 17×17 grid of IT responses across a given image. A total of 14 images were shown, for a total of 4046 responses per neuron. 31 neurons were recorded. As described in Linsley et al. (2023), responses are binned in 40ms bins between 50-250ms post stimulus presentation. Within each bin, a noise ceiling is calculated for each neuron (maximum correlation achievable between any two neurons within that time interval). The responses for comparison are chosen from the time bins with the highest average noise ceiling.

For these same images, we obtain spatial feature map responses (up-sampled to 17×17) for our VITO model and standard supervised ResNet-50 model. For both models, we choose the final feature map in the encoder backbone. Given model responses to each of the 4046 spatial locations (across the 14 images), we follow the evaluation protocol in Linsley et al. (2023), which

again leverages the standard PLS regression approach from BrainScore to learn a linear mapping between model and IT responses.

The regression is trained with leave-one-out cross validation, training on 13 images (3757 responses) and testing on the held out image (289 responses). We find that the standard ResNet-50 network has a median explained variance over the 31 neurons of approximately 42%. Surprisingly, VITO far outperforms this with 52% explained variance. This suggests, that by only evaluating on standard IT datasets (recording single neural responses to single object images at a fixed position), we may be missing components of neural responses that can in fact better discriminate between existing models.

While this is one particular experiment, we hope that this result provides a reminder for the community that different DNNs may capture different aspects of cortical responses that are *highly dependent on the stimuli that are used*. It is worth noting that in this vain, there have also been recent studies showing that neural alignment of DNNs become far worse when specifically measured on out-of-distribution images (Bagus et al., 2022; Ren and Bashivan, 2023).

5.3 A SPARSE REGRESSION APPROACH TO MEASURING NEURAL ALIGNMENT

5.3.1 MOTIVATION

We next study another potential confounding factor in standard linear regression-based approaches for measuring neural alignment of high-dimensional models. Assume we are attempting to predict the activity of a single neuron. This neuron’s response lies in a specific low-dimensional space, constrained by it’s receptive field selectivity. An ideal model will be one which has a single model unit with the same feature selectivity and response to stimuli. However, it is not plausible to expect this of any model and thus we allow for weighting multiple model units (via a learned

regression) to fit the given neuron. In an unconstrained linear regression (or even PLS regression); however, we generally assume that *all model units can be used to fit any given biological neuron*. Given a low-dimensional model, this would not be an issue. However, with sufficiently large numbers of *random* model bases (units), it can still be possible to learn a weighting to perfectly predict a given biological neuron. This critically assumes that the random basis spans the space defined by the low-dimensional neuron feature selectivity. We hypothesize that in practice this assumption is often valid, especially when we 1) leverage extremely high-dimensional deep networks (thousands of model units) and 2) predict neural responses in early visual areas (V1) with simple feature selectivities (closer to being linear functions of the stimulus). While this theory may be demonstrated mathematically and through simulation, it is more instructive to analyze a real scenario. Specifically, we evaluate the V1 (dataset from [Ziemba et al. \(2016\)](#)) BrainScore explained variance for a randomly initialized ResNet-50 network and a trained, supervised ResNet-50 network.

Model	V1 BrainScore Explained Variance
ResNet-50 Untrained 1	0.263
ResNet-50 Untrained 2	0.262
ResNet-50 Untrained 3	0.275
ResNet-50 Supervised	0.278

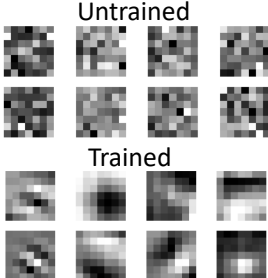


Figure 5.2: Right: We show example filters from the first convolutional layer of the randomly initialized ResNet-50 and trained ResNet-50. Left: Across many random initializations, explained variance is nearly the same as for the trained ResNet-50.

We find that randomly initialized networks (over multiple initializations) perform nearly the same as the trained network in V1 response explained variance, despite the trained network clearly having filters more aligned with the expected selectivity properties of V1 neurons.

Clearly, we would like an evaluation protocol that is able to distinguish between high-dimensional random models and those with learned feature selectivities that align with neuronal tuning.

One option is to simply measure how well the single neuron tuning properties match between the artificial and biological neurons. This approach has been taken by [Marques et al. \(2021\)](#); however, it is difficult to enumerate all of the desired tuning properties, especially in cortical areas beyond V1.

5.3.2 METHOD

We propose an alternative simple modification of the existing regression-based BrainScore procedures to address the aforementioned issues. Specifically, we suggest using a sparse regression method to enforce that models use a small number of units to predict the response of a given neuron. Conceptually, a high performing model will be one that tiles the dimensions of feature-selectivity such that *a given biological neuron (which lies in this space) can be locally interpolated by a small number of model units.*

To be concrete, assume for image i , we are given a d dimensional model response vector: $x_i \in \mathcal{R}^{d \times 1}$, and a single scalar output neuron response y_i , we minimize the following objective:

$$\min_{\beta} \sum_{i=1}^N (y_i - x_i^\top \beta)^2 + \lambda * \frac{\|\beta\|_1}{\|\beta\|_2} \quad (5.1)$$

where N is the total number of training images and λ is a parameter which controls the strength of the sparsity regularizer. Instead of the standard l_1 lasso penalty, we utilize the ratio of l_1 to l_2 norms to enforce sparsity on the regression weights as in [Hoyer \(2004\)](#). While this makes the optimization problem non-convex, the benefits of such a regularizer are that it is scale-invariant and thus does not force all weights to be small (as is common in standard lasso regression). Instead of selecting a specific λ , we sweep λ over a range for each neuron, such that we obtain a model prediction over a range of sparsity constraints. For each λ value, we count the “number of model units used” by finding the number of non-zero weights (weights with absolute magnitude $> 1e^{-6}$). We repeat this procedure for each output neuron independently.

To verify our modified sparse regression method, we evaluate three models that we hypothesize should have very different properties in terms of their ability to locally-interpolate V1 selectivities:

1. **V1-Steerable Pyramid (V1-SteerPyr)**: this is the same model used in Chapter 4, (details in Sec. C.1). On account of using steerable Gaussian derivative basis filters (Simoncelli and Freeman, 1995), this model tiles the four dimensional (orientation, scale, x (spatial), y (spatial)) space, and thus is particularly suited to interpolating V1 responses with small numbers of model units.
2. **ResNet50 (random)**: the ResNet-50 model initialized with random weights.
3. **ResNet50 (supervised)**: the ResNet-50 model trained with object recognition supervision.

As done in the BrainScore method, for both ResNet models, we select the best layer for predicting V1 responses (testing all layers).

5.3.3 PRELIMINARY RESULTS

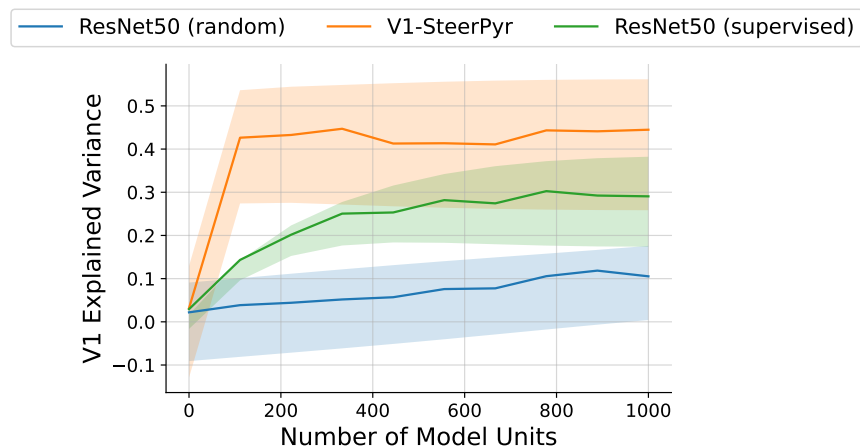


Figure 5.3: For each model, at a given λ , we compute the mean explained variance (y-axis), standard deviation across neurons, and number of model units used for the prediction (x-axis).

We use the same V1 dataset as before (Ziemba et al., 2016) which consists of 102 V1 neurons. For each model, at a given λ , we compute the mean explained variance, standard deviation across neurons, and number of model units used for the prediction. For each model, we then plot the mean explained variance (with standard deviation bands) vs. number of model units used (Fig. 5.3). These curves describe *how efficiently a given model interpolates V1 neural responses*. Unlike the small difference between the untrained and trained ResNet models using the PLS regression method (Fig. 5.2), we now see a drastic difference between the two models, as the trained model, captures significantly more variance when using limited numbers of model responses. On the other hand, the random model improves linearly as individual model units are added and performs poorly even with 1000 units. Strikingly, the V1-SteerPyr model is far more efficient than either DNN and reaches nearly 40% explained variance with approximately 100 model units. This result confirms our hypothesis that this model does in fact interpolate well in the the space of selectivities commonly associated with V1 neurons.

Due to instabilities in the optimization procedure and complexities of choosing λ for each neuron and dataset, we have yet to be able to study the impact of the sparse regression on discriminating between models of other cortical areas. However, we hope that future work can build on this to develop model-brain alignment metrics that better correlates with a model's ability to *both* predict cortical responses and capture their fundamental dimensions of tuning or selectivity.

6 | DISCUSSION

6.1 TEMPORALLY-INFORMED MODELS OF IMAGE PERCEPTION

There has long been evidence that temporal learning has a large impact on human object perception. As described in the introduction, there is a long line of behavioral and psychological evidence for the fact that infant object perception is driven by learning how objects move (Kellman and Spelke, 1983; Spelke, 1990; Spelke and Kinzler, 2007). More surprisingly, even in adults it has been shown that learning from specific spatiotemporal experiences can alter position-invariant recognition (Cox et al., 2005). Given this, it is particularly puzzling that while there have been many successful examples of learning computer vision models of motion and video understanding (Dave et al., 2022; Dorkenwald et al., 2022; Qian et al., 2021; Recasens et al., 2021; Sermanet et al., 2018), it has been extremely difficult to learn spatial representations from natural video that are competitive with standard image-based training.

In the first chapter of this dissertation, we verify that many related prior works, severely under-perform standard ImageNet pretraining on general spatial understanding benchmarks (See Table 2.1). We then propose VITO, a simple method for learning general, robust, and human-aligned spatial visual representations from natural temporally-evolving scenes.

One of the key features of our method is the use of a more diverse curated video dataset (VideoNet). Concurrently, there have been efforts in the psychology community to collect very carefully controlled video datasets that capture infant viewing experience (Sullivan et al., 2021),

with the hope of training models on more naturalistic data. However, while these efforts are potentially useful for many studies, it is hard to obtain the diversity of content in these controlled settings that we have in our dataset- a feature which seems to have a large impact on generalization and human-alignment of models.

The second key feature of our method is the use of a novel learned attention (or spatial-gain modulation) architecture to discover temporally co-occurring content via a self-supervised contrastive loss. This discovery is interesting as gain-modulation is a pervasive canonical computation studied in the visual neuroscience literature (Carandini and Heeger, 1994; Lee et al., 2012; Ohshiro et al., 2011; Reynolds and Heeger, 2009; Treue and Trujillo, 1999). While our implementation is not currently tied to any biological predictions, we believe it may be interesting to revisit this in future work. In a similar direction, while our work clearly demonstrates the impact of learning from natural temporal deformations on achieving more robust and human-aligned visual representations, there are still many open questions on how to link our method and architecture better to the biology. Our approach leverages end-to-end self-supervised learning, so there is a natural question of how to achieve similar results with layerwise or local learning (similar to our efforts in Chapter 4). While experimental evidence has shown that neurons in the early ventral stream may be implicitly optimizing for things such as temporal straightness or predictability (Hénaff et al., 2021a; Wiskott and Sejnowski, 2002), there have yet to be convincing demonstrations that layerwise or local objectives can learn hierarchical representations from natural video data.

6.2 LAYERWISE COMPLEXITY-MATCHED LEARNING

Chapters 3 and 4 are primarily concerned with the problem of how to learn hierarchical visual representations that better align with neural representations by leveraging more biologically-plausible layerwise objectives. Focusing on Chapter 4, we demonstrate that a self-supervised

feature-contrastive layerwise learning paradigm can indeed lead to learned representations that are more predictive of neural responses in early visual areas (V1 and V2). Our bottom-up self-supervised training methodology operates independently on successive layers to maximize feature similarity between pairs of spatially-restricted locally-deformed natural image patches, while decorrelating features across patches sampled from other images. The extent of spatial restriction and the amplitude deformation are adjusted proportionally to receptive field sizes in each layer, thus matching the complexity of content to the computational capacity at each stage of processing. Conceptually, we see this method as a principled and general hypothesis for how “selectivity” and “tolerance” for visual features are developed in a hierarchical representation. While these concepts have mostly been discussed within specific settings such as late-stage object representation (Rust and DiCarlo, 2010) and more recently mid-visual texture representation (Ziemba et al., 2016), the two terms in our objective can be seen as optimizing for selectivity (decorrelation term) and tolerance (invariance term) over image content and deformations that progressively become more complex through stages of the visual hierarchy. As a result, we hope that a generalization of our method may provide a way to probe and perhaps discover selectivities that emerge in intermediate model (and potentially neural) representations. Briefly, it is worth noting that while there has obviously been extensive work in the machine learning and neuroscience communities in the space of local and layerwise learning, our results stand in contrast with the still popular notion that end-to-end optimization may be sufficient for learning biologically-aligned network representations (Yamins et al., 2014; Zhuang et al., 2021).

The key innovation of our method lies in the conceptual proposal and implementation of “complexity-matching” objective functions with the model capacity at a given stage of computation. Although we do not provide a method for quantitatively characterizing the complexity of an objective or the exact computational capacity of a given network architecture, we believe our results signal that these lines of theoretical research are worth pursuing. This is especially important as we demonstrate in Sec. 4.4.4, how a mismatch in these parameters can lead to

drastically less biologically-aligned learned representations. It is additionally possible that further work along these lines may provide more general insights into why many of the learning approaches based on normative theories (Atick and Redlich, 1990; Barlow et al., 1961; Bell and Sejnowski, 1997; Karklin and Lewicki, 2009; Karklin and Simoncelli, 2011; Olshausen and Field, 1996) have failed to generalize well beyond a single layer (V1-like) network stage.

Finally, another potentially useful interpretation of our layerwise learning approach comes from the perspective of model overfitting. Given the enormous capacity of most DNNs, their failure to capture basic aspects of human perception (Berardino et al., 2017; Feather et al., 2023; Szegedy et al., 2013) suggests that the models overfit on their training task, learning uninterpretable or unreliable features. To date, the best methods for reducing this overfitting issue have been to either use fixed (not-learned) models (Berardino et al., 2017; Dapello et al., 2020) or computationally expensive adversarial training procedures (Madry et al., 2017). Our work suggests that there may be another more biologically-plausible alternative to reducing this overfitting problem (imposing layerwise constraints), and that this may enable future development of more neurally-aligned DNNs.

6.3 EVALUATING DNN ALIGNMENT WITH NEURONS AND BEHAVIOR

While this dissertation is primarily focused on normative theories and implementations of methods for learning visual representations, a core underlying problem we deal with is how to evaluate the alignment of DNN models with both neural responses and behavior.

We choose to focus primarily on behavioral evaluations that indicate how well models capture human object recognition behavior (Geirhos et al., 2021) and human perception of object saliency (Fel et al., 2022; Linsley et al., 2018). We see these as relevant benchmarks given the fact that these networks are known to succeed at tasks centered on object discrimination. We find that both of our approaches to improving DNN alignment (video pretraining and layerwise learning) succeed

in these benchmarks, but acknowledge that this is still a very limited set of evaluations. In future work, we hope to evaluate our models on a wider range of behavioral and perceptual metrics such as eigendistortions (Berardino et al., 2017), metamers (Feather et al., 2023), representational similarity on different types of visual tasks (Muttenthaler et al., 2022) etc.

Regarding neural evaluations, we acknowledge the relatively limited nature of our evaluations. Although the BrainScore (Schrimpf et al., 2018) regression-based benchmark is a current standard, it is limited in ways that we probe in Chapter 5. We hope these analyses help in developing the benchmark further to be more robust and comprehensive.

6.4 CONCLUDING REMARKS

This dissertation proposes multiple novel self-supervised learning methods to train neural network models that better align with both human behavior and early visual neurons. In Chapter 2, we propose VITO, a contrastive video-pretraining method, that improves drastically on prior work to learn general, robust, and more human-aligned representations from natural video data. We make an observation; however, that end-to-end task optimized models (even those that better predict human visual task performance), may not adequately constrain internal representations in intermediate model stages. As a result, in Chapters 3 and 4, we show that we can improve models learned models of early vision by imposing layerwise constraints through self-supervised complexity-matched objective functions. We achieve state-of-the-art predictions of cortical responses in area V2, with the potential for extending these ideas to learning more stages. Finally, we provide some preliminary analyses probing the limitations of current neural benchmarking evaluations. In sum, this work lays the foundation for future research in using learned DNNs to reveal new organizing principles for understanding the mysteries of biological vision.

A | SELF-SUPERVISED VIDEO PRETRAINING

YIELDS ROBUST AND MORE

HUMAN-ALIGNED REPRESENTATIONS

A.1 APPENDIX: IMPLEMENTATION DETAILS

A.1.1 SELF-SUPERVISED LEARNING

Data pre-processing. Each frame is randomly augmented by composing the following operations, each applied with a given probability:

1. random cropping: a random patch of the image is selected, whose area is uniformly sampled in $[s \cdot \mathcal{A}, \mathcal{A}]$, where \mathcal{A} is the area of the original image, and whose aspect ratio is logarithmically sampled in $[3/4, 4/3]$. s is a scale hyper-parameter set to 0.08 when learning from ImageNet, and 0.4 when learning from videos. Regardless, the patch is then resized to 224×224 pixels using bicubic interpolation;
2. horizontal flipping;
3. color jittering: the brightness, contrast, saturation and hue are shifted by a uniformly distributed offset;

4. color dropping: the RGB image is replaced by its grey-scale values;
5. gaussian blurring with a 23×23 square kernel and a standard deviation uniformly sampled from $[0.1, 2.0]$;
6. solarization: a point-wise color transformation $x \mapsto x \cdot \mathbb{1}_{x < 0.5} + (1 - x) \cdot \mathbb{1}_{x \geq 0.5}$ with pixels x in $[0, 1]$.

The augmented frames v^1 and v^2 result from augmentations sampled from distributions \mathcal{A}_1 and \mathcal{A}_2 respectively. These distributions apply the primitives described above with different probabilities, and different magnitudes. The following table specifies these parameters for the BYOL framework (Grill et al., 2020), which we adopt without modification. When learning from three views, we use the distribution \mathcal{A}_1 to generate the third view.

Parameter	\mathcal{A}_1	\mathcal{A}_2
Random crop probability		1.0
Flip probability		0.5
Color jittering probability		0.8
Color dropping probability		0.2
Brightness adjustment max		0.4
Contrast adjustment max		0.4
Saturation adjustment max		0.2
Hue adjustment max		0.1
Gaussian blurring probability	1.0	0.1
Solarization probability	0.0	0.2

Optimization. We pretrain ResNet-50 using the LARS optimizer (You et al., 2017) with a batch size of 4096 split across 128 Cloud TPU v3 workers. We adopt the optimization details of BYOL, scaling the learning rate linearly with the batch size and decaying it according to a cosine schedule. The base learning rate is 0.3 and the weight decay is 10^{-6} .

A.1.2 TRANSFER TO PASCAL AND ADE20K SEMANTIC SEGMENTATION

Architecture. We evaluate ResNet models by attaching a fully-convolutional network (FCN, Long et al. (2015)) and fine-tuning end-to-end, following He et al. (2020). When evaluating Swin transformers we instead use the UperNet segmentation architecture (Xiao et al., 2018).

Data pre-processing. During training, images are randomly flipped and scaled by a factor in $[0.5, 2.0]$. Training and testing are performed with 512×512 -resolution images. When fine-tuning on ADE20K, we additionally use photometric transformations from the mmseg¹ codebase.

Optimization. We fine-tune for 45 epochs on the PASCAL train_aug2012 set or 60 epochs on the ADE20K train set. We use stochastic gradient descent with a batch size of 16 and weight decay of 0.005. The learning rate is initially set to 0.04 and decayed exponentially with a factor of 0.9^n where n is the iteration number. When fine-tuning external models, we sweep over the base learning rate and weight decay and report their performance given the optimal configuration. In all cases we report mIoU on the val set averaged across 5 runs.

A.1.3 TRANSFER TO COCO AND LVIS OBJECT DETECTION

Architecture. We evaluate both ResNet and Swin transformers using the FCOS* architecture, following Hénaff et al. (2022). FCOS* is the implementation of a single-stage detector based on FCOS (Tian et al., 2019), and improved with the collection of techniques from Wu et al. (2020), Zhang et al. (2020a), and Feng et al. (2021), full details can be found in Hénaff et al. (2022).

Data pre-processing. The target resolution is 800×1024 . During testing, an image is resized by a factor s while preserving the aspect ratio, such that it is tightly contained inside the target resolution, and then padded. When fine-tuning, the image is rescaled by a factor of $u \cdot s$ where u is uniformly sampled in $[0.8, 1.25]$, and is then cropped or padded to the target resolution.

¹<https://github.com/open-mmlab/mmdetection>

Optimization The network is fine-tuned for 30 epochs on the COCO train2017 set or the LVIS v1_train set. We use AdamW (Loshchilov and Hutter, 2019) with weight decay 10^{-4} , base learning rate of 10^{-3} , and batch size 128 split across 16 workers. The learning rate rises linearly for $\frac{1}{4}$ of an epoch, and is dropped twice by a factor of 10, after $\frac{2}{3}$ and $\frac{8}{9}$ of the total training time. We report mAP on the COCO val2017 set and the LVIS v1_val set, averaged across 5 runs.

A.1.4 TRANSFER TO DAVIS VIDEO SEGMENTATION

As a further test of scene understanding, we assess whether learned representations can continue to recognize parts of an object as they evolve over time. Video object segmentation, specifically in its semi-supervised setting, captures this ability, which we evaluate on the DAVIS'17 benchmark. Having evaluated a learned representation on a video independently across frames, we segment these features with nearest neighbor matching from frame to frame, given a segmentation of the first frame. In this way, the segmentation is propagated according to the similarity of the representation across space and time. We reuse the segmentation procedure from Xu and Wang (2021) without modification, and report region (\mathcal{J}) and boundary quality (\mathcal{F}).

A.1.5 TRANSFER TO UCF-101 ACTION RECOGNITION

We evaluate action recognition classification on the UCF101 dataset (Soomro et al., 2012). We follow the procedure for finetuning used in (Wu and Wang, 2021) which is based on (Morgado et al., 2021). We utilize clips of 2 seconds in length at 12fps. Each frame is processed by the ResNet-50 backbone. Clip representations are obtained by one of three methods for temporal integration:

1. Average pooling is the standard baseline, producing a 2048-d vector output for a clip which is then fed to and one fully connected (2048×101) layer for predicting the action class.
2. MS avg-pool: we pool the block3 representations (1024-d) over the two subclips of 1s each

that make up the larger clip. This is done because the features at this scale have smaller receptive fields and are selective for less complex content. Then we concatenate the two (1024-d) vectors with the average pooled feature from the block4 output to get a single 4096-d vector for each clip that again is fed through a fully-connected layer to predict the action class. By concatenating the two subclip representations, the fully-connected layer can in fact compute complex temporal relationships such as differences etc. along with the final layer’s invariant representation that is pooled for the full clip.

3. MS temp-attn: We perform the same methodology as above for integrating multiple scales, but replace the average pooling over time with an attention pooling layer. Given representations for an L-frame clip $z \in \mathbb{R}^{B \times C \times L}$ at a given scale, we compute temporal attention weights $w_t \in \mathbb{R}^L$ where $w = f(z)$. We choose f to be $\tanh(Wz)$ where $W \in \mathbb{R}^{C \times 1}$ is a linear weighting of channels. Finally the pooled representation $v = \sum_L w_t \cdot z$

We show results using method 1 in the main text and demonstrate the improvements from methods 2 and 3 in Appendix Table A.5. 10 clips are sampled from each video and the predictions of the clips are averaged for the final results. We fine-tune for 16 epochs using the ADAM optimizer with a multi-step LR decay schedule at epochs 6, 10, and 14. The initial learning rate is set to 0.0001. The implementation is adopted from <https://github.com/facebookresearch/AVID-CMA>.

A.1.6 TRANSFER TO IMAGENET CLASSIFICATION

For all models we freeze the ResNet-50 encoder (which outputs 2048-d embeddings). We then train a linear head to classify the 1000 categories in the ImageNet training set using the standard split. To train the classifier, we use the SGD optimizer with nesterov momentum and momentum parameter equal to 0.9. We use weight-decay of 0 and sweep the learning rate for each model in the range [0.4, 0.3, 0.2, 0.1, 0.05] and pick the best classifier based on ImageNet validation accuracy.

A.1.7 TRANSFER TO OUT-OF-DISTRIBUTION EVALUATIONS

For all OOD evaluations, we evaluate on datasets that utilize all (or subsets) of the ImageNet validation set. Therefore, for these evaluations we use the pre-trained encoder and linear classifier (trained as in Sec A.1.6). We freeze the encoder and linear classification head and evaluate task performance on images from either the ImageNet-A, ImageNet-vid-robust, and Imagenet-3DCC datasets. For ImageNet-A and ImageNet-vid-robust, we use the evaluation code and method from (Taori et al., 2020).

For ImageNet-3DCC, we do not use the entire corruption set because we wanted to specifically test models under the more natural 3-d corruptions. As is described in (Kar et al., 2022), the dataset can be broken down into two sets of corruptions: 3-d informed corruptions (using a depth model to generate natural corruptions informed by 3-d information) and standard 2-d noise and artifacts (like in ImageNet-C). For our experiments, we chose to evaluate specifically on the 3-d corruptions, which were found to induce larger robustness effects for evaluating standard networks (Kar et al., 2022). Nevertheless, we found similar results when evaluating robustness to 2-d noise and artifacts. All images from the following classes of corruptions were used for evaluation: far focus, near focus, fog, flash, xy motion blur, z motion blur, view jitter.

A.1.8 ALIGNMENT WITH HUMAN SALIENCY

Human saliency measurements are obtained from the ClickMe dataset. Alignment is measured as the Spearman rank correlation between model and human saliency averaged over the dataset, normalized by inter-rater alignment of humans.

A.1.9 HUMAN ERROR CONSISTENCY EVALUATION

We evaluate accuracy and human error consistency on shape-bias datasets using the code from <https://github.com/bethgelab/model-vs-human/tree/master>. We choose the subset

of images that remove textural cues in different ways (forcing humans and models to utilize global shape during discrimination): edge drawings, cue-conflict stimuli, graded low-pass filtering, and uniform gaussian noise. We report three metrics from (Geirhos et al., 2021):

1. Accuracy difference: measure of human vs model classification accuracy on each OOD dataset and then averaged.
2. Observed consistency: measures the fraction of samples for which humans and a model get the same sample either both right or both wrong.
3. Error consistency: Score that measures whether there is above-chance consistency. This is important because e.g. two decision makers with 95% accuracy each will have at least 90% observed consistency, even if their 5% errors occur on non-overlapping subsets of the test data (intuitively, they both get most images correct and thus observed overlap is high). Error consistency indicates whether the observed consistency is larger than what could have been expected given two independent binomial decision makers with matched accuracy (Geirhos et al., 2020b).

The mathematical details on each of these metrics are provided in (Geirhos et al., 2020b).

A.2 APPENDIX: ADDITIONAL RESULTS

A.2.1 SEMANTIC BINDING WITH CONTRASTIVE ATTENTION POOLING

The ablation study demonstrated that multi-scale attention improves the performance of VITO in semantic segmentation. To probe why this may be, we visualize and interpret the learned attention masks (Figure A.1). For simplicity, we only visualize the masks from the coarsest scale (output feature map), but the interpretation naturally extends to the multi-scale version as these masks are learned with independent attention modules.



Figure A.1: Example augmented frames with overlaid (resized) learned attention masks. Attention is computed from the output of the final block of the VITO trained ResNet-50. Crucially, the attention masks are computed independently, such that the attention module can only use spatial cues.

Because the attention masks are not computed jointly across each view, for a given video frame, the attention module must marginalize over the training data to make a statistical prediction: what should be attended to in the first view in order to minimize the contrastive loss across possible second views? Specifically, the attention must focus on content that is most likely to be stable across time while still being discriminative (or unique) relative to other frames from other videos. Different examples appear to trade-off these criteria differently, yet systematically. For example, in the third column of Figure A.1 even though the animated characters on the right side of both frames may be discriminative content, the attention module has learned to focus on the static picture on the left as it is the content that is most likely to be stable across time. For this pair of frames the prediction is correct—the attention disregards content that is changing too abruptly—despite not having access to motion cues. On the other hand, the example in the fourth column demonstrates a scenario where the model has attended to stable, but primarily discriminative content (the bird) rather than the background, which is also very stable but most likely less unique relative to other videos.

Even beyond the ability to localize stable, yet discriminative content, it seems that our method also enables “semantic binding” of visually different, but semantically related features. This can be seen in the first pair of frames, as the model has learned to associate an arm or elbow (in the

first frame) with the dumbbell (in the second frame), demonstrating an understanding that these two semantically related concepts co-occur and thus are predictive of one another given the right embedding.

Binding co-occurring features appears as an intuitive explanation for why these representations would perform well on semantic segmentation. It is particularly interesting that training end-to-end with a standard contrastive loss can produce complex behavior reminiscent of the DINO approach (Caron et al., 2021) even though we use a single, two-layer MLP attention module as opposed to large-scale transformer architectures which use attention throughout the network.

A.2.2 ABLATING THE COMPONENTS OF VITO

In Figure A.2 we demonstrate on an example scene understanding task (PASCAL) how VITO is impacted by crop-scale, clip length, and the type of attention pooling used (or not used). In Figure A.3 we additionally do a deeper analysis of the temporal sampling scheme and demonstrate that our choice performs best across tasks and is arguably the most natural.

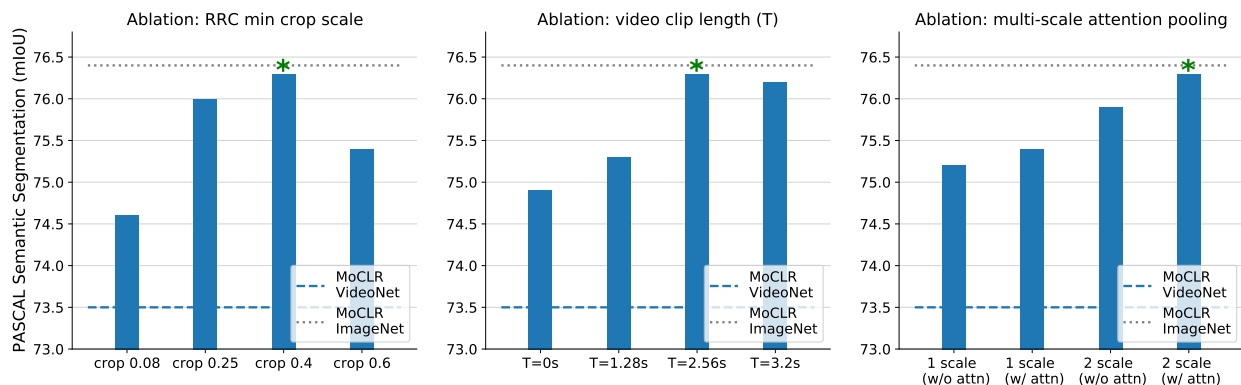


Figure A.2: Effects of crop scale, natural augmentations, and multi-scale attention on representation quality. All ablations are performed relative to VITO’s configuration (denoted by a green asterisk) which uses 2-scale attention pooling, a less aggressive crop scale of 40%, and natural augmentations uniformly sampled in a window of length $T = 2.56s$. We also compare to our baseline MoCLR model trained on single frames, either from ImageNet (dotted gray line) or VideoNet (dashed blue line). All models are evaluated by transferring to PASCAL semantic segmentation.

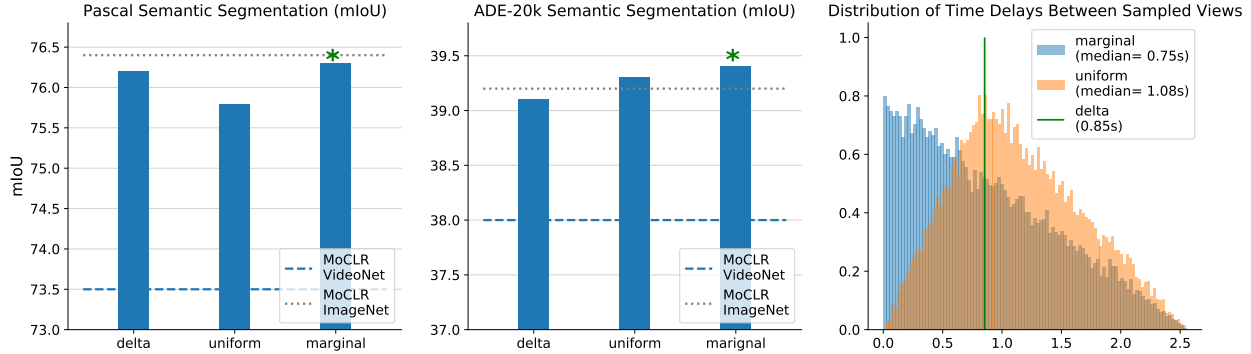


Figure A.3: Ablating different temporal sampling schemes. *Delta* refers to fixed time sampling between frames as in [Gordon et al. \(2020\)](#). *Uniform* refers to chunking time into non-overlapping blocks and uniformly sampling within each chunk as in [Xu and Wang \(2021\)](#). *Marginal* sampling (ours) refers to simple uniform sampling from the full video clip of length $T = 2.56s$. First two panels show that marginal sampling is best overall across transfer to PASCAL and ADE20K. Third panel shows the distribution of absolute time-differences between any two pairs of frames under each sampling scheme (assuming 3 views are sampled per clip). Our marginal sampling scheme is arguably the most natural as the mode of the distribution is at 0, meaning that it is not biased to over-represent any specific time difference (similarly to the random-resized crop operation in space).

A.2.3 DATASET AND METHOD ABLATIONS

In [Table A.1](#) we show that both our learning objective VITO and choice of dataset, VideoNet, are important for achieving top performance. However, these results also show that we can outperform existing video pretraining even when using standard datasets like Audioset and YT8M. In addition, by comparing to MoCLR trained on JFT-300M, we demonstrate the benefits of our method are not the result of simply having more frames of training data.

A.2.4 SCALING ARCHITECTURES

Here we demonstrate that VITO scales effectively to more powerful Swin transformer architectures. Results on scene understanding benchmarks improve greatly over ResNet-50 models and are competitive with specialized fine-grained scene understanding models from recent literature ([Hénaff et al., 2021b](#)). See [Table. A.2](#)

Pretraining	Dataset	Epochs	Semantic segmentation		Object detection	
			PASCAL	ADE20K	COCO	LVIS
MoCLR	VideoNet	200	72.8	37.5	42.6	24.6
VITO	YT8M	200	71.8	37.8	42.7	24.6
VITO	AudioSet	200	73.6	38.5	43.2	25.0
VITO	VideoNet	200	75.5	39.2	43.6	25.6
MoCLR	JFT-300M	200	74.3	38.7	43.2	25.4

Table A.1: VITO dataset and method ablations. We compare the baseline method MoCLR trained on VideoNet to demonstrate the impact of our methodology. VITO on VideoNet performs significantly better due to the methodological improvements (attention pooling, adaptation of spatial and temporal augmentations). We also evaluate VITO on traditional video datasets such as YT8M and AudioSet. We note that these numbers still greatly outperform prior video pretraining (See Table A.3. However the impact of the VideoNet dataset is clear as the best model is VITO trained on VideoNet. Finally, we show that VideoNet *does not* simply provide benefits due to increased number of total frames vs. ImageNet. In fact, MoCLR trained on JFT-300M has an order of magnitude more frames and yet still underforms.

Pretraining	Dataset	Backbone	Semantic segmentation		Object detection	
			PASCAL	ADE20K	COCO	LVIS
VITO	VideoNet	R50	76.3	39.4	44.0	25.7
MoCLR	VideoNet	Swin-S	78.6	43.7	48.4	32.7
VITO	VideoNet	Swin-S	81.3	46.1	49.8	33.5
Detcon _B	ImageNet	Swin-S	81.4	46.1	50.4	33.1

Table A.2: VITO scales to larger model architectures (Swin-S), improving performance compared to the ResNet-50 baseline and remaining competitive with a strong ImageNet pretrained baseline (Detcon) from [Hénaff et al. \(2021b\)](#).

A.2.5 COMPARISONS ON ADDITIONAL SCENE UNDERSTANDING TASKS

VITO outperforms all prior video pretraining (of image representations) on scene understanding tasks. In addition to the evaluations in the main text, we add PASCAL segmentation, LVIS object detection, ImageNet-1K classification. VITO remains highly competitive with the best ImageNet pretraining on these tasks. (See Table A.3).

A.2.6 COMPARISON TO IMAGE PRETRAINING ON VIDEO-BASED TASKS

Here we demonstrate more thoroughly that compared with image pretraining methods (image backbones), we perform significantly better on video-level tasks. On both DAVIS segmenta-

Video Pretraining	Dataset	Semantic segmentation		Object detection		Classif.
		PASCAL	ADE20K	COCO	LVIS	IN-1K
Random Init		53.0	27.9	39.0	21.1	-
<i>Methods pretraining on video datasets</i>						
R3M (Nair et al., 2022)	-	-	-	-	-	13.3
VFS (Xu and Wang, 2021)	K400	63.9	31.4	41.6	23.2	-
VIVI (Tschannen et al., 2020)	YT8M	65.8	34.2	41.3	23.2	62.6
VINCE (Gordon et al., 2020)	R2V2	69.0	35.7	42.4	24.4	54.4
CycleContrast (Wu and Wang, 2021)	R2V2	69.2	35.6	42.8	24.5	55.6
MMV TSM (Alayrac et al., 2020)	AS + HT	70.6	32.5	41.3	24.2	51.4
VITO	VidNet	76.3	39.4	44.0	25.7	66.2
<i>Methods pretraining on ImageNet</i>						
Supervised	IN-1K	71.3	33.5	44.2	25.2	76.1
BYOL (Grill et al., 2020)	IN-1K	76.1	38.8	43.7	25.5	-
MoCLR (Tian et al., 2021)	IN-1K	76.4	39.2	43.9	25.8	71.4
DINO (Caron et al., 2021)	IN-1K	76.1	39.0	44.3	26.4	75.3

Table A.3: Image and pretraining evaluated on object-detection, semantic segmentation, and ImageNet-1K classification.

tion (Table A.4) and UCF-101 action recognition (Table A.5), VITO outperforms strong ImageNet trained baselines and methods pretrained on video datasets.

Pretraining	Dataset	\mathcal{J}_m	\mathcal{F}_m
<i>ImageNet pretraining</i>			
Supervised	ImageNet	63.7	68.4
MoCo (He et al., 2020)	ImageNet	63.2	67.6
DetCon _B (Hénaff et al., 2021b)	ImageNet	63.1	66.4
MoCLR (Tian et al., 2021)	ImageNet	63.1	67.8
BYOL (Grill et al., 2020)	ImageNet	63.8	69.4
<i>Video pretraining</i>			
VINCE (Gordon et al., 2020)	Kinetics	63.4	67.8
TimeCycle (Wang et al., 2019)	VLOG	41.9	39.4
UVC (Li et al., 2019)	Kinetics	54.5	58.1
CRW (Jabri et al., 2020)	K400	64.8	70.2
VFS (Xu and Wang, 2021)	K400	65.3	70.2
VITO	VideoNet	65.5	70.8

Table A.4: VITO significantly outperforms all image-pretraining baselines on DAVIS 2017 video segmentation. VITO also outperforms many recent successful video pretraining methods.

Pretraining	Dataset	Backbone	Top-1
<i>Video architectures</i>			
Supervised (Wang et al., 2021a)	ImageNet	I3D	67.1
VideoMoCo (Pan et al., 2021)	K400	R(2+1)D	78.7
Temporal-ssl (Jenni et al., 2020)	K400	R(2+1)D	81.6
VTHCL (Yang et al., 2020a)	K400	3D-R50	82.1
CoCLR (Han et al., 2020)	K400	S3D	87.9
CVRL (Qian et al., 2021)	K400	3D-R50	92.9
ρ -BYOL (Feichtenhofer et al., 2021)	K400	3D-R50	95.5
Supervised (Carreira and Zisserman, 2017)	K400	I3D	95.1
<i>Image architectures</i>			
OPN (Lee et al., 2017)	UCF101	VGG-M	59.8
TCE (Knights et al., 2021)	K400	R50	71.2
CycleContrast (Wu and Wang, 2021)	R2V2	R50	82.1
MoCLR (Tian et al., 2021)	ImageNet	R50	85.5
BYOL (Grill et al., 2020)	ImageNet	R50	85.6
VITO (avgpool)	VideoNet	R50	87.4
VITO (MS-avgpool)	VideoNet	R50	88.5
VITO (MS-attnpool)	VideoNet	R50	89.4

Table A.5: VITO outperforms all image representations when finetuning for UCF101 action recognition, using temporally-pooled frame-level representations. VITO’s performance is even competitive with many video architectures.

B | SELF-SUPERVISED LEARNING OF A BIOLOGICALLY-INSPIRED VISUAL TEXTURE MODEL

B.1 TRANSFER LEARNING WITH PRE-TRAINED NETWORKS

In addition to comparing networks trained from scratch on our modified KTH dataset, we also tested the performance of features transferred from pre-trained versions of our V2Net model, VGG-16, and ResNet-18. We pre-trained our model on a dataset of 11000 unlabeled image patches using our self-supervised objective. Example images from this dataset are provided in Fig. B.1:

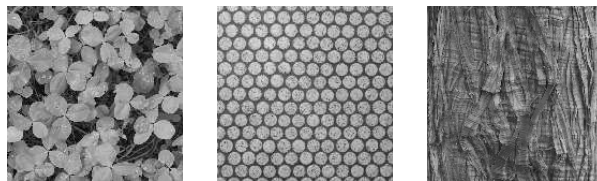


Figure B.1: Example texture images from our hand-curated dataset, comprised of a large collection of natural textures that are unlabelled, but diverse in content and homogeneous across their spatial extent.

The VGG and ResNet networks are pre-trained on the supervised task of object recognition using 1 million images from the ImageNet database.

We used these pre-trained networks as feature extractors, and retrained the respective clas-

sifiers (See Fig. 3.2) for texture classification. Results are shown in Fig. B.2.

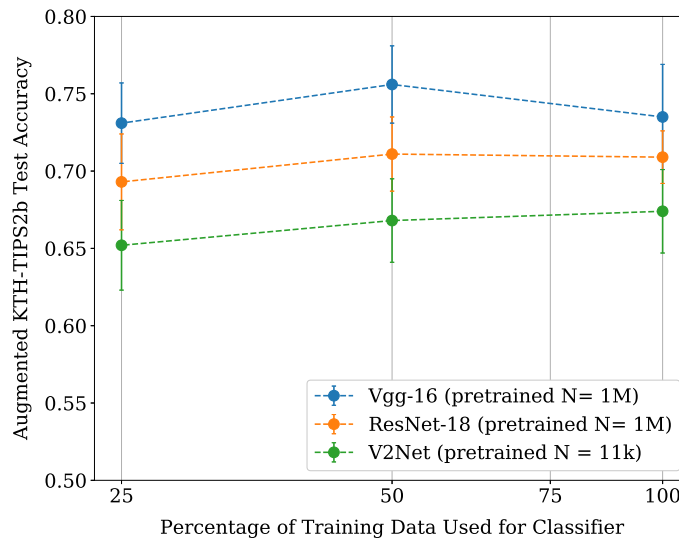


Figure B.2: V2Net vs. DNN texture classification efficiency. Mean and standard error computed across the 4 train/test splits on our KTH dataset (for each experiment where 25 %, 50 % and 100 % of training data is used to train the classifier weights). N refers to the number of images used to pre-train each model.

For all of the models, we find the classifier does not require large amounts of training data - performance is relatively constant across the different amounts of training data used. For the full (100%) classifier training set, our model achieves 67% - the performance gap (5-10%) relative to the pre-trained CNNs is surprisingly small given that our model is pre-trained without supervision, using two orders of magnitude fewer images (11k vs. 1M).

B.2 SELECTIVITY FOR NATURAL TEXTURE VS. SPECTRALLY-SHAPED NOISE

For each of the 11 texture families in the test dataset, we plot the mean accuracy our model trained on natural images (V2Net (Natural) vs. our model trained on phase-scrambled images V2Net (PS)). Fig. B.3 shows that the model trained on natural images performs better for most

texture families, since it is able to capture higher-order natural statistics. If we visualize an example of one of these classes (aluminum foil), we see that this is because the scrambling of phase destroys content that is critical in defining that texture. However, for a few families, the performance of the V2Net (PS) model is about the same as the V2Net (Natural) model because certain texture families (e.g. wood) are primarily defined by their spectral content (and thus not altered significantly by phase-scrambling).

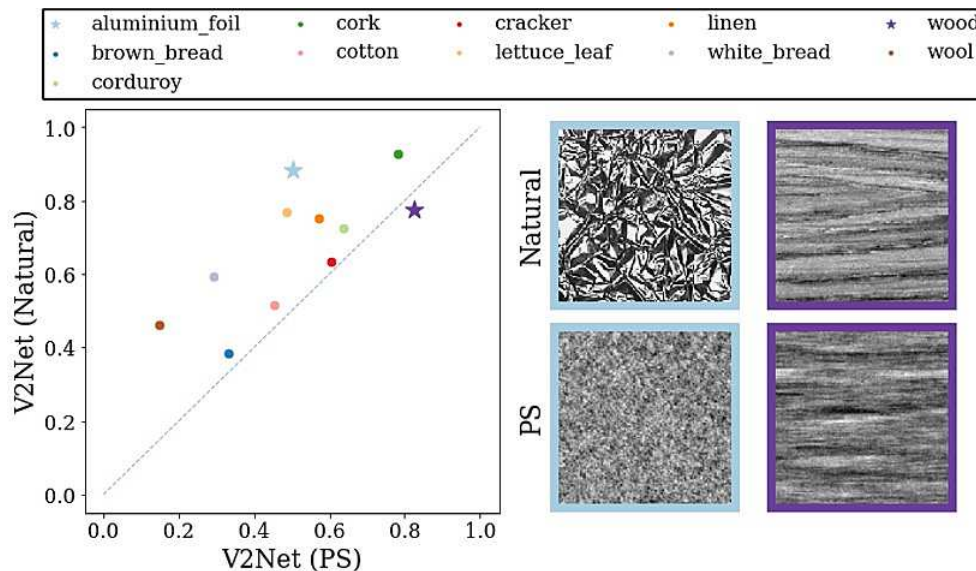


Figure B.3: V2Net discriminates natural texture from spectrally matched noise. Left: Average KTH test accuracy (averaged over 4 splits with 100 % training data) for the V2Net (Natural) vs. V2Net (PS) models. Right: example images (both natural and phase-scrambled) for two texture families. For the ‘aluminum foil’ family, the phase-scrambled image removes the higher-order content that is necessary for identifying the texture. For the ‘wood’ family, the phase-scrambling does not alter perception of the texture significantly, because its appearance is primarily determined by spectral content.

B.3 V2 PHYSIOLOGY COMPARISONS

B.3.1 DATA AND METHODS

Here, we provide more details about the dataset and methods used for the representational similarity analysis presented in the main text. The neural data taken from [Ziamba et al. \(2016\)](#)

consists of electrophysiological recordings of 103 V2 neurons from anesthetized adult macaque monkeys. As is done in the original analysis, we averaged spike counts within 100-ms time windows aligned to the response onset for each single unit. To gaussianize the neural responses, we applied a variance-stabilizing transformation to the spike counts for each neuron ($r_{gauss} = \sqrt{r_{poiss}} + \sqrt{r_{poiss} + 1}$).

The visual stimuli used in the experiment are synthetic texture stimuli generated using the procedure described in [Portilla and Simoncelli \(2000\)](#). A set of 15 grayscale texture photographs are used as the examples for 15 different texture families. From these seed images, 15 samples are generated for each family to provide sample variation across the family. The original stimuli have a size of 320 x 320 pixels and are presented to every V2 unit at a size of 4°, within a raised cosine aperture (this window was larger than all of the receptive fields of the neurons at the recorded eccentricities). For our representational similarity experiments, we thus pre-processed the images for input to the models such that they are resized to the appropriate pixel dimensions (224 x 224) and presented within a 4° raised cosine aperture.

B.3.2 T-SNE VISUALIZATION

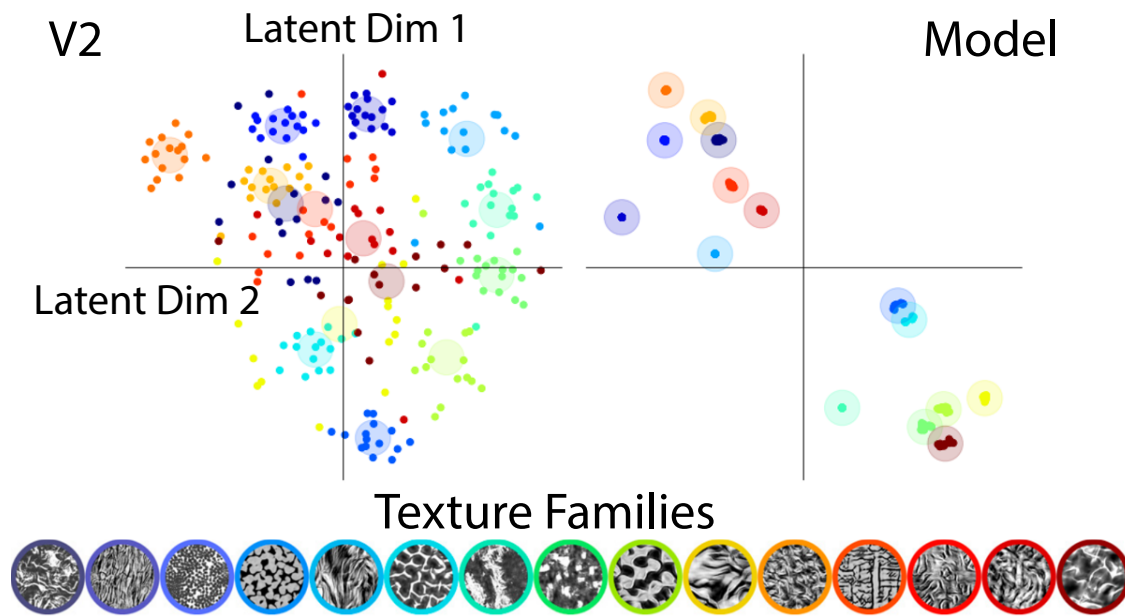


Figure B.4: V2 vs Model T-SNE representation. Left: V2 neural responses projected into a 2-d latent space using the T-SNE method. Right: Our model responses projected into a 2-d latent space using the T-SNE method. Individual dots are responses to individual images from a given texture family. Each of the 15 texture families is given a separate color.

C | LAYERWISE COMPLEXITY-MATCHED SELF-SUPERVISED LEARNING YIELDS IMPROVED MODELS OF CORTICAL AREA V2

C.1 ARCHITECTURE DETAILS

We summarize the details of all architectures used in both the biological-alignment evaluations and the human behavior evaluations.

C.1.1 OUR ARCHITECTURES

For the LCL-V2 two-stage model, and or for all of the ablation studies in Sec. 4.4.4, we use the following architecture:

We use this block as the architecture for all of the ablation studies in Sec. 4.4.4.

As shown in Fig. 4.3, we use projector networks $g(\theta_1), g(\theta_2)$ during training to encourage learning ‘equivariant’ representations:

For the full LCL-V2Net architecture (used in Sec. 4.4.5), we fix the LCL-V2 block and train the

Layer 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)
Layer 2	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)

Table C.1: LCL-V2 Two-layer Architecture. We use the same channel dimensions and non-linearities take from the first two layers of the AlexNet architecture (along with BatchNorm layers). In addition, because we train our model with small image patches, aliasing artifacts impact model responses more than with large images. As a result, we replace standard MaxPooling with anti-aliasing blurring followed by max-pooling (as done in (Zhang, 2019))

$g(\theta_1)$	Linear (in_channels=64, out_channels=64) BatchNorm2d (64) ReLU () Linear (in_channels=64, out_channels=2048)
$g(\theta_2)$	Linear (in_channels=192, out_channels=192) BatchNorm2d (192) ReLU () Linear (in_channels=192, out_channels=2048)

Table C.2: LCL-V2 projector network architectures. Projector networks used during self-supervised pre-training for each layer are MLP networks with single hidden layers. Following (Siddiqui et al., 2023), we use output dimensionalities of 2048 for each projector.

following subsequent stages shown in Table. C.3:

For the neural response prediction evaluations, we implement a baseline model using the Steerable Pyramid (Simoncelli and Freeman, 1995). We use a 5 scale, 4 orientation complex pyramid (based on 3rd-order oriented derivative filters, and their Hilbert Transforms) as implemented in the Plenoptic package (Duong et al., 2023). We rectify (ReLU) both the real and imaginary channels to generate a total of 40 ‘simple cells’. We additionally create 20 ‘complex-cell’ channels by computing the modulus of each complex-valued filter response: $r_{complex} = \sqrt{r_{real}^2 + r_{imag}^2}$. For the V1-baseline model, we subsample the output spatial feature map by a factor of 4 to reduce the

Downstream features	Conv2d (192, 384, ks=3, padding=1) BatchNorm2d (384) ReLU () Conv2d (384, 256, ks=3, padding=1) BatchNorm2d (256) ReLU () Conv2d (256, 256, ks=3, padding=1) BatchNorm2d (256) ReLU () MaxPool2d (ks=3, stride=2)
Classifier	AdaptiveAvgPool2d (1,1) Dropout (p=0.5) Linear (D, 4096) BatchNorm2d (4096) ReLU () Dropout (p=0.5) Linear (4096, 4096) BatchNorm2d (4096) ReLU () Linear (4096, 1000)

Table C.3: LCL-V2Net Downstream Architecture. On top of the LCL-V2 stage, we train the subsequent stages indicated here based on the AlexNet architecture. We include BatchNorm layers as we find this speeds up convergence.)

total number of responses. For the V2-baseline model, we use the response of the V1-stage after applying L_2 spatial energy pooling in a channel-independent way with a 3x3 kernel and additional subsampling by a factor 2. Given a 200x200 grayscale input image, the V1-layer response vector therefore has shape (60, 50, 50) while the V2-layer response vector has shape (60, 25, 25).

C.1.1.1 VONE NET (HAND-CRAFTED / E2E)

We use the VOneNet-AlexNet network developed in (Dapello et al., 2020). We note that this network does not use BatchNorm layers; however, due to inability to re-train this model effectively we use the published pre-trained version. Briefly, this network consist of a VOneBlock front-end model that contains 256 channels (128 simple cell, 128 complex cell) created from a Ga-

bor filter bank basis set with parameters sampled from distributions of recorded macaque V1 cells. This front-end is followed by the full standard AlexNet network architecture from the Pytorch library (Paszke et al., 2019).

C.1.1.2 BARLOWTWINS (LAYERWISE)

We use the same architecture as our LCL-V2 block defined above. However, we utilize the E2E augmentation training parameters defined in (Zbontar et al., 2021) and (Siddiqui et al., 2023) (see Sec. C.3 for details).

We use this model for comparison in both neural response prediction (Sec. 4.4.1) and as a baseline for model ablations (Sec. 2.4.3).

C.1.1.3 LPL (LAYERWISE)

We use the training code (<https://github.com/fmi-basel/latent-predictive-learning>) and methods provided in (Halvagal and Zenke, 2023). While we attempted to re-train the LPL method on the ImageNet-1k dataset, we found that this network would not converge. As a result, we use the AlexNet model trained on the STL-10 dataset for 800 epochs (as done in the original work).

We only use this model for comparison in the neural response prediction experiment (Sec. 4.4.1) to provide an additional layerwise learning baseline.

C.1.1.4 BARLOWTWINS (E2E)

Due to the fact that there is not an available pre-trained AlexNet-based version of the Barlow Twins E2E training method from (Zbontar et al., 2021), we pre-train our own version based on the standard AlexNet architecture (with BatchNorm layers after each convolution/linear layer). We pool the final ‘feature’ layer such that each image is represented by a single 256-d responses vector. As was done in (Zbontar et al., 2021), before the loss computation, this vector is propagated

through a standard MLP projector network with 2 hidden layers. We varied different projector sizes and found the best projector setting to be one with dimensionalities: (1024, 1024, 1024) for the 2 hidden layers and output layer. We pre-train this network for 100 epochs.

For the comparisons in Sec. 4.4.5, we train a classifier stage (as defined in Table C.3), that operates on the output of the fixed network.

C.1.1.5 SUPERVISED ALEXNET

For the results in this work, we tested two variations of the standard AlexNet architecture (one with BatchNorm (Ioffe and Szegedy, 2015) and one without). Interestingly, although we found benefits of BatchNorm in convergence for our LCL training, we find that the standard AlexNet, without BatchNorm, provides a slightly better account of the neural data (and similar performance on the OOD behavioral benchmarks). As a result, use the standard network (without BatchNorm) for comparison. Because this network is fully-supervised on the ImageNet-1K recognition task (provided in the Pytorch library), we do not perform any extra training for the results in Sec. 4.4.5.

C.1.1.6 L2-AT ($\epsilon = 3.0$)

We use an existing fully pre-trained version of the AlexNet architecture, trained with L2-AT adversarial training (Madry et al., 2017). We use the specific pre-trained model from (Chen et al., 2020a), which uses $\epsilon = 3.0$ as the perturbation threshold. Because this network is fully-supervised on the ImageNet-1K recognition task (provided in the Pytorch library), we do not perform any extra training for the results in Sec. 4.4.5.

C.1.1.7 ANT

We use the code provided at <https://github.com/bethgelab/game-of-noise> to train the standard AlexNet model with the adversarial noise training method provided in (Rusak et al.,

2020). This method differs from the L2-AT standard adversarial training as it uses a parameterized noise distribution (Gaussian) rather than arbitrary pixel perturbations within a fixed budget.

C.2 BARLOW TWINS (E2E) DIAGRAM

For completeness, we briefly depict the original Barlow Twins method in case the pictorial diagram is helpful for interpreting the objective function (which we adapt here in our layerwise setting).

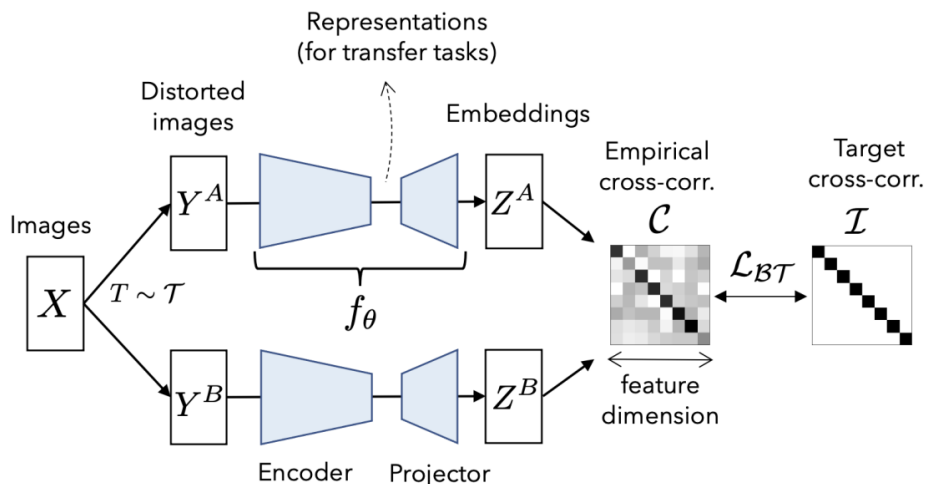


Figure C.1: Diagram depicting the Barlow Twins method. Given an original image X , distorted versions Y^A, Y^B are generated from the original image. These augmented images are passed through the encoder and projector networks to produce embeddings Z^A, Z^B . The objective then attempts to maximize the cross-correlation on the diagonal (across the two views of the same image) and minimize the off-diagonal elements of the cross-correlation matrix (feature dot products between different images)

C.3 AUGMENTATION DETAILS

Standard E2E training

The standard E2E augmentation scheme that we use to both train the Barlow (E2E) (Zbontar et al., 2021) and Barlow (layerwise) (Siddiqui et al., 2023) baseline models is defined as follows:

Each image is randomly augmented by composing the following operations, each applied with a given probability:

1. random cropping: a random patch of the image is selected, whose area is uniformly sampled in $[s \cdot \mathcal{A}, \mathcal{A}]$, where \mathcal{A} is the area of the original image, and whose aspect ratio is logarithmically sampled in $[3/4, 4/3]$. s is a scale hyper-parameter set to 0.08. The patch is then resized to 224×224 pixels using bicubic interpolation;
2. horizontal flipping;
3. color jittering: the brightness, contrast, saturation and hue are shifted by a uniformly distributed offset;
4. color dropping: the RGB image is replaced by its grey-scale values;
5. gaussian blurring with a 23×23 square kernel and a standard deviation uniformly sampled from $[0.1, 2.0]$;
6. solarization: a point-wise color transformation $x \mapsto x \cdot \mathbb{1}_{x < 0.5} + (1 - x) \cdot \mathbb{1}_{x \geq 0.5}$ with pixels x in $[0, 1]$.

The augmented frames x^A and x^B result from augmentations sampled from distributions \mathcal{A}_A and \mathcal{A}_B respectively. These distributions apply the primitives described above with different probabilities, and different magnitudes. The following table specifies these parameters.

LCL-V2 Augmentations

Given the limited capacity of our two-stage network, we drastically reduce the number and strength of these augmentations. As described in the main text, we additionally apply a complexity-matched set of augmentations to generate the inputs for training each layer. For our training, each image is first resized such that the smallest side is resized to 224 pixels. It is then randomly augmented by composing the following operations, each applied with a given probability.:

Parameter	\mathcal{A}_A	\mathcal{A}_B
Random crop probability		1.0
Flip probability		0.5
Color jittering probability		0.8
Color dropping probability		0.2
Brightness adjustment max		0.4
Contrast adjustment max		0.4
Saturation adjustment max		0.2
Hue adjustment max		0.1
Gaussian blurring probability	1.0	0.1
Solarization probability	0.0	0.2

1. center cropping: a center crop of a given size is first selected from the image.
2. random cropping: a random patch of this central crop is selected, whose area is uniformly sampled in $[s \cdot \mathcal{A}, 0.9 \cdot \mathcal{A}]$, where \mathcal{A} is the area of the original image, and whose aspect ratio is logarithmically sampled in $[0.9, 1.1]$. s is a scale hyper-parameter set to a different value for each layer of the LCL-V2 architecture. The patch is then resized to $p \times p$ where p is again dependent on the layer.
3. contrast and luminance jittering: the brightness and contrast are shifted by a uniformly distributed offset.
4. Gaussian noise: additive Gaussian noise is added independently to each channel of the RGB image. The noise is generated to be mean 0 with random standard deviation uniformly sampled from the range (0.04, 0.1).

The parameters to generate the first layer patches x_1^A and x_1^B are defined below:

Parameter	\mathcal{A}_A	\mathcal{A}_B
Central crop size	56	
Minimum random crop scale	0.6	
Random crop resize output size	48	
Color jittering probability	0.8	
Color dropping probability	0.2	
Brightness adjustment max	0.2	
Contrast adjustment max	0.2	
Gaussian noise probability	1.0	0.0

The parameters to generate the second layer patches x_2^A and x_2^B are defined below:

Parameter	\mathcal{A}_A	\mathcal{A}_B
Central crop size	112	
Minimum random crop scale	0.3	
Random crop resize output size	96	
Color jittering probability	0.8	
Color dropping probability	0.2	
Brightness adjustment max	0.2	
Contrast adjustment max	0.2	
Gaussian noise probability	1.0	0.0

C.4 NEURAL EVALUATION DETAILS

C.4.1 STIMULUS PRE-PROCESSING

We use the stimuli from (Freeman et al., 2013; Ziemba et al., 2016), which consist of 225 naturalistic texture images from 15 texture families and 225 corresponding spectrally-matched noise images. We use the pre-processing in the BrainScore benchmark <https://github.com/brain-score/brain-score> which applies a circular aperture to the original images and resizes the image such that it covers the central 4° , within a raised cosine aperture (112 pixel diameter), to

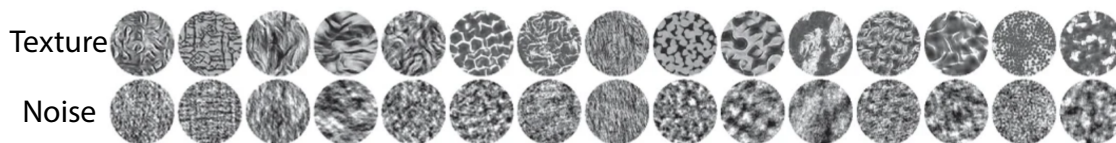


Figure C.2: Example samples of each of the 15 texture families and their corresponding spectrally-matched noise images. Taken from (Freeman et al., 2013).

align with the fact that the experimental protocol presented images at a 4° field-of-view to the individual V1/V2 neurons. However, because we assume an 8° field-of-view for all models (224 pixels). Each image is then padded with gray pixels beyond the central 4° .

C.4.2 BRAINSCORE NEURAL PREDICTION

Following the BrainScore pipeline, we preprocess the neural responses binning raw spike-counts within 10ms windows and averaging these spike-counts over the 50ms to 150ms range post stimulus presentation. Because each stimulus is presented 20 times, the average spike-count for each sample is calculated as the average over these 20 trials. To evaluate neural predictivity we use the API provided in <https://github.com/brain-score/brain-score>. We show scores on the private split of the data which consists of approximately 70% of the original images (official BrainScore split). Briefly, $N \times D_m$ model responses are regressed onto the $N \times D_n$ neural responses (V1: $D_n = 102$, V2: $D_n = 103$). This is done using the PLS regression method with 25 components. The regression is computed using 10-fold cross-validation and pearson correlations r_n (between model predictions and neural responses across all test images) are obtained for each neuron and for each split. A measure of internal consistency $r_{ceil,n}$ is also computed by splitting neural responses in half across repeated presentations of the same image and computing the Pearson correlation coefficient between the two splits across images for each neuron. For more details, see (Schrimpf et al., 2018).

For the overall scores presented in Fig. 4.4 (Left), The final explained variance for a given

model is calculated as $median_n(r_n^2/r_{ceil,n}^2)$. For the subset explained variance scores in Fig. 4.4 (Right), these medians are computed over the specific subset of neurons identified as ‘V1-like’ or ‘not-V1-like’ (based on the V1-SteerPyr model explained variances).

We further estimate the internal consistency between neural responses by splitting neural responses in half across repeated presentations of the same image and computing Spearman-Brown-corrected Pearson correlation coefficient between the two splits across images for each neuroid.

C.4.3 TEXTURE MODULATION ANALYSIS

The analysis on texture modulation uses the same stimuli and image pre-processing. For the neural responses, we follow the same pre-processing outlined above; however, instead of binning each neural responses over the fixed 50ms to 150ms window post presentation, we use the method in (Freeman et al., 2013) which still selects a 100ms window (but now aligned to the specific response-latency of each neuron). This is done for consistency with the measured texture modulation indices in (Freeman et al., 2013), but we do not find that this significantly changes our results.

C.5 OOD AND HUMAN BEHAVIOR BENCHMARK

Dataset information. We evaluate accuracy and human error consistency using the code from <https://github.com/bethgelab/model-vs-human/tree/master>. We report the OOD accuracy (averaged over samples and distortions) on the full dataset from (Geirhos et al., 2021), which consists of 17 OOD distortions applied to ImageNet images. We additionally report the behavior error consistency metric summarized below.

Behavior error consistency metric. This metric, derived in (Geirhos et al., 2020b) measures whether there is above-chance consistency with human per-sample recognition decisions.

let $c_{h,m}(s)$ be 1 if both a human observer h and m decide either correctly or incorrectly on a given sample s , and 0 otherwise. The observed consistency $c_{h,m}$ is the average of $c_{h,m}(s)$ over all samples. The error consistency then measures whether this observed consistency is larger than the expected consistency given two independent binomial decision makers with matched accuracy. For the details on this exact computation, see (Geirhos et al., 2020b).

C.6 V1 RECEPTIVE FIELD COMPARISONS

In addition to the overall predictivity of the V1 data, we also use qualitative and quantitative analyses to probe the selectivities of the learned receptive fields in the first convolutional layer of our model. We compare these learned filters to those learned via adversarial-training and standard supervised training.

Filter Visualization. We visualize the 64 filters of each learned model in Fig. C.3. We visualize the filters in grayscale (even though the filters operate on color channels), to draw comparisons specifically between the spatial receptive field properties. By inspection, none of the models perfectly capture the nature of real V1 receptive fields; however, they all learn a set of reasonably diverse of multi-scale oriented filters. Our model and the L2-Robust model seem to better capture the number of cycles within the oriented filters, whereas the supervised network receptive fields are too high-frequency. Both LCL-V1 and Supervised models, however, seem to learn more localized blob filters than the adversarially-trained model.

V1 Receptive Field Properties. We briefly explore further a couple of the canonical receptive field properties highlighted in the data collected from macaque V1 neurons in (Ringach, 2002). For the three models, we fit Gabor receptive fields to the filters shown in Fig. C.3 by minimizing the mean squared error between a parameterized 2-d gabor function and the model receptive field. We remove those that were not fit well by the fitting procedure.

We first measure the spatial-phase of each receptive field and compare these to the distribu-

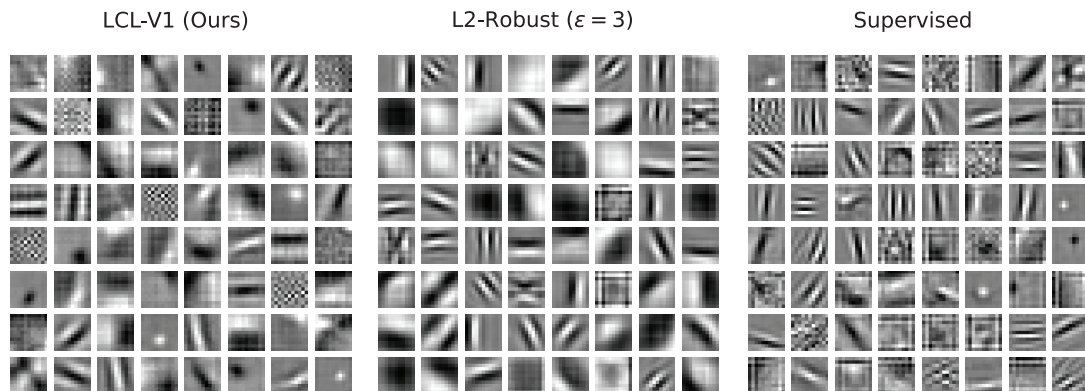


Figure C.3: Comparing Learned Receptive Fields. We demonstrate that our single-layer (layerwise) trained model (LCL-V1), learns a diverse set of oriented receptive fields. All 64 filters are shown for our method, adversarial robust training, and standard supervised training. Qualitatively, both our model and the adversarially-trained network learn more low-frequency filters than the standard supervised-trained network. Original filters are RGB, but we show grayscale versions here to focus on the spatial structure comparisons.

tion of spatial phase of the macaque data (Fig. C.4). None of the models perfectly match the neural data distribution, but our LCL-V1 model seems to qualitatively best capture the relative bi-modal structure around even and odd-symmetry (0 and $\pi/2$). We next compare the receptive field structure via the method in (Ringach, 2002), plotting the number of cycles in each dimension of 2-d Gabor function (n_y , n_x), against each other for each filter. These two parameters control receptive field shape by changing the structure of the Gabor sub-fields. We see in Fig. C.5, that while there are some outliers, most of our model receptive fields fall within a similar distribution as the macaque V1 receptive fields. Compared with the other two models, our model does a better job capturing the density of the more 'blob-like' filters near the origin.

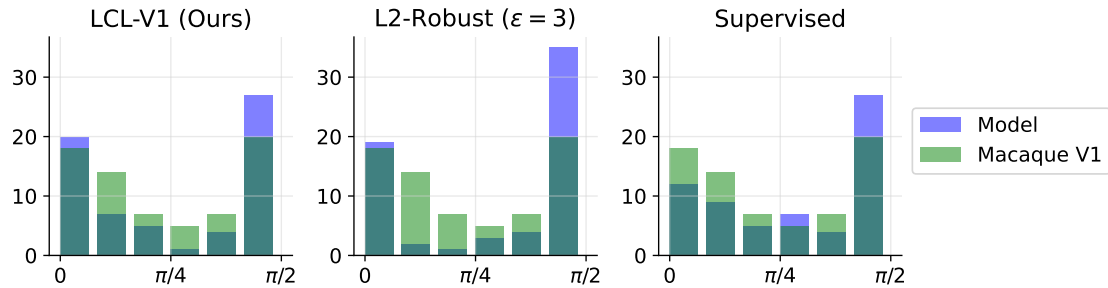


Figure C.4: Comparing spatial phase selectivity between models and macaque V1 neurons. We compare our LCL-V1 receptive field tuning to the equivalent receptive fields extracted from the L2-Robust and Supervised networks. Model histograms of neuron spatial phase are shown in blue and real macaque data from (Ringach, 2002) is shown in green.

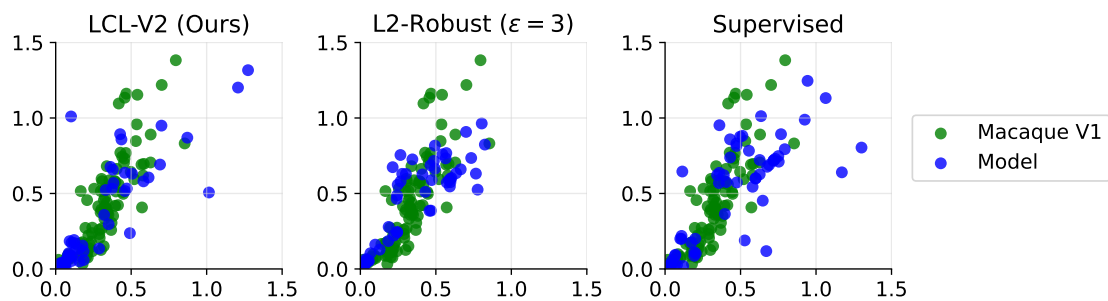


Figure C.5: N_y vs. N_x V1 receptive field structure. N_y vs. N_x parameters for model receptive fields and macaque V1 receptive fields (obtained via fitted Gabors to each filter). Model neurons are plotted in blue and Macaque V1 data is shown in green.

C.7 ADDITIONAL HUMAN BEHAVIOR RESULTS

Method	IN-1K acc. \uparrow	OOD acc. \uparrow	obs. consistency \uparrow	error consistency \uparrow
LCL-V2Net	0.527	0.492	0.643	0.211
Barlow Twins (Zbontar et al., 2021)	0.459	0.451	0.607	0.166
Supervised (Krizhevsky et al., 2012)	0.590	0.443	0.597	0.165
VOneNet (Dapello et al., 2020)	0.491	0.407	0.585	0.168
L2-Robust (Madry et al., 2017)	0.399	0.391	0.573	0.176

Table C.4: OOD accuracy and consistency with human judgments on 17 different OOD recognition tests, including multiple types of noise, phase-scrambling, and shape-biased stimuli. In both accuracy and human-alignment, our model trained with the LCL-V2 front-end improves over all end-to-end training approaches.

BIBLIOGRAPHY

- Abou-Moustafa, K. T., De La Torre, F., and Ferrie, F. P. (2010). Designing a metric for the difference between gaussian densities. In *Brain, Body and Machine*, pages 57–70. Springer.
- Adelson, E. H. and Bergen, J. R. (1985). Spatiotemporal energy models for the perception of motion. *Josa a*, 2(2):284–299.
- Agrawal, P., Carreira, J., and Malik, J. (2015). Learning to see by moving. In *ICCV*.
- Alayrac, J.-B., Recasens, A., Schneider, R., Arandjelović, R., Ramapuram, J., De Fauw, J., Smaira, L., Dieleman, S., and Zisserman, A. (2020). Self-supervised multimodal versatile networks. *Advances in Neural Information Processing Systems*, 33:25–37.
- Arcaro, M. J., Ponce, C., and Livingstone, M. (2020). The neurons that mistook a hat for a face. *Elife*, 9:e53798.
- Atick, J. J. and Redlich, A. N. (1990). Towards a theory of early visual processing. *Neural computation*, 2(3):308–320.
- Bagus, A. M. I. G., Marques, T., Sanghavi, S., DiCarlo, J. J., and Schrimpf, M. (2022). Primate inferotemporal cortex neurons generalize better to novel image distributions than analogous deep neural networks units. In *SVRHM 2022 Workshop@ NeurIPS*.
- Bai, Y., Chen, X., Kirillov, A., Yuille, A., and Berg, A. C. (2022). Point-level region contrast for object detection pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16061–16070.
- 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.
- Bansal, N., Chen, X., and Wang, Z. (2018). Can we gain more from orthogonality regularizations in training deep cnns? In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pages 4266–4276. Curran Associates Inc.
- Bányai, M., Nagy, D. G., and Orbán, G. (2019). Hierarchical semantic compression predicts texture selectivity in early vision. In *Proceedings of the Conference on Cognitive Computational Neuroscience*.

- Barlow, H. B. et al. (1961). Possible principles underlying the transformation of sensory messages. *Sensory communication*, 1(01):217–233.
- Battaglia, P. W., Hamrick, J. B., and Tenenbaum, J. B. (2013). Simulation as an engine of physical scene understanding. *Proceedings of the National Academy of Sciences*, 110(45):18327–18332.
- Becker, S. and Hinton, G. E. (1995). Spatial coherence as an internal teacher for a neural network. *Backpropagation: Theory, architecture and applications*, pages 313–349.
- 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. (1997). The “independent components” of natural scenes are edge filters. *Vision research*, 37(23):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.
- Bergen, J. R. and Adelson, E. H. (1986). Visual texture segmentation based on energy measures (a). *J. Opt. Soc. Am. A*, vol. 3, page P99, 3.
- Beyer, L., Hénaff, O. J., Kolesnikov, A., Zhai, X., and Oord, A. v. d. (2020). Are we done with imagenet? *arXiv preprint arXiv:2006.07159*.
- Bhattacharyya, A. (1946). On a measure of divergence between two multinomial populations. *Sankhyā: the indian journal of statistics*, pages 401–406.
- Bian, Z., Jabri, A., Efros, A. A., and Owens, A. (2022). Learning pixel trajectories with multiscale contrastive random walks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6508–6519.
- Bovik, A. C., Clark, M., and Geisler, W. S. (1990). Multichannel texture analysis using localized spatial filters. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (1):55–73.
- Bowers, J. S., Malhotra, G., Dujmović, M., Montero, M. L., Tsvetkov, C., Biscione, V., Puebla, G., Adolfi, F., Hummel, J. E., Heaton, R. F., et al. (2022). Deep problems with neural network models of human vision. *Behavioral and Brain Sciences*, pages 1–74.
- Bruna, J. and Mallat, S. (2013). Invariant scattering convolution networks. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1872–1886.
- Bures, D. (1969). An extension of kakutani’s theorem on infinite product measures to the tensor product of semifinite w^* -algebras. *Transactions of the American Mathematical Society*, 135:199–212.

- Cadieu, C. F. and Olshausen, B. A. (2012). Learning intermediate-level representations of form and motion from natural movies. *Neural computation*, 24(4):827–866.
- Caputo, B., Hayman, E., and Mallikarjuna, P. (2005). Class-specific material categorisation. In *Tenth IEEE International Conference on Computer Vision (ICCV’05) Volume 1*, volume 2, pages 1597–1604. IEEE.
- Carandini, M. and Heeger, D. J. (1994). Summation and division by neurons in primate visual cortex. *Science*, 264(5163):1333–1336.
- Carandini, M., Heeger, D. J., and Movshon, J. A. (1997). Linearity and normalization in simple cells of the macaque primary visual cortex. *Journal of Neuroscience*, 17(21):8621–8644.
- Carlini, N., Athalye, A., Papernot, N., Brendel, W., Rauber, J., Tsipras, D., Goodfellow, I., Madry, A., and Kurakin, A. (2019). On evaluating adversarial robustness. *arXiv preprint arXiv:1902.06705*.
- Caron, M., Bojanowski, P., Joulin, A., and Douze, M. (2018). Deep clustering for unsupervised learning of visual features. In *The European Conference on Computer Vision (ECCV)*.
- 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.
- Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., and Joulin, A. (2021). Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9650–9660.
- Carreira, J. and Zisserman, A. (2017). Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308.
- Chellappa, R. and Chatterjee, S. (1985). Classification of textures using gaussian markov random fields. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 33(4):959–963.
- 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.
- 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.
- Cimpoi, M., Maji, S., and Vedaldi, A. (2015). Deep filter banks for texture recognition and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3828–3836.
- 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.
- Cox, D. D., Meier, P., Oertelt, N., and DiCarlo, J. J. (2005). ‘breaking’ position-invariant object recognition. *Nature neuroscience*, 8(9):1145–1147.
- Cross, G. R. and Jain, A. K. (1983). Markov random field texture models. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (1):25–39.
- Danon, D., Averbuch-Elor, H., Fried, O., and Cohen-Or, D. (2019). Unsupervised natural image patch learning. *Computational Visual Media*, 5(3):229–237.
- Dapello, J., Marques, T., Schrimpf, M., Geiger, F., Cox, D., and DiCarlo, J. J. (2020). Simulating a primary visual cortex at the front of cnns improves robustness to image perturbations. *Advances in Neural Information Processing Systems*, 33:13073–13087.
- Dave, I., Gupta, R., Rizve, M. N., and Shah, M. (2022). Tclr: Temporal contrastive learning for video representation. *Computer Vision and Image Understanding*, 219:103406.
- De la Torre, F. and Kanade, T. (2005). Multimodal oriented discriminant analysis. In *Proceedings of the 22nd international conference on Machine learning*, pages 177–184. ACM.
- Dehghani, M., Djolonga, J., Mustafa, B., Padlewski, P., Heek, J., Gilmer, J., Steiner, A., Caron, M., Geirhos, R., Alabdulmohsin, I., et al. (2023). Scaling vision transformers to 22 billion parameters. *arXiv preprint arXiv:2302.05442*.
- Derin, H. and Elliott, H. (1987). Modeling and segmentation of noisy and textured images using gibbs random fields. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (1):39–55.
- DiCarlo, J. J., Zoccolan, D., and Rust, N. C. (2012). How does the brain solve visual object recognition? *Neuron*, 73(3):415–434.
- Dorkenwald, M., Xiao, F., Brattoli, B., Tighe, J., and Modolo, D. (2022). Scvrl: Shuffled contrastive video representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4132–4141.
- Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

- Douglas, R. J., Martin, K. A., and Whitteridge, D. (1989). A canonical microcircuit for neocortex. *Neural computation*, 1(4):480–488.
- Dryden, I. L., Koloydenko, A., Zhou, D., et al. (2009). Non-euclidean statistics for covariance matrices, with applications to diffusion tensor imaging. *The Annals of Applied Statistics*, 3(3):1102–1123.
- Dumoulin, V., Shlens, J., and Kudlur, M. (2016). A learned representation for artistic style. *arXiv preprint arXiv:1610.07629*.
- 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.
- Duong, L. R., Zhou, J., Nassar, J., Berman, J., Olieslagers, J., and Williams, A. H. (2022). Representational dissimilarity metric spaces for stochastic neural networks. *arXiv preprint arXiv:2211.11665*.
- 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.
- Everingham, M., Eslami, S. A., Van Gool, L., Williams, C. K., Winn, J., and Zisserman, A. (2015). The pascal visual object classes challenge: A retrospective. *International journal of computer vision*, 111(1):98–136.
- Faraki, M., Harandi, M. T., and Porikli, F. (2016). Image set classification by symmetric positive semi-definite matrices. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1–8. IEEE.
- Feather, J., Leclerc, G., Mađry, A., and McDermott, J. H. (2023). Model metamers reveal divergent invariances between biological and artificial neural networks. *Nature Neuroscience*, pages 1–18.
- Feichtenhofer, C., Fan, H., Li, Y., and He, K. (2022). Masked autoencoders as spatiotemporal learners. *arXiv preprint arXiv:2205.09113*.
- Feichtenhofer, C., Fan, H., Xiong, B., Girshick, R., and He, K. (2021). A large-scale study on unsupervised spatiotemporal representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3299–3309.
- Fel, T., Felipe, I., Linsley, D., and Serre, T. (2022). Harmonizing the object recognition strategies of deep neural networks with humans. *arXiv preprint arXiv:2211.04533*.
- Feng, C., Zhong, Y., Gao, Y., Scott, M. R., and Huang, W. (2021). TOOD: Task-aligned one-stage object detection. In *Int. Conf. Comput. Vis.*

- Foster, K., Gaska, J. P., Nagler, M., and Pollen, D. (1985). Spatial and temporal frequency selectivity of neurones in visual cortical areas v1 and v2 of the macaque monkey. *The Journal of physiology*, 365(1):331–363.
- Freeman, J. and Simoncelli, E. P. (2011). Metamers of the ventral stream. *Nature neuroscience*, 14(9):1195–1201.
- 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.
- Fujieda, S., Takayama, K., and Hachisuka, T. (2018). Wavelet convolutional neural networks. *arXiv preprint arXiv:1805.08620*.
- Fukushima, K. (1980). Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological cybernetics*, 36(4):193–202.
- Garrido, Q., Chen, Y., Bardes, A., Najman, L., and Lecun, Y. (2022). On the duality between contrastive and non-contrastive self-supervised learning. *arXiv preprint arXiv:2206.02574*.
- Gatys, L., Ecker, A. S., and Bethge, M. (2015). Texture synthesis using convolutional neural networks. In *Advances in neural information processing systems*, pages 262–270.
- Geirhos, R., Jacobsen, J.-H., Michaelis, C., Zemel, R., Brendel, W., Bethge, M., and Wichmann, F. A. (2020a). Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673.
- Geirhos, R., Meding, K., and Wichmann, F. A. (2020b). Beyond accuracy: quantifying trial-by-trial behaviour of cnns and humans by measuring error consistency. *Advances in Neural Information Processing Systems*, 33:13890–13902.
- 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. (2018). Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. *arXiv preprint arXiv:1811.12231*.
- Ghodrati, M., Farzmahdi, A., Rajaei, K., Ebrahimpour, R., and Khaligh-Razavi, S.-M. (2014). Feed-forward object-vision models only tolerate small image variations compared to human. *Frontiers in computational neuroscience*, 8:74.
- Gidaris, S., Singh, P., and Komodakis, N. (2018). Unsupervised representation learning by predicting image rotations. *International Conference on Learning Representations (ICLR)*.
- Girard, P. and Bullier, J. (1989). Visual activity in area v2 during reversible inactivation of area 17 in the macaque monkey. *Journal of neurophysiology*, 62(6):1287–1302.

- Golan, T., Raju, P. C., and Kriegeskorte, N. (2020). Controversial stimuli: Pitting neural networks against each other as models of human cognition. *Proceedings of the National Academy of Sciences*, 117(47):29330–29337.
- Goodfellow, I. J., Shlens, J., and Szegedy, C. (2014). Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*.
- Gordon, D., Ehsani, K., Fox, D., and Farhadi, A. (2020). Watching the world go by: Representation learning from unlabeled videos. *arXiv preprint arXiv:2003.07990*.
- Goroshin, R., Bruna, J., Tompson, J., Eigen, D., and LeCun, Y. (2015). Unsupervised learning of spatiotemporally coherent metrics. In *Proceedings of the IEEE international conference on computer vision*, pages 4086–4093.
- Greenspan, H., Goodman, R., and Chellappa, R. (1991). Texture analysis via unsupervised and supervised learning. In *IjCNN-91-Seattle International Joint Conference on Neural Networks*, volume 1, pages 639–644. IEEE.
- 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.
- Gross, C. G. (1973). Inferotemporal cortex and vision. *Progress in physiological psychology*, 5:77–123.
- Halvagal, M. S. and Zenke, F. (2023). The combination of hebbian and predictive plasticity learns invariant object representations in deep sensory networks. *Nature Neuroscience*, pages 1–10.
- Han, T., Xie, W., and Zisserman, A. (2020). Self-supervised co-training for video representation learning. *Advances in Neural Information Processing Systems*, 33:5679–5690.
- He, K., Chen, X., Xie, S., Li, Y., Dollár, P., and Girshick, R. (2021). Masked autoencoders are scalable vision learners. *arXiv preprint arXiv:2111.06377*.
- 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.
- Heeger, D. J. and Bergen, J. R. (1995). Pyramid-based texture analysis/synthesis. In *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pages 229–238. Citeseer.
- Heeger, D. J., Simoncelli, E. P., and Movshon, J. A. (1996). Computational models of cortical visual processing. *Proceedings of the National Academy of Sciences*, 93(2):623–627.

- Hénaff, O. J., Bai, Y., Charlton, J. A., Nauhaus, I., Simoncelli, E. P., and Goris, R. L. (2021a). Primary visual cortex straightens natural video trajectories. *Nature communications*, 12(1):5982.
- Hénaff, O. J., Ballé, J., Rabinowitz, N. C., and Simoncelli, E. P. (2014). The local low-dimensionality of natural images. *arXiv preprint arXiv:1412.6626*.
- Hénaff, O. J., Goris, R. L., and Simoncelli, E. P. (2019a). Perceptual straightening of natural videos. *Nature neuroscience*, 22(6):984–991.
- Hénaff, O. J., Koppula, S., Alayrac, J.-B., Oord, A. v. d., Vinyals, O., and Carreira, J. (2021b). Efficient visual pretraining with contrastive detection. In *ICCV*.
- Hénaff, O. J., Koppula, S., Shelhamer, E., Zoran, D., Jaegle, A., Zisserman, A., Carreira, J., and Arandjelović, R. (2022). Object discovery and representation networks. *arXiv preprint arXiv:2203.08777*.
- Hénaff, O. J. and Simoncelli, E. P. (2015). Geodesics of learned representations. *arXiv preprint arXiv:1511.06394*.
- Hénaff, O. J., Srinivas, A., De Fauw, J., Razavi, A., Doersch, C., Eslami, S., and Oord, A. v. d. (2019b). Data-efficient image recognition with contrastive predictive coding. *arXiv preprint arXiv:1905.09272*.
- Hendrycks, D., Basart, S., Mu, N., Kadavath, S., Wang, F., Dorundo, E., Desai, R., Zhu, T., Parajuli, S., Guo, M., et al. (2021a). The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8340–8349.
- Hendrycks, D. and Dietterich, T. (2019). Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*.
- Hendrycks, D., Zhao, K., Basart, S., Steinhardt, J., and Song, D. (2021b). Natural adversarial examples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15262–15271.
- Hermann, K. and Lampinen, A. (2020). What shapes feature representations? exploring datasets, architectures, and training. *Advances in Neural Information Processing Systems*, 33:9995–10006.
- Hinton, G. E., Osindero, S., and Teh, Y.-W. (2006). A fast learning algorithm for deep belief nets. *Neural computation*, 18(7):1527–1554.
- Hosoya, H. and Hyvärinen, A. (2015). A hierarchical statistical model of natural images explains tuning properties in v2. *Journal of Neuroscience*, 35(29):10412–10428.
- Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., and Adam, H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*.

- Hoyer, P. O. (2004). Non-negative matrix factorization with sparseness constraints. *Journal of machine learning research*, 5(9).
- Hoyer, P. O. and Hyvärinen, A. (2002). A multi-layer sparse coding network learns contour coding from natural images. *Vision research*, 42(12):1593–1605.
- Huang, Z., Wang, R., Shan, S., Li, X., and Chen, X. (2015). Log-euclidean metric learning on symmetric positive definite manifold with application to image set classification. In *International conference on machine learning*, pages 720–729.
- Hubel, D. H. and Wiesel, T. N. (1959). Receptive fields of single neurones in the cat’s striate cortex. *The Journal of physiology*, 148(3):574.
- Hubel, D. H. and Wiesel, T. N. (1962). Receptive fields, binocular interaction and functional architecture in the cat’s visual cortex. *The Journal of physiology*, 160(1):106.
- Hubel, D. H. and Wiesel, T. N. (1968). Receptive fields and functional architecture of monkey striate cortex. *The Journal of physiology*, 195(1):215–243.
- Hurri, J. and Hyvärinen, A. (2003). Simple-cell-like receptive fields maximize temporal coherence in natural video. *Neural Computation*, 15(3):663–691.
- 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. *arXiv preprint arXiv:1502.03167*.
- Jabri, A., Owens, A., and Efros, A. (2020). Space-time correspondence as a contrastive random walk. *Advances in neural information processing systems*, 33.
- Jacobsen, J.-H., van Gemert, J., Lou, Z., and Smeulders, A. W. (2016). Structured receptive fields in cnns. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2610–2619.
- Jaini, P., Clark, K., and Geirhos, R. (2023). Intriguing properties of generative classifiers. *arXiv preprint arXiv:2309.16779*.
- Jean, N., Wang, S., Samar, A., Azzari, G., Lobell, D., and Ermon, S. (2019). Tile2vec: Unsupervised representation learning for spatially distributed data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3967–3974.
- Jenni, S., Meishvili, G., and Favaro, P. (2020). Video representation learning by recognizing temporal transformations. In *European Conference on Computer Vision*, pages 425–442. Springer.
- Ji, X., Henriques, J. F., and Vedaldi, A. (2018). Invariant information distillation for unsupervised image segmentation and clustering. *arXiv preprint arXiv:1807.06653*.

- Jing, L., Vincent, P., LeCun, Y., and Tian, Y. (2021). Understanding dimensional collapse in contrastive self-supervised learning. *arXiv preprint arXiv:2110.09348*.
- Jozwik, K. M., Kriegeskorte, N., and Mur, M. (2016). Visual features as stepping stones toward semantics: Explaining object similarity in it and perception with non-negative least squares. *Neuropsychologia*, 83:201–226.
- Julesz, B. (1962). Visual pattern discrimination. *IRE transactions on Information Theory*, 8(2):84–92.
- Kar, O. F., Yeo, T., and Zamir, A. (2022). 3d common corruptions for object recognition. In *ICML 2022 Shift Happens Workshop*.
- Karklin, Y. and Lewicki, M. S. (2009). Emergence of complex cell properties by learning to generalize in natural scenes. *Nature*, 457(7225):83–86.
- 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.
- Kellman, P. J. and Spelke, E. S. (1983). Perception of partly occluded objects in infancy. *Cognitive psychology*, 15(4):483–524.
- Kheradpisheh, S. R., Ghodrati, M., Ganjtabesh, M., and Masquelier, T. (2016). Deep networks can resemble human feed-forward vision in invariant object recognition. *Scientific reports*, 6(1):32672.
- Khosla, P., Teterwak, P., Wang, C., Sarna, A., Tian, Y., Isola, P., Maschinot, A., Liu, C., and Krishnan, D. (2020). Supervised contrastive learning. *Advances in neural information processing systems*, 33:18661–18673.
- Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
- Knights, J., Harwood, B., Ward, D., Vanderkop, A., Mackenzie-Ross, O., and Moghadam, P. (2021). Temporally coherent embeddings for self-supervised video representation learning. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 8914–8921. IEEE.
- Kong, N. and Norcia, A. (2021). Are models trained on temporally-continuous data streams more adversarially robust? In *SVRHM 2021 Workshop@ NeurIPS*.
- Kriegeskorte, N., Mur, M., and Bandettini, P. A. (2008). Representational similarity analysis—connecting the branches of systems neuroscience. *Frontiers in systems neuroscience*, 2:4.
- Kriegeskorte, N. and Wei, X.-X. (2021). Neural tuning and representational geometry. *Nature Reviews Neuroscience*, 22(11):703–718.
- Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105.

- Kubilius, J., Schrimpf, M., Kar, K., Rajalingham, R., Hong, H., Majaj, N., Issa, E., Bashivan, P., Prescott-Roy, J., Schmidt, K., et al. (2019). Brain-like object recognition with high-performing shallow recurrent anns. *Advances in neural information processing systems*, 32.
- Kulkarni, T. D., Gupta, A., Ionescu, C., Borgeaud, S., Reynolds, M., Zisserman, A., and Mnih, V. (2019). Unsupervised learning of object keypoints for perception and control. *Advances in neural information processing systems*, 32.
- Kumar, M., Houlsby, N., Kalchbrenner, N., and Cubuk, E. D. (2022). Do better imagenet classifiers assess perceptual similarity better? *Transactions of Machine Learning Research*.
- Laskar, M. N. U., Giraldo, L. G. S., and Schwartz, O. (2020). Deep neural networks capture texture sensitivity in v2. *Journal of vision*, 20(7):21–1.
- LeCun, Y. et al. (1989). Generalization and network design strategies. *Connectionism in perspective*, 19(143-155):18.
- Lee, H.-Y., Huang, J.-B., Singh, M., and Yang, M.-H. (2017). Unsupervised representation learning by sorting sequences. In *Proceedings of the IEEE international conference on computer vision*, pages 667–676.
- Lee, S.-H., Kwan, A. C., Zhang, S., Phoumthipphavong, V., Flannery, J. G., Masmanidis, S. C., Taniguchi, H., Huang, Z. J., Zhang, F., Boyden, E. S., et al. (2012). Activation of specific interneurons improves v1 feature selectivity and visual perception. *Nature*, 488(7411):379–383.
- Lennie, P. (1998). Single units and visual cortical organization. *Perception*, 27(8):889–935.
- Levitt, J. B., Kiper, D. C., and Movshon, J. A. (1994). Receptive fields and functional architecture of macaque v2. *Journal of neurophysiology*, 71(6):2517–2542.
- Li, X., Liu, S., De Mello, S., Wang, X., Kautz, J., and Yang, M.-H. (2019). Joint-task self-supervised learning for temporal correspondence. *Advances in Neural Information Processing Systems*, 32.
- Li, Z. (1996). A theory of the visual motion coding in the primary visual cortex. *Neural computation*, 8(4):705–730.
- Lin, T.-Y., RoyChowdhury, A., and Maji, S. (2015). Bilinear cnn models for fine-grained visual recognition.
- Linsley, D., 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. *arXiv preprint arXiv:2306.03779*.
- Linsley, D., Shiebler, D., Eberhardt, S., and Serre, T. (2018). Learning what and where to attend. *arXiv preprint arXiv:1805.08819*.

- Liu, L., Chen, J., Fieguth, P., Zhao, G., Chellappa, R., and Pietikäinen, M. (2019). From bow to cnn: Two decades of texture representation for texture classification. *International Journal of Computer Vision*, 127(1):74–109.
- Liu, L., Fieguth, P., Wang, X., Pietikäinen, M., and Hu, D. (2016). Evaluation of lbp and deep texture descriptors with a new robustness benchmark. In *European Conference on Computer Vision*, pages 69–86. Springer.
- Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., and Guo, B. (2021). Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022.
- Long, J., Shelhamer, E., and Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440.
- Loshchilov, I. and Hutter, F. (2019). Decoupled weight decay regularization. In *Int. Conf. Learn. Represent.*
- 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.
- Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. (2017). Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*.
- 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.
- Margalit, E., Lee, H., Finzi, D., DiCarlo, J. J., Grill-Spector, K., and Yamins, D. L. (2023). A unifying principle for the functional organization of visual cortex. *bioRxiv*, pages 2023–05.
- 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.
- McLean, J. and Palmer, L. A. (1989). Contribution of linear spatiotemporal receptive field structure to velocity selectivity of simple cells in area 17 of cat. *Vision research*, 29(6):675–679.
- Mehrer, J., Spoerer, C. J., Kriegeskorte, N., and Kietzmann, T. C. (2020). Individual differences among deep neural network models. *bioRxiv*.
- Mishkin, M. and Ungerleider, L. G. (1982). Contribution of striate inputs to the visuospatial functions of parieto-preoccipital cortex in monkeys. *Behavioural brain research*, 6(1):57–77.
- Misra, I., Zitnick, C. L., and Hebert, M. (2016). Shuffle and learn: unsupervised learning using temporal order verification. In *European conference on computer vision*, pages 527–544. Springer.

- Morgado, P., Vasconcelos, N., and Misra, I. (2021). Audio-visual instance discrimination with cross-modal agreement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12475–12486.
- Movshon, J. A., Thompson, I. D., and Tolhurst, D. J. (1978a). Receptive field organization of complex cells in the cat’s striate cortex. *The Journal of physiology*, 283(1):79–99.
- Movshon, J. A., Thompson, I. D., and Tolhurst, D. J. (1978b). Spatial summation in the receptive fields of simple cells in the cat’s striate cortex. *The Journal of physiology*, 283(1):53–77.
- Muttenthaler, L., Dippel, J., Linhardt, L., Vandermeulen, R. A., and Kornblith, S. (2022). Human alignment of neural network representations. *arXiv preprint arXiv:2211.01201*.
- Muzellec, B. and Cuturi, M. (2018). Generalizing point embeddings using the wasserstein space of elliptical distributions. In *Advances in Neural Information Processing Systems*, pages 10237–10248.
- Nair, S., Rajeswaran, A., Kumar, V., Finn, C., and Gupta, A. (2022). R3m: A universal visual representation for robot manipulation. *arXiv preprint arXiv:2203.12601*.
- Nayebi, A., Kong, N. C., Zhuang, C., Gardner, J. L., Norcia, A. M., and Yamins, D. L. (2023a). Mouse visual cortex as a limited resource system that self-learns an ecologically-general representation. *PLOS Computational Biology*, 19(10):e1011506.
- Nayebi, A., Rajalingham, R., Jazayeri, M., and Yang, G. R. (2023b). Neural foundations of mental simulation: Future prediction of latent representations on dynamic scenes. *arXiv preprint arXiv:2305.11772*.
- Nenadic, Z. (2007). Information discriminant analysis: Feature extraction with an information-theoretic objective. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (8).
- Nili, H., Wingfield, C., Walther, A., Su, L., Marslen-Wilson, W., and Kriegeskorte, N. (2014). A toolbox for representational similarity analysis. *PLoS computational biology*, 10(4).
- Noroozi, M. and Favaro, P. (2016). Unsupervised learning of visual representations by solving jigsaw puzzles. In *European Conference on Computer Vision*, pages 69–84. Springer.
- Ohshiro, T., Angelaki, D. E., and DeAngelis, G. C. (2011). A normalization model of multisensory integration. *Nature neuroscience*, 14(6):775–782.
- Ojala, T., Pietikäinen, M., and Mäenpää, T. (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (7):971–987.
- Olshausen, B. A. and Field, D. J. (1996). Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381(6583):607.

- Oord, A. v. d., Li, Y., and Vinyals, O. (2018). Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.
- Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., et al. (2023). Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*.
- Orhan, E., Gupta, V., and Lake, B. M. (2020). Self-supervised learning through the eyes of a child. *Advances in Neural Information Processing Systems*, 33:9960–9971.
- Pagan, M., Simoncelli, E. P., and Rust, N. C. (2016). Neural quadratic discriminant analysis: Non-linear decoding with v1-like computation. *Neural computation*, 28(11):2291–2319.
- Palmer, S. E., Marre, O., Berry, M. J., and Bialek, W. (2015). Predictive information in a sensory population. *Proceedings of the National Academy of Sciences*, 112(22):6908–6913.
- Pan, T., Song, Y., Yang, T., Jiang, W., and Liu, W. (2021). Videomoco: Contrastive video representation learning with temporally adversarial examples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11205–11214.
- Parthasarathy, N., Eslami, S. M. A., Carreira, J., and Hénaff, O. J. (2023a). Self-supervised video pretraining yields human-aligned visual representations.
- Parthasarathy, N., Hénaff, O. J., and Simoncelli, E. P. (2023b). Layerwise complexity-matched learning yields an improved model of cortical area v2. *arXiv preprint arXiv:2312.11436*.
- Parthasarathy, N. and Simoncelli, E. P. (2020). Self-supervised learning of a biologically-inspired visual texture model. *arXiv preprint arXiv:2006.16976*.
- Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. (2019). Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32.
- Pathak, D., Girshick, R., Dollár, P., Darrell, T., and Hariharan, B. (2017). Learning features by watching objects move. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2701–2710.
- 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(1):49–70.
- Qian, R., Meng, T., Gong, B., Yang, M.-H., Wang, H., Belongie, S., and Cui, Y. (2021). Spatiotemporal contrastive video representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6964–6974.
- Qian, Y., Vazquez, E., and Sengupta, B. (2017). Differential geometric retrieval of deep features. In *2017 IEEE International Conference on Data Mining Workshops (ICDMW)*, pages 539–544. IEEE.

- Qin, L., Zheng, Q., Jiang, S., Huang, Q., and Gao, W. (2008). Unsupervised texture classification: Automatically discover and classify texture patterns. *Image and Vision Computing*, 26(5):647–656.
- Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. (2021). Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR.
- Raghu, P., Poongodi, R., and Yegnanarayana, B. (1997). Unsupervised texture classification using vector quantization and deterministic relaxation neural network. *IEEE Transactions on Image Processing*, 6(10):1376–1387.
- Rajalingham, R., Issa, E. B., Bashivan, P., Kar, K., Schmidt, K., and DiCarlo, J. J. (2018). Large-scale, high-resolution comparison of the core visual object recognition behavior of humans, monkeys, and state-of-the-art deep artificial neural networks. *Journal of Neuroscience*, 38(33):7255–7269.
- 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.
- Recasens, A., Luc, P., Alayrac, J.-B., Wang, L., Hemsley, R., Strub, F., Tallec, C., Malinowski, M., Patraucean, V., Altché, F., Valko, M., Grill, J.-B., van den Oord, A., and Zisserman, A. (2021). Broaden your views for self-supervised video learning. In *Int. Conf. Comput. Vis.*
- Ren, Y. and Bashivan, P. (2023). How well do models of visual cortex generalize to out of distribution samples? *bioRxiv*, pages 2023–05.
- Reynolds, J. H. and Heeger, D. J. (2009). The normalization model of attention. *Neuron*, 61(2):168–185.
- Riesenhuber, M. and Poggio, T. (1999). Hierarchical models of object recognition in cortex. *Nature neuroscience*, 2(11):1019–1025.
- 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.
- Robinson, J., Sun, L., Yu, K., Batmanghelich, K., Jegelka, S., and Sra, S. (2021). Can contrastive learning avoid shortcut solutions? *Advances in neural information processing systems*, 34:4974–4986.
- Rodriguez, M. X. B., Gruson, A., Polania, L., Fujieda, S., Prieto, F., Takayama, K., and Hachisuka, T. (2020). Deep adaptive wavelet network. In *The IEEE Winter Conference on Applications of Computer Vision*, pages 3111–3119.
- Rowekamp, R. J. and Sharpee, T. O. (2017). Cross-orientation suppression in visual area v2. *Nature communications*, 8(1):15739.

- Rusak, E., Schott, L., Zimmermann, R. S., Bitterwolf, J., Bringmann, O., Bethge, M., and Brendel, W. (2020). A simple way to make neural networks robust against diverse image corruptions. In *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, pages 53–69. Springer.
- Rust, N. C. and DiCarlo, J. J. (2010). Selectivity and tolerance (“invariance”) both increase as visual information propagates from cortical area v4 to it. *Journal of Neuroscience*, 30(39):12978–12995.
- Sanghavi, S. and DiCarlo, J. J. (2021). Sanghavi2020: Documentation pdf of dataset. <https://doi.org/10.17605/OSF.IO/CHWDK>.
- Sanghavi, S., Jozwik, K. M., and DiCarlo, J. J. (2021a). Sanghavijozwik2020: Documentation pdf of dataset. <https://doi.org/10.17605/OSF.IO/FHY36>.
- Sanghavi, S., Murty, N. A. R., and DiCarlo, J. J. (2021b). Sanghavimurty2020: Documentation pdf of dataset. <https://doi.org/10.17605/OSF.IO/FCHME>.
- Schiller, P. H. and Malpeli, J. G. (1977). The effect of striate cortex cooling on area 18 cells in the monkey. *Brain research*.
- 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*.
- Schütt, H. H., Kipnis, A. D., Diedrichsen, J., and Kriegeskorte, N. (2023). Statistical inference on representational geometries. *Elife*, 12:e82566.
- Schwartz, O. and Simoncelli, E. P. (2001). Natural signal statistics and sensory gain control. *Nature neuroscience*, 4(8):819–825.
- Sermanet, P., Lynch, C., Chebotar, Y., Hsu, J., Jang, E., Schaal, S., Levine, S., and Brain, G. (2018). Time-contrastive networks: Self-supervised learning from video. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1134–1141. IEEE.
- Shankar, V., Dave, A., Roelofs, R., Ramanan, D., Recht, B., and Schmidt, L. (2021). Do image classifiers generalize across time? In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9661–9669.
- Shapley, R. and Victor, J. D. (1979). The contrast gain control of the cat retina. *Vision research*, 19(4):431–434.
- Sharma, Y., Zhu, Y., Russell, C., and Brox, T. (2022). Pixel-level correspondence for self-supervised learning from video. *arXiv preprint arXiv:2207.03866*.

- Siddiqui, S. A., Krueger, D., LeCun, Y., and Deny, S. (2023). Blockwise self-supervised learning at scale. *arXiv preprint arXiv:2302.01647*.
- Sifre, L. and Mallat, S. (2013). Rotation, scaling and deformation invariant scattering for texture discrimination. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1233–1240.
- 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.
- Simonyan, K. and Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. *International Conference on Learning Representations (ICLR)*.
- Sincich, L. C. and Horton, J. C. (2005). The circuitry of v1 and v2: integration of color, form, and motion. *Annu. Rev. Neurosci.*, 28:303–326.
- Song, Y., Zhang, F., Li, Q., Huang, H., O’Donnell, L. J., and Cai, W. (2017). Locally-transferred fisher vectors for texture classification. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4912–4920.
- Soomro, K., Zamir, A. R., and Shah, M. (2012). Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*.
- Spelke, E. S. (1990). Principles of object perception. *Cognitive science*, 14(1):29–56.
- Spelke, E. S. and Kinzler, K. D. (2007). Core knowledge. *Developmental science*, 10(1):89–96.
- Srivastava, N., Mansimov, E., and Salakhudinov, R. (2015). Unsupervised learning of video representations using lstms. In *International conference on machine learning*, pages 843–852. PMLR.
- Subramanian, A., Sizikova, E., Majaj, N. J., and Pelli, D. G. (2023). Spatial-frequency channels, shape bias, and adversarial robustness. *arXiv preprint arXiv:2309.13190*.
- Sucholutsky, I., Muttenthaler, L., Weller, A., Peng, A., Bobu, A., Kim, B., Love, B. C., Grant, E., Achterberg, J., Tenenbaum, J. B., et al. (2023). Getting aligned on representational alignment. *arXiv preprint arXiv:2310.13018*.
- Sullivan, J., Mei, M., Perfors, A., Wojcik, E., and Frank, M. C. (2021). Saycam: A large, longitudinal audiovisual dataset recorded from the infant’s perspective. *Open mind*, 5:20–29.
- Sun, C., Shrivastava, A., Singh, S., and Gupta, A. (2017). Revisiting unreasonable effectiveness of data in deep learning era. In *Proceedings of the IEEE international conference on computer vision*, pages 843–852.
- 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*.

- Taori, R., Dave, A., Shankar, V., Carlini, N., Recht, B., and Schmidt, L. (2020). Measuring robustness to natural distribution shifts in image classification. *Advances in Neural Information Processing Systems*, 33:18583–18599.
- Tian, Y., Henaff, O. J., and van den Oord, A. (2021). Divide and contrast: Self-supervised learning from uncurated data. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10063–10074.
- Tian, Y., Krishnan, D., and Isola, P. (2020). Contrastive multiview coding. In *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16*, pages 776–794. Springer.
- Tian, Z., Shen, C., Chen, H., and He, T. (2019). FCOS: Fully convolutional one-stage object detection. In *Int. Conf. Comput. Vis.*
- Tramèr, F., Papernot, N., Goodfellow, I., Boneh, D., and McDaniel, P. (2017). The space of transferable adversarial examples. *arXiv preprint arXiv:1704.03453*.
- Treue, S. and Trujillo, J. C. M. (1999). Feature-based attention influences motion processing gain in macaque visual cortex. *Nature*, 399(6736):575–579.
- Tschannen, M., Djolonga, J., Ritter, M., Mahendran, A., Houlsby, N., Gelly, S., and Lucic, M. (2020). Self-supervised learning of video-induced visual invariances. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13806–13815.
- Ulyanov, D., Vedaldi, A., and Lempitsky, V. (2017). Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6924–6932.
- Ungerleider, L. G. and Haxby, J. V. (1994). ‘what’ and ‘where’ in the human brain. *Current opinion in neurobiology*, 4(2):157–165.
- Valente, A. C., Perez, F. V., Megeto, G. A., Cascone, M. H., Gomes, O., Paula, T. S., and Lin, Q. (2019). Comparison of texture retrieval techniques using deep convolutional features. *Electronic Imaging*, 2019(8):406–1.
- Van der Maaten, L. and Hinton, G. (2008). Visualizing data using t-sne. *Journal of machine learning research*, 9(11).
- Van Hateren, J. H. and van der Schaaf, A. (1998). Independent component filters of natural images compared with simple cells in primary visual cortex. *Proceedings of the Royal Society of London. Series B: Biological Sciences*, 265(1394):359–366.
- Wallisch, P. and Movshon, J. A. (2008). Structure and function come unglued in the visual cortex. *Neuron*, 60(2):195–197.

- Wang, J., Gao, Y., Li, K., Lin, Y., Ma, A. J., Cheng, H., Peng, P., Huang, F., Ji, R., and Sun, X. (2021a). Removing the background by adding the background: Towards background robust self-supervised video representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11804–11813.
- Wang, X. and Gupta, A. (2015). Unsupervised learning of visual representations using videos. In *Proceedings of the IEEE international conference on computer vision*, pages 2794–2802.
- Wang, X., Jabri, A., and Efros, A. A. (2019). Learning correspondence from the cycle-consistency of time. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2566–2576.
- Wang, X., Zhang, R., Shen, C., Kong, T., and Li, L. (2021b). Dense contrastive learning for self-supervised visual pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3024–3033.
- Williams, A. H., Kunz, E., Kornblith, S., and Linderman, S. (2021). Generalized shape metrics on neural representations. *Advances in Neural Information Processing Systems*, 34:4738–4750.
- Willmore, B. D., Prenger, R. J., and Gallant, J. L. (2010). Neural representation of natural images in visual area v2. *Journal of Neuroscience*, 30(6):2102–2114.
- Wiskott, L. and Sejnowski, T. J. (2002). Slow feature analysis: Unsupervised learning of invariances. *Neural computation*, 14(4):715–770.
- Wu, H. and Wang, X. (2021). Contrastive learning of image representations with cross-video cycle-consistency. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10149–10159.
- Wu, S., Li, X., and Wang, X. (2020). IoU-aware single-stage object detector for accurate localization. *Image and Vision Computing*.
- 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., Liu, Y., Zhou, B., Jiang, Y., and Sun, J. (2018). Unified perceptual parsing for scene understanding. In *Proceedings of the European conference on computer vision (ECCV)*, pages 418–434.
- Xie, J., Zhan, X., Liu, Z., Ong, Y., and Loy, C. C. (2021a). Unsupervised object-level representation learning from scene images. *Advances in Neural Information Processing Systems*, 34.
- Xie, Q., Luong, M.-T., Hovy, E., and Le, Q. V. (2020). Self-training with noisy student improves imagenet classification. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10687–10698.

- Xie, Z., Lin, Y., Yao, Z., Zhang, Z., Dai, Q., Cao, Y., and Hu, H. (2021b). Self-supervised learning with swin transformers. *arXiv preprint arXiv:2105.04553*.
- Xie, Z., Lin, Y., Zhang, Z., Cao, Y., Lin, S., and Hu, H. (2021c). Propagate yourself: Exploring pixel-level consistency for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16684–16693.
- Xiong, Y., Ren, M., and Urtasun, R. (2020). Loco: Local contrastive representation learning. *Advances in neural information processing systems*, 33:11142–11153.
- Xiong, Y., Ren, M., Zeng, W., and Urtasun, R. (2021). Self-supervised representation learning from flow equivariance. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10191–10200.
- Xu, J. and Wang, X. (2021). Rethinking self-supervised correspondence learning: A video frame-level similarity perspective. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10075–10085.
- Xue, J., Zhang, H., and Dana, K. (2018). Deep texture manifold for ground terrain recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 558–567.
- Xue, J., Zhang, H., Dana, K., and Nishino, K. (2017). Differential angular imaging for material recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 764–773.
- 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.
- Yang, C., Xu, Y., Dai, B., and Zhou, B. (2020a). Video representation learning with visual tempo consistency. *arXiv preprint arXiv:2006.15489*.
- Yang, Q., Walker, E., Cotton, R. J., Tolia, A. S., and Pitkow, X. (2020b). Revealing nonlinear neural decoding by analyzing choices. *BioRxiv*, page 332353.
- You, Y., Gitman, I., and Ginsburg, B. (2017). Large batch training of convolutional networks. *arXiv preprint arXiv:1708.03888*.
- 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.
- Zhai, W., Cao, Y., Zhang, J., and Zha, Z.-J. (2019). Deep multiple-attribute-perceived network for real-world texture recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3613–3622.

- Zhang, R. (2019). Making convolutional networks shift-invariant again. In *International conference on machine learning*, pages 7324–7334. PMLR.
- Zhang, S., Chi, C., Yao, Y., Lei, Z., and Li, S. Z. (2020a). Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In *IEEE Conf. Comput. Vis. Pattern Recog.*
- Zhang, W., Ma, K., Zhai, G., and Yang, X. (2020b). Uncertainty-aware blind image quality assessment in the laboratory and wild. *arXiv preprint arXiv:2005.13983*.
- Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., and Torralba, A. (2017). Scene parsing through ade20k dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 633–641.
- Zhuang, C., Xiang, Z., Bai, Y., Jia, X., Turk-Browne, N., Norman, K., DiCarlo, J. J., and Yamins, D. (2022). How well do unsupervised learning algorithms model human real-time and life-long learning? *Advances in Neural Information Processing Systems*, 35:22628–22642.
- 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.
- Zhuang, C., Zhai, A. L., and Yamins, D. (2019). Local aggregation for unsupervised learning of visual embeddings. *arXiv preprint arXiv:1903.12355*.
- Ziemba, C. M. (2016). Neural representation and perception of naturalistic image structure. *Unpublished doctoral dissertation*. New York: Center for Neural Science, New York University.
- 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.