Comparing brain-like representations learned by vanilla, residual, and recurrent CNN architectures

Cara Van Uden

Senior Thesis (High Honors) in Computer Science, Dartmouth College

Dartmouth Computer Science Technical Report TR2019-871

May 30, 2019

Contents

1 Introduction **4**

2 Background **6**

2.1 What do we know about human vision? . . . 6

2.2 Hyperalignment . . . 6

2.3 DNN models for vision and object recognition . . . 8

2.3.1 VGG . . . 8

2.3.2 ResNet . . . 8

2.3.3 DenseNet . . . 8

2.3.4 CORnet . . . 9

2.3.4.1 CORnet-Z . . . 10

2.3.4.2 CORnet-R . . . 10

2.3.4.3 CORnet-S . . . 10

2.4 Representational similarity analysis . . . 11

2.4.1 Pearson correlation . . . 11

2.4.2 Centered kernel alignment . . . 12

2.5 Forward encoding . . . 13

3 Related work **14**

4 Previous work: Why use hyperalignment when comparing representations? **16**

4.1 Improved model performance . . . 17

4.2 Improved spatial specificity . . . 18

5 Materials and methods **21**

5.1 fMRI data . . . 22

5.1.1 Participants . . . 22

5.1.2 Stimuli and design . . . 22

5.1.3 MRI acquisition . . . 22

5.1.4 MRI preprocessing . . . 23

5.1.5 Hyperalignment . . . 23

5.2 CNN model data . . . 24

5.2.1 ImageNet dataset and ILSVRC . . . 24
5.2.2 Pretrained models . . . 24
5.2.3 Image activation extraction and preprocessing . . . 25
5.2.4 Representational similarity analysis . . . 25
5.2.5 Forward encoding . . . 26

6 Results **27**
6.1 Representational similarity analysis . . . 27
6.2 Forward encoding . . . 31

7 Discussion **34**
7.1 Two approaches to measuring "brain-like" representation similarity . . . 35
7.2 CKA as a similarity metric for representational similarity analysis . . . 35
7.3 Residual and recurrent CNNs learn more brain-like representations . . . 36
7.3.1 Representational similarity analysis . . . 36
7.3.2 Forward encoding . . . 36
7.4 Why do early visual areas have such low similarities and prediction scores? . . . 37
7.5 Using the simplest models possible as comparators . . . 37
7.6 More "brain-like" (or just "better") features? . . . 38

8 Future work **39**

9 Code and data availability **39**

10 Acknowledgements **40**

1 Introduction

Humans have an apparently effortless ability to rapidly recognize objects in various naturalistic visual contexts. Understanding how the brain implements this ability is a topic that unites vision researchers in both computer science and neuroscience. With the recent explosion in the field of deep learning, deep neural networks (DNNs) have been increasingly extended to various applications in understanding the brain. Recently, deep convolutional neural networks (CNNs), optimized to solve object (or action) recognition tasks, have excelled as models of visual processing in the human ventral (or dorsal) stream, respectively. CNNs whose parameters are optimized to achieve high performance tend to develop internal "neural" representations that are close to those observed in low-level (V1), intermediate (V2), and the highest levels of the ventral visual stream (V4 and IT) [69] [70]. Although they are inspired by hierarchical processing in biological visual systems [26], CNNs optimized for vision differ from the brain in many ways, most notably in terms of depth and recurrence.

While state-of-the-art deep neural networks typically have between 100 and 150 layers, and can have as many as 1202 layers [23], biological systems seem to have two orders of magnitude less, if we make the customary assumption that a layer in the neural network architecture corresponds to a cortical area. In fact, there are about half a dozen areas in the ventral stream of visual cortex from the retina to the inferior temporal cortex. The evolutionary advantage of having fewer layers is apparent: it supports rapid (100msec from image onset to meaningful information in IT neural population) visual recognition, which is a key ability of human and non-human primates [61]. It is intriguingly possible to account for this discrepancy by taking into account recurrent connections within and between visual areas in the brain.

Unlike CNNs, the brain achieves robust visual perception by using feedforward, feedback and recurrent connections [9] [64]. In the brain, information is not only processed through a bottom-up pathway running from lower to higher visual areas, like in feedforward neural networks. Recurrent connections, or connections between nodes that form a directed graph along a temporal sequence, allow the network to exhibit temporal dynamic behavior. Recurrent neural networks can use their internal state (memory) to process and understand sequences of inputs. In addition to these recurrent connections, the brain also has feedback connections—top-down pathways running from higher to lower visual areas. Such bi-directional and recurrent processes enable humans to perform a wide range of visual tasks, including object recognition, and are believed to mediate some attentional effects [3] and even learning—such as backpropagation [40]. For human vision, feedforward processing is essential for object recognition [7]. However, recurrent and feedback processing improves object recognition and enables cognitive processes to influence perception [43] [68].

How can we connect the feedforward structure we see in today's DNNs with the more complex recurrent structure we see in the brain? Recent work by Liao and Poggio (2016) has found a conneection between the brain's recurrent networks and the state-of-the-art residual networks common in the machine learning community. If we "unroll" (through time) the recurrent computations carried out by the visual cortex, we find an equivalent "ultra-deep" feedforward residual network with the same (that is, shared) weights among the layers. The number of layers in the unrolled network corresponds to the discrete time iterations of the dynamical system. This deep residual network represents a more appropriate comparison with the state-of-the-art computer vision models [41].

Though it has been hypothesized that state-of-the art residual networks approximate the recurrent visual system, it is yet to be seen if the representations learned by these "biologically inspired" CNNs actually have closer representations to neural data. It is likely that CNNs and DNNs that are most functionally similar to the brain will contain mechanisms that are most like those used by the brain. In this thesis, we investigate how different CNN architectures approximate the representations learned through the ventral—object recognition and processing—stream of the brain. We specifically evaluate how recent approximations of biological neural recurrence—such as residual connections, dense residual connections, and a biologically-inspired implementation of recurrence—affect the representations learned by each CNN. We first investigate the representations learned by layers throughout a few state-of-the-art CNNs—VGG-19 (vanilla CNN), ResNet-152 (CNN with residual connections), and DenseNet-161 (CNN with dense connections). To control for differences in model depth, we then extend this analysis to the CORnet family of biologically-inspired CNN models with matching high-level architectures. The CORnet family has three models: a vanilla CNN (CORnet-Z), a CNN with biologically-valid recurrent dynamics (CORnet-R), and a CNN with both recurrent and residual connections (CORnet-S).

We compare the representations of these six models to functionally aligned (with hyperalignment) fMRI brain data acquired during a naturalistic visual task. We take two approaches to comparing these CNN and brain representations. We first use forward encoding, a predictive approach that uses CNN features to predict neural responses across the whole brain. We next use representational similarity analysis (RSA) and centered kernel alignment (CKA) to measure the similarities in representation within CNN layers and specific brain ROIs. We show that, compared to vanilla CNNs, CNNs with residual and recurrent connections exhibit representations that are even more similar to those learned by the human ventral visual stream. We also achieve state-of-the-art forward encoding and RSA performance with the residual and recurrent CNN models.

2 Background

2.1 What do we know about human vision?

Human beings are extremely adept at recognizing complex objects based on elementary visual sensations. Object recognition appears to be solved in the primate brain via a cascade of neural computations along the visual ventral stream that represents increasingly complex stimulus features, which derive from the retinal input [65]. That is, neurons in early visual areas have smaller receptive fields (RFs) and respond to simple features such as edge orientation [25], whereas neurons further along the ventral pathway have larger RFs and are more invariant to transformations and can be selective for complex shapes [28]. Despite a consensus concerning a steady progression in feature complexity, it remains nontrivial to quantify such a progression across multiple regions in the human ventral stream. Furthermore, while the RFs in early visual area V1 have been characterized in terms of preferred orientation, location, and spatial frequency [32], exactly what stimulus features are represented in downstream areas is less clear [5].

Most neuroscience research on human perception has focused on early vision, using highly-controlled stimuli such as orientation and motion direction. Even tasks used to investigate later-stage visual processing are often unnaturally constrained. For example, DiCarlo et al (2012) defined the core object recognition (COR) task, in which each subject must discriminate from all other possible objects a dominant object with the viewing duration of a natural fixation (about 200 ms) in the central visual field under high view and background variation [7]. Humans, however, perceive and act on the world in terms of both semantically rich representations and complex behavioral goals. However, naturalistic stimuli serve to convey richer perceptual and semantic information, and have been shown to reliably drive neural responses [22]. Natural vision paradigms can provide complementary insights relative to traditional visual experiments.

2.2 Hyperalignment

The brain encodes complex naturalistic visual information in high-dimensional representational spaces, called representational geometries, that are grounded in the collective activity of distributed populations of neurons [21]. Multivariate decoding analyses of human neuroimaging data have allowed us to leverage distributed patterns of cortical activation to provide a window into the representation of high-level semantic information [22]. However, these functional topographies are not aligned across individual brains. Because this mapping between anatomy and function is unique across individuals, anatomical alignment can only approximate functional alignment. In previous work, Haxby et al 2011 created a high-dimensional model of the representational space in human ventral temporal (VT) cortex [22]. Dimensions are response-tuning functions that

are common across individuals, and patterns of response are modeled as weighted sums of basis patterns associated with these response tunings. They mapped response-pattern vectors, measured with fMRI, from individual subjects' voxel spaces into this common model space. Hyperalignment effectively rotates each participant's idiosyncratic anatomical space into optimal alignment (with some local searchlight constraints) based on their responses to rich, naturalistic stimulus. The searchlight hyperalignment algorithm learns a locally constrained whole-cortex transformation rotating each individual's anatomical coordinate space into a common space that optimizes the correspondence of representational geometry (in this case, the response patterns to the movie stimulus at each time point) across brains [19]. In the current project, we use nested cross-validation to derive hyperalignment parameters and transform the representational geometries across the whole brain into the resulting shared feature space. See Figure 1 for a comparison of inter-subject correlations and representational geometries before and after hyperalignment.

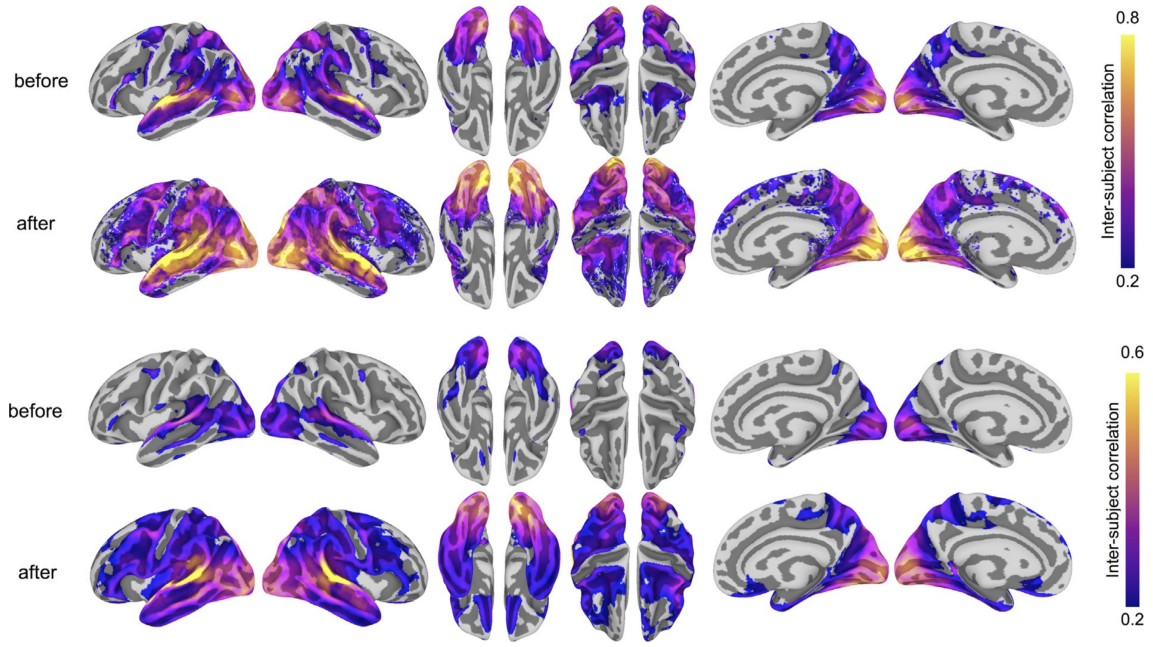

Figure 1: Hyperalignment improves inter-subject correlation (ISC) of response profiles and representational geometry. (A) ISC of vertex-wise response time series before and after hyperalignment. Colored vertices reflect the mean ISC across subjects, thresholded at a mean correlation of 0.2. ISCs are highest in the superior temporal gyrus (in the vicinity of auditory cortex), as well as the dorsal and ventral visual pathways, comprising early visual, lateral occipitotemporal, ventral temporal, posterior parietal, and intraparietal cortices. (B) ISC of searchlight representational geometries (time-point RDMs in 9 mm radius searchlights) before and after hyperalignment. Colored vertices reflect the mean pairwise correlation across subjects for each searchlight, thresholded at a mean correlation of 0.2, revealing a broader extent of cortex with improved alignment of functional topography after hyperalignment. ISCs were Fisher z-transformed before averaging across all subjects and inverse Fisher transformed before mapping onto the cortical surface for visualization. All maps are rendered on the fsaverage6 surface template [66]

.

2.3 DNN models for vision and object recognition

2.3.1 VGG

VGG [62] is one of the most popular "vanilla" CNN models—it won ILSVRC 2014 and was one of the first networks to make deep learning the gold standard for image recognition. The architecture of a CNN is designed to take advantage of the 2D structure of an input image (or other 2D input). This is achieved with local connections and tied weights within each convolutional "block", followed by pooling, which results in translation invariant features. In the VGG network, an image is passed through a stack of convolutional layers, each of which uses filters with a very small receptive field: 3×3 pixels (which is the smallest size to capture the notion of left/right, up/down, and center). While VGG achieves a good accuracy on the ImageNet dataset and learns unsupervised features that generalize well to other datasets, its deployment can be a problem because of huge computational requirements, both in terms of memory and time.

2.3.2 ResNet

Deep residual networks [23] were a breakthrough idea which enabled the development of much deeper networks (hundreds of layers as opposed to tens of layers). It is a generally accepted principle that deeper networks are capable of learning more complex functions and representations of the input which should lead to better performance. However, many researchers observed that adding more layers eventually had a negative effect on the final performance—this phenomenon is referred to as the degradation problem. In the degradation problem, although better parameter initialization techniques and batch normalization allow for deeper networks to converge, they often converge at a higher error rate than their shallower counterparts. In the limit, simply stacking more layers degrades the model's ultimate performance.

Residual blocks, as seen in residual networks like ResNet, offer a remedy to this degradation problem [23]. Within residual blocks, intermediate layers of a block learn a residual function with reference to the block input. The output of each residual block is its input summed with a convolution of its input. This residual function can be thought of as a refinement step, in which we learn how to adjust the input feature map for higher quality features. This compares with a "vanilla" network in which each layer is expected to learn new and distinct feature maps.

2.3.3 DenseNet

DenseNets [24] are the natural extension of ResNets. The idea behind dense convolutional networks is simple: it may be useful to reference feature maps from earlier in the network. Unlike ResNet, where each residual

block's output is its input summed with a convolution of its input, each DenseNet block's output is its input concatenated with a convolution of its input. This allows later layers within the network to directly leverage the features from earlier layers, encouraging feature reuse within the network. The authors of DenseNet state, "concatenating feature-maps learned by different layers increases variation in the input of subsequent layers and improves efficiency."

The authors refer to the number of filters used in each convolutional layer as a "growth rate", k, since each successive layer will have k more channels than the last (as a result of accumulating and concatenating all previous layers to the input). Intuitively, it seems that DenseNet would have an absurd number of parameters to support the dense connections between layers. However, because the network is capable of directly using any previous feature map, the authors found that they could work with very small output channel depths (for example, 12 filters per layer), vastly reducing the total number of parameters needed [24].

When compared with ResNet models, DenseNet achieves better performance with less complexity.

2.3.4 CORnet

From the neuroscientist's point of view, the relationship between such very deep architectures and the ventral visual pathway is incomplete in at least two ways. On one hand, current state-of-the-art DNNs appear to be too complex (e.g. now over 100 levels) compared with the relatively shallow cortical hierarchy (4-8 levels), which makes it difficult to map their elements to those in the ventral visual stream and makes it difficult to understand what they are doing. On the other hand, current state-of-the-art DNNs appear to be not complex enough in that they lack recurrent connections and the resulting neural response dynamics that are commonplace in the ventral visual stream.

Rather than just seeking high object recognition performance, the CORnet project [38] instead tries to reduce DNN models to their most important elements and then gradually build in recurrent and skip connections, while monitoring both performance and the match between each model and primate brain and behavioral data. The CORnet family of deep neural network architectures has four computational areas, conceptualized as analogous to the visual areas V1, V2, V4, and IT, and a linear category decoder that maps from the population of neurons in the model's last visual area to its behavioral choices. Each visual area implements a particular neural circuitry with neurons performing simple canonical computations: convolution, addition, nonlinearity, response normalization, or pooling over a receptive field. The decoder part of the model implements a simple linear classifier – a set of weighted linear sums with one sum for each object category.

2.3.4.1 CORnet-Z

CORnet-Z is the simplest CORnet model, derived by observing that (1) simple networks like AlexNet are already nearly as good in predicting neural responses as deeper models like VGG and (2) multiple fully-connected layers do not appear necessary to achieve good ImageNet performance, as most architectures proposed after VGG contain only a single 1000-way linear classification layer. CORnet-Z is a simple implementation of a vanilla feedforward convolutional neural network. Therefore, CORnet-Z's area circuits consist of only a single convolution, followed by a ReLU nonlinearity and max pooling.

2.3.4.2 CORnet-R

CORnet-R is a simple recurrent neural network. In standard machine learning implementations of recurrent models, information is passed instantly at each time step directly from inputs to outputs. However, CORnet-R's recurrent dynamics propagate through the network in a biologically-valid manner. In biological systems, inputs are transformed to outputs in a stepwise fashion. At t = 0, only the first area (V1 in this case) processes an input and sends the output of this computation to V2. At t = 1, V2 processes V1's output, while V1 is already processing a new input with an updated internal state. It takes several time steps for the original input to finally reach higher visual areas like IT and VT, whereas in a non-biological unrolling of a network the input reaches the highest layers at the same time as it is fed to the lowest layer.

The difference between the two kinds of unrolling becomes the most dramatic when a feedback connection is introduced. In standard recurrent networks, information from the highest layers would be sent down to the lowest layer, but since the lowest layer has already processed its inputs, this information would not be utilized. In biological systems, at t = 1 feedback from higher visual areas like V4 or VT would be combined with inputs to V1, so this feedback would readily affect further processing downstream of V1.

In CORnet-R, recurrence is introduced only within an area (no feedback connections between areas), so the particular way of unrolling has little effect (apart from consuming much more memory), but the model uses biologically-valid unrolling to make it useful for investigating neural dynamics. The input is first passed it through a convolution, followed by normalization and a nonlinearity. The state (initially zero) is added to the result and passed through another convolution, normalization, and nonlinearity, and the result is saved as a new state of the area. This model uses group normalization [67] and ReLU nonlinearity.

2.3.4.3 CORnet-S

Another model, called CORnet-S, draws inspiration from ResNet and combines CORnet-R's shallow recurrent model with residual connections. Liao and Poggio 2016 proposed that ResNet structure could be

thought of as an unrolled recurrent network, and recent studies further demonstrated that weight sharing was possible without a significant loss in object classification (CIFAR and ImageNet) performance. The model also includes a residual connection, such that the result of adding the state to the input is combined with the output of the last convolution just prior to applying a nonlinearity. Given that CORnet-S only has within-area recurrent connections, in order to minimize memory footprint this model was trained without making use of any biological network unrolling in time (but weights are still shared over repeated computations). CORnet-S commits to a straightforward mapping between model and brain areas (Layer 1 to V1, Layer 2 to V2, Layer 3 to V4, and Layer 4 to IT cortex).

2.4 Representational similarity analysis

Representational similarity analysis (RSA) [36] is commonly used, especially in systems and cognitive neuroscience, to characterize the information carried by a given representation in a brain or model. With RSA, we abstract from the activity patterns themselves by computing representational dissimilarity matrices (RDMs). These RDMs contain distance measures (usually correlation) between pairs of distributed activity patterns that represent different experimental conditions. In neuroscience, these RDMs are used to analyze the response similarity between evoked fMRI responses in selected regions-of-interest (ROIs). Note that the calculated similarity structure between conditions can itself be related to RDMs from other ROIs, producing second-level RDMs that help to understand the specific representational principles of a brain region by revealing what distinctions between stimuli are emphasized and what distinctions are de-emphasized in a specific ROI (or at a certain level of a computational model). Since the comparison of first-level RDMs does not require voxel-level correspondence, data from other sources can be easily integrated in second-level analyses, including integration of RDMs from multiple subjects, other measurement modalities, and computational models.

Measuring similarity between the representations learned by biological and artificial neural networks is an ill-defined problem, since it is not entirely clear what aspects of the representation a similarity index should focus on. We first discuss correlation as a similarity metric (the standard in neuroscience) and then discuss centered kernel alignment, a new similarity metric that consistently identifies relevant representations across different networks.

2.4.1 Pearson correlation

The motivation for using Pearson correlation distance for comparing brain-activity patterns is is to describe to what extent two experimental conditions push the baseline activity pattern in different directions in

multivariate response space: The correlation distance is 1 minus the cosine of the angle the two patterns span (after the regional-mean activation has been subtracted out from each). Correlation distance normalizes out to the regional-mean component and also normalizes out the pattern variance across voxels. However, the correlation distance is not an ideal dissimilarity measure because a large correlation distance does not indicate that two stimuli are distinctly represented. If a region does not respond to either stimulus, for example, the correlation of the two patterns (due to noise) will be close to 0 and the correlation distance will be close to 1, a high value that can be mistaken as indicating a decodable stimulus pair.

2.4.2 Centered kernel alignment

Kornblith et al (2019) proposed centered kernel alignment (CKA) as a method for comparing representations of neural networks [35].

CKA is a normalized version of the Hilbert-Schmidt Independence Criterion (HSIC) [13]. Let $K_{ij} = k(x_i, x_j)$ and let $L_{ij} = l(y_i, y_j)$ where k and l are two kernels. The empirical estimator of HSIC is:

$$HSIC(K, L) = \frac{1}{(n-1)^2} tr(KHLH) \tag{1}$$

where H is the centering matrix $H_n = I_n - \frac{1}{(n-1)^2}\mathbf{1}\mathbf{1}^T$. For linear kernels $k(x, y) = l(x, y) = x^T y$, which we use in our analysis, HSIC reduces to:

$$\frac{1}{(n-1)^2} tr(XX^T YY^T) = ||cov(X^T, Y^T)||_F^2 \tag{2}$$

As a note, distance covariance is a specific instance of HSIC with a particular kernel (Sejdinovic et al., 2013). HSIC can be made invariant to isotropic scaling through normalization. This normalized index is k centered kernel alignment [4]:

$$CKA(K, L) = \frac{HSIC(K, L)}{HSIC(K, K)HSIC(L, L)} \tag{3}$$

CKA is invariant to orthogonal transformation and isotropic scaling. Kornblith et al show that CKA outperforms other similarity metrics: it consistently identifies correspondences between layers, not only in the same network trained from different initializations, but across entirely different architectures, whereas other methods do not [35]. It is also able to discover meaningful relationships betweens layers of the same and different architectures, while other similarity metrics are not. For example, after applying CKA to ResNet layers, the authors observed an intuitive grid pattern that originates from the architecture: post-residual activations are similar to other post-residual activations, but activations within blocks are not [35].

2.5 Forward encoding

Machine learning approaches to relating stimuli and neural representations have become more common in recent years. The benefits over a traditional stimulus-contrast approach include the ability to make predictions about new datasets [51], to take a multivariate approach to fitting model weights [30], and to use multiple feature representations within a single, complex stimulus set [27]. These machine learning models come in two complementary flavors. "Encoding" models are a regression problem, where stimulus features are used to predict patterns of brain activity. Encoding models have grown in popularity in fMRI [50], electrocorticography [47], and EEG/MEG [6]. On the other hand, "decoding" models are a classification problem, where patterns of brain activity are used to predict stimulus features [50].

Encoding models are useful for exploring multiple levels of abstraction within a complex stimulus, and investigating how each affects activity in the brain. For example, natural vision is a continuous stream of input with a hierarchy of complex information embedded within it. A single glance contains many representations of information, such as brightness and color, objects, actions/movement, and semantics. The neural signal is a continuous response to this input with multiple embedded streams of information in it, due to recording the activity from many neurons spread across a relatively large region of cortex. Naturalistic stimuli pose a challenge for univariate event-related analysis, but are naturally handled in a predictive modeling framework. In the predictive modeling approach, the solution takes the form of a linear regression problem:

$$activity(t) = \Sigma_i^N feature_i(t) \cdot weight_i + error(t) \tag{4}$$

where the neural activity at time t is modeled as a weighted sum of N stimulus features. It is common to use ridge regression (penalized L_2-norm regression) to estimate neural responses. It is also common to either convolve the features using the hemodynamic response function (HRF) [42] or include several time-lagged versions of each feature [29]. Previous work performed these preprocessing steps to account for the fact that the sluggish hemodynamic response to neural activity (measured by the BOLD signal), not the actual neural activity itself, is the only thing visible to fMRI analysis.

Encoding models describe how dynamic stimulus features are encoded into patterns of neural activity. Encoding models of sensory cortex attempt to model cortical activity as a function of stimulus features. These features may be complex and applied to naturalistic stimuli, allowing one to study the brain under conditions observed in the real world. This provides a flexible framework for estimating the neural tuning to particular features, and assessing the quality of a feature set for predicting brain activity.

3 Related work

Since their inception, DNNs have been compared explicitly with their biological counterparts. Connectionist models were widely used in the 1980s to explain various psychological phenomena, particularly by the parallel distributed processing (PDP) movement, which stressed the parallel nature of neural processing and the distributed nature of neural representations [45]. For example, neural networks have been used to explain grammar acquisition [8], category learning [37], and the organization of the semantic system [56].

DNNs have also been related explicitly to brain function. For example, the perceptron has been used in the modeling of various neuronal systems, including sensorimotor learning in the cerebellum [44] and associative memory in cortex [12], sparse coding has been used to explain receptive field properties [53], topographic maps have been used to explain the formation of cortical maps [52], Hebbian learning has been used to explain neural tuning to face orientation [39], and networks trained by backpropagation have been used to model the response properties of posterior parietal neurons [71]. Furthermore, reinforcement learning algorithms used to train neural networks for action selection have strong ties with the brain's reward system [60]. It has been shown that recurrent neural networks (RNNs) trained to solve a variety of cognitive tasks using reinforcement learning replicate various phenomena observed in biological systems [63] [48]. Crucially, these efforts go beyond descriptive approaches in that they may explain why the human brain is organized in a certain manner [1].

Recent work in various domains has focused on fitting DNNs to neural responses. DNNs have been used to directly predict neural responses, but they have also been used indirectly to both predict neural responses [46] [18] and understand the representations underlying these responses. In this approach, neural networks are first trained to solve a task of interest. Subsequently, the trained network's responses are fitted to neurobehavioral data obtained as participants engage in the same task. Using this approach, deep convolutional neural networks trained on object recognition, action recognition and music tagging have been used to explain the functional organization of visual as well as auditory cortex [15] [16] [17].

We will discuss a specific example of this approach in depth. To probe how visual features of varying complexity are mapped across the cortical sheet, Güçlü and van Gerven (2015) used a vanilla CNN (CNN-S) trained on ImageNet. Like the approach in this thesis, they used the representations that emerge after training the CNN to predict BOLD responses to complex naturalistic stimuli [16]. They showed that this framework yields state-of-the-art encoding and decoding performances, improving on results from earlier studies that used nonlinear feature models as the basis for neural encoding and decoding [31-33]. Predictions were made in progressively downstream areas of the ventral stream, moving from striate area V1 along

extrastriate areas V2 and V4, all the way up to downstream area LO. Individual neural network layers were used to predict single-voxel responses to natural images. This allowed the authors to isolate different voxel groups, whose population receptive fields are best predicted by a particular neural network layer. Using this approach, the authors were able to determine how receptive field properties, such as complexity, invariance, and size, correlate with the position of voxels in the visual hierarchy. Next, by using individual features in the neural network to predict voxel responses, the authors were able to map how individual low-, mid-, and high-level stimulus features are represented across the ventral stream. This mapping procedure provided detailed insight into how stimulus features are represented across cortex and indicates that particular visual areas show a fine-grained functional specialization. Their results show that DNNs accurately predict neural responses to naturalistic stimuli and suggest that object categorization is a guiding principle for the formation of receptive field properties in ventral stream.

Güçlü and van Gerven (2015) took a predictive (encoding and decoding) approach to understanding brain and CNN representations. On the other hand, Khaligh-Razavi and Kriegeskorte (2014) used a representational similarity analysis approach to investigate a wide range of computational model representations [33]. They tested these models' categorization performance and their ability to account for the IT representational geometry. The models include well-known neuroscientific object-recognition models (HMAX, VisNet) along with several models from computer vision (SIFT, GIST, self-similarity features, and a deep convolutional neural network). They compared the representational dissimilarity matrices (RDMs) of the model representations with the RDMs obtained from human IT (measured with fMRI) and monkey IT (measured with cell recording) for the same set of stimuli. Better performing models were more similar to IT in that they showed greater clustering of representational patterns by category.

While Güçlü and van Gerven (2015) and Khaligh-Razavi and Kriegeskorte (2014) took predictive and RSA approaches, respectively, to understanding representational similarities, Schrimpf et al 2018 [59] asked a similar question to the one we ask in this thesis: as DNNs have continued to evolve, are they becoming more or less brain-like? They therefore developed Brain-Score—a composite of multiple neural (V4 and IT EEG recordings from monkeys) and behavioral benchmarks that score any DNN on how similar it is to the brain's mechanisms for object recognition—and they deployed it to evaluate a wide range of state-of-the-art DNNs. Using this scoring system, they report that: (1) DenseNet, CORnet-S and ResNet are the most brain-like DNNs, but that there remains considerable variability in neural and behavioral responses that is not predicted by any DNN, suggesting that no DNN model has yet captured all the relevant mechanisms. (2) Gains in ANN ImageNet performance led to gains on Brain-Score. However, correlation weakened at $\geq$ 70% top-1 ImageNet performance, suggesting that additional guidance from neuroscience is needed to make further advances in capturing brain mechanisms.

In the approach discussed here, and unlike previous work, we are able to leverage a huge amount of data and do not focus on any specific CNN model, brain region of interest, or similarity analysis method. We are unique in our methodological approach—first, we use hyperalignment to leverage whole-brain naturalistic data from many people (20 total), not just constrained vision data from two or three. Second, we combine predictive and RSA (with a novel, state-of-the-art similarity metric) approaches to understanding similarities in representation. This thesis differs from previous work by specifically focusing on the role of residual and recurrent connections in the representations learned by CNN models, and comparing these CNN representations to the representations in the brain. Iin addition to this main focus, we also compare the representations uncovered by the predictive and RSA approaches, and disentangle "powerful" visual features from "brain-like" ones.

4 Previous work: Why use hyperalignment when comparing representations?

In our previous work, we evaluated the impact of hyperalignment on forward encoding model performance [66]. We outlined an approach for estimating encoding models that can make detailed predictions of responses to novel stimuli in novel individuals at the specificity of cortical vertices. To accommodate idiosyncratic functional topographies, we used hyperalignment to derive transformations to map each individual's responses into a common representational space [19] [22]. We used a dynamic, naturalistic stimulus – the *Life* nature documentary narrated by David Attenborough – for the dual purpose of deriving hyperalignment transformations and fitting the encoding model. Using a naturalistic paradigm that thoroughly samples both stimulus space and neural response space is critical for robustly fitting the encoding model and ensuring the hyperalignment transformations generalize to novel experimental contexts [19] [22].

To evaluate hyperalignment in the context of encoding models, we compared within-subject encoding models and between-subject encoding models. Our model, trained on three-fourths of the movie in a left-one-out subset of participants, was used to predict responses at each cortical vertex for the left-out fourth of the movie in a left-out participant. We compared within-subject models, between-subject models using high-performing surface-based anatomical normalization [34] [10] and surface-based searchlight whole-cortex hyperalignment [19]. We modeled semantic tuning at each cortical vertex based on distributed word embeddings (word2vec; [49]) assigned to each imaging volume based on an annotation of the documentary.

We first showed that constructing between-subject models using anatomical alignment reduced the spatial specificity of vertex-wise semantic tuning relative to within-subject models. Next, we demonstrated that

hyperalignment generally leads to improved between-subject model performance, exceeding within-subject models. We have highlighted the key results below to give some insight on the benefits of using hyperalignment to functionally align fMRI data.

In this analysis, we showed that hyperalignment affords aggregation of data across individuals that aligns fine-scale variations. In constructing a common representational space, we decouple functional tuning from anatomy, registering representational geometries rather than anatomical features. This result is important for this thesis, because we are directly comparing representations in aggregated fMRI brain data to representations in deep neural networks—we do not want these representations to be confounded or concealed by differences in brain anatomy.

4.1 Improved model performance

Hyperalignment improves forward encoding model performance (see Figure 2 for example prediction performance maps for two subjects). We summarized differences in model performance across the entire cortex in two ways. To constrain our analysis to well-predicted vertices, for each subject we selected the 10,000 vertices with highest model performance separately for each model. We then considered only the union of well-predicted vertices across all three models (on average 15,724 vertices per subject, SD = 1,293 across subjects). First, for each pair of models, we computed the proportion of vertices with greater model prediction performance (i.e., correlation between predicted and actual time series for the test data) for one model relative to the other. We calculated these proportions per subject, then computed a paired t-test to assess statistical significance per model pair. When comparing the model performance for the within-subject and the between-subject models, the between-subject model using anatomical alignment yielded higher correlations in 50.7% of selected cortical vertices [$t(17) = 0.717$, $p = 0.483$]. The between-subject model using hyperalignment yielded better performance than the within-subject model in 58.9% of selected cortical vertices [$t(17) = 8.539$, $p<0.001$]. The between-subject model using hyperalignment also yielded better performance than the between-subject model using anatomical alignment [58.7% of cortical vertices; $t(17) = 20.736$, $p<0.001$].

Second, we assessed the difference in model prediction performance averaged across the same subset of well-predicted vertices. The between-subject model using anatomical alignment performed similarly to the within-subject model [0.120 and 0.124, respectively; $t(17) = 1.866$, $p = 0.079$]. The between-subject model using hyperalignment performed better than the within-subject model [0.135 and 0.124, respectively; $t(17) = 8.547$, $p<0.001$]. Additionally, hyperalignment exceeded anatomical alignment when comparing the performance of between-subject models [0.135 and 0.120, respectively; $t(17) = 15.800$, $p<0.001$]. To visualize

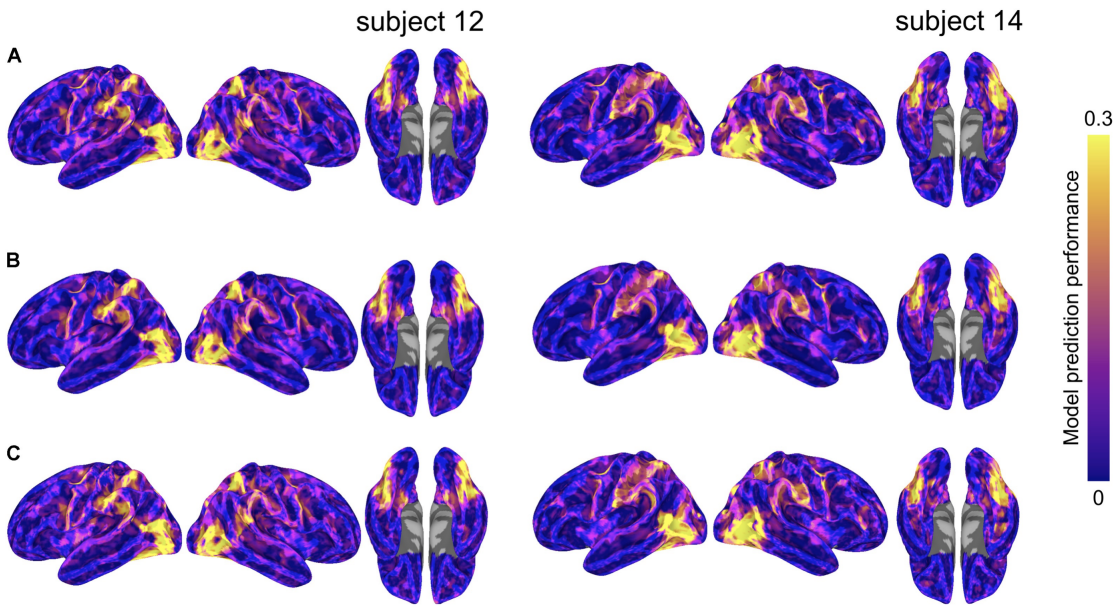

Figure 2: Model prediction performance maps for two example subjects (left and right). Colored vertices reflect the Pearson correlation between the predicted and actual time series averaged across the four test runs. Three types of models are presented: (A) within-subject model performance maps; (B) between-subject anatomically aligned model performance maps; and (C) between-subject hyperaligned model performance maps. Maximum correlations are 0.37 (both subjects 12 and 14, area LO, all three maps) and correlations at the 95% level for vertices within the mask ranged from 0.29 to 0.31. All maps are unthresholded and uncorrected for multiple tests. Note that all of these prediction maps look very similar across the three types of models; however, see Figure 3 for statistically significant differences in performance between the three types of models [66].

differences in model performance, we compared model performance maps on the cortical surface (Figure 3). We computed vertex-wise paired t-tests for each of the three model comparisons. For visualization, we thresholded maps at a t-value of 2.11 ($p<0.05$, two-tailed test, uncorrected for multiple tests).

4.2 Improved spatial specificity

Hyperalignment effectively recovers the specificity of within-subject models, allowing us to leverage a large volume of group data for individualized prediction at the specificity of individual voxels or cortical vertices. To compare the spatial specificity of semantic tuning across model types, we computed the spatial point spread function (PSF) of the semantic model predictions (Figure 4). To constrain our analysis to well-predicted vertices, for each subject we again selected the 10,000 vertices with highest model performance separately for each model and considered only the union of these well-predicted vertices across all three models.

For each well-predicted vertex, we computed the model prediction performance (Pearson correlation

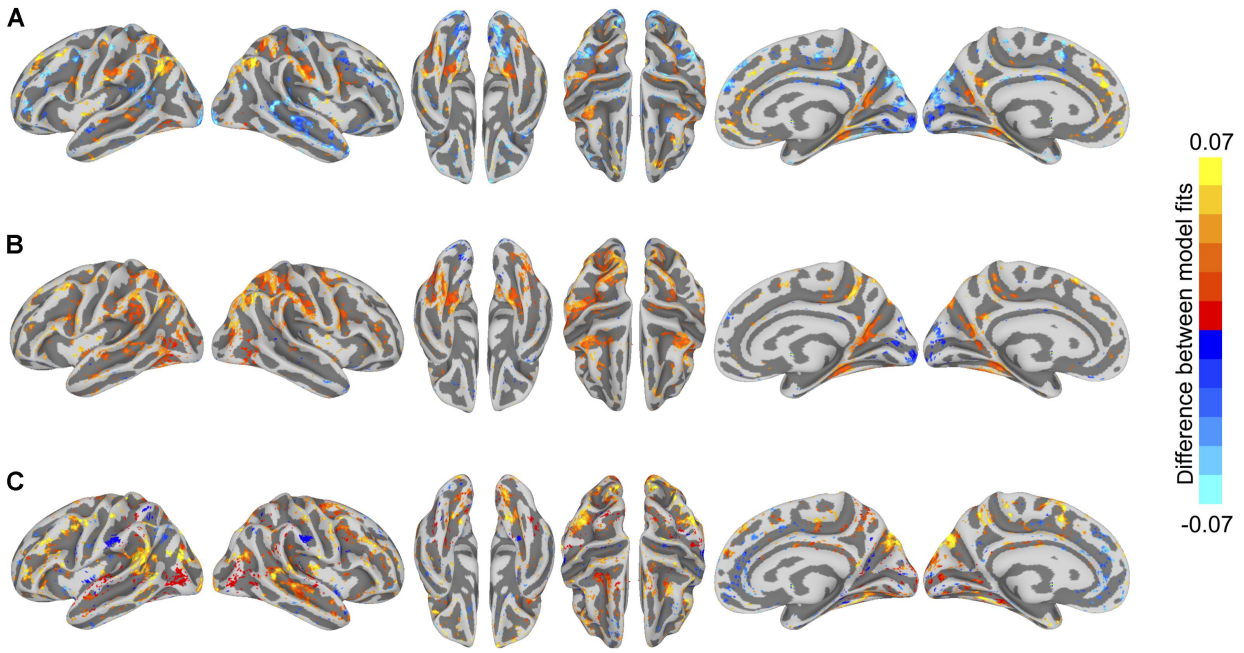

Figure 3: Differences in semantic encoding model performance maps. (A) Paired differences in model performance between between-subject models using anatomical normalization and within-subject models. Warm colors indicate vertices where the anatomically aligned between-subject model performance exceeds within-subject model performance, and cool colors indicate where within-subject model performance exceeds anatomically aligned between-subject model performance. (B) Paired differences in model performance between between-subject models using hyperalignment and within-subject models. Warm colors indicate vertices where the hyperaligned between-subject model performance exceeds within-subject model performance, and cool colors indicate where within-subject model performance exceeds hyperaligned between-subject model performance. (C) Paired differences in model performance for between-subject models using hyperalignment and anatomical normalization. Warm colors indicate vertices where the hyperaligned between-subject model performance exceeds anatomically aligned between-subject model performance, and cool colors indicate where anatomically aligned between-subject model performance exceeds hyperaligned between-subject model performance. Colored vertices reflect mean paired differences in model performance, thresholded at an absolute t-value of $t(17) = 2.11$, $p<0.05$, uncorrected for multiple comparisons [66].

between predicted and actual time series) for that vertex, and for neighboring vertices using the same prediction equation at 2 mm intervals up to 12 mm. That is, we used the encoding model at each vertex to predict the actual time series at neighboring, increasingly distant vertices. Each "ring" of vertices (e.g., the ring of vertices at a radius 10–12 mm from the central vertex of interest) was 2 mm wide and excluded vertices sampled at smaller radii. For a given ring of vertices, model performance was computed at each vertex in the ring and averaged across those vertices. Model performances at each radius per vertex were then averaged across the set of selected well-predicted vertices. To statistically assess PSFs, we computed bootstrapped confidence intervals around the model performance estimates at each radius by resampling subjects with replacement. To quantify the decline in spatial specificity of model performance over radii, we fit a logarithmic function to the PSF for each model at the midpoint of each ring (i.e., the vertex of

interest, 1 mm, 3 mm, etc.) and reported the slope of this fit. The spatial point-spread function of the model predictions for the between-subject model using anatomical alignment was relatively flat [negative slope of the logarithmic fit = 0.0172 (0.0166, 0.0178)]. The within-subject and hyperaligned between-subject models had steeper slopes [0.0447 (0.0409, 0.0488) and 0.0396 (0.0375, 0.0418), respectively; both $p < 0.001$], indicating greater spatial specificity in semantic tuning.

The prediction performance maps for each model varied in their spatial smoothness (Figure 5). We computed the full width at half maximum (FWHM) of the model performance maps using SUMA's Surf-FWHM. Spatial smoothness was computed per run in each hemisphere in each participant and averaged across hemispheres. Model performance maps for the between-subject model using anatomical alignment were significantly more spatially blurred than for the within-subject model [5.034 and 4.428 mm FWHM, respectively; $t(17) = 27.617$, $p < 0.001$]. The between-subject model using hyperalignment recovered the spatial specificity of the within-subject maps, and in fact yielded less smooth model performance maps (3.697 mm FWHM) than the within-subject model [$t(17) = 24.650$, $p < 0.001$].

We also assessed how well the between-subject model performance maps approximated the spatial organization of the within-subject model performance maps by computing the Pearson correlation between model performance maps (Figure 6). Correlations were computed across both cortical hemispheres within each participant and run. The spatial correlation between the model performance maps for the within-subject and between-subject models was 0.542 using anatomical normalization and 0.719 when using hyperalignment. That is, the spatial correlation between the map of within-subject model fits and the map of between-subject model fits increased by 0.177 after hyperalignment [a 33% increase; $t(17) = 22.432$, $p < 0.001$].

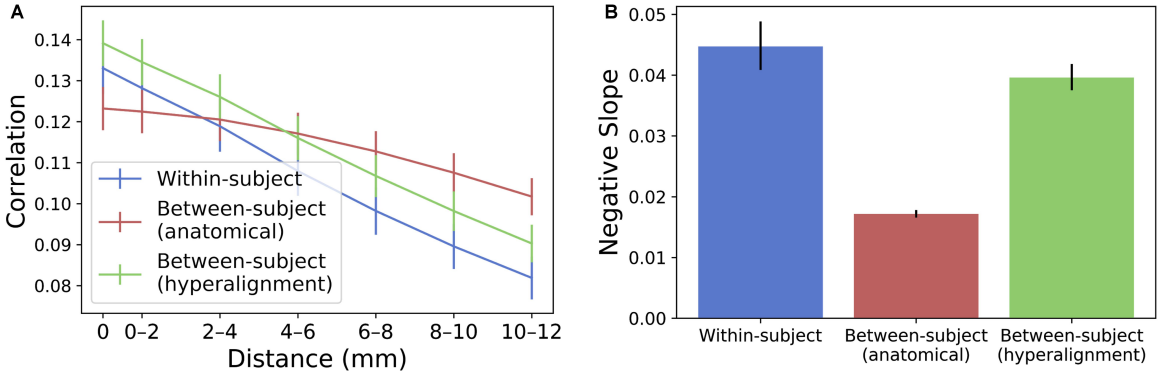

Figure 4: Spatial PSF of semantic tuning. (A) Correlation between the predicted time series of one vertex and the actual time-series of its neighboring vertices up to 12 mm away. Correlations were aggregated based on distance from the central vertex of interest and averaged across vertices and subjects. Error bars denote 68% confidence intervals (standard error of the mean). (B) The within-subject and hyperaligned between-subject models have the steepest slopes (negative of the slope based on logarithmic curve fitting). Error bars denote 95% confidence intervals obtained by bootstrapping subjects 20,000 times with replacement.

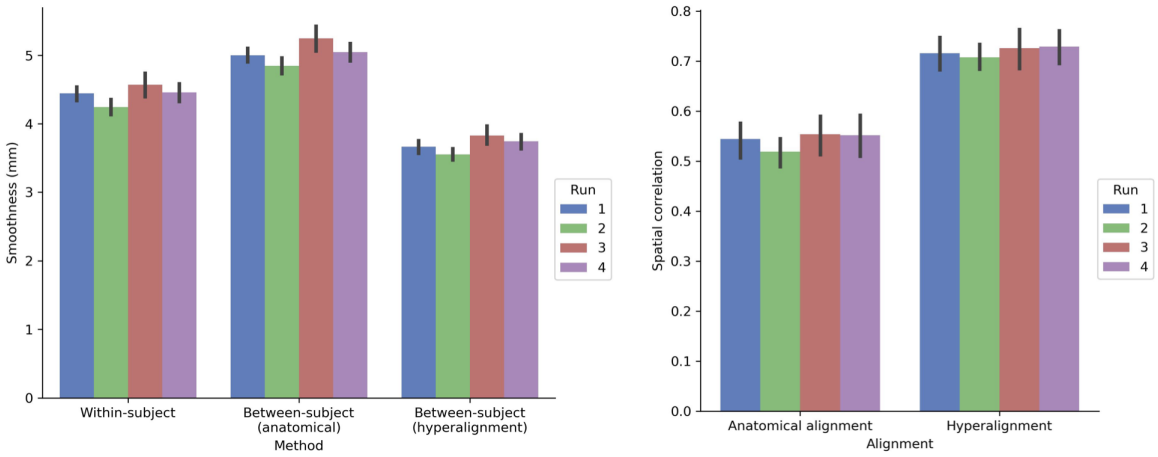

Figure 5: (Left) Spatial smoothness (FWHM) of model prediction performance maps on the cortical surface. The between-subject model performance maps using anatomical alignment are blurred relative to the within-subject model performance maps, while the hyperaligned between-subject model recovers the spatial specificity of the within-subject model. The height of each bar indicates spatial smoothness averaged across hemispheres and participants for each run. Error bars indicate bootstrapped 95% confidence intervals estimated by resampling participants (1,000 bootstrap samples). (Right) Spatial correlation between within-subject model prediction performance maps and between-subject model prediction performance maps using either anatomical alignment or hyperalignment. The between-subject model using hyperalignment yielded a model performance map that is more similar to the within-subject model than the model performance map of the between-subject model using anatomical alignment. The height of each bar indicates mean spatial correlation across participants for each run. Error bars indicate bootstrapped 95% confidence intervals estimated by resampling participants (1,000 bootstrap samples) [66].

5 Materials and methods

In our approach, we first show people a naturalistic movie, collect their fMRI neural response, and functionally align their neural responses using hyperalignment. We then show the six CNN models the same movie and extract intermediate layer feature activations. We take two approaches for comparing the CNN layer activations and the fMRI activations (both whole-brain and in ROIs). We first compare the brain ROI—and CNN layer-wise similarities in representation using representational distance matrices (RDMs) generated using representational similarity analysis (RSA). We compare the representations captured through RSA using two similarity metrics: correlation and centered kernel alignment (CKA). With this approach, we generate an RDM for each brain-CNN model pair, and we find which CNN model has the highest similarities with brain ROIs and most sensible ROI-layer assignments (lower layers should maps to lower visual areas, and higher layers should map to higher visual areas). Next, we use the activation outputs from each layer of each CNN to directly predict fMRI activations using forward encoding (with ridge regression). Here, the model with the best ability to predict neural responses has representations closest to those in the brain.

5.1 fMRI data

We collected and subsequently hyperaligned fMRI data from 20 participants watching *Raiders of the Lost Ark*. We call this dataset "Raiders" for short.

5.1.1 Participants

Twenty healthy young adults (age: mean ± standard deviation: 24.4 ± 3.4 years, 12 females) participated in this study. All participants were right-handed, with normal hearing and normal or corrected-to-normal vision, and no known history of neurological illness. They gave written, informed consent, and were paid for their participation. The study was approved by the Institutional Review Board of Dartmouth College.

5.1.2 Stimuli and design

Participants watched a full-length audiovisual movie, *Raiders of the Lost Ark*, while fMRI data were collected. The movie was divided into eight parts, each 14–15 min in duration. We used the second half of the movie (four parts with duration 339 TRS, 344 TRs, 344 TRs, and 340 TRs). Participants viewed four parts of the movie in each of the two scanning sessions, and were taken out of the scanner between the two sessions for a break.

The video was projected from an LCD projector onto a rear projection screen, and then reflected through a mirror on the head coil. The corresponding visual angles subtended approximately 22.7 ° horizontally and 17 ° vertically. The audio was played through MR-compatible headphones (MR confon GmbH, Magdeburg, Germany). Participants were instructed to pay attention to the movie and enjoy.

5.1.3 MRI acquisition

MR images were acquired using a 3T Philips Intera Achieva scanner with a 32-channel head coil at the Dartmouth Brain Imaging Center.

Functional images comprised 80 × 80 × 42 3 mm isotropic voxels, providing whole brain coverage. They were acquired every 2.5 s with an echo planar imaging (EPI) sequence (TR = 2.5 s, TE = 35 ms, flip angle = 90 °, 80 × 80 matrix, FOV = 240 mm × 240 mm, SENSE reduction factor = 2, 42 interleaved axial slices). The length of a run was adjusted to match the length of the corresponding movie part, consisting of 326–344 vol each. In total, we acquired 2718 functional images per participant during 8 runs of movie watching (approximately 2 h of functional data per participant).

A high resolution T1-weighted image (0.9375 mm × 0.9375 mm × 1.0 mm voxel resolution) was also acquired in each session with an MPRAGE sequence (TR = 8.2 ms, TE = 3.7 ms, flip angle = 8 °, 256 ×

256 matrix, FOV = 240 mm × 240 mm, 220 axial slices), except for one session of one participant.

5.1.4 MRI preprocessing

MRI data were first preprocessed using the fMRIPrep software version 1.0.0-rc2 (Esteban et al., 2018; https://github.com/poldracklab/fmriprep). T1-weighted images were corrected for bias field (Tustison et al., 2010) and skullstripped using antsBrainExtraction.sh. High resolution cortical surfaces were reconstructed with FreeSurfer ([10], learn more at http://surfer.nmr.mgh.harvard.edu/) using all available anatomical images, and registered to the fsaverage template [11]. Functional data were motion corrected using MCFLIRT [31], and resampled to the fsaverage template based on boundary-based registration [14]. After these steps, functional data from all participants were in alignment with the fsaverage template based on cortical folding patterns.

Further preprocessing steps were performed using Python scripts based on PyMVPA ([20], learn more at http://www.pymvpa.org/). First, functional data were further downsampled to a standard cortical surface mesh with 18,742 vertices across both hemispheres (3 mm vertex spacing; 20,484 vertices before removing non-cortical vertices), and data acquired during overlapping movie segments were discarded (8 TRs, 20 s, for each of runs 2–8). Then, nuisance regressors—6 motion parameters and their derivatives, global signal, framewise displacement [55], 6 principal components from cerebrospinal fluid and white matter [2], and up to second order polynomial trends—were partialed out from functional data separately for each run. Finally, the residual time series of each surface vertex in each run was normalized to zero mean and unit variance.

5.1.5 Hyperalignment

First, we created a common representational space to hyperalign functional data based on an independent dataset. The dataset comprised responses to the same movie from 20 different participants [19] [22]. We preprocessed the 20 participants' data through the same pipeline, and hyperaligned responses to the entire movie for all subjects using searchlight hyperalignment [19] with a 20-mm searchlight radius. For a given searchlight, the hyperalignment algorithm uses the Procrustes transformation to rotate each subject's feature space so as to best align response trajectories across individuals. The local transformations for each searchlight are then aggregated, forming a single, sparse transformation matrix for each cortical hemisphere. The hyperaligned data of the 20 participants were then averaged and normalized to unit variance to serve as the final common representational space.

Then, we derived hyperalignment transformations for each of the 20 participants to the common representational space based on their responses to the first half of the movie (runs 1–4), and applied these transformations to data from the second half of the movie (runs 5–8). The following analyses of individual

differences were based on the second half of the movie for these 20 participants, and are thus completely independent of the data used for deriving the common space and hyperalignment parameter estimation. Note that all data were anatomically aligned according to sulcal curvature on the cortical surface prior to hyperalignment.

5.2 CNN model data

5.2.1 ImageNet dataset and ILSVRC

The models used here were trained on the ImageNet dataset as part of the ImageNet Large Scale Visual Recognition Challenge, or ILSVRC for short [57]. The goal of this image classification challenge is to train a model that can correctly classify an input image into 1,000 separate object categories. Models are trained on the ImageNet dataset—approximately 1.2 million training images with another 50,000 images for validation and 100,000 images for testing. These 1,000 image categories represent object classes that we encounter in our day-to-day lives, such as species of dogs, cats, various household objects, vehicle types, and much more. When it comes to image classification, the ImageNet challenge is the de-facto benchmark for computer vision classification algorithms — and the leaderboard for this challenge has been dominated by Convolutional Neural Networks and deep learning techniques since 2012.

5.2.2 Pretrained models

We used VGG-19, ResNet-152, DenseNet-161 models, pretrained on the ImageNet dataset, from the PyTorch TorchVision library [54]. The state-of-the-art pre-trained networks included in this library represent some of the highest performing CNNs on the ImageNet challenge over the past few years, and each model we used here is the highest-performing model in its type (ie, we used VGG-19 instead of VGG-11). These networks also demonstrate a strong ability to generalize to images outside the ImageNet dataset via transfer learning, such as feature extraction and fine-tuning. From VGG, we extracted activations from the last 3x3 convolutional layer of each of the four blocks. From ResNet, we extracted activations from the 3x3 convolutional layer of the last bottleneck layer of each of the four residual blocks. From DenseNet, we extracted activations from the last 3x3 convolutional layer of each of the four dense blocks.

We also used CORnet-Z, CORnet-R, and CORnet-S models [38], again pretrained on the ImageNet dataset. For these models, we extracted activations from the final output layer of each of the four blocks.

5.2.3 Image activation extraction and preprocessing

We sampled an image from every half-second of the *Raiders of the Lost Ark*—since the movie has a frame rate of 30 frames per second, we sampled once every 15 frames. For each of the four parts of the movie we extracted images from, we ended up with 1694, 1720, 1720, and 1700 images, respectively. We then extracted CNN activations from the specified layers of each of the pretrained CNN models in response to these images, resulting in four sets of activations from each model.

To preprocess these activations for comparison to fMRI data, we averaged these extracted image activations in groups of five in order to obtain the average activation for every 2.5 s (to match the sampling rate of fMRI). We then convolved these averaged image activations with the canonical hemodynamic response function (HRF) [42] to construct a continuous function that is close to the hemodynamic response we observe for a single neural response in the brain. Before feeding these activations into the forward encoding models, we applied kernel PCA [58] to the activations to reduce their dimensionality to the same dimensionality as the number of timepoint neural responses, so as to speed up computation (ridge regression also reduces the feature space automatically, so this was not necessarily needed for preventing overfitting, though that is an additional use of kernel PCA). We did not apply kernel PCA for the RSA analyses.

5.2.4 Representational similarity analysis

We computed representational distance matrices between nine brain ROIS in the visual stream (all in the ventral object/scene recognition stream except for MT for comparison—MT is implicated in action recognition). We also computed RDMs within-model for the four extracted blocks of each CNN model. After computing these within-model RDMs, we computed RDMs comparing the brain ROIs and the four blocks of each CNN model (Figure 9). We used both correlation and CKA as similarity matrices; they gave almost identical results, but since CKA is the current state-of-the-art we include and discuss the RSA results using the CKA similarity metric here.

I then computed DNN layer assignments for each ROI. We first computed the CKA similarity between each layer of each DNN model and each brain ROI. We then saw which layer had the highest CKA similarity with the voxel responses in each ROI; this layer was then assigned to the ROI as its "most representationally similar layer." We compare the results of the RSA and forward encoding approaches to brain ROI layer assignment below.

I also am interested in how the differences between CNN model performance on object classification tasks (such as ImageNet) and on similarity with neural responses as measured through RSA. It has been hypothesized [70] that model performance on object classification tasks and on representational similiar-

ity with visual processing areas in the brain are positively correlated; the models that perform better on both tasks simply are more complex and capable, learning better features that are not necessarily more representationally similar to the visual features learned by the brain.

5.2.5 Forward encoding

In this thesis, we estimated voxel-wise forward encoding models—we created a forward encoding model for each cortical surface voxel, and used CNN layer activation data to predict that voxel's fMRI response over time.

The stimuli (input) for the forward encoding models was the preprocessed layer activations for each model. We estimated forward encoding models using activations from all of the layers (to measure the overall predictive ability of each DNN model's activations), and from each of the layers separately (for performing brain ROI to DNN layer assignments later). The responses (output to predict) were voxel-wise fMRI responses. We first anatomically aligned and then hyperaligned all of the participants' time series fMRI responses, and then averaged these time series responses across all participants. The forward encoding models were trained to predict these hyperaligned and averaged fMRI responses over three of the four parts of the movie. The models were then tested on the left-out fourth movie part. We repeated this process over all four movie parts to perform leave-one-run-out cross-validation for the forward encoding models.

We used L_2-norm penalized linear least squares regression (ridge regression) to estimate regression coefficients (weights) for the DNN feature predictor variables so as to best predict the response time series at each vertex. We used a modified implementation of ridge regression for sped-up computation [30]. Ridge regression uses a regularization hyperparameter to control the magnitude of the regression coefficients, where a larger regularization parameter yields greater shrinkage and reduces the effect of collinearity among predictor variables. The regularization parameter was chosen using leave-one-run-out cross-validation nested within each set of training runs. We estimated regression coefficients for a grid of 20 regularization parameters log-spaced from 1 to 1,000 at each vertex within each set of two runs in the training set of three runs. We then predicted the responses for the held-out third run (within the training set) and evaluated model prediction performance by computing the correlation between the predicted and actual responses. These correlations were averaged over the three cross-validation folds nested within the training set, then averaged across all vertices. We then selected the regularization parameter with the maximal model performance across runs and vertices. Selecting a single regularization parameter across all vertices ensures that estimated regression coefficients are comparable across vertices. This regularization parameter was then used at the final stage when estimating the encoding model across all three training runs for evaluation on the left-out fourth run. Note, however, that different regularization parameters were chosen for each of the four leave-one-run-out

cross-validation folds (where stimuli differed for each set of training runs).

To evaluate the vertex-wise forward encoding models, we used the regression coefficients from the model trained on three training runs to predict the response time series for the left-out fourth run. Both the hyperalignment transformations and encoding models were cross-validated to previously unseen data; the averaged subjects' data on the left-out test run played no role in estimating the hyperalignment transformations or the regression weights of the encoding model. For each vertex, we then computed the Pearson correlation between the predicted time series and actual time series for that run to measure model prediction performance [30].

We then computed DNN layer assignments for each ROI. We first estimated forward encoding models for each layer of each DNN model. We then saw which layer best predicted the voxel responses in each ROI; the layer that made the most accurate prediction (averaged across the voxels in each ROI) was then assigned to the ROI as its "most representationally similar layer."

We are also interested in how the differences between CNN model performance on object classification tasks (such as ImageNet) and on predicting neural responses through forward encoding. Just as with the comparable RSA analysis, we aim to determine if the models that perform better on both tasks simply are more complex and capable, and learn better features that are not necessarily more representationally similar to the visual features learned by the brain.

6 Results

To demonstrate the validity of performing both predictive and RSA analyses of DNN-brain representational similarity, we compared each DNN model's forward encoding performance (the predictive approach to understanding DNN-brain representational similarities) and CKA similarity (the RSA approach to understanding DNN-brain representational similarities) with brain ROIs, and demonstrated that these approaches both capture underlying information about representational similarities (Figure 6). Models that explicitly showed higher representational similarities with brain ROIs also had improved forward encoding performance for these same brain ROIs.

6.1 Representational similarity analysis

We first computed RSA RDMs between the brain regions of interest, using both correlation and CKA as similarity metrics (Figure 7). Both similarity metrics find relevant similarities between early visual areas (V1-V3), but CKA finds stronger representational similarities in theese areas, as well as meaningful representational similarities between higher visual areas implicated in object recognition and processing

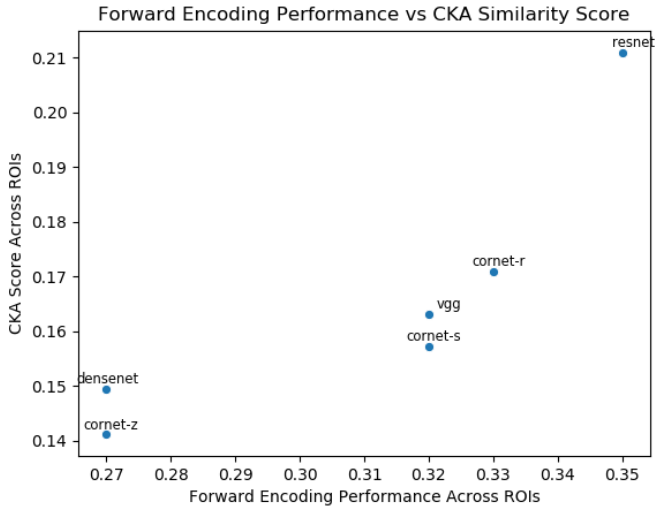

Figure 6: Comparison of each model's forward encoding performance (the predictive approach) and CKA similarity (the RSA approach) with brain ROIs. The forward encoding score measured here is the average forward encoding model performance across all of the ROIs, while the CKA similarity score measured here is the maximum CKA between brain ROIs and model layers, again averaged over brain ROIs. The approaches are correlated (as CKA similarity increases, so does forward encoding performance) and capture similar high-level differences in representational similarity.

(V3b, LO, VT, MT). We will discuss this similarity metric difference further in the discussion. Combined with Kornblith et al's findings that CKA uncovers more meaningful representational similarities for CNNs [35], we hereafter use only CKA as a similarity metric in these analyses.

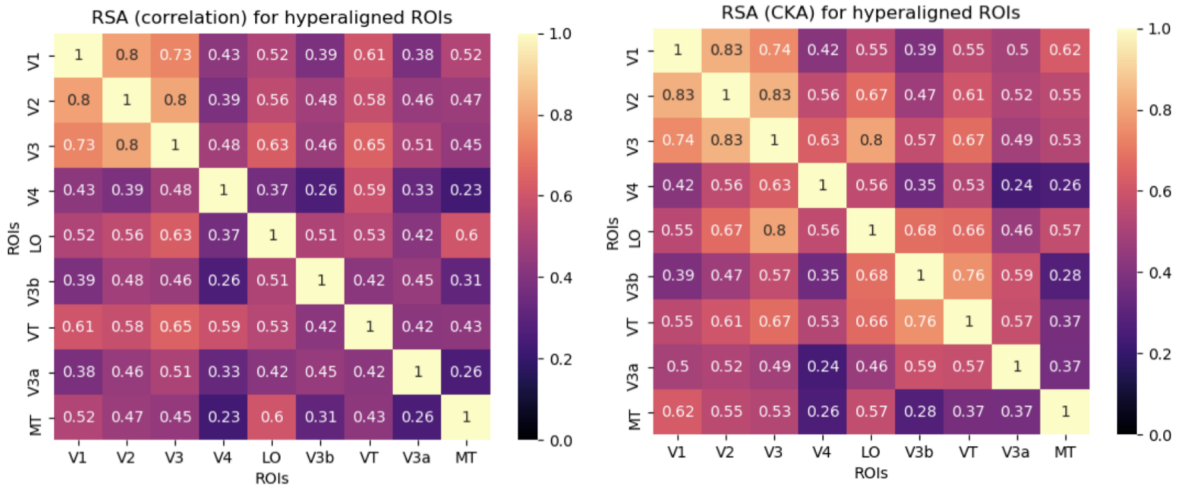

Figure 7: Comparison of representational dissimilarity matrices (RDMs) created with representational similarity analysis (RSA) using correlation and centered kernel alignment (CKA) similarity metrics.

We next computed RDMs of CKA similarity between layers within each DNN model (Figure 8) to discover how layer representations change with layer depth. Note that VGG maintains a more consistent representa-

tion throughout its layers, while ResNet and DenseNet learn highly different representations throughout their layers. This dissimilarity between ResNet and DenseNet layers is consistent with the idea that ResNets (and to a greater degree DenseNets) are more efficient than classic feedforward nets (like VGG) because their dense inter-layer connections encourage feature reuse, preventing layers from learning redundant feature maps.

Figure 8: RDMs of CKA similarity between layers within each DNN model.

Similarly, when we computed RDMs of CKA similarity between layers within each CORnet DNN model (Figure 9), we found that CORnet-S (recurrent model with residual connections) learns the representations that differ the most throughout its layers, followed by CORnet-R and CORnet-Z.

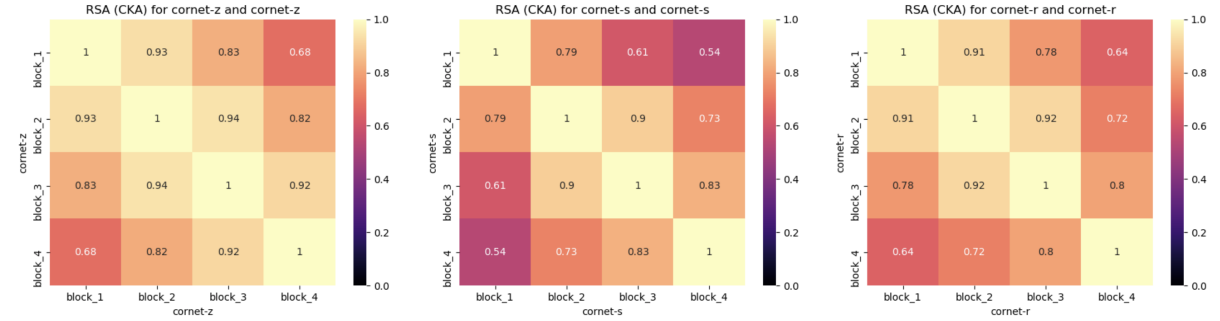

Figure 9: RDMs of CKA similarity between layers within each CORnet DNN model.

Increased layer dissimilarity is positively correlated with improved ImageNet performance (Figure 10). Additionally, DNN layer dissimilarity is positively correlated with forward encoding performance and CKA similarity score with brain ROIs, except for DenseNet (Figure 10).

I next computed the CKA RDMs comparing hyperaligned neural responses to activations extracted from CORnet-Z, CORnet-S, CORnet-R, VGG, ResNet, and DenseNet (Figure 11). All of the models have their highest CKA similarities with early visual areas (V1-V3) and VT. Interestingly, and as we will cover in the discussion, lower layers across all of the models have lower CKA score than expected. In particular,

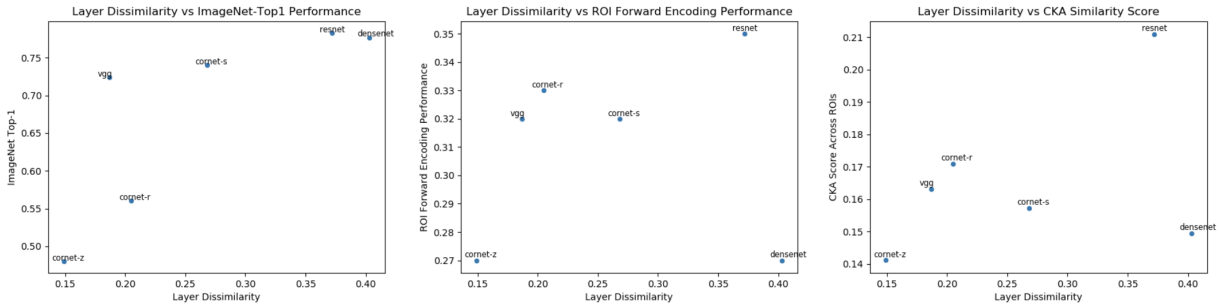

Figure 10: Layer dissimilarity, measured as 1 - the mean CKA score between all pairs of layers, vs (left) ImageNet top-1 performance, (center) forward encoding performance, and (right) CKA similarity score with brain ROIs. Note that increased layer dissimilarity is correlated with improved ImageNet performance, and is usually correlated with improved CKA score and forward encoding performance (except for DenseNet).

we expected the lower layers of the CNN models to have very high CKA scores with the lower visual area brain ROIs. Among the state-of-the-art models, ResNet learns the most brain-like representations, which are especially notable for V3B and VT. For the CORnet models, CORnet-R has a higher CKA similarity than CORnet-S, which has a higher similarity than CORnet-Z, to the brain data.

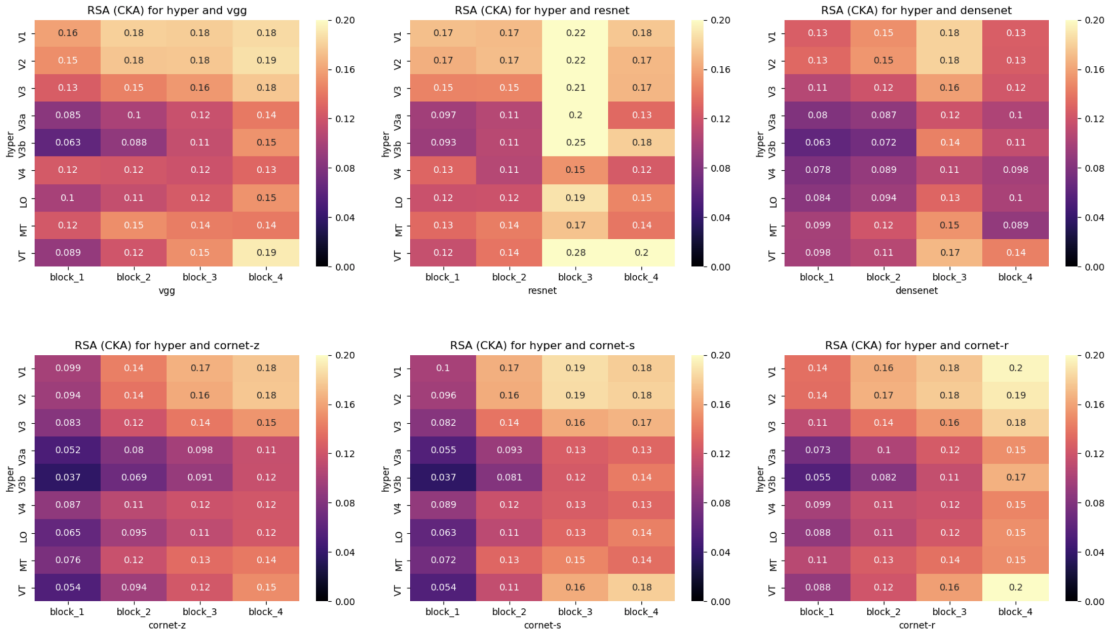

Figure 11: CKA RDMs comparing hyperaligned neural responses to activations extracted from CORnet-Z, CORnet-S, CORnet-R, VGG, ResNet, and DenseNet. For each model, we compare each layer's representation to each brain ROIs representation using the CKA similarity metric. Note that "hyper" refers to the hyperaligned data from the brain ROIs. See Figure 14 for the comparable analysis with forward encoding.

We also computed the "layer assignment" for each brain ROI based on CKA similarity score (Figure 12). Most of the CNN models had one layer that had the highest similarity score with all of the brain ROIs;

only VGG and CORnet-S had more than one layer assigned across all of the brain ROIs. We will discuss this layer assignment issue, and especially why lower layers across all of the CNN models had lower CKA similarity scores than their higher layer counterparts, in the discussion.

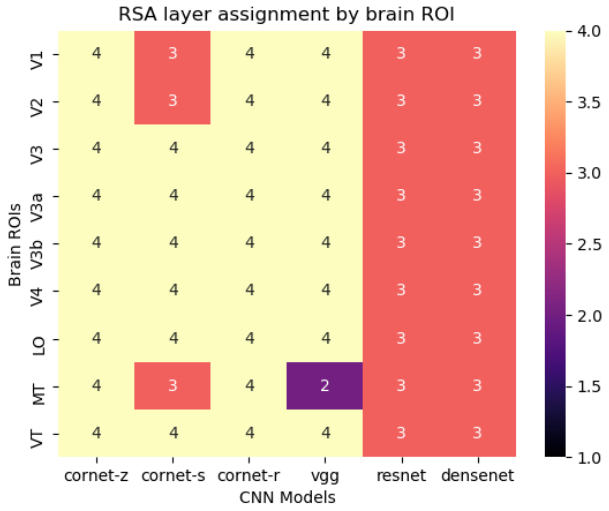

Figure 12: DNN layer assignments for each ROI as computed with RSA (CKA similarity metric). We first computed the CKA similarity between each layer of each DNN model and each brain ROI. We then saw which layer had the highest CKA similarity with the voxel responses in each ROI; this layer was then assigned to the ROI as its "most representationally similar layer."

6.2 Forward encoding

For whole-brain forward encoding model performance (Figure 13), ResNet outperformed the other models (mean correlation between predicted and true voxel activation of 0.19, standard deviation 0.14), followed by CORnet-R (mean 0.17, standard deviation 0.13), CORnet-S and VGG (both with mean 0.16, standard deviation 0.13), CORnet-Z (mean 0.14, standard deviation 0.12), and DenseNet (mean 0.13, standard deviation 0.13). As expected, all models have their highest prediction performance in early visual and ventral stream visual areas. ResNet outperforms all other models in terms of voxel activation prediction performance, followed by CORnet-R and CORnet-S. All models, except for ResNet, have their maximum prediction performance in VT (ranging from 0.57 for VGG to 0.63 for CORnet-R), while ResNet has its maximum prediction performance in V1 (0.69, but note that ResNet's maximum prediction performance in VT is 0.67).

For forward encoding model performance focusing only on the visual stream ROIs (Figure 15), ResNet again outperformed the other models (mean correlation between predicted and true voxel activation of 0.35 across voxels in ROIs, standard deviation 0.14), followed again by CORnet-R (mean 0.33, standard deviation 0.14), CORnet-S (mean 0.32, standard deviation 0.14), VGG (mean 0.31, standard deviation 0.13), CORnet-

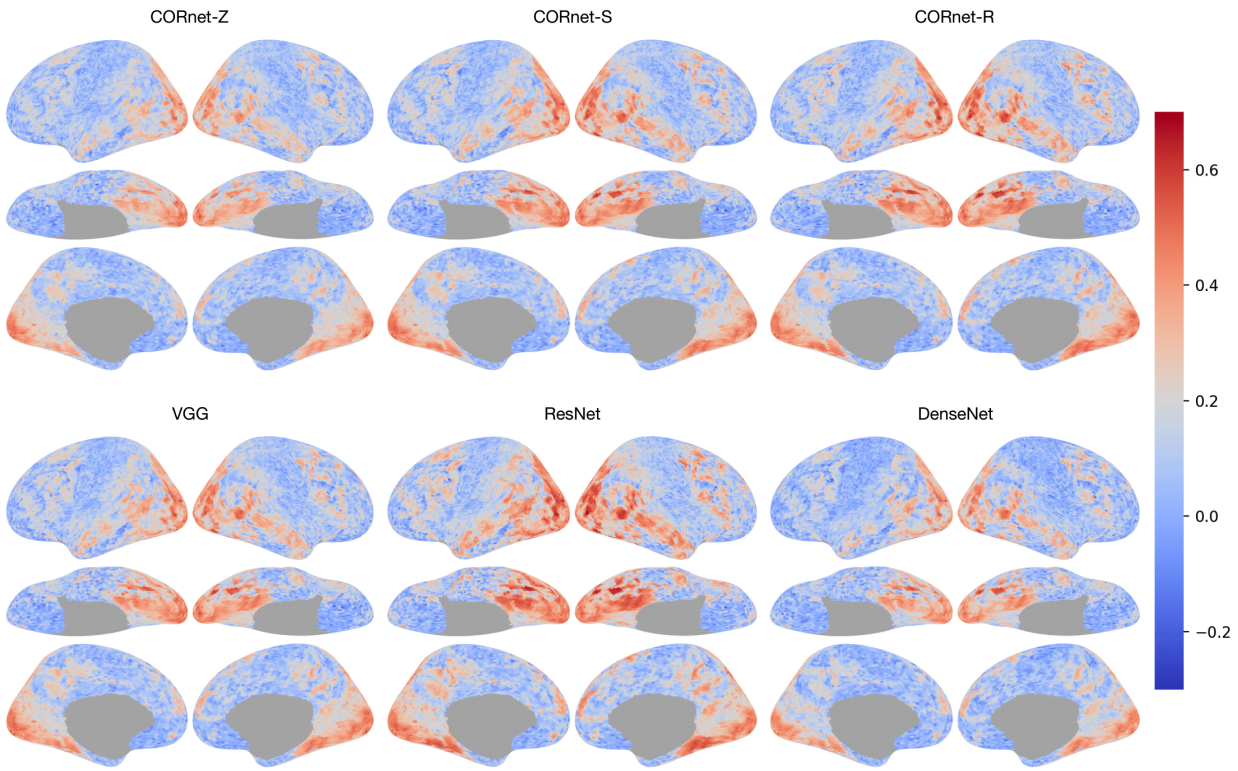

Figure 13: Whole-brain vertex-wise forward encoding model prediction performance map. Color indicates the correlation between predicted and true voxel activation value, averaged across timepoint samples.

Z and DenseNet (both with mean 0.27, standard deviation 0.13). V3b's voxel responses were the most well-predicted across all of the models, followed by LO and MT. VT was the least well-predicted across all models. All models, except for ResNet, have their maximum prediction performance in VT (ranging from 0.57 for VGG to 0.63 for CORnet-R), while ResNet has its maximum prediction performance in V1 (0.69, but note that ResNet's maximum prediction performance in VT is 0.67).

We also computed the "layer assignment" for each brain ROI (Figure 16). Most of the CNN models had one layer that best predicted all of the ROI voxel responses; only ResNet had more than one layer assigned across all of the brain ROIs. We will discuss this layer assignment issue, and especially why lower layers across all of the CNN models had lower forward encoding performance than their higher layer counterparts, in the discussion.

We also plotted each model's ImageNet top-1 performance (the object classification task the models were trained for) against their performance on the forward encoding task—both across the whole brain and also only in regions of interest (Figure 17). For both comparisons, ImageNet performance and forward encoding performance were positively correlated. However, CORnet-R and DenseNet's performances stand out as outliers, which we will elaborate on in the discussion below.

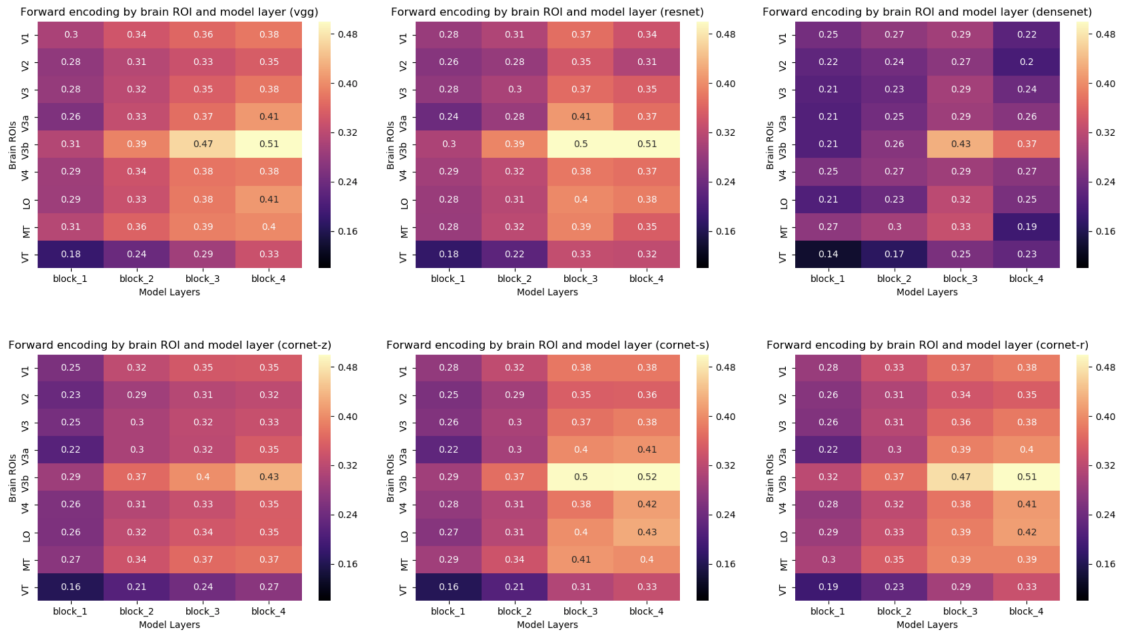

Figure 14: Comparing forward encoding model performance across brain ROIs and model layers for CORnet-Z, CORnet-S, CORnet-R, VGG, ResNet, and DenseNet. For each model, we compare each layer's representation to each brain ROIs representation using the performance of a forward encoding model trained only on that layer's activations, averaged across all of the voxels within each brain ROI. See Figure 11 for the comparable analysis with representational similarity analysis.

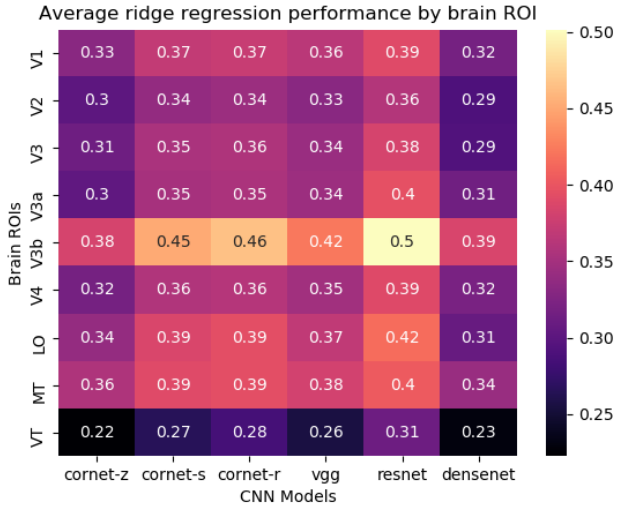

Figure 15: The forward encoding (ridge regression) performance by brain ROI (performance score is the averaged accuracy across all of the voxels in each ROI). V3B, LO, and MT are notably well-predicted, followed unexpectedly by the early visual areas. VT, though less well predicted than the other brain ROIs shown here, is actually predicted with state-of-the-art accuracy for a visual area so high in the visual stream hierarchy. Note the improved performance of ResNet, CORnet-R, and CORnet-S across these brain ROIs.

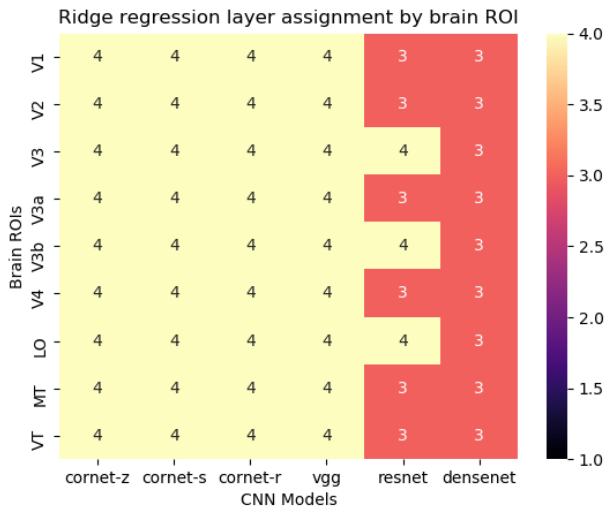

Figure 16: Forward encoding layer assignment by brain ROI and CNN model. We trained separate forward encoding models for each layer of each DNN model, and then saw which layer individually best predicted the voxel responses in each ROI; this layer was then assigned to the ROI as its "most representationally similar layer."

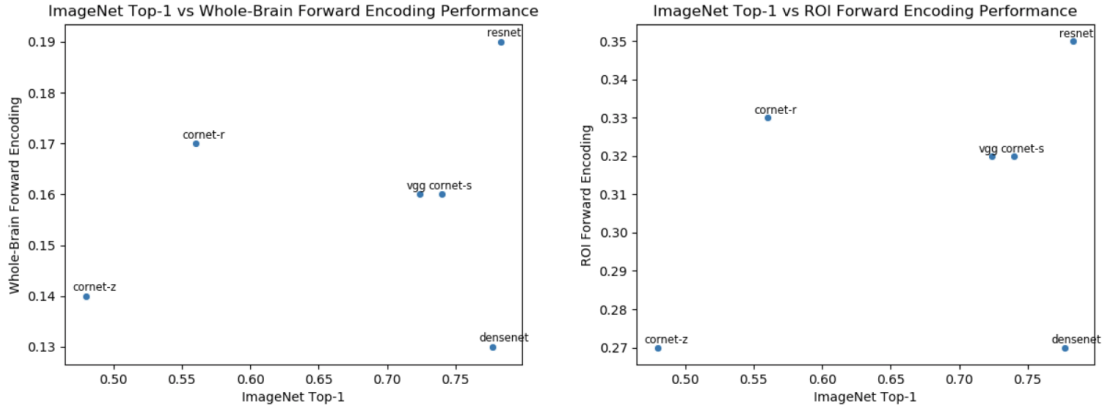

Figure 17: ImageNet top-1 vs forward encoding (whole-brain on left and ROI-only on right) performance plotted for each CNN model.

7 Discussion

Our results support the hypothesis that residual and recurrent CNNs learn more brain-like representations than vanilla CNNs. Unlike previous work, we were able to leverage functionally-aligned naturalistic fMRI data from twenty people, and we compared the representations stored in this between-subject data to representations across various layers of different CNN architectures. We found that the predictive (forward encoding) and representational similarity (RSA) approaches both offer compelling and parallel measures of brain-DNN feature similarity. We also achieve high RSA similarity scores for various brain ROIs across

all of the CNN models, and achieve state-of-the-art forward encoding performance, especially within higher visual areas in the ventral stream. We also disentangled "powerful" or "informative" visual features from "brain-like" features. Models that performed well on an image classification task, and learned "powerful" visual features did not necessarily learn "brain-like" visual features that corresponded to or predicted brain representations. Additionally, model performance on image classification tasks is correlated with layer dissimilarity—we suggest that models that take advantage of feature reuse (and avoid relearning the same features from layer to layer) were able to learn a larger space of hierarchical visual features. We suggest that these hierarchical visual features mirror those seen in the brain, as layer dissimilarity is also correlated with forward encoding performance and RSA similarity with ROIs in the ventral stream.

7.1 Two approaches to measuring "brain-like" representation similarity

We evaluated two approaches to understanding similarities between DNN and brain visual feature representations. We took a predictive approach (forward encoding) and a representational similarity approach (RSA). In the literature, previous approaches take either one approach or the other, but here We evaulate the utility of both approaches in tandem. The approaches are correlated (as CKA similarity increases, so does forward encoding performance) and capture similar high-level differences in representational similarity between the CNN models and brain data (Figure 6)—DenseNet has the lowest overall similarity on both metrics, while ResNet, CORnet-R and CORnet-S have the highest overall similarity on both metrics. However, upon diving deeper into the data, we do see differences in the patterns of similarity structures found by forward encoding and representational similarity analysis (compare Figures 11 and 14). We do see increases in representational similarity on both metrics as layer depth increases. However, while early visual areas and VT are very similar to CNN model activations (across all layers) as measured by RSA, early visual areas are not notably more similar, and VT is actually far less similar (as compared to any other ROIs) with the forward encoding similarity metric. These results suggest that there is no one-to-one mapping between the patterns in representation found by forward encoding and RSA. In the future, researchers should use both approaches for a more well-rounded understanding of representational similarities.

7.2 CKA as a similarity metric for representational similarity analysis

CKA is a novel similarity metric that has only been used for investigating CNN representations. Here, we show that, as with with CNN representations, CKA uncovers meaningful representational similarity differences that classic similarity metrics, such as Pearson correlation, do not. We performed the RSA analyses with both Pearson correlation and CKA as our similarity metrics. Both similarity metrics find relevant sim-

ilarities between early visual areas (V1-V3), but CKA finds meaningful representational similarities between higher visual areas implicated in object recognition and processing (V3b, LO, VT). CKA also shows the similarities between V3 specifically and LO and VT (V3 projects to LO and VT directly). Importantly, CKA shows the dissimilarities in representation between the the higher object-recognition visual areas (LO and VT) and the higher action-recognition visual areas (MT), with these two sets of visual processing pathways both showing similarities only at the early visual processing level. CKA uncovers more meaningful representational similarities for brain ROIs than correlation.

7.3 Residual and recurrent CNNs learn more brain-like representations

7.3.1 Representational similarity analysis

In the representational distance matrices created with RSA, we expected to see a "diagonal" of CKA similarities—we expected lower CNN layers to be more similar to lower visual areas like V1 and V2, and higher CNN layers to be more similar to higher visual areas like LO and VT. We also expected to see very high CKA similarities between lower brain ROIs (V1-V3) and lower layers in all of the networks. Instead, all of the CNN models seem to improve brain ROI similarity with increased layer depth, and we will discuss why we believe the models (for both RSA and forward encoding) did not perform as well as expected in lower visual areas in the future work section. Most of these models have the highest similarities with V1-V3 (though we did expect the CKA similarities for these lower visual areas to be higher) and VT across the board—this may be because these are generalized visual processing areas, while the ROIs with lower similarities may be more specialized for certain visual tasks.

ResNet had the highest CKA similarity with the brain ROIs, followed by CORnet-R, then CORnet-S, CORnet-Z, and VGG (all similar), and then finally DenseNet. Below, we will discuss why DenseNet performs so poorly, and why CORnet-R and ResNet perform so well, on this CKA measure.

7.3.2 Forward encoding

Compared to previous work, we achieve state-of-the-art forward encoding performance, especially within higher-order visual brain ROIs such as V3a and b, V4, LO, and VT (lower visual area performance was worse than expected, which we will discuss below). As expected, all models have their highest forward encoding prediction performance in early visual and ventral stream visual areas. The forward encoding results discussed in this thesis support the hypothesis that recurrent and residual networks learn more brain-like representations. The three residual and recurrent models (ResNet, CORnet-S, and CORnet-R) outperform the vanilla CNN models (VGG and CORnet-Z). It is also notable that the shallow recurrent networks

outperform the shallow vanilla network (CORnet-Z) and match the performance of the deep vanilla network (VGG). Specifically, ResNet (the deep residual network) outperforms all other models in terms of voxel activation prediction performance, followed by CORnet-R (the shallow recurrent network) and CORnet-S (the shallow residual/recurrent network).

7.4 Why do early visual areas have such low similarities and prediction scores?

For both the RSA and forward encoding analyses, we expected better performance and higher similarity scores in early visual areas like V1-V3. In the RSA analyses, we believe that this lower-than-expected performance is partially why we do not see the expected "diagonal" of lower brain ROIs and lower layers to higher brain ROIs and higher layers. We also believe that this lower performance accounts for the odd layer assignments (all ROIs assigned to the higher DNN layers) encountered above. This performance mismatch boils down to one issue: the CNN models and the people scanned watching the movie received different inputs. While the people watching the movie were able to freely view the movie as they pleased (moving their eyes around the screen to take in input from different areas), we fed the CNN models a center-cropped version of the movie. Therefore, people and the CNN models saw very different low-level visual features—they were sometimes looking at different parts of the movie! This difference in stimulus input equalled out as the brains and models processed higher-level features. For example, two people (and models) looking at different parts of a face (say, the upper and lower halves of a face) can both say that even though they are seeing completely different low-level input, at a high level they are both looking at a face. We are currently working on incorporating eye-tracking data into how we crop the stimulus input for the CNN models. We expect that this will greatly improve the CNN models' similarity and predictive ability for lower visual areas.

7.5 Using the simplest models possible as comparators

Both the RSA and forward encoding approaches are simple comparators of representations learned. Even with the CKA similarity metric, RSA does not perform any nonlinear transformations of the data while comparing CNN and brain data. For forward encoding, note that we used a simple linear (ridge) regression model in our analyses when comparing similarities in representation. This linear regression comparison operator is purposefully not complex to retain validity of comparing representations between CNN layers and brain ROI data; a more complex model could potentially absorb any non-linear transformation necessary to map any layer features to output brain data space. Nonlinear transformations of the data would obscure the actual underlying representations. For example, if we used a DNN as a forward encoding comparator, we could potentially find high representational similarities between early layers of a DNN model and later

ROIs in the ventral stream. The DNN would perform additional representational transformations, on top of the actual representations stored in the CNN layer, and give a false appraisal of representational similarity between these areas.

7.6 More "brain-like" (or just "better") features?

We need to disentangle learning more complex and powerful visual features from learning features that are similar to those we see in the brain. To do this, we introduced and accounted for performance on an object classification task (ILSVRC). Hypothetically, models that perform better on both object classification and RSA/forward encoding could hypothetically just learn more "powerful" features in general, not more brain-like features. However, we found that it is possible to disentangle more "powerful" visual features from more "brain-like" visual features—these are not necessarily the same thing. Though object classification and forward encoding/RSA performance are often positively correlated, there are two key cases where models perform far better on one task or the other.

Even though CORnet-R does not perform well on the ILSVRC task, especially when compared to the deeper models (VGG, ResNet, and DenseNet), it performs very well on the forward encoding and RSA tasks. CORnet-R showed poor ImageNet performance (0.56 top-1 performance) but the second-best forward encoding neural response prediction performances (0.17 for whole-brain and 0.33 for ROI-only) and CKA similarity scores (0.17 max CKA similarity on average across ROIs). Similarly, CORnet-S and VGG have similar forward encoding performances, though CORnet-S is much shallower than VGG.

On the other hand, while DenseNet showed one of the best ImageNet performances (0.78 top-1 performance) it had the worst forward encoding neural response prediction performances (0.13 for whole-brain and 0.27 for ROI-only) and CKA similarity scores (0.15 max CKA similarity on average across ROIs). This casts some doubt on the idea that models that perform better on both object classification and brain similarity (forward encoding) tasks simply learn more "powerful" features in general, not more brain-like features. DenseNet is more complex than the other models evaluated and has one of the best performances on the ImageNet object recognition challenge, but the representations it learns are less similar to the representations learned by the brain.

Why would a densely residual CNN like DenseNet learn features that are so unlike those we see in the brain? Similar to residual and recurrent networks, dense networks improve over vanilla CNNs by taking advantage of previous layers' outputs for feature reuse. However, we hypothesize that dense networks extend this feature reuse too far and do not perform it in a biologically-valid manner. Unlike a dense residual network, each area in the brain's visual processing hierarchy does not take in input from all of the areas

preceding it. Recurrent (and their unrolled-through-time counterpart — residual) networks, have a more local feature reuse that is far more biologically valid.

8 Future work

In addition to incorporating eye-tracking data to better sample the movie for CNN model stimulus inputs as mentioned above, we have several other ideas in mind for future work.

We plan to run the RSA and forward encoding analyses discussed here on more CNN model architectures, such as SqueezeNet and ResNext, and on more layers from each model. We would also like to look more closely at the brain ROIs —for example, VT is a large swath of cortex, which can be split further into different functional regions. It would be very interesting, for example, to compare RSA and forward encoding performance within specialized ROIs like the fusiform face area (FFA, specialized for faces), parahippocampal place area (PHC/PPA, specialized for places/scenes), and subsections of ventral/lateral occipital cortex (VO/LO, specialized for objects).

In these analyses, we worked with a shallow residual model (CORnet-S), a deep residual model (ResNet), and a shallow recurrent model (CORnet-R)—in the future we plan to work with a deeper recurrent model and work to incorporate feedback connections. Additionally, recurrent models take advantage of temporal dependence and are often used to process sequences of inputs. We plan to extend our analyses to other models that take advantage of temporal dependence, such as video processing models. We hypothesize that deeper recurrent or video processing models would perform far better on our RSA and forward encoding tasks than any of the models analyzed in this thesis.

Additionally, we plan to combine the approaches for evaluating representational similarity discussed here with Schrimpf et al 2018's Brain-Score [59]. While we use predictive and RSA approaches to compare representations in fMRI data to CNN representations, Brain-Score uses EEG neural predictivity and behavioral scores to compare these representations. In the future, it will become increasingly valuable to use the most multimodal approach possible for understanding representational similarities.

9 Code and data availability

All of the code for this thesis can be found at http://github.com/caravanuden/thesis. The pretrained CNN models are available either through PyTorch (specifically the TorchVision module) or the CORnet project (http://github.com/dicarlolab/CORnet). The fMRI datasets generated and analyzed for this study can be found in the DataLad repository for Raiders (http://datasets.datalad.org/?dir=/labs/haxby/raiders).

10 Acknowledgements

I would like to thank my thesis advisors Yaroslav Halchenko, James Haxby, and Lorenzo Torresani for their invaluable guidance, feedback, and (more than) occasional debugging assistance on this thesis. I would also like to thank Temiloluwa Prioleau for participating in my thesis defense committee. I would like to thank Feilong Ma, Andy Connolly, Sam Nastase, and everyone else in the HaxbyLab for thoughtful discussions and code snippets. I especially would like to thank all of the open source developers and ressearchers without whose work this thesis would not have been possible (notably, the DiCarlo Lab at MIT for their CORnet models, the Huth Lab at UT Austin for their ridge regression framework, the HaxbyLab at Dartmouth for PyMVPA and hyperalignment, and the PyTorch community). Lastly, I could not have not done this without the unwavering support of my friends and family.

References

[1] Omri Barak. Recurrent neural networks as versatile tools of neuroscience research. *Current opinion in neurobiology*, 46:1–6, 2017.

[2] Yashar Behzadi, Khaled Restom, Joy Liau, and Thomas T Liu. A component based noise correction method (compcor) for bold and perfusion based fmri. *Neuroimage*, 37(1):90–101, 2007.

[3] Christian Büchel and Karl J Friston. Modulation of connectivity in visual pathways by attention: cortical interactions evaluated with structural equation modelling and fmri. *Cerebral cortex (New York, NY: 1991)*, 7(8):768–778, 1997.

[4] Corinna Cortes, Mehryar Mohri, and Afshin Rostamizadeh. Algorithms for learning kernels based on centered alignment. *Journal of Machine Learning Research*, 13(Mar):795–828, 2012.

[5] David Daniel Cox. Do we understand high-level vision? *Current opinion in neurobiology*, 25:187–193, 2014.

[6] Giovanni M Di Liberto, James A O'Sullivan, and Edmund C Lalor. Low-frequency cortical entrainment to speech reflects phoneme-level processing. *Current Biology*, 25(19):2457–2465, 2015.

[7] James J DiCarlo, Davide Zoccolan, and Nicole C Rust. How does the brain solve visual object recognition? *Neuron*, 73(3):415–434, 2012.

[8] Jeffrey L Elman. Distributed representations, simple recurrent networks, and grammatical structure. *Machine learning*, 7(2-3):195–225, 1991.

[9] Daniel J Felleman and DC Essen Van. Distributed hierarchical processing in the primate cerebral cortex. *Cerebral cortex (New York, NY: 1991)*, 1(1):1–47, 1991.

[10] Bruce Fischl. Freesurfer. *Neuroimage*, 62(2):774–781, 2012.

[11] Bruce Fischl, Martin I Sereno, Roger BH Tootell, and Anders M Dale. High-resolution intersubject averaging and a coordinate system for the cortical surface. *Human brain mapping*, 8(4):272–284, 1999.

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

[13] Arthur Gretton, Olivier Bousquet, Alex Smola, and Bernhard Schölkopf. Measuring statistical dependence with hilbert-schmidt norms. In *International conference on algorithmic learning theory*, pages 63–77. Springer, 2005.

[14] Douglas N Greve and Bruce Fischl. Accurate and robust brain image alignment using boundary-based registration. *Neuroimage*, 48(1):63–72, 2009.

[15] Umut Güçlü, Jordy Thielen, Michael Hanke, and Marcel Van Gerven. Brains on beats. In *Advances in Neural Information Processing Systems*, pages 2101–2109, 2016.

[16] Umut Güçlü and Marcel AJ van Gerven. Deep neural networks reveal a gradient in the complexity of neural representations across the ventral stream. *Journal of Neuroscience*, 35(27):10005–10014, 2015.

[17] Umut Güçlü and Marcel AJ van Gerven. Increasingly complex representations of natural movies across the dorsal stream are shared between subjects. *NeuroImage*, 145:329–336, 2017.

[18] Umut Güçlü and Marcel AJ van Gerven. Modeling the dynamics of human brain activity with recurrent neural networks. *Frontiers in computational neuroscience*, 11:7, 2017.

[19] J Swaroop Guntupalli, Michael Hanke, Yaroslav O Halchenko, Andrew C Connolly, Peter J Ramadge, and James V Haxby. A model of representational spaces in human cortex. *Cerebral cortex*, 26(6):2919–2934, 2016.

[20] Michael Hanke, Yaroslav O Halchenko, Per B Sederberg, Stephen José Hanson, James V Haxby, and Stefan Pollmann. Pymvpa: a python toolbox for multivariate pattern analysis of fmri data. *Neuroinformatics*, 7(1):37–53, 2009.

[21] James V Haxby, M Ida Gobbini, Maura L Furey, Alumit Ishai, Jennifer L Schouten, and Pietro Pietrini. Distributed and overlapping representations of faces and objects in ventral temporal cortex. *Science*, 293(5539):2425–2430, 2001.

[22] James V Haxby, J Swaroop Guntupalli, Andrew C Connolly, Yaroslav O Halchenko, Bryan R Conroy, M Ida Gobbini, Michael Hanke, and Peter J Ramadge. A common, high-dimensional model of the representational space in human ventral temporal cortex. *Neuron*, 72(2):404–416, 2011.

[23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[24] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

[25] David H Hubel and Torsten N Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of physiology*, 160(1):106–154, 1962.

[26] David H Hubel and Torsten N Wiesel. Receptive fields and functional architecture of monkey striate cortex. *The Journal of physiology*, 195(1):215–243, 1968.

[27] Patrick W Hullett, Liberty S Hamilton, Nima Mesgarani, Christoph E Schreiner, and Edward F Chang. Human superior temporal gyrus organization of spectrotemporal modulation tuning derived from speech stimuli. *Journal of Neuroscience*, 36(6):2014–2026, 2016.

[28] Chou P Hung, Gabriel Kreiman, Tomaso Poggio, and James J DiCarlo. Fast readout of object identity from macaque inferior temporal cortex. *Science*, 310(5749):863–866, 2005.

[29] Alexander G Huth, Wendy A de Heer, Thomas L Griffiths, Frédéric E Theunissen, and Jack L Gallant. Natural speech reveals the semantic maps that tile human cerebral cortex. *Nature*, 532(7600):453, 2016.

[30] Alexander G Huth, Shinji Nishimoto, An T Vu, and Jack L Gallant. A continuous semantic space describes the representation of thousands of object and action categories across the human brain. *Neuron*, 76(6):1210–1224, 2012.

[31] Mark Jenkinson, Peter Bannister, Michael Brady, and Stephen Smith. Improved optimization for the robust and accurate linear registration and motion correction of brain images. *Neuroimage*, 17(2):825–841, 2002.

[32] Judson P Jones and Larry A Palmer. An evaluation of the two-dimensional gabor filter model of simple receptive fields in cat striate cortex. *Journal of neurophysiology*, 58(6):1233–1258, 1987.

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

[34] Arno Klein, Satrajit S Ghosh, Brian Avants, BT Thomas Yeo, Bruce Fischl, Babak Ardekani, James C Gee, J John Mann, and Ramin V Parsey. Evaluation of volume-based and surface-based brain image registration methods. *Neuroimage*, 51(1):214–220, 2010.

[35] Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. *arXiv preprint arXiv:1905.00414*, 2019.

[36] Nikolaus Kriegeskorte, Marieke Mur, and Peter A Bandettini. Representational similarity analysis-connecting the branches of systems neuroscience. *Frontiers in systems neuroscience*, 2:4, 2008.

[37] John K Kruschke. Alcove: an exemplar-based connectionist model of category learning. *Psychological review*, 99(1):22, 1992.

[38] Jonas Kubilius, Martin Schrimpf, Aran Nayebi, Daniel Bear, Daniel LK Yamins, and James J DiCarlo. Cornet: Modeling the neural mechanisms of core object recognition. *BioRxiv*, page 408385, 2018.

[39] Joel Z Leibo, Qianli Liao, Fabio Anselmi, Winrich A Freiwald, and Tomaso Poggio. View-tolerant face recognition and hebbian learning imply mirror-symmetric neural tuning to head orientation. *Current Biology*, 27(1):62–67, 2017.

[40] Qianli Liao, Joel Z Leibo, and Tomaso Poggio. How important is weight symmetry in backpropagation? In *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[41] Qianli Liao and Tomaso Poggio. Bridging the gaps between residual learning, recurrent neural networks and visual cortex. *arXiv preprint arXiv:1604.03640*, 2016.

[42] Martin A Lindquist, Ji Meng Loh, Lauren Y Atlas, and Tor D Wager. Modeling the hemodynamic response function in fmri: efficiency, bias and mis-modeling. *Neuroimage*, 45(1):S187–S198, 2009.

[43] Nikos K Logothetis and David L Sheinberg. Visual object recognition. *Annual review of neuroscience*, 19(1):577–621, 1996.

[44] David Marr and W Thomas Thach. A theory of cerebellar cortex. In *From the Retina to the Neocortex*, pages 11–50. Springer, 1991.

[45] James L McClelland and Timothy T Rogers. The parallel distributed processing approach to semantic cognition. *Nature reviews neuroscience*, 4(4):310, 2003.

[46] Lane McIntosh, Niru Maheswaranathan, Aran Nayebi, Surya Ganguli, and Stephen Baccus. Deep learning models of the retinal response to natural scenes. In *Advances in neural information processing systems*, pages 1369–1377, 2016.

[47] Nima Mesgarani, Connie Cheung, Keith Johnson, and Edward F Chang. Phonetic feature encoding in human superior temporal gyrus. *Science*, 343(6174):1006–1010, 2014.

[48] T Miconi. Biologically plausible learning in recurrent neural networks for flexible decision tasks. biorxiv, 2016.

[49] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

[50] Thomas Naselaris, Kendrick N Kay, Shinji Nishimoto, and Jack L Gallant. Encoding and decoding in fmri. *Neuroimage*, 56(2):400–410, 2011.

[51] Shinji Nishimoto, An T Vu, Thomas Naselaris, Yuval Benjamini, Bin Yu, and Jack L Gallant. Reconstructing visual experiences from brain activity evoked by natural movies. *Current Biology*, 21(19):1641–1646, 2011.

[52] Klaus Obermayer, Helge Ritter, and Klaus Schulten. A principle for the formation of the spatial structure of cortical feature maps. *Proceedings of the National Academy Of Sciences*, 87(21):8345–8349, 1990.

[53] Bruno A Olshausen and David J Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381(6583):607, 1996.

[54] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

[55] Jonathan D Power, Kelly A Barnes, Abraham Z Snyder, Bradley L Schlaggar, and Steven E Petersen. Spurious but systematic correlations in functional connectivity mri networks arise from subject motion. *Neuroimage*, 59(3):2142–2154, 2012.

[56] Helge Ritter and Teuvo Kohonen. Self-organizing semantic maps. *Biological cybernetics*, 61(4):241–254, 1989.

[57] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

[58] Bernhard Schölkopf, Alexander Smola, and Klaus-Robert Müller. Kernel principal component analysis. In *International conference on artificial neural networks*, pages 583–588. Springer, 1997.

[59] Martin Schrimpf, Jonas Kubilius, Ha Hong, Najib J Majaj, Rishi Rajalingham, Elias B Issa, Kohitij Kar, Pouya Bashivan, Jonathan Prescott-Roy, Kailyn Schmidt, et al. Brain-score: which artificial neural network for object recognition is most brain-like? *BioRxiv*, page 407007, 2018.

[60] Wolfram Schultz, Peter Dayan, and P Read Montague. A neural substrate of prediction and reward. *Science*, 275(5306):1593–1599, 1997.

[61] Thomas Serre, Aude Oliva, and Tomaso Poggio. A feedforward architecture accounts for rapid categorization. *Proceedings of the national academy of sciences*, 104(15):6424–6429, 2007.

[62] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[63] H Francis Song, Guangyu R Yang, and Xiao-Jing Wang. Reward-based training of recurrent neural networks for cognitive and value-based tasks. *Elife*, 6:e21492, 2017.

[64] Olaf Sporns and Jonathan D Zwi. The small world of the cerebral cortex. *Neuroinformatics*, 2(2):145–162, 2004.

[65] Keiji Tanaka. Inferotemporal cortex and object vision. *Annual review of neuroscience*, 19(1):109–139, 1996.

[66] Cara E Van Uden, Samuel A Nastase, Andrew C Connolly, Ma Feilong, Isabella Hansen, Maria Ida Gobbini, and James V Haxby. Modeling semantic encoding in a common neural representational space. *Frontiers in neuroscience*, 12:437, 2018.

[67] Yuxin Wu and Kaiming He. Group normalization. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 3–19, 2018.

[68] Dean Wyatte, David J Jilk, and Randall C O'Reilly. Early recurrent feedback facilitates visual object recognition under challenging conditions. *Frontiers in psychology*, 5:674, 2014.

[69] Daniel L Yamins, Ha Hong, Charles Cadieu, and James J DiCarlo. Hierarchical modular optimization of convolutional networks achieves representations similar to macaque it and human ventral stream. In *Advances in neural information processing systems*, pages 3093–3101, 2013.

[70] Daniel LK Yamins, Ha Hong, Charles F Cadieu, Ethan A Solomon, Darren Seibert, and James J DiCarlo. Performance-optimized hierarchical models predict neural responses in higher visual cortex. *Proceedings of the National Academy of Sciences*, 111(23):8619–8624, 2014.

[71] David Zipser and Richard A Andersen. A back-propagation programmed network that simulates response properties of a subset of posterior parietal neurons. *Nature*, 331(6158):679, 1988.