Deep fMRI Encoding Models of Human Vision

by

Shawn Giacomo Carere

A thesis submitted in conformity with the requirements
for the degree of Master of Science

Department of Medical Biophysics
University of Toronto

Deep fMRI Encoding Models of Human Vision

Shawn Giacomo Carere
Master of Science

Department of Medical Biophysics
University of Toronto
2023

Abstract

In this work, we analyze artificial intelligence models that predict fMRI-measured brain activity in response to visual stimuli (ie. encoders). First, we introduce capsule encoders, the first alternative to convolutional neural networks (CNNs) for deep image-fMRI encoding. We show that although capsule networks are not yet effective for complex naturalistic images, with simpler visual stimuli, they improve upon existing CNN encoders. We also introduce new metrics and visualizations that provide a more complete picture of how encoders behave when able or unable to predict voxel activity. With this, we separate these behaviors using a novel thresholding approach and show it is more robust than p-value significance. We also find that encoders may be able to surpass the noise ceiling, revealing how the SNR of the validation set may be limiting our ability to evaluate encoders and highlighting a trade-off between the number of unique and repeated samples.

Acknowledgements

I'd like to thank my supervisor, Dr. Kamil Uludag, for his overall continued support as well as his leniency when I decided to shift my focus from research in order to take twice as many courses as I was supposed to. Thank you for being a wise and understanding mentor.

I'd like to thank the various members of my lab for their support and friendship over the past few years. Thank you for putting up with my various scientific rants and playful teasing while in the lab. A special thanks to Uzair and Sayan who were often the sturdy walls off which I could bounce my ideas and Sri for lending his expertise when needed.

Lastly, I'd like to thank my family and friends who provided me with much-needed distractions and breaks when I was at my busiest. Thank you for everything, I know I can always rely on you all.

Sincerely,

Shawn Carere

Contents

1 Introduction **1**

1.1 Information in the Brain 1

1.2 Computational Modelling as a Tool 2

1.3 Convolutional Neural Networks for fMRI 2

1.4 Data Limitations 3

1.5 Model Limitations 7

1.6 Contributions 8

2 Background **10**

2.1 Functional Magnetic Resonance Imaging 10

2.2 Brief Overview of The Visual Pathway 12

2.3 Defining Encoding Models 14

2.4 A History of Image to fMRI Encoding Models 16

2.4.1 Linearly Mapping Handcrafted Features 16

2.4.2 The Introduction of Deep Learning 17

2.4.3 A More Biologically Inspired Use of Deep Learning 19

2.4.4 Fully Embracing Deep Learning 19

2.4.5 Combining Approaches to Achieve State of the Art 20

2.5 Relation to Decoding Models 21

2.6 Capsule Networks 22

3 Materials and Methods **27**

3.1 Data . . . 27

3.2 Model Architecture . . . 28

3.3 Model Training . . . 30

3.4 Model Evaluation . . . 32

3.4.1 Voxelwise versus Samplewise Statistics . . . 33

3.4.2 Metrics . . . 33

3.4.3 Bootstrapping . . . 37

4 Results **39**

4.1 Encoding Naturalistic Images . . . 39

4.2 Encoding Handwritten Digits . . . 41

4.3 Characterizing Model Behavior . . . 42

5 Discussion **54**

5.1 Capsule Networks as Encoding Models . . . 54

5.2 Characterizing Encoder Behaviour . . . 57

5.3 Novel Methods Describing Encoder Behaviour . . . 59

5.4 Evaluating Encoders with Improved Methods . . . 62

5.5 Characterizing Effects of Validation Set SNR . . . 63

6 Conclusion **67**

6.1 Summary . . . 67

6.2 Future Work . . . 69

A Supplementary Materials **72**

A.1 Capsule Encoder Experiments . . . 72

A.2 Existing Methods for Encoder Evaluation . . . 75

A.3 Further Model Characterization . . . 76

B List of Abbreviations **82**

C Glossary **85**

List of Tables

3.1 Summary of Datasets 29

4.1 Metric-Based Results on Imagenet-fMRI 41

4.2 Metric-Based Results on MNIST-fMRI 42

4.3 Improved Metric Based Results on Imagenet-fMRI 51

4.4 Improved Metric Based Results on MNIST-fMRI 51

A.1 Capsule Network on MNIST Classification 72

A.2 Capsule Encoder Architectures 72

A.3 Capsule Encoder Experiments on Imagenet-fMRI 73

A.4 Capsule Encoder Experiments on MNIST-fMRI 74

A.5 Outline of Model Sizes Measured in Number of Parameters 76

List of Figures

1.1 Comparison of CNN autoencoders and fMRI-image decoders 4

2.1 Diagram of Dorsal and Ventral Streams 14
2.2 Overview of Image-fMRI Encoding and Decoding 15
2.3 Basic Approach to Computational Encoders 16

3.1 Model Architecture Diagram 31
3.2 Visualization of Voxelwise vs. Samplewise Vectors 34
3.3 Overview of Bagging Aggregation Evaluation 38

4.1 Predicted Activations in Anatomical Space 40
4.2 Model Robustness to Voxel Subset Changes 43
4.3 Voxelwise Correlation and Response Sensitivity in Anatomical Space 44
4.4 Voxel Response Sensitivity vs Voxelwise Correlation on Imagenet-fMRI 46
4.5 Voxel Response Sensitivity vs Voxelwise Correlation on MNIST-fMRI 47
4.6 Visualization of Response Weighted Correlation (RWC) and RWC Skew 49
4.7 Comparison of RWC vs P-Value Thresholding 50
4.8 Effect of Validation SNR on Model Evaluation 53

A.1 Voxelwise Correlation of Beliy Encoder on Imagenet-fMRI Projected onto Cortical Surface 75
A.2 Voxelwise Correlation Advantage Plots 76
A.3 RWC and Voxelwise Correlation for Different ROI's 77

A.4 Voxel Response Sensitivity vs Voxelwise Correlation with Training Set 77
A.5 Histogram of Voxelwise Correlations Categorized by p-value 78
A.6 Robustness of Voxel Response Sensitivity 78
A.7 Comparison of Voxelwise Correlation on Training and Validation Sets 79
A.8 Voxelwise Correlation with Predictions versus Between Repeated Measurements 80
A.9 Effect of Validation SNR for Different ROI's 81

Chapter 1

Introduction

1.1 Information in the Brain

It is somewhat known that in physics there exists a disparity between the frameworks of quantum mechanics and general relativity. There also exists a different, yet perhaps similarly vast, dichotomy in cognitive neuroscience, between the physical world and the seemingly intangible mind. While physicists continue to seek a falsifiable 'theory of everything' to bridge the two existing frameworks of their field, cognitive neuroscientists need only look at the biological brain. This presents a contrasting challenge however, one in which a 'theory of everything' must be reverse-engineered. In the brain, stimuli from the physical world are *encoded* into neural activations (electrical signals generated by the 'firing' of neurons) until they are eventually perceived. Conversely, the brain interacts with the world by *decoding* neural activations into motor function. These processes are effectively transfers of information. Information in the physical world takes on one physical form (eg. light, sound, etc.) and information in the brain takes on another (neural activations). Together, encoding and decoding form the basis of our ability to perceive and interact with the world, making them critical components towards understanding the fundamental mechanisms of the brain

[1] [2].

1.2 Computational Modelling as a Tool

The brain is a structurally complex, self-organizing, and information-dense biological entity. These properties make the brain difficult to observe and experiment on without interrupting its natural function or irreparably damaging it. As a solution, many researchers have turned to computational modelling. Computational encoding and decoding models can offer new insights into the specific operations underlying brain function. As our understanding of brain function improves, so too do our computational models. A common starting point has been to model the brain activity of the visual pathway as measured using functional magnetic resonance imaging (fMRI) in response to visual stimuli (e.g. images, videos, etc.). Accurately modelling the visual pathway is widely considered a milestone in cognitive neuroscience [3][4], yet has proven to be extremely challenging. Despite there being much yet left to learn within the field, these models have already begun to see various uses. Decoding models have been used as a window into perception with visual recall [5] and imagery [5][6][7]. Encoding models have also proven useful, enabling retinotopic mapping of visual features [8][9][10]. Additionally, advancements made with fMRI-based models are an important step towards brain-computer interfaces (BCIs) as well as treating and diagnosing neurological diseases [11][12].

1.3 Convolutional Neural Networks for fMRI

Recently, the use of deep learning methods, in particular convolutional neural networks (CNNs), has become standard practice for state-of-the-art modelling of the human visual pathway [8][9][10][13][14]. CNN-based architectures have previously been shown to be extremely effective in complex domains involving computer vision

such as image classification [15], segmentation [16], and generation [17][18]. However, they do not achieve comparable performance for similar fMRI encoding and decoding tasks. For example, both autoencoders and fMRI decoders can reconstruct images from latent space representations: for the former, this latent space representation is derived from a CNN-based image encoder, and for the latter, it is derived from a biological encoder in the form of fMRI measured brain activity. Existing CNN based models can decode significantly better image reconstructions from a CNN-derived latent vector than from an fMRI-measured latent vector. An example is shown in Figure 1.1. A similar comparison can be made for the inverse problem. Given an image as an input, existing CNN-based models can better predict label-derived latent vectors (such as in image classification) than those derived from fMRI. These differences in performance can possibly be explained by two major factors: limitations in the data used to train and test the models and limitations of the CNN models themselves.

1.4 Data Limitations

Obtaining fMRI data in response to a diverse set of visual stimuli is expensive and time-consuming [14]. This results in relatively small datasets (a few thousand samples) when compared to other computer vision tasks which typically rely on datasets with millions and sometimes hundreds of millions of samples. Additionally, while these large computer vision datasets generally contain high-quality images and labels, fMRI is an imperfect measure of brain activity. Technical limitations create a trade-off between spatiotemporal resolution and signal-to-noise ratio (SNR). Therefore, it is likely that a significant amount of potentially relevant information is lost during data acquisition. Furthermore, fMRI only indirectly measures brain activity using the blood oxygenation level-dependent (BOLD) signal. This results in other

(a) VGG16 Based Autoencoder

(b) VGG-19 Based fMRI-Image Decoders

Figure 1.1: Comparison of CNN's for autoencoding and fMRI-Image decoding. (a) Ground truth images and corresponding reconstructions (obtained from [20]) of a CNN based autoencoder from a compressed latent space representation. (b) Ground truth images and corresponding reconstructions of two state-of-the-art decoders from fMRI measured brain activity

sources of error such as physiological noise, in addition to the information lost due to aggregating the metabolic cost of direct neuronal activity into a single scaler number for some finite region in space and time. Although many of these limitations can be partially addressed via post-acquisition processing, the quantity and quality of image-fMRI datasets is still a major limitation. In trying to address the limited quantity of data, Gaziv et al. [19] proposed a self-supervised learning technique wherein samples artificially generated using a pretrained encoder are used to increase the amount of training data for a decoding model. This highlights the value of encoding models not only for modelling the visual pathway but also for addressing fMRI data limitations. Despite this, there remains a comparatively large focus on decoding and the quality of reconstructed images in existing work. In fact, to the best of our knowledge, the state-of-the-art encoder is the encoder used for self-supervision (see description above) in Gaziv et al., despite decoding being the focus of the paper. In contrast, and most importantly, my thesis focuses on the encoding of images into fMRI space. Furthermore, I attempt to address the underrepresentation of encoding in existing work by providing a framework for understanding and evaluating encoding models. Perhaps the greater emphasis on decoding in existing work is in part due to the difficulty in evaluating and interpreting an encoder's predicted fMRI activations. Contrastingly, the image reconstructions of decoders can be immediately and intuitively compared by the reader (with machine and human-based n-way identification experiments as a quantitative measure). Self-supervised learning is just one of the reasons we want to address the underrepresentation of encoders in existing work. From a neuroscientific perspective, encoding models have more potential for revealing the functional mechanisms of the visual pathway than decoders. Firstly, encoders model the actual process we are interested in, which is how the brain encodes visual information. By observing how encoders map images to brain activity we can extrapolate how it is done in the brain. Conversely, with decoders, the biological relevance is less straightforward,

being more closely related to tasks such as visual imagery and dreaming, which are of less neuroscientific interest than the much more common task of visual perception. In fact, it is conceivable that a high decoding accuracy does not necessarily require a high performance encoder if only a subset of the voxels are needed for image reconstruction. Additionally, decoding is much more likely to be an extrapolative process. There is, at most, equal amounts of visual information between the physical world and neural activations, however, it is much more likely that the brain does not perceive all the information contained within a visual stimulus and instead encodes only what is most relevant. The concept of attention is an example of this. Therefore, it is possible that decoding models do not reconstruct images using only the information present in fMRI, but additionally draw upon their knowledge of naturalistic images in general (eg. discussed as predictive coding in neuroscientific literature). Although this would result in more accurate reconstructions of the original visual stimulus, it may not be entirely accurate for the biological processes of interest such as visual imagery and dreams. Since an encoder's goal is to predict brain activity directly, we can be sure that they will only become more biologically relevant as they improve. However, in order to improve encoding models we must first address the limitations associated with interpreting and evaluating predicted fMRI brain activity. In this work, we focus on fMRI data (both predicted and ground truth) in order to gain a deeper understanding of encoding models. We also aim to make encoders more approachable by exploring how to better interpret and evaluate their results so that future work may use these methods when comparing and evaluating their own encoding models.

1.5 Model Limitations

Although data limitations certainly affect encoding and decoding models, it is likely that if these limitations were resolved, CNNs would still not perform as well on fMRI encoding and decoding as they do for other domains of computer vision. If this were the case, it would suggest that the differences between encoding/decoding and other computer vision tasks are too significant for existing fMRI encoding models to overcome. It is possible that CNNs are just not well suited for encoding and decoding fMRI data. For encoding, the current approach is generally to extract high-dimensional features using convolutional layers and then to map those features to each voxel individually. The best models [13][19] extract features from the images using CNN classifiers pretrained on large image datasets such as ImageNet [21]. These pretrained networks, designed for and pretrained on image classification, may not necessarily extract all the information relevant for encoding. If this is the case, then perhaps a more biologically plausible approach would be more suitable. Although neural networks, and consequentially deep learning, were originally inspired by the anatomy of the brain and physiology of neurons, it is generally accepted that they are not biologically plausible on a mechanistic level [22][23]. That being said, deep learning approaches can still be used to model biological processes from a functional perspective [3]. However, deep learning is a broad field, and the degree to which different types of neural networks are functionally similar to the brain varies. For example, one of the properties that makes CNNs so effective for computer vision is that they are translationally *equivariant*. When it comes to other transformations however, CNNs use pooling to ignore small changes making them partially (but not fully) viewpoint *invariant*. For image classification, invariance might be sufficient; the subject of an image remains the same regardless of the perspective of the camera. Biological vision, on the other hand, involves a deeper understanding of the objects within an image and how they relate to both the viewer and one another. For exam-

ple, it has been shown that different neurons activate depending on the orientation of the stimulus [24][25], demonstrating a form of rotational equivariance in the visual pathway. This provides part of the motivation for exploring capsule networks [26][27], which have been proposed to share more functional similarities with biological vision, as a basis for encoding models. Capsule networks are a novel type of neural network inspired by the cortical columns in the brain, which are considered to be elementary units of brain function. Unlike CNNs, they are mechanistically capable of full viewpoint equivariance. Additionally, they attempt to propagate information between layers in such a way that encodes a deeper understanding of the hierarchical relationship between features (see Section 2.6). CNNs, on the other hand, use pooling to create a hierarchy of features based primarily on scale. The so-called picasso problem is an example of when the latter naive approach can become problematic [28]. To the best of our knowledge, CNNs have so far been the only deep-learning approach used to model the visual pathway. In this paper, we explore capsule networks as the first deep-learning alternative to CNNs in order to determine if they are more or less suitable for modelling biological vision.

1.6 Contributions

In light of this introduction, this work aims to accomplish two main goals. The first, motivated by the model limitations described above, is to design a capsule network encoding model to predict fMRI-measured brain activity in response to images. Since no deep-learning based alternative to CNN's has ever been explored within the context of fMRI response to visual stimuli, this capsule encoder will provide a meaningful perspective on how different supervised approaches affect encoding performance. The second is to provide a better understanding of encoding models in general, allowing future works to draw better insights from their encoders. We believe encoding models

hold more potential for extrapolating the functional mechanisms of the brain. Additionally, encoding models have been shown to help address the data limitations by making use of existing image-only datasets to generate 'artificial' fMRI-image pairs. Hence, encoding models are critical to the quality of these artificial samples. We summarize the specific contributions of this work below:

- A capsule network-based encoder trained directly on paired image-fMRI data
- Hypotheses describing the behaviour of deep fMRI encoding models as well as limitations in evaluation methods
- An improved set of metrics and visualizations for evaluating and comparing encoders

Chapter 2

Background

2.1 Functional Magnetic Resonance Imaging

Functional magnetic resonance imaging (fMRI) has quickly become one of the most common methods for non-invasively measuring brain activity. Despite having low temporal resolution compared to other non-invasive methods such as electroencephalography (EEG), its relatively high spatial resolution and signal-to-noise ratio (SNR) makes it the most popular choice for computational models of the visual pathway. It is worth noting that compared to invasive methods, fMRI has a larger field of view (FOV) but lower spatiotemporal resolution and signal-to-noise ratio (SNR). However, the use of invasive methods is generally not an option for studies containing human subjects. fMRI is a special type of magnetic resonance imaging (MRI). An MRI image is a 3-dimensional volume composed of cubic points in space called voxels (the 3-dimensional version of pixels). MRI images are acquired using a phenomenon called nuclear magnetic resonance, which refers to the process of measuring the absorption and emission of radio-frequency (RF) energy from nuclear spins in an external magnetic field. For images of biological entities, including the brain, these nuclei are primarily hydrogen protons since water is the most abundant molecule in biological

tissue. Therefore, an anatomical MRI image, which generally refers to an image that was taken over a longer period of time in order to obtain higher spatial resolution and SNR, measures the presence of water and T1 properties in a volume such as the brain. Hydrogen, however, is not the only atom whose nucleus has a magnetic moment (i.e., spin number unequal to zero). Therefore, although water accounts for the majority of the NMR signal measured by MRI, there are slight variations in the signal due to presence of paramagnetic molecules. One such molecule is deoxyhemoglobin (dHb), the deoxygenated form of hemoglobin (Hb). Hb is a protein molecule present in the blood that is responsible for the transportation of oxygen between cells. In its oxygenated state, it is diamagnetic and does not have a strong effect on the NMR signal. However, after delivering its oxygen to a nearby cell, it becomes paramagnetic and disturbs the magnetic field, leading to a faster decay and hence increasing relaxation in $T2^*$ imaging. This results in a small measurable effect on the NMR signal. This small measurable effect, called the blood oxygenation level-dependent (BOLD) signal, is the signal of interest in fMRI. Neurons require energy to release neurotransmitter and establish ionic gradients through the cell membrane (or *fire*), energy for which they require oxygen to generate. The more rapidly a neuron fires, for example, the more energy it consumes. The more energy it consumes, the more oxygen it uses to generate more, resulting in a higher concentration of dHb in the surrounding blood. However, the concomitant increase in cerebral blood flow overcompensates for this oxygen consumption and leads to an increase in blood oxygenation. Therefore, one can see that the BOLD signal can be used as a proxy of brain activity. In fMRI, we take MRI images, generally in response to external stimuli, and use them to measure this BOLD signal at each voxel. In order to obtain the change in BOLD signal in response to the external stimuli, we must take multiple $T2^*$-weighted fMRI images in quick succession, which generally results in lower spatial resolution and SNR than T1-weighted anatomical MRI images. That being said, the change in BOLD signal

relative to baseline allows us to obtain an estimate of which areas of the brain were more or less active (and by what magnitude) in response to the given stimuli.

2.2 Brief Overview of The Visual Pathway

The visual pathway of the brain refers to the anatomical neural systems which have evolved over time to process visual information. Here we provide a brief summary of the human visual pathway based on the review from Prasad et al. [29] as needed to understand the work done in this thesis. The eye is the primary sensory organ for visual pathway and is responsible for collecting light, focusing it and encoding it into the lowest-level neural signals of the visual pathway. These neural signals then travel posteriorly along the left and right optic nerves (one for each eye). The optic nerves intersect with one another in the optic chiasm, located more centrally within the brain just above the hypothalamus. This intersection of the left and right optic nerves serves to combine information from the parts of the visual field that both eyes are able to see. Due to this combination of information, as the optic nerves exit this x-shaped intersection, they become referred to as the left and right optic tracts. The optic tracts then relay visual information to multiple regions of the brain. However most of this information arrives at the lateral geniculate nucleus (LGN), which acts as a bottleneck of information flow, filtering out visual information that is deemed irrelevant to the present behavioural state. The information that does manage to pass through the LGN eventually finds its way to the visual cortex. The visual cortex, located at the posterior of the brain in both the left and right the occipital lobes, is responsible for most of the processing of visual information. The primary visual cortex, also referred to as V1, is the first part of the visual cortex to recieve information and is often considered the lowest-level region of interest (ROI) for fMRI encoding and decoding. Neurons in V1 have been shown to perform the initial visual

processing of color composition, brightness, direction of motion and edge detection. From V1, visual information ascends two roughly parallel pathways referred to as the *dorsal* and *ventral* streams. The complexity of visual information as well as the receptive field of neurons, progressively increase along these parallel pathways. A diagram from Prasad et al. is shown in Figure 2.1. The dorsal stream is largely responsible for locating objects and processing the spatial relationships between them (eg. motion). Information from V1 progresses to the thick stripe areas of V2 and V3, then to the V5 region and eventually the medial superior temporal area (MT or MST). MT has been shown to process high-level information about an objects motion. Since motion is heavily dependent on time, processing in the dorsal stream has been shown to occur significantlly faster than in the ventral stream. The ventral stream on the other hand is largely responsible for visual recognition of objects. Information from V1 first progresses to the thin and interstripe regions of V2. The thin region has been shown to process color information while the interstripe region has been shown to process shape information. This information is then processed by neurons in the V4 region before progressing to regions in the cortex of the inferior temporal lobe where highly specialized neurons perform visual object processing. Examples include the fusiform face area (FFA), responsible for the visual processing of faces, and the parahippocampal place area (PPA), responsible for the visual processing of scenes. The ventral stream also contains the lateral occipital cortex (LO or LOC) which is believed to recieve information directly from regions such as V2, V3, V4 and V5 and is primarily responsible for detecting objects (as opposed to non-objects). In the datasets used in this work, the lower visual cortex (LVC) is defined as containing V1, V2 and V3, and the higher visual cortex (HVC) is defined as containing LOC, FFA and PPA. Additionally, we note that the visual pathway does not exclusively process information in a *bottom-up* approach from lower-level to higher level regions, but in fact contains feedback mechanisms where information is transferred in the

reverse direction. This feedback information is believed to provide context to lower level regions. Both feedback and the interconnectivity of LOC demonstrate that although we can simplify the visual pathway into major streams of information, in reality it is an extremely complex, interconnected network with various parallel, skip, and feedback connections that transfer and process information in a way that is yet to be fully understood.

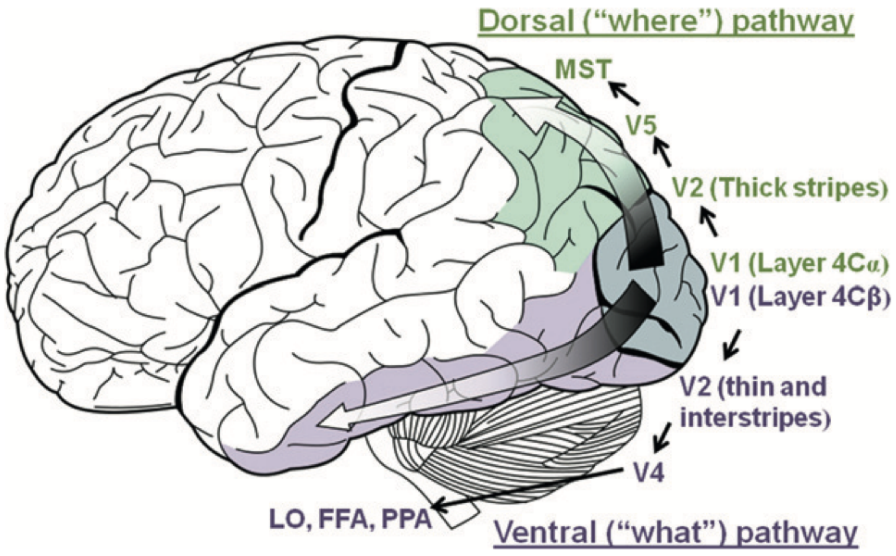


Figure 2.1: Diagram outlining the dorsal and ventral streams of the human visual pathway

2.3 Defining Encoding Models

Encoding models, or encoders, attempt to model processes in the brain by predicting brain activity in response to external physical stimuli. Existing work has largely focused on visual encoders, or more specifically image encoders with fMRI as the modality for measuring brain activity, as outlined in Figure 2.2. When brain activity is measured using fMRI, a 3-dimensional brain volume is sectioned off into (typically cubic) voxels, each of which has its own measured activation derived from the

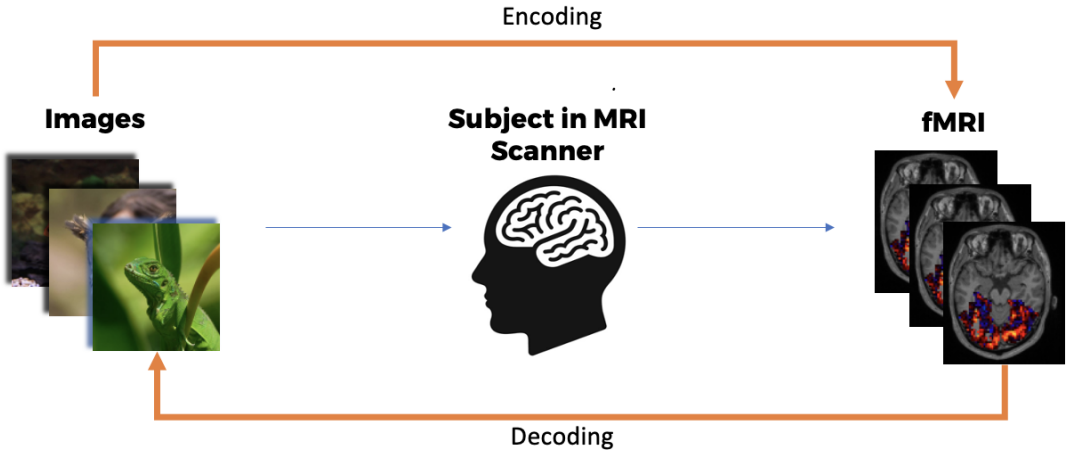


Figure 2.2: Diagram outlining image-fMRI encoding and decoding. The subject's brain encodes the images into brain activity which is measured using functional magnetic resonance imaging (fMRI). Each image is accompanied by an fMRI measurement of the brain activity it elicited. Computational models of this encoding process are referred to as encoders or encoding models. Conversely the inverse process is called decoding. Computational models of the decoding process are referred to as decoders or decoding models

BOLD signal. Positive and negative activation values typically indicate increased or decreased neuronal firing (brain activity) in response to the image respectively. The magnitude of the activation value is proportional to the change in brain activity from baseline. Although several fMRI brain volumes are usually acquired during the stimulus presentation it is common practice in computational neuroscience to ignore the time dimension and average these volumes to obtain a static estimate. Hence, we can define an image-fMRI encoder as a model that takes an image as an input and predicts a set of fMRI-measured activations of voxels in a brain volume. From this point onward we omit the prefix 'image-fMRI' for readability, one may assume that all encoders in this work are image-fMRI encoders.

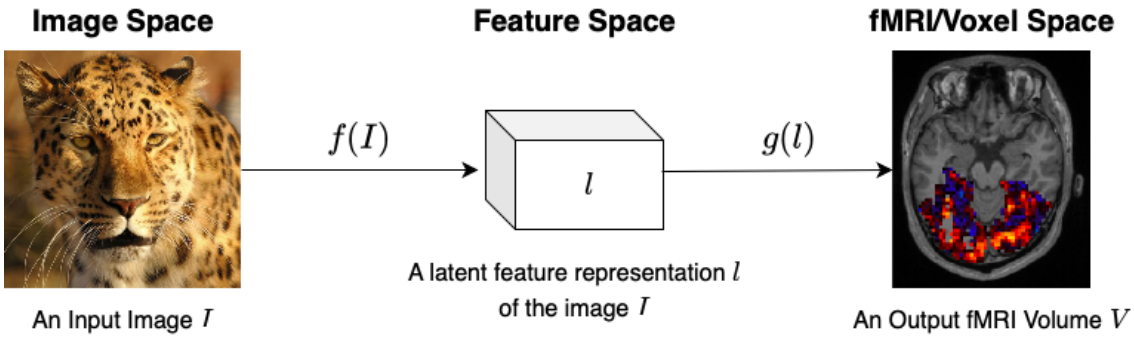


Figure 2.3: Diagram outlining a high level summary of the encoding process for computational models. The function f represents the mapping from images I to features l. The function g represents the mapping from features l to a volume of fMRI voxels V.

2.4 A History of Image to fMRI Encoding Models

2.4.1 Linearly Mapping Handcrafted Features

Predicting fMRI brain activity in response to stimulus images has, thus far, largely taken the following form. First, features are extracted from the stimulus images into some latent feature space representation. Then, these latent representations of the original images are mapped to the fMRI voxel activations. We can therefore conceptualize the encoding process as sequentially traversing three distinct mathematical spaces:

1. Input/Image Space: *defines images composed of pixel intensities.*
2. Feature Space: *defines image features using some abstract latent space*
3. Output/fMRI Space: *defines brain activity represented by voxel activations*

These are further outlined in Figure 2.3. Although encoding models have used a variety of methods to extract features from images (f in Figure 2.3), for the process of mapping these features to voxel activations (g in Figure 2.3), linear regression has been used by all but the most recent encoders (St. Yves 2018 [13], Beliy 2019 [30], Gaziv 2022 [19]). Therefore, the mapping from image to feature space (f) is

what differentiates much of the existing work. Many of the earliest encoding models approached this problem by using handcrafted features. Some of the first attempts at fMRI encoding used discrete labels to represent features within the images (eg. face versus houses). Kay et al. [31] were the first to extract more complex features from images using the gabor wavelet pyramid (GWP), a series of gabor wavelets with varying spatial frequency, orientation, field-of-view, and position. Although this approach was able to learn the basic retinotopic properties (angle and eccentricity) of voxels early within the visual pathway (V1, V2, V3), it was still limited in its predictive power; test predictions for only a small subset of the voxels ($< 30\%$) were able to meet the threshold for statistical significance. Naselaris et al. [32] later showed that although this approach was better for voxels in the early visual cortex, a model that used categorical semantic labels as features could better predict voxels higher in the visual processing hierarchy. Guclu et al. [33] were able to slightly improve upon a gabor wavelet based approach by using unsupervised sparse coding, however, this model was also only tested using voxels in the early visual pathway. Naselaris et al. later used a GWP-based encoder as part of their decoding approach for mental imagery [5].

2.4.2 The Introduction of Deep Learning

The introduction of deep learning brought many advances to the field of computer vision, namely image classification. These advances were largely driven by deep CNN models which, in combination with increasingly large image datasets, were able to learn better and better feature representations of images. Although these networks were designed and trained for image classification, the features they extracted from images proved to be richer and more relevant to fMRI encoding than previously used handcrafted features. Guclu et al. were one of the first to use a pretrained CNN for fMRI encoding [10]. Image features could be extracted using a pretrained image

classifier by taking the activations from various layers of the network in response to the input image. Their model outperformed the previous state-of-the-art (gabor wavelet pyramids) in all areas of the brain, particularly those associated with higher-level visual processing (V4 and LO). Guclu et al. compared various different pretrained networks including several VGG-net based classifiers, but found that all of them performed similarly to a much simpler Alex-net based model. This suggested that given the same pretraining dataset (Imagenet 2012), architectural differences between pretrained models did not have a significant effect on encoding models that used linear regression for the mapping g in Figure 2.3. Eickenberg et al. [9] used a different CNN-based classifier (Large version of OverFeat model from Sermanet et al. [34]) which was also pretrained on ImageNet. With their model, they showed that encoders could generalize across different datasets consisting of previously unseen stimuli. These datasets included novel image types as well as video stimuli. Wen et al. [8] also encoded video stimuli with the help of an Alex-net based model pretrained on Imagenet. Unlike Eickenberg et al., who used the average activity across 30 frames, Wen et al. extracted features individually from each frame of the stimulus video. Principal component analysis was then used to reduce the dimensionality of these features prior to using linear regression to map them to fMRI voxel activations. In all of these works, it was found that the pretrained CNN classifiers followed a similar processing hierarchy to the biological visual pathway. Features extracted by layers earlier in the pretrained networks were better for predicting voxels in the early visual pathway whereas features extracted from layers deeper in the networks were better for predicting voxels higher in the visual pathway. This suggested that CNNs might share some functional similarities with biological neural networks in the ventral stream.

2.4.3 A More Biologically Inspired Use of Deep Learning

As existing work explored how to best use CNNs to map images to features (f in Figure 2.3), when it came to mapping these features to voxel activations (g in Figure 2.3), the comparatively simple approach of linear regression remained the sole approach. That was until St Yves et al. [13] introduced their feature-weighted receptive field (fwRF) model which, inspired by retinotopic experiments, assigned a Gaussian receptive field to each individual voxel. The position and variance of these Gaussian receptive fields were selected using a grid search and then used to weight the 2d feature maps extracted from the stimulus images. Linear regression was then used to map these now-modified features maps to voxel activations. The authors used both GWP and an Alexnet CNN pretrained on Imagenet to extract the initial 2d feature maps. They found that using the CNN-extracted features resulted in better encoding performance than the GWP-extracted features, particularly in regions associated with more complex visual processing (V3, V4, LO, and AOC). They also observed that layers deeper in the pretrained CNN contributed more to these higher-order visual regions. Both of these findings agree with the observations of existing work discussed in the previous section (Section 2.4.2). When using the same pretrained CNN for image feature extraction, however, their fwRF approach encoded voxels significantly better than linear regression in all regions of the brain. This showed that linear regression does not necessarily capture all the information when mapping from feature space to fMRI space and that this mapping could be improved by introducing non-linearities such as fwRFs.

2.4.4 Fully Embracing Deep Learning

Although St. Yves et al. were the first to use a non-linear method to map image features to voxel activations, their approach was still relatively rigid. The Gaussian receptive fields introduced only a few new parameters which were learned using a

grid search. Conversely, Beliy et al. [30] designed an encoding model which had significantly more freedom to learn the mapping from features to voxel activations on its own. Their encoder consisted of a single pretrained Alexnet layer followed by two trainable convolutional layers and a linear regression layer. Each of the convolutional layers was accompanied by a ReLU activation function and batch normalization. This model was trained directly on image-fMRI data after which it was used to support a decoding model in a self-supervised manner by providing fMRI labels to unpaired images. Although the work of Beliy et al. was focused on their decoding model, their encoder proved to be unique. It was the first encoding model that used deep learning methods trained directly on image-fMRI data. Previous works had only ever used deep learning layers pretrained on image classification. When compared to the fwRFs introduced by St. Yves, the CNN layers introduced by Beliy et al. introduced significantly more trainable parameters with which their encoder could non-linearly transform the image features. Although this gave their model more freedom when training, it blurred the lines between image space, feature space, and fMRI space by having a single model perform the f and g mappings from Figure 2.3. The trainable CNN layers could be interpreted as extracting more image features to create a richer feature space, or as being part of the mapping from features to voxel activations.

2.4.5 Combining Approaches to Achieve State of the Art

In a follow-up work to Beliy et al. [30], the same authors introduced a new encoding model in Gaziv et al. [19] which combined their previous approach with the fwRFs of St. Yves et al. [13]. They used feature maps extracted from multiple layers of a VGG19 CNN pretrained on ImageNet. The feature maps from each layer were non-linearly transformed using trainable convolutional layers accompanied by the ReLU activation function and batch normalization. This was followed by a custom space-feature locally-connected layer which stacked the 8 neighbors of each spatial

coordinate along the channel dimension resulting in 8 additional channels for each feature map. The resultant feature maps were then weighted using the fwRF approach from St. Yves et al. before being linearly mapped to the voxel activations. This model significantly improved encoding performance when compared to the encoder from Beliy et al. and resulted in state-of-the-art decoding performance when used with the author's self-supervised approach. It is worth noting however that the encoders of St. Yves et al. and Gaziv et al. were trained with different datasets and have yet to be directly compared in existing literature. Therefore, we consider both of them to be the current state-of-the-art for encoding naturalistic images into fMRI voxel activations.

2.5 Relation to Decoding Models

If encoders predict fMRI brain activity in response to visual stimuli, then decoders can be defined as reconstructing the original visual stimuli given the fMRI brain activity as an input, as outlined in Figure 2.2. This is a closely related inverse problem that, compared to encoding, has been more extensively explored in existing work. Similar to encoding, recent state-of-the-art decoders have made extensive use of deep learning. These include various different approaches such as self-supervised learning [30][19], U-Nets [35], pretrained CNNs [6] [36], generative adversarial networks [7][37][38][35][13][39][40] and variational autoencoders [37] [38]. That being said, although state-of-the-art decoders have used a wide variety of different model types and architectures, all of these have remained CNN-based. The exception is Qiao et al. [41], which also explores the use of capsule networks within the field of image-fMRI encoding and decoding. However, this work differs from our work significantly in that their capsule network was pretrained on image classification and used as a feature extractor. To the best of our knowledge, we are the first to train capsule networks, or

any DNN alternative to CNNs, directly on image-fMRI pairs.

2.6 Capsule Networks

In this paper we explore capsule networks as a deep learning alternative to CNN's for modelling the human visual pathway. Capsule networks attempt to address some of the fundamental limitations of CNNs. The first is a lack of hierarchical understanding between features. Using pooling, CNNs attempt to create a shallow hierarchical understanding of how features relate to one another based on scale. Larger features rely on the presence of smaller features within the feature detection kernel and so on. However, in order to ascend from one scale to the next, pooling reduces information about the precise spatial location of features at the previous scale. The spatial information within layers therefore gets coarser and coarser resulting in neurons being forced to rely solely on the presence of features within their receptive field as opposed to any interaction between them. The picasso problem, the phenomenon in which a CNN classifier cannot discriminate between a real face and one which has been distorted to have the eyes, nose, and mouth in the wrong locations, is an intuitive example of how the presence of features alone is not sufficient for a true visual understanding. This lack of understanding is related to another limitation of CNNs, which is their lack of viewpoint equivariance. A system or function can be described as equivariant if transforms to the input result in *equivalent* transforms to the output. One of the properties of CNNs that makes them incredibly effective is the fact that they can be translationally equivariant. Translating features in an image will result in those features being translated in the output feature map since the convolutional kernel slides across the entire image. We note that some CNNs use pooling to make themselves translationally invariant across spatial scales. Invariance can be described as producing the same output for a given input regardless of what

transformations are done to it. This is important for image classification as we want the output prediction to be 'dog' regardless of where the dog is in the image. For vision, we are generally interested in viewpoint transformations as a result of viewing a 3-dimensional object from different points in space, or even non-homogeneous transforms such as those caused by lens distortions. Translation represents just one of these many possible types of transforms, the rest of which CNNs have no form of invariance or equivariance. One way that CNNs try to make up for this is by learning multiple rotated copies of the same feature detection kernel in order to detect that feature regardless of its orientation. However, from the perspective of the CNN, these kernels have no relation to one another, and it can only do this for simple features which naturally occur at various different orientations in the training dataset. One can see from these various limitations that CNNs learn the patterns of images statistically, but do not have a true visual understanding of the images. Additionally, these limitations are inherent to the fundamental mechanisms of CNNs, and are difficult to address through traditional means such as new model architectures. This is what Sabour et al. wanted to address when they introduced their idea of capsule networks in [26]. Although the inspiration behind capsule networks is extensive, they can effectively be reduced down to two major novelties. We first list these below for clarity and subsequently discuss how these properties work together to address the aforementioned limitations.

1. Scalar units or 'neurons' are replaced with vectors called *capsules*
2. Information between layers of capsules is propagated forward using a *routing* algorithm

The use of vectors as opposed to scalar activations allows capsules to carry additional information about the entities or features that they represent. Traditionally, the activations within a CNN's feature map represent the presence of a particular feature at

that location. By introducing additional dimensions capsule networks encourage the model to learn not only how to detect the presence of features, but to detect the properties of those features as well. In order to make use of this additional information, Sabour et al. proposed the concept of routing algorithms. In CNNs, information is propagated between layers using a weighted sum of the scalar activations within the previous layer. Routing algorithms however use the additional information contained within capsules to force the network to be more discriminative. Namely, parent capsules (capsules in a subsequent layer) dynamically weight child capsules (capsules in a previous layer) differently for each input using *agreement*. The concept of capsule agreement is best explained with an example. Say there are several child capsules that represent the part features (features that are part of some larger feature) of a face such as a mouth, nose, eyes, etc., and that their additional dimensionality contains information about the position of that feature (ie. its location, orientation, etc.). For the parent capsule that represents faces, it receives a vote from each of the child capsules on its own position. The mouth would predict the face to be slightly above it, the nose would predict the face to be centered on it, and so on. If the child capsules make very different predictions about the position of the face, then this would imply that their positions relative to one another is not as we would expect for the faces we have seen before. Therefore, these child capsules are said to disagree with one another about the position of the parent capsule. Routing algorithms take advantage of this by iteratively weighting the child capsules in a way that places greater importance on capsules that agree with each other. Hence, in order to activate the parent capsule that detects faces, not only do the part features need to be present, but they also need to be oriented in such a way that they agree on the position of the face they are a part of. Since the agreement between child capsules changes depending on the input, they are effectively being weighted dynamically. This example uses position as the additional information contained within the child capsules, but in fact,

the capsule network could learn to use these additional dimensions to represent any kind of information, such as the entire pose of the feature. Since positions and viewpoint transformations in 3D space can be fully represented using 4x4 pose matrices, this makes capsule networks mechanistically capable of full viewpoint equivariance given only 16 dimensions. Additionally, this forces capsule networks to learn how part-features relate to one another and the object they are a part of. This approach was largely inspired not only by computer graphics, but by biological vision as well, which is also thought to impose perceptual coordinate frames when interpreting visual stimuli [42]. In the paper that introduced capsule networks [26], Sabour et al. used a convolutional layer to generate the lowest-level capsules, referred to as primary capsules. These capsules were then connected to their parent capsules using *dynamic routing*. This routing algorithm uses the scalar product between the votes from each child capsule and the current estimate of the parent capsule as the measure of agreement. Both the value of the parent capsule and the agreement of each child capsule are updated over several iterations. Each dimension of the 8-dimensional capsules were used to represent the properties of the feature, and the capsule's activation, which represents whether or not the feature it represents is present, was calculated by using a non-linear squash function on the values within the capsule. The authors also included the addition of a feed-forward reconstruction network which using the last layer of capsules, referred to as the output capsules, reconstructed the original image. The difference between the original image and the reconstructed image was added to the loss as a form of regularization. Using this approach, Sabour et al. achieved state-of-the-art performance in the classification of both handwritten digits from the MNIST dataset and simple 3D objects from the smallNORB dataset. In a follow-up work by Hinton et al. a new state of the art was achieved on the smallNORB dataset. In this work, they proposed a deeper capsule network with convolutional capsules where, reminiscent of convolution, only capsules within a specific

receptive field were routed to the parent capsule. They also introduced a new routing algorithm called EM routing based on the expectation maximization (EM) clustering algorithm. The value of the parent capsule can be thought of as the mean of the cluster, its activation can be thought of as the compactness of that cluster and the agreement of the child capsules can be thought of as their distance from the mean. Additionally, they encouraged the capsules to learn pose information by making them 4x4 matrices instead of vectors and using a technique called coordinate addition to account for their position in the feature map. The votes were therefore calculated by doing matrix multiplication with 4x4 transformation matrices on the child capsules in order to encourage the network to learn pose transformations. Both of the capsule networks from Sabour et al. and Hinton et al. proved to be more robust to adversarial attacks than existing state-of-the-art CNN-based networks. The main disadvantage of capsule networks is that they tend to be computationally expensive despite generally having fewer parameters than CNN's, however, various works have already begun to address this and improve the computational efficiency of capsules [43][44]. In this thesis, I have applied various forms of capsule networks and applied them as encoders of images into fMRI activations.

Chapter 3

Materials and Methods

3.1 Data

We first validated my implementation of capsules by training a capsule network to classify handwritten digits from the MNIST dataset as in Sabour et al. [26]. The results from this validation are included in Appendix A.1.

We then trained both CNN and capsule based encoding models using two different and publicly available image-fMRI datasets. We summarize these datasets in Table 3.1. Notably, each dataset differs in the complexity of the stimulus images. The 'MNIST-fMRI' dataset [45] uses MNIST images [46] of handwritten digits from two distinct classes (9 and 6). The 'ImageNet-fMRI' dataset [36] uses more complex naturalistic images from 200 distinct ImageNet [21] categories. Note that we downsample these images from their original resolution of 224x224 down to 112x112 in order to accommodate the pretrained Alexnet layer of the Beliy model [30] used as a baseline in this work. Each of the 50 test stimuli in the Imagenet-fMRI dataset were drawn from categories not present in the training set. For both datasets, we use the steady-state voxel activations provided by the original authors. The steady-state activations are derived from the BOLD signal by averaging fMRI volumes acquired within specific

time periods (shifted to account for hemodynamic delays) corresponding to a specific stimulus presentation. This results in a single steady state volume for each stimulus image where the intensity values (*activations*) at each voxel within the volume represent increased or decreased BOLD signal in response to the image. Additionally, as with existing work, we standardize these activations to zero mean and unit variance for each voxel individually. For the MNIST-fMRI dataset, we standardize the test fMRI, which were randomly separated from the original dataset, using statistics calculated from the remaining training set samples. This is a standard approach that ensures information from the test data does not leak into the training set. For the ImageNet-fMRI dataset, which is much larger and was acquired over multiple sessions, we standardize each run (separate sessions within the MRI scanner) individually. We take this approach in order to stay consistent with the preprocessing of Beliy et al. [30]. We note, however, that for this dataset, testing and training samples were acquired in separate sessions, and therefore the statistics used to standardize the training data were derived from training samples alone. There were a total of 35 test sessions during which the same 50 images were presented each time to obtain repeated fMRI measurements. We averaged across these repeats to obtain a single, higher SNR, fMRI measurement for each of the 50 test stimuli images. Lastly, we note that the MNIST-fMRI dataset does not contain any spatial information such as voxel coordinates or functional masks and therefore can not be mapped to anatomical space.

3.2 Model Architecture

In our experiments, we use the same original capsule network architecture from Sabour et al. [26] except we modify the number of output capsules to better fit the fMRI datasets being used (see the end of this paragraph for more details). The

	Image-fMRI Dataset	
	MNIST-fMRI [45]	ImageNet-fMRI [36]
N train samples (K repeats)	80(1)	1,200(1)
N test samples (K repeats)	20(1)	50(35)
N voxels	3063	4643
N image categories (total/train/test)	2/2/2	200/150/50
Image Resolution	28x28	112x112

Table 3.1: Datasets used in experiments. *Samples* refers to a stimulus image paired with an fMRI recording of brain activity. *Repeats* refers to the number of times each stimulus was presented. All values in the table are 'per subject'

primary capsule layer converts the 2D scalar feature maps into capsule form. It is essentially a linear convolutional layer that reshapes its outputs into capsule vectors and then uses the same squash function from Sabour et al. [26] as its non-linearity. The primary capsules are then passed through a fully connected capsule layer to generate the output capsules. The fully connected capsule layer linearly weights the primary capsules and uses dynamic routing as its non-linearity. Due to computational constraints, there are too many voxels to have them be represented by their own individual output capsule. Instead, we map a computationally feasible number of output capsules to the voxel activations using 3 fully connected layers. For these layers, we use the same number of hidden units as the reconstruction network in Sabour et al. [26]. A summary of this architecture can be seen in Figure 3.1. For the MNIST-fMRI dataset, we use two output capsules (A=2) since there are only two image classes in the dataset (see Table 3.1). For the ImageNet-fMRI dataset, having an output capsule for each image class exceeds computational constraints. Furthermore, the test images are from novel classes, therefore encouraging output capsules to represent image classes from the training set would not improve generalization. Hence, in our experiments, we leave the number of output capsules unchanged (A=10) from the original implementation by Sabour et al. [26]. Although we experimented with various different architectures, as seen in Appendix A.1, these did not result in any

significant improvements.

As a baseline for CNN-based encoders, we considered the models from both Beliy et al. [30] as well as St Yves et al. [13]. To the best of our knowledge, these were the state-of-the-art models at the time. We chose to use the former since it had already been implemented with the ImageNet-fMRI dataset.

Since our capsule network has significantly more network parameters than the Beliy encoder, we also include a larger CNN-based model in some of our experiments for comparison. This model has the same architecture as Beliy et al. [30] except the number of channels in the convolutional layers (excluding the first layer which is pretrained) was increased to 192. This modification resulted in the large CNN having a comparable number of parameters to the capsule-based encoder (10M and 9.3M respectively).

Lastly, we also include results from what we refer to as a *naive* model. This model first uses the training set to calculate the mean activation for each voxel. It then uses this set of means as its prediction of fMRI BOLD activity regardless of the input stimulus image. The only difference between predictions is a small amount of added noise drawn from an exponential distribution. This model is meant to emulate the behavior observed by more complex models when they are unable to find a meaningful relationship between the stimulus images and a particular voxel. We chose exponential noise with a rate parameter of $\lambda = 25$ because it most closely resembled our observations of this behavior

3.3 Model Training

For the MNIST-fMRI dataset, we train all the encoders using the Adam optimizer with its default TensorFlow settings ($\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 1 \times 10^{-7}$, $lr = 0.001$) as in Sabour et al. [26]. We use a batch size of 8 since this dataset has few training

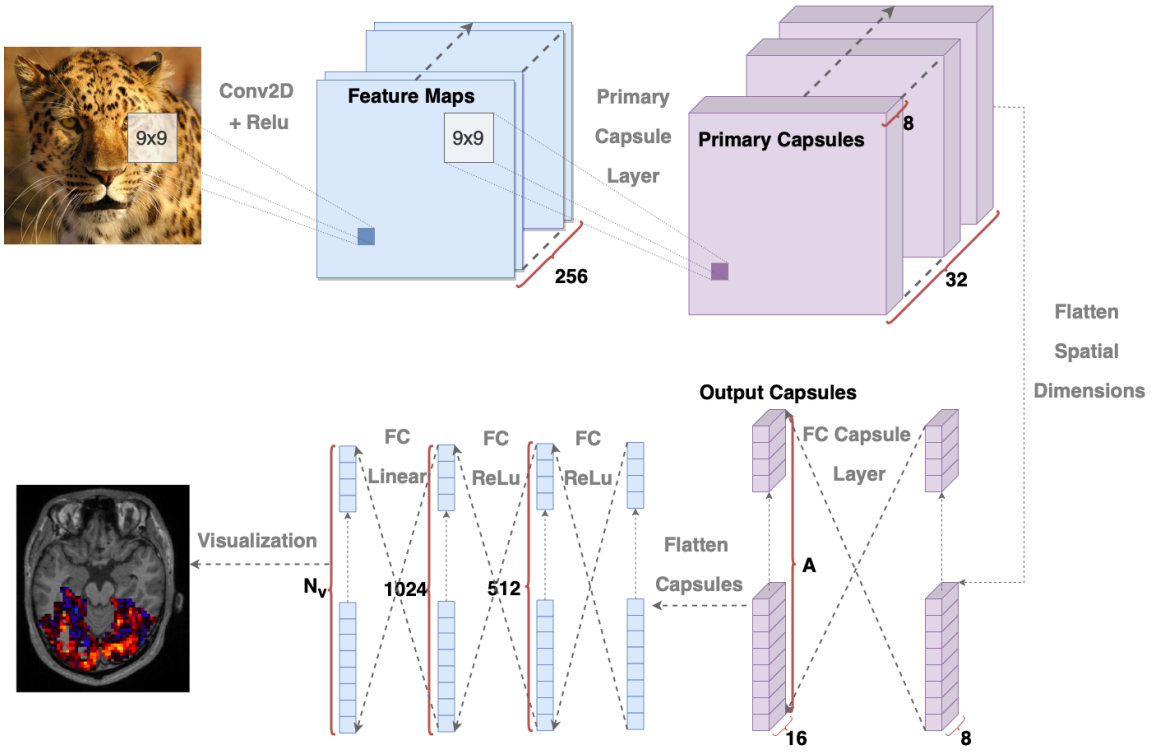


Figure 3.1: Capsule Encoder Architecture. Blue layers are scalar feature maps and purple layers are capsule feature maps. A is the number of output capsules and N_v is the number of voxels being predicted. Routing occurs between the primary and output capsules after flattening the spatial dimensions (In the Fully Connected Capsule Layer)

samples and only two image classes. Note that we step down the learning rate by a factor of 10 at various milestones throughout training in order to prevent the model from overfitting too quickly. We trained the capsule network for 400 epochs with learning rate step-downs at epochs 10 and 20. The Beliy encoder was trained for 1200 epochs with learning rate step-downs at epochs 25 and 45. For the larger CNN encoder, we train for 200 epochs with a learning rate step-down at epoch 25. These training schedules were chosen based on how fast the models converged relative to one another.

For the ImageNet-fMRI dataset, we train the CNN-baseline encoder from Beliy et al. [30] with its original hyperparameters. These were to train the model for 80 epochs using the SGD optimizer with a Nesterov momentum of 0.9, a batch size of

64, an initial learning rate of 0.1, and learning rate step-downs (by a factor of 10) at epochs 25, 45 and 65. In order to stay consistent with Beliy et al. [30], we use the same optimizer, batch size, and initial learning rate when training the capsule encoder. However, instead we train the model for 200 epochs with a single learning rate step-down (by a factor of 10) at epoch 20. We chose this schedule because the larger capsule network was more prone to overfitting when compared to the much smaller encoder from Beliy et al. [30].

For all models, we use the same loss as in Beliy et al. [30] which was a combination of mean squared error (MSE) and the cosine similarity between the ground truth and predicted fMRI volumes. The MSE was weighted by a factor of 1 and the cosine similarity was weighted by a factor of 0.1. Additionally, during training, we randomly shift the stimulus images in each batch by up to 5 pixels in any direction. This was initially used by Beliy et al. [30] to account for eye movement, therefore we use it here for consistency when training all supervised models.

3.4 Model Evaluation

One of the major challenges in developing fMRI encoding models is the difficulty in interpreting the data. Both images and fMRI data are complex and high dimensional, and mapping one to the other is a difficult multi-regression problem. In order to address this we need efficient and effective methods for evaluating model outputs. For decoders, since their outputs are in image space, our own biological vision allows us to quickly and intuitively compare them to the original stimulus images. Encoders, on the other hand, output predictions in the fMRI space, making interpretation unintuitive and evaluation difficult. To resolve this we look closely at existing metrics for evaluating encoders to identify how they behave in the fMRI space. Furthermore, we introduce new metrics that capture important aspects of model performance pre-

viously overlooked. For clarity, we explicitly describe the metrics used in this paper below. We also use our analysis to capture the most relevant information about models and visualize them as efficiently as possible. Our goal is to make encoding models more accessible by demonstrating how to best and most intuitively interpret their results.

3.4.1 Voxelwise versus Samplewise Statistics

When quantitatively evaluating fMRI data, the method by which one computes their metrics significantly affects not only the value of the metric, but its interpretation as well. Here we define two methods by which to compute quantitative metrics using fMRI data; *voxelwise* and *samplewise*. We define voxelwise statistics as those that are computed using all the activations for a single voxel, which we refer to as voxelwise vectors, in response to some set of images. Therefore, when computing voxelwise metrics using some validation set, a single value is obtained for each voxel predicted by the model. We define samplewise statistics as those that are computed using the activations for all of the voxels in response to a single stimulus image, which we refer to as samplewise vectors. A samplewise vector can also be thought of as a single fMRI brain volume. When computing a samplewise metric using some validation set, a single value is obtained for each stimulus image, or sample, in the set. We note that for some metrics, such as mean squared error (MSE), the average voxelwise MSE is equivalent to the average samplewise MSE; however this is not the case for metrics such as Pearson's correlation coefficient. Figure 3.2 shows a visualization of the difference between samplewise and voxelwise vectors.

3.4.2 Metrics

Voxelwise Correlation We define voxelwise correlation as the Pearson's correlation coefficients between the predicted and ground truth fMRI activations, computed

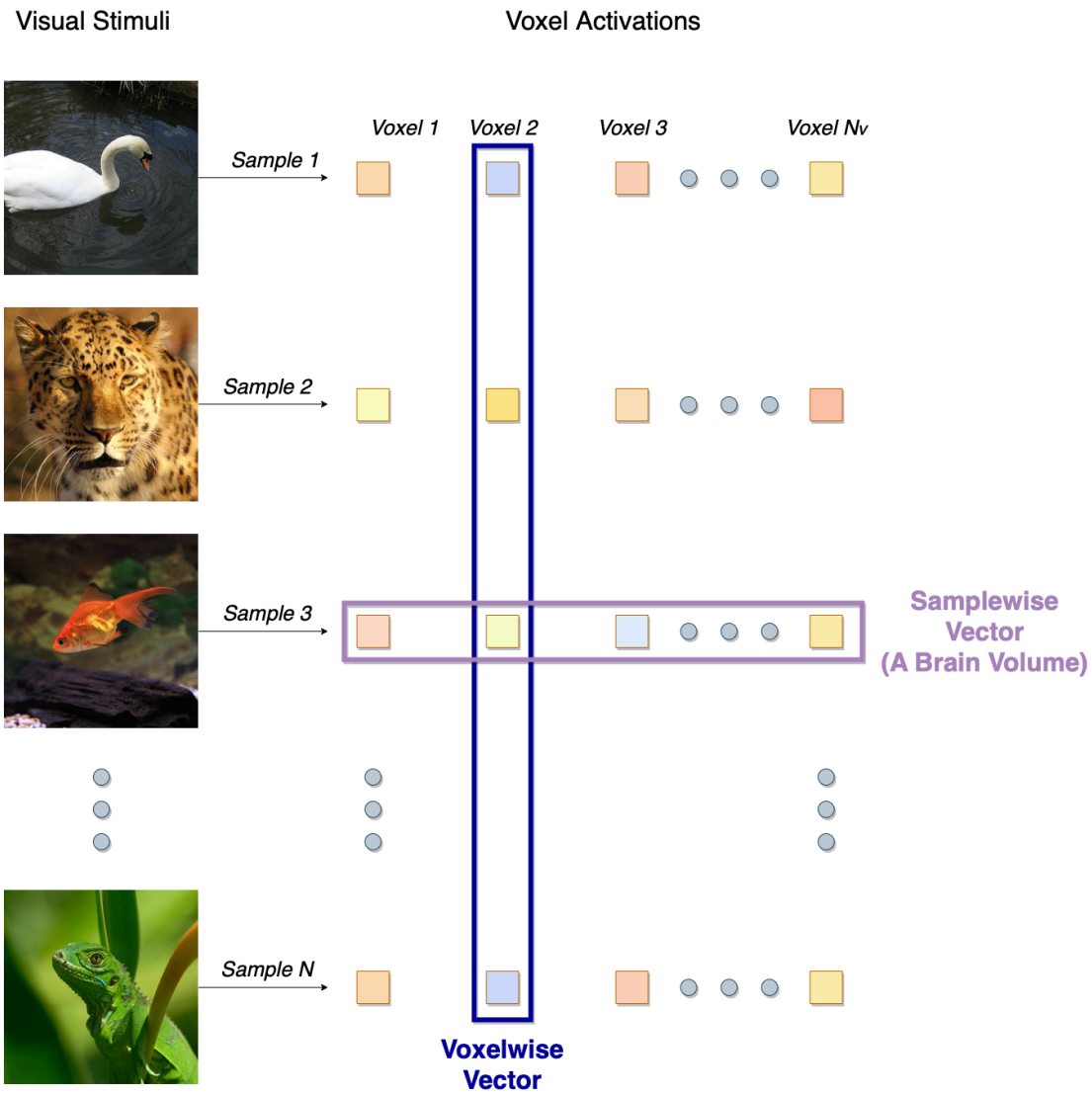


Figure 3.2: Illustration of the difference between voxelwise and samplewise vectors. Voxelwise metrics are computed using voxelwise vectors and samplewise metrics are computed using samplewise vectors

voxelwise. In this paper, we abbreviate voxelwise correlation as *Voxel Corr*. Furthermore, when presented as a scalar value, this represents the mean voxelwise correlation for all voxels unless stated otherwise. We note that in most existing work, including the state-of-the-art encoders from Gaziv et al.[19] and St. Yves et al.[13], voxelwise correlation is often referred to as *accuracy*. However, we argue that this simplification is inappropriate. *Accuracy* implies an absolute measure of correctness for which vox-

elwise correlation is insufficient. Voxelwise correlation captures whether the predicted activation for a specific voxel is varied appropriately in response to the input stimuli, however, it ignores its absolute magnitude. For a single-voxel encoding model, one might propose linearly scaling the predicted activations. However, voxels are often predicted with varying relative magnitudes and would therefore each require a different linear scaling. This example demonstrates a lack of understanding in the encoding model and a level of *inaccuracy* not captured by voxelwise correlation. To the best of our knowledge, nearly all existing work uses voxelwise correlation as the only performance metric when evaluating encoding models. The works from authors such as Gaziv et al. [19], Allen et al. [14], Beliy et al. [30], St. Yves et al. [13], Wen et al. [8], Guclu et al. [10], Naselaris et al. [32] and Kay et al. [31] use voxelwise correlation as the sole performance metric when evaluating their encoders. All of these works except Kay et al. refer to voxelwise correlation as *accuracy*. Eickenberg et al. [9] and Guclu et al. [33] differ slightly by using the squared voxelwise correlation. The algonauts challenge outlined by Cichy et al. [47][48] was the only work we found which did not use voxelwise correlation. Instead, they used representational similarity analysis (RSA) which we note is derived from samplewise correlation (discussed below) and also does not take into account the magnitude of predictions.

Samplewise Correlation We define samplewise correlation as the pearson's correlation coefficients between the predicted and ground truth fMRI activations, computed samplewise. In this paper we abbreviate samplewise correlation as *Sample Corr*. Furthermore, when presented as a scalar value, this represents the mean samplewise correlation for all samples in the validation set. Unlike voxelwise correlation, this metric gives a measure of how well fMRI responses to individual stimuli are predicted. However, it also does not take into account the magnitude of the predictions in relation to the ground truth.

Voxel Response Sensitivity We define a model's voxel response sensitivity as the standard deviation of its predicted activations for a set of stimuli computed voxelwise. Unlike the correlation metrics, voxel response sensitivity exclusively contains information about the magnitude of the predictions. A large response sensitivity indicates the voxel's predicted activations are more varied in response to the input images, whereas a low response sensitivity indicates the model predicts approximately the same value for that voxel regardless of the input image. In this paper, we abbreviate voxel response sensitivity as *Voxel RS*

Response Weighted Correlation In order to take into account both the pattern and magnitude of model predictions at once we use a voxel's response weighted correlation (RWC). We define RWC as a voxelwise metric that is calculated by multiplying a voxel's voxelwise correlation with its response sensitivity and then taking the square root of the resultant value. Note that sign is ignored when taking the square root and reapplied afterwards. This is summarized in Equation 3.1 where ρ is the voxel's voxelwise correlation, σ is the voxel's response sensitivity and $sgn()$ is the signum function.

$$RWC = sgn(\rho) \cdot \sqrt{|\rho\sigma|} \tag{3.1}$$

RWC Skew Since response weighted correlation measures how well a voxel is predicted taking into account both the pattern and magnitude of predictions, we define the skew of the voxel's RWC to characterize the contributions of these components relative to one another. RWC is calculated using the formula shown in Equation 3.2 where θ is the RWC skew measured in radians, ρ is the voxelwise correlation and σ is the response sensitivity. The RWC skew ranges from -45 degrees to approximately 45 degrees (assuming $\sigma \leq 1$) with 0 degrees representing an equal voxelwise correlation and response sensitivity. A negative skew indicates that the voxel's predicted pattern

or correlation is more dominant whereas a positive skew indicates that the voxel's response sensitivity is more dominant.

$$\theta = tan^{-1}\left(\frac{1-\sigma}{1-\rho}\right) - \frac{\pi}{4} \tag{3.2}$$

3.4.3 Bootstrapping

In this work we use a bagging aggregation approach to test the significance of our results as well as generate confidence intervals for the various metrics. To calculate a metric using bagging aggregation we first create a number of bootstrap replicates of the validation set. The bootstrap replicates were created by sampling the validation set with replacement in order to create new validation sets with different combinations of the original validation samples. We sample the same number of image-fMRI pairs contained as are contained in the original validation for each bootstrap replicate so that they have the same number of samples as the validation set. For the ImageNet-fMRI dataset this was 50 samples and for the MNIST-fMRI dataset this was 20 samples. Since the image-fMRI pairs are sampled with replacement, they may appear multiple times in a bootstrap replicate. The metric of interest is then calculated using the bootstrap replicates as the validation set, resulting in a value for each bootstrap replicate. The metric is then reported as the mean of these values. We note that throughout this thesis we use 1000 bootstrap replicates and a confidence interval of 95 percent. If a metric is not reported with a confidence interval then it may be assumed that it was calculated using the original validation set. An overview of bagging aggregation is shown in Figure 3.3.

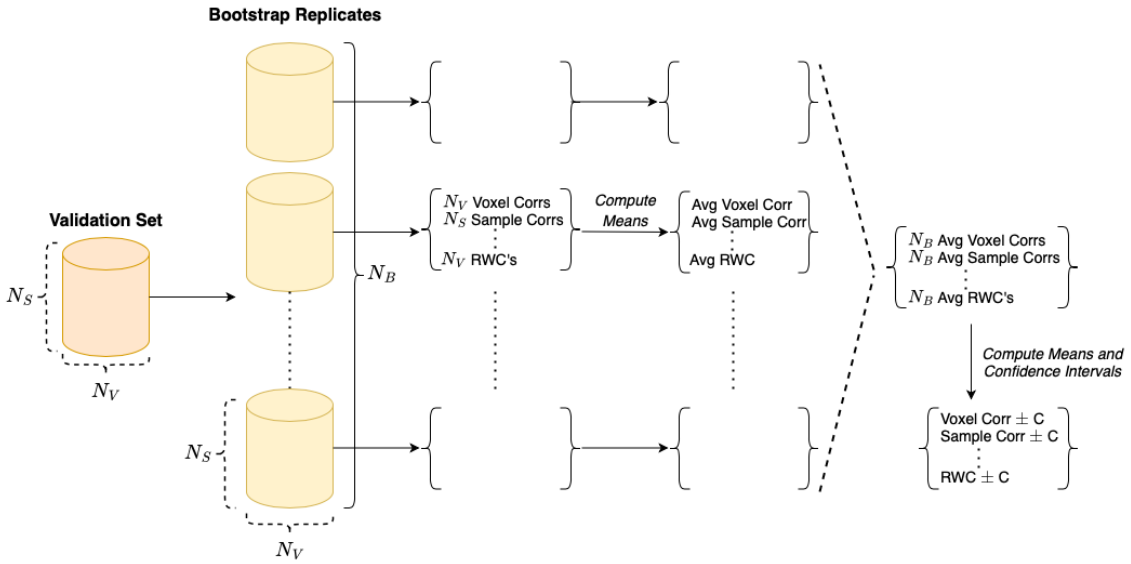


Figure 3.3: Diagram outlining the approach for calculating metrics using bagging aggregation. N_B is the number of bootstrap replicates, N_V is the number of voxels and N_S is the number of stimulus images in the validation set.

Chapter 4

Results

4.1 Encoding Naturalistic Images

To compare the performance and behavior of the capsule-based encoder to the CNN baseline encoder from Beliy et al. [30] we trained and validated both models using the Imagenet-fMRI dataset from Horikawa et al. [36]. Both models take a 112x112 resolution stimulus image as an input and predict a samplewise vector of voxel activations. Using the associated coordinates for each voxel, these output activations can be reconstructed into three-dimensional volumes representing predicted voxel activity in response to the input image. Figure 4.1 shows two examples of stimulus images, their ground truth BOLD activations and the associated predicted activations from both the capsule and CNN encoding models. Note that this figure shows only a single axial slice out of the three-dimensional ground truth and predicted fMRI volumes. The Beliy and Capsule encoders had samplewise correlations of 0.527/0.351 respectively for stimulus image (a) and 0.578/0.017 respectively for stimulus image (b). One can see that for the first stimulus image (4.1a), both models make similar predictions that resemble the fMRI pattern in the ground truth. Notably, however, the predictions for both encoders have a dampened magnitude. This is observed for all samples in

the training and validation sets. The second example (4.1e) shows an instance where the capsule encoder completely fails in comparison to the Beliy encoder. One can see from the ground truth (4.1f) that the stimulus image elicits significantly decreased BOLD activity in the posterior of the right hemisphere. The CNN encoder accurately predicts this decreased activity from the stimulus image (4.1g) whereas the capsule encoder does not (4.1h).

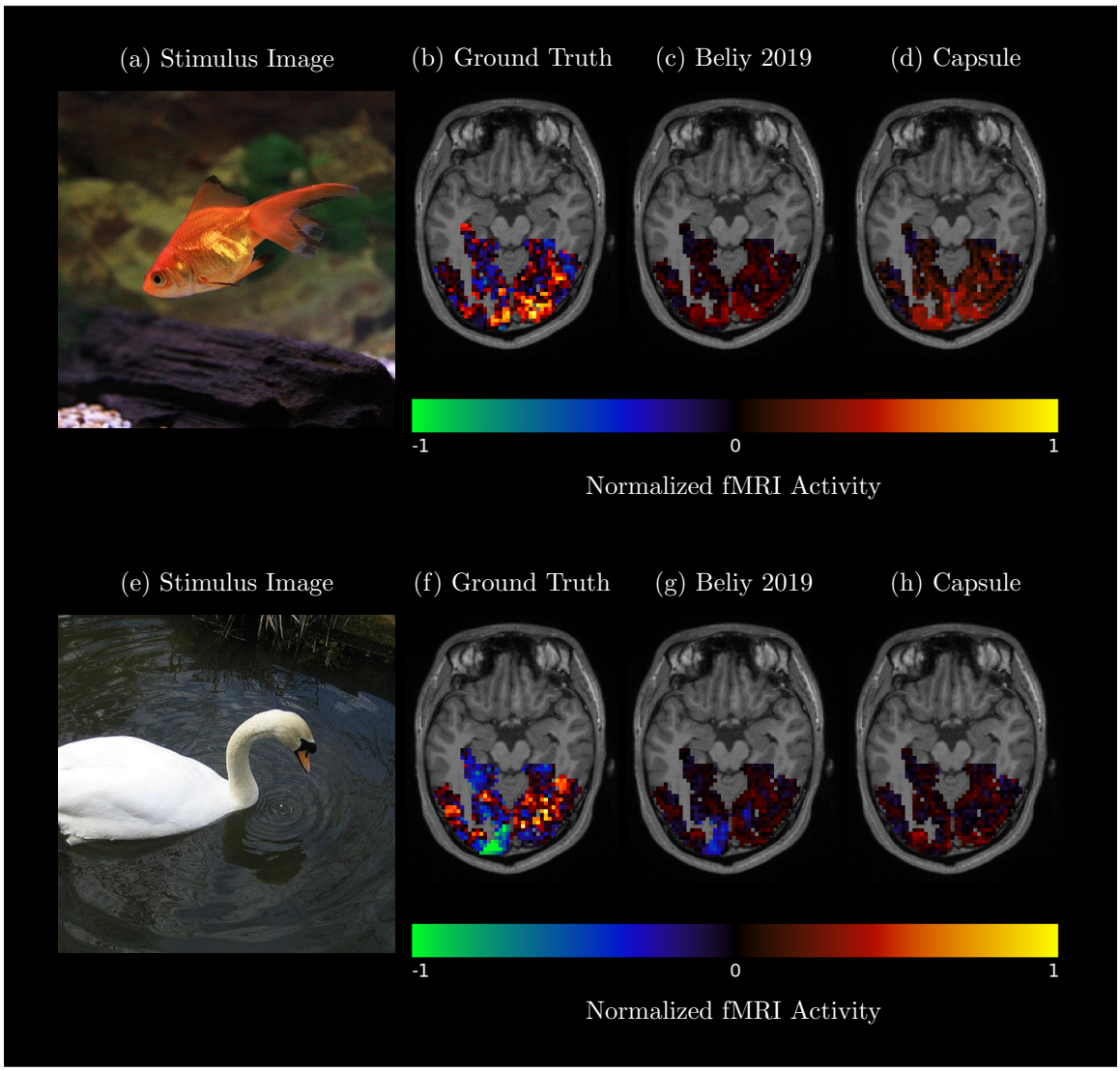


Figure 4.1: Predicted and ground truth fMRI activations for the Imagenet-fMRI dataset. The normalized fMRI activity is derived from the BOLD signal as discussed in Section 3.1. All volumetric slices are shown in radiological space.

Using 1000 bootstrap replicates derived from the validation set (see Section 3.4.3), various metrics were calculated to evaluate the performance of both encoders on the ImageNet-fMRI dataset. These can be seen in Table 4.1. We also include the results from a naive model which is described in Section 3.2. Although the capsule network does manage to outperform the naive model by learning some *meaningful* mappings between the image and fMRI space, it is significantly outperformed by the CNN-based encoder from Beliy et al. [30]. We also note that the response sensitivity for both models is relatively low.

Model	**MSE**	**Voxel Corr**	**Sample Corr**	**Voxel RS**
Naive	0.0888 ± 0.0003	0.001 ± 0.0001	0.160 ± 0.001	0.0379 ± 0.00001
Capsule	0.0825 ± 0.0002	0.089 ± 0.002	0.219 ± 0.001	0.0564 ± 0.0002
Beliy 2019 [30]	**0.0629** ± 0.0001	**0.292** ± 0.002	**0.499** ± 0.001	**0.0736** ± 0.0002

Table 4.1: Metric based results on the Imagenet-fMRI dataset for various different encoding models. The best values for each column are in bold. Confidence intervals are for 95% confidence.

4.2 Encoding Handwritten Digits

To determine the effect of image resolution and complexity on the encoding performance of our models we trained both encoders with the MNIST-fMRI dataset. We also trained a larger CNN-based model to account for the difference in the number of parameters between the Beliy 2019 and capsule encoders. Table 4.2 shows validation metrics for these encoders calculated using 1000 bootstrap replicates of the held-out validation set from the MNIST-fMRI dataset (see Section 3.4.3). One can see that the capsule encoder now significantly outperforms the encoder from Beliy et al. [30]. However, it seems to perform equally well as the CNN-based encoder with a similar number of network parameters.

Model	MSE	Voxel Corr	Sample Corr	Voxel RS
Naive	1.093 ± 0.003	-0.003 ± 0.0003	0.0001 ± 0.0006	0.036 ± 0.00004
Beliy 2019 [30]	1.046 ± 0.003	0.145 ± 0.002	0.216 ± 0.002	0.0621 ± 0.0004
Capsule	**1.011** ± 0.003	0.185 ± 0.001	0.245 ± 0.002	**0.213** ± 0.001
Large CNN	1.017 ± 0.003	**0.192** ± 0.002	**0.255** ± 0.002	0.200 ± 0.001

Table 4.2: Metric based results on the MNIST-fMRI dataset for various different encoding models. The best values for each column are in bold.

4.3 Characterizing Model Behavior

To explore whether negatively correlated voxels could be attributed to random noise we retrained the Beliy encoder using only the voxels which previously had negative voxelwise correlations. We did the same with the subset of remaining voxels that previously had positive correlations. Figure 4.2 compares the results from these experiments to the original encoder trained with all the voxels. One can see that there is no association between the voxelwise correlations of the original model and those of the model trained only on the negative subset. Instead, the voxelwise correlations of the negative subset model seem to be normally distributed around a mean of zero. The voxels in the positive subset however retain a similar voxelwise correlation to what was previously achieved by the model trained with all voxels.

To further analyze model behaviour and the robustness of its predictions for specific voxels, we introduced various new metrics, the first of which was voxel response sensitivity, which, unlike voxelwise correlation, takes into account the magnitude of predictions. More specifically it measures how much the predictions vary in response to different stimuli. Figure 4.3 compares the voxelwise correlations and voxel response sensitivity of the Beliy and capsule encoders trained on Imagenet-fMRI in anatomical space. One can see that both models are more effective at predicting voxels in the lower visual cortex (LVC), this can be seen in ROI 1 which has high voxelwise correlation and voxel response sensitivity. Notably, however, the voxelwise correlation and voxel response sensitivity do not always agree with one another. For example, in

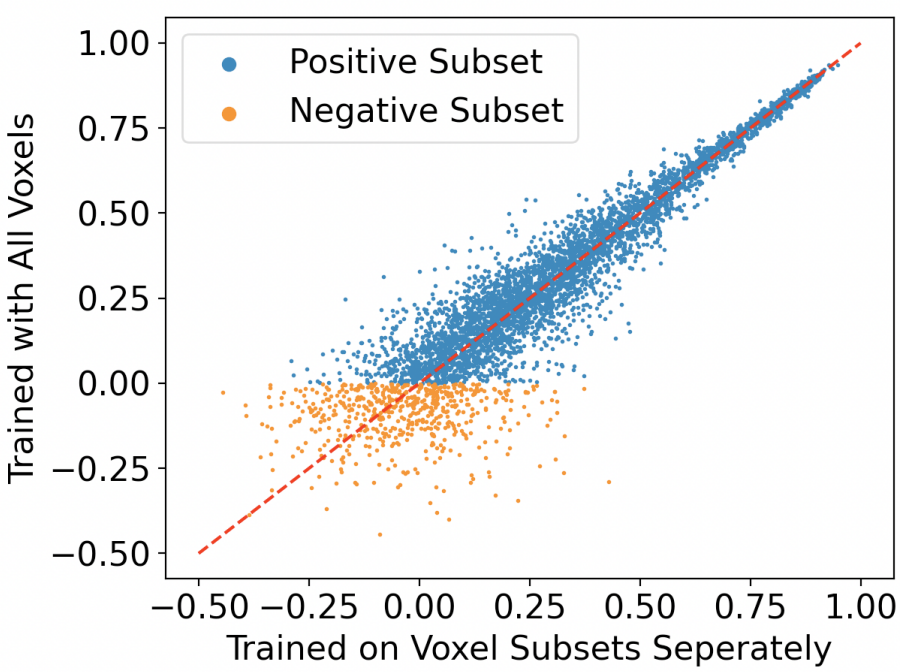


Figure 4.2: Comparison of Beliy 2019 encoder when trained with different subsets of voxels. The y-axis is the original voxelwise correlation when trained with all 4643 voxels. The x-axis is the voxelwise correlation achieved when the model is trained on the positive and negative voxel subsets separately. The positive (or negative) voxel subsets are the voxels for which the original model (trained with all voxels) achieves positive (or negative) voxelwise correlations. X=Y is denoted by the red dashed line

ROI 2 the Beliy encoder obtains relatively large voxelwise correlations (≈ 0.3 to 0.5) yet its response sensitivity for the voxels in this region is nearly zero.

To explore the relationship between voxelwise correlation and voxel response sensitivity across all voxels we display both metrics simultaneously using hexbin plots shown in Figures 4.4 and 4.5. One can see that negatively correlated voxels consistently have lower response sensitivity. This is reflective of the naive model which has a low response sensitivity for all voxels since it predicts nearly the same signal regardless of the input image. However, voxels that are positively correlated by the other encoders tend to have increasing response sensitivity as the magnitude of their correlation increases. Figure 4.4 shows the voxelwise correlation versus voxel response sensitivity using hexbin plots for the models trained on the Imagenet-fMRI dataset.

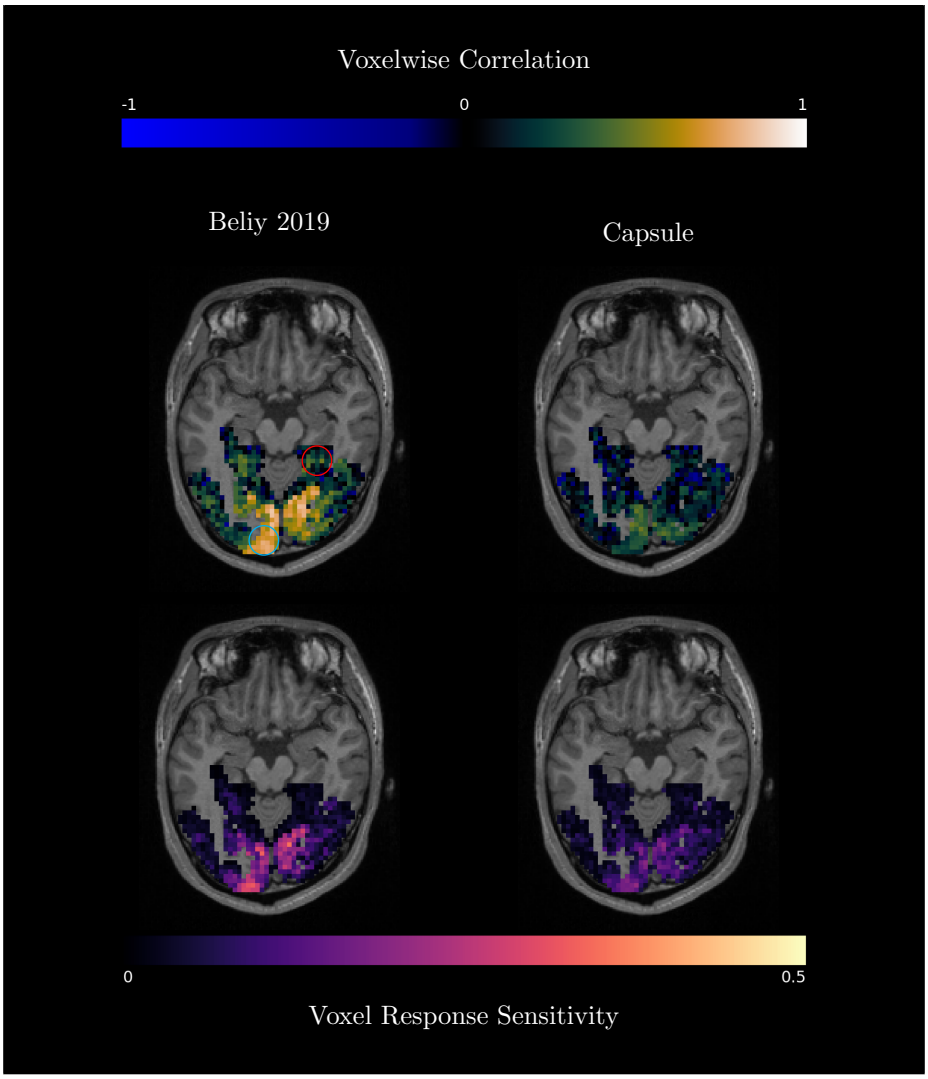
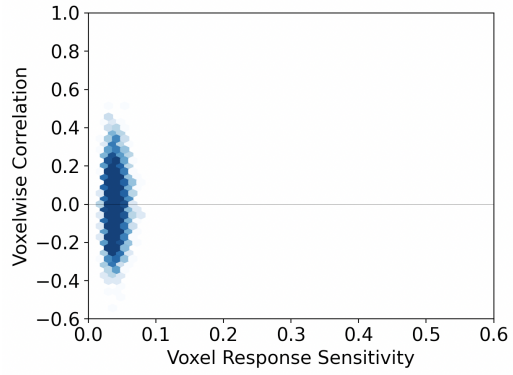


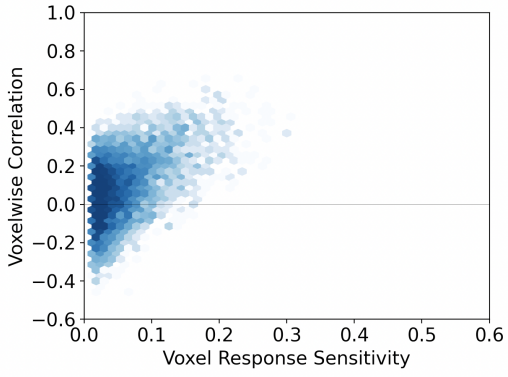
Figure 4.3: Comparison of voxelwise correlations and voxel response sensitivity for both the Beliy 2019 and Capsule encoders on the Imagenet-fMRI dataset. Volumetric slices are shown in radiological space. ROI 1 is encircled in cyan and ROI 2 is encircled in red.

One can see that the capsule network does in fact go beyond the naive approach by varying its predictions in response to the input stimuli, indicated by the larger response sensitivities primarily for voxels which are positively correlated. However, it is outperformed by the CNN-based encoder from Beliy 2019. We observe the same association between voxelwise correlation and voxel response sensitivity for the encoders trained on the MNIST-fMRI dataset as seen in Figure 4.5. In this figure, we

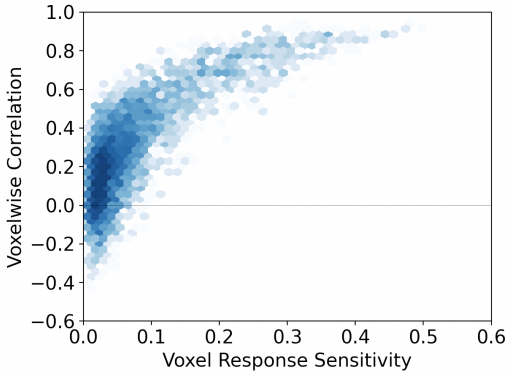
see that the encoder from Beliy et al. is outperformed by the larger capsule and CNN encoders. Notably, both of the larger models exhibit an increased overall response sensitivity, even for the negatively correlated voxels, when compared to the naive model and the Beliy encoder. However, the increased voxelwise correlation and voxel response sensitivity of the positively correlated voxels still differentiate themselves from those which are negatively correlated. Additionally, although the capsule encoder and large CNN encoder seemed to perform similarly according to the metrics from Table 4.2, one can see from Figure 4.5 that the capsule encoder exhibits a stronger association between voxelwise correlation and voxel response sensitivity than its CNN counterpart.



(a) Naive (Mean + Noise)



(b) Capsule



(c) Beliy 2019

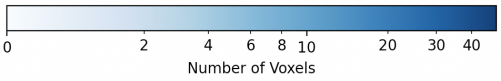
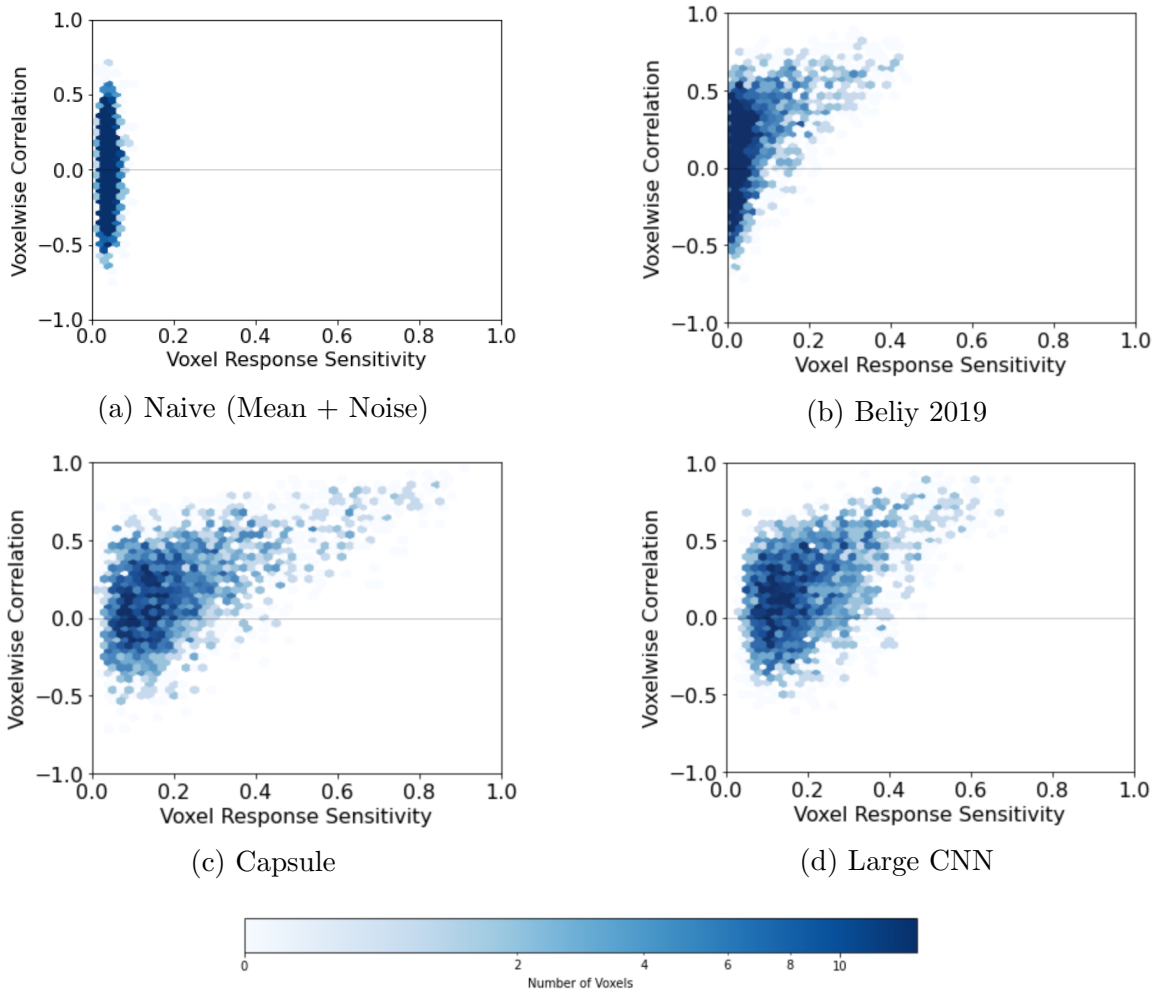


Figure 4.4: Hexbin plots of voxelwise correlation versus voxel response sensitivity for models trained on Imagenet-fMRI



(a) Naive (Mean + Noise)

(b) Beliy 2019

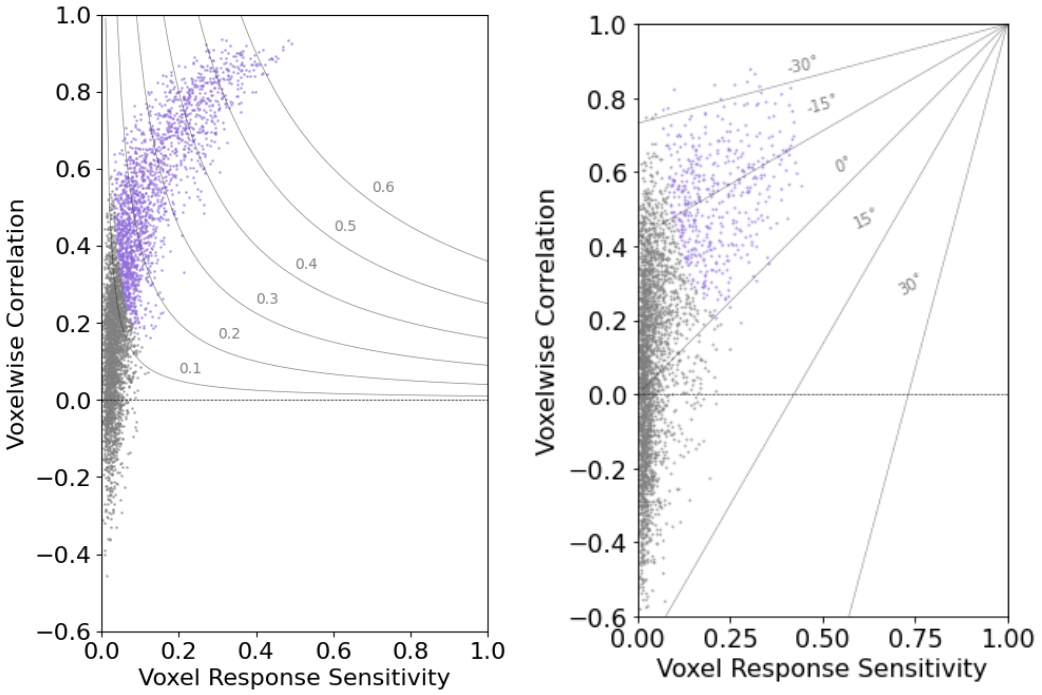
(c) Capsule

(d) Large CNN

Figure 4.5: Hexbin plots of voxelwise correlation versus voxelwise response sensitivity for models trained on MNIST-fMRI Dataset

P-value thresholding is a common approach for seperating voxels into significant and insignificant clusters using the magnitude of their voxelwise correlation. In this work, we sought to improve upon this method to obtain a better differentiation of which voxels were likely to have been predicted robustly by the encoder and which weren't. Firstly, we introduced a new metric that took into account both the magnitude and pattern of predictions which we refer to as response weighted correlation (RWC), see Section 3.4.2. Our approach separated the voxels for each model into two clusters using a threshold RWC value. The threshold RWC was determined by taking the voxel with the most negative RWC and using the absolute value of its RWC as the threshold. This thresholding approach was chosen in order to be robust to differences between models and datasets. For example, one can see from the naive models in Figures 4.4 and 4.5 that the MNIST-fMRI dataset results in a larger variance along the voxelwise correlation dimension. Similarly, the capsule encoder and large CNN seen in Figure 4.5 exhibit larger voxel response sensitivity across all voxels which could skew the results if not taken into account. We note that we did not observe any voxels with RWC values that were extreme negative outliers, however, this could be addressed on a case-by-case basis in future work. Figure 4.6 shows the separation of voxels after RWC thresholding for the Beliy model on both datasets in addition to visualizing how RWC and RWC skew changes across the 2D plane of voxel response sensitivity versus voxelwise correlation using various contours. One can see that our thresholding approach is robust to the changes in voxelwise correlation and voxel response sensitivity caused by the different datasets.

To compare the effectiveness of RWC thresholding to the current approach of p-value thresholding, we categorized the voxels in Figure 4.2. This can be seen in Figure 4.7. For the p-value thresholding, we used a voxelwise correlation threshold corresponding to a significance of 0.1% ($p = 0.001$), voxels above the threshold would therefore be deemed significant ($p < 0.001$). This p-value is low enough to ensure no



(a) Imagenet-fMRI

(b) MNIST-fMRI

Figure 4.6: Scatter plots of voxel response sensitivity versus voxelwise correlation for the Beliy 2019 encoder on both datasets. Purple points represent voxels above the RWC threshold and gray points represent voxels below the threshold. The RWC threshold values of the Beliy encoder were 0.123 and 0.209 for the ImageNet-fMRI and MNIST-fMRI datasets respectively. RWC is visualized using the contours in plot 4.6a and RWC skew is visualized using the contours in plot 4.6b

voxels with negative correlations were above the significance threshold, as in St Yves et al. [13]. However, one can see that in order to achieve this, p-value thresholding must unnecessarily exclude well-predicted voxels which were generally robust and retained a similar voxelwise correlation for both models. RWC thresholding, however, manages to include many of these voxels and separate them from those that are less robust. Notably, all the voxels which are above the RWC threshold but below the p-value threshold (orange points in Figure 4.7) had positive voxelwise correlations for both models. We also note that the RWC threshold omits a small number of voxels that are above the p-value threshold (red points in Figure 4.7)

Tables 4.4 and 4.3 show various metrics that summarize the results from Figures

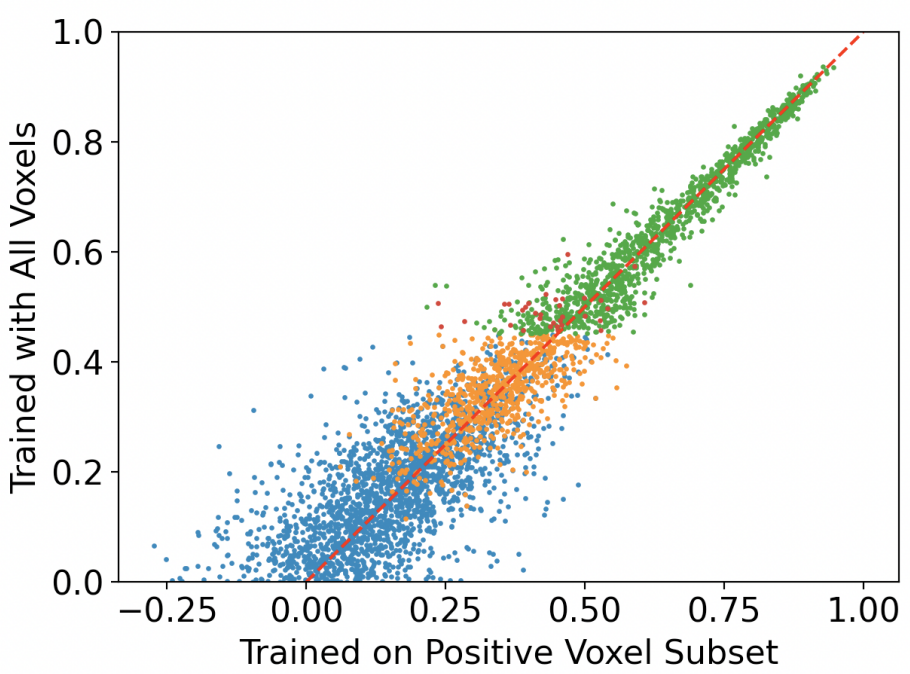


Figure 4.7: Comparison of RWC vs P-Value Thresholding for Beliy 2019 encoder on ImagenetfMRI. The y-axis is the voxelwise correlation when trained with all 4643 voxels. The x-axis is the voxelwise correlation achieved when the model is trained using the positive voxel subset. The RWC and p-value threshold values were both calculated using the predictions from the model trained with all voxels. Blue points are voxels that were below both thresholds. Green points are voxels that were above both thresholds. Orange points are voxels above the RWC threshold but below the p-value threshold. Red points are voxels below the RWC threshold but above the p-value threshold. X=Y is denoted by the red dashed line

4.4 and 4.5 using our RWC thresholding approach. The RWC skew for each model was calculated by averaging the RWC skews of only the voxels above their RWC threshold. This subset of voxels, which we refer to as the *intentional* set, is different for each model. The model's above threshold percentage is the percentage of voxels in the intentional set. The rest of the metrics in the table (Voxel RWC, Voxel Corr, and Voxel RS) are averages across the set of voxels formed by taking the union of all the intentional sets for each model in the table. Therefore, voxels which all the models predicted below their respective RWC thresholds we're the only voxels not used when calculating these metrics. This approach was taken to ensure a model was not disadvantaged for having a higher above-threshold percentage. Additionally, we

calculated RWC, RWC Skew, Voxel Corr and Voxel RS using 1000 boostrap replicates of the validation set as discussed in Section 3.4.3. Note that the set of voxels on which the metrics were calculated was determined using the original validation set and remained the same for each bootstrap replicate. By comparing the metrics in these tables to the more detailed visualizations in Figures 4.4 and 4.5, one can see that the RWC threshold value itself gives information about the size of the central cluster of voxels. Additionally, it is now clear from these metrics that the MNIST capsule encoder outperforms the large CNN encoder. In this case, primarily due to an increased voxel response sensitivity. This is also made clear by the fact that the capsule encoder's RWC skew is significantly closer to zero indicating a more even balance of voxelwise correlation and response sensitivity. We also did a one-tailed wilcoxon signed rank test of the RWC values obtained by both the Capsule and Large CNN encoders on the 1000 boostrap replicates of the validation set. The results confirmed that the Voxel RWC of the capsule encoder was greater than that of the Large CNN with a significance level of $p = 6.62 \cdot 10^{-118}$.

Model	**RWC Thresh**	**Above Thresh (%)**	**Voxel RWC**	**RWC Skew**	**Voxel Corr**	**Voxel RS**
Naive	0.155	0.0	0.001 ± 0.0001	N/A	0.004 ± 0.0002	0.038 ± 0.00002
Beliy 2019	0.123	**42.2**	**0.258** ± 0.001	-18.0° ± 0.1°	**0.530** ± 0.002	**0.134** ± 0.0004
Capsule	0.177	9.5	0.099 ± 0.001	-6.3° ± 0.2°	0.163 ± 0.002	0.102 ± 0.0003

Table 4.3: Various metrics for the Imagenet-fMRI dataset characterizing the plots from Figure 4.4. The best values for each column are bolded

Model	**RWC Thresh**	**Above Thresh (%)**	**Voxel RWC**	**RWC Skew**	**Voxel Corr**	**Voxel RS**
Naive	0.194	0.0	0.002 ± 0.0002	N/A	0.006 ± 0.0004	0.035 ± 0.00004
Beliy 2019	0.209	15.6	0.276 ± 0.002	-15.2° ± 0.1°	0.486 ± 0.003	0.180 ± 0.001
Capsule	0.377	**16.3**	**0.475** ± 0.002	-7.7° ± 0.1°	**0.546** ± 0.002	**0.449** ± 0.002
Large CNN	0.388	13.4	0.425 ± 0.002	-13.5° ± 0.2°	0.545 ± 0.002	0.364 ± 0.002

Table 4.4: Various metrics for the MNIST-fMRI dataset characterizing the plots from Figure 4.5. The best values for each column are bolded.

Lastly, we decided to explore how the SNR of our validation set affects model

evaluation and whether our RWC thresholding approach would be robust to changes in SNR. Figure 4.8 shows how the SNR of the validation set affects the voxelwise correlation of the Beliy model trained on the Imagenet-fMRI dataset. We use the original validation data which was averaged over 35 repeats for each image (see Section 3.1) as the high SNR validation set. For the low-SNR validation set, we take a single repeat for each image instead of averaging. This low-SNR validation set is the same SNR as the training data set since the training samples also only had a single repeat. Both the low and high SNR validation sets have the same number of samples. One can see that voxels with negative correlations on the high-SNR validation set were not robust to the change in SNR. Instead, they form part of a central cluster that is approximately normally distributed around the center of the graph (0, 0). For the positively correlated voxels, however, we see a significant increase in voxelwise correlation when evaluated using the higher SNR measurements. We also note that the voxels above our RWC threshold but below the typical p-value threshold were all robust to the change in SNR, with none of them being negatively correlated with either validation set. Importantly, the thresholds were calculated using the voxelwise correlations with the low-SNR validation set.

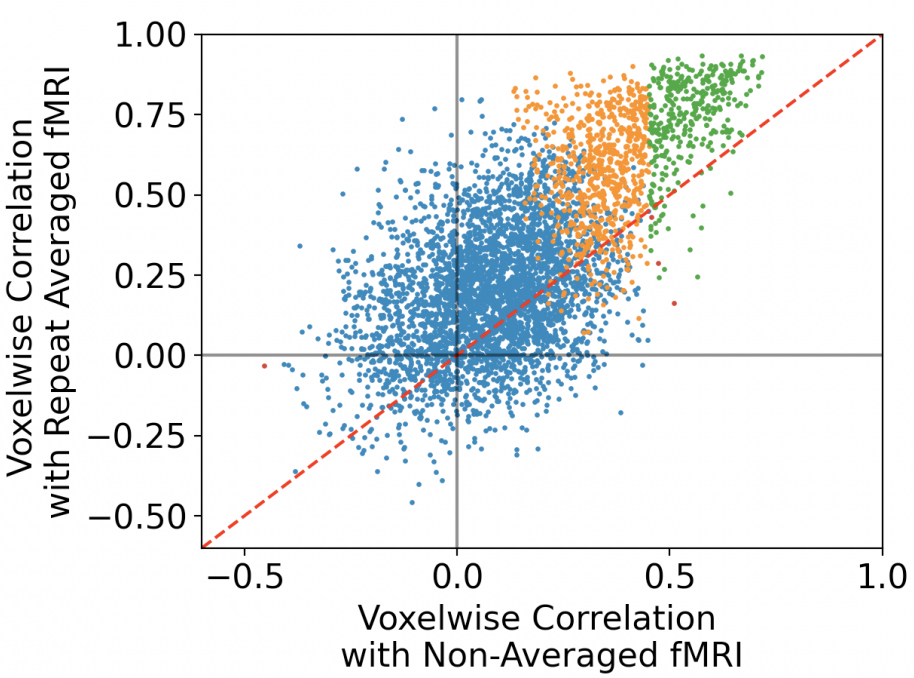


Figure 4.8: Scatter plot showing how the SNR of the validation set affects the voxelwise correlation of the Beliy 2019 model trained on the Imagenet-fMRI dataset. The y-axis uses the original high-SNR validation set, obtained by averaging 35 repeated fMRI measurements for each stimulus image, to compute the voxelwise correlations for each voxel. The x-axis uses only a single repeat for each image as the ground truth to compute voxelwise correlation (low-SNR validation set). RWC and significance ($p < 0.001$) threshold values were both calculated using the low-SNR validation set. Blue points are voxels which were below both thresholds. Green points are voxels which were above both thresholds. Orange points are voxels above the RWC threshold but below the p-value threshold. Red points are voxels below the RWC threshold but above the p-value threshold. X=Y is denoted by the red dashed line.

Chapter 5

Discussion

In this work we include several findings. Firstly, we show that capsule networks are more effective at encoding images into fMRI activity than state-of-the-art CNN-based encoders when the images are simple and low-resolution, yet less effective when the images are complex, naturalistic and higher-resolution. We then show that the robustness of an encoder's predictions for a particular voxel can be determined based on the encoder's predictive behavior for that voxel, which we categorize as being *intentional* or *unintentional*, terms defined in this work (see Section 5.2 for more detail). The characteristics of these two types of predictive behaviors are revealed using the various novel metrics which we introduce. These metrics are used to categorize voxels as *intentional* or *unintentional* more effectively than p-value thresholding in addition to helping better evaluate and compare the capsule and CNN based encoders. Lastly, we introduce our *Partial Averaging* hypothesis to explain the increase in performance associated with evaluating one's encoder using higher-SNR validation data.

5.1 Capsule Networks as Encoding Models

In this work, we introduced a novel encoding model using the recently proposed capsule based approach from Sabour et al. [26]. We showed, using the Imagenet-fMRI

dataset, that although capsule networks may share functional similarities with biological vision, they are more difficult to train than existing CNN-based encoders when it comes to small datasets with large complex naturalistic images (See Table 4.1). One can see that the existing CNN-based state-of-the-art encoder from Beliy et al. [30] significantly outperforms the capsule encoder. Figure A.3 shows that this is true across various regions or interest (ROI's) of the visual pathway. However, we noted that this experiment changed two major variables in comparison to MNIST [26] or smallNORB [27] classification, for which capsule networks achieve state-of-the-art over CNNs. These are the type of task (fMRI-encoding versus classification) and the types of images (complex, naturalistic and high-resolution versus simple and low-resolution). To determine which variable was negatively affecting capsule network performance, we trained both capsule and CNN based encoders on the MNIST-fMRI dataset, which uses simple low-resolution images for fMRI encoding. Logically, if capsule networks performed well on this task, it would indicate that the types of images are what resulted in poor performance in comparison to CNNs on the Imagenet-fMRI dataset. Using existing metrics and visualizations from works such as St. Yves et al. [13], Gaziv et al. [19], etc., we showed that our capsule encoder is comparable in performance to a similarly sized CNN model based on the state-of-the-art CNN encoder from Beliy et al. [30] (See Table 4.2 and Figures A.1 and A.2) for encoding MNIST images into fMRI brain activity. Furthermore, after evaluating the models using novel metrics (See Section 5.4) that take into account the magnitude of predictions, we showed that the capsule encoder in fact significantly outperformed the CNN based encoders on the MNIST-fMRI dataset (See Table 4.4). This demonstrates that capsule-based networks can in fact effectively map visual stimuli to BOLD activity, highlighting the image complexity as their limitation on the Imagenet-fMRI dataset. We would therefore predict similar capsule based networks to perform poorly on the classification of large complex images in comparison to state-of-the-art CNN

based models. This limitation has since been acknowledged by Sabour et al. in a more recent work that attempts to partially address it [49]. Although this limitation may be intrinsic to capsule networks themselves, it could also be a manifestation of computational constraints. The functional properties of capsule networks that make them extremely effective for simple low resolution images do not scale efficiently with respect to network parameters and memory consumption as the task and images become more complex. This limitation will need to be addressed in order for capsule networks to be useful for tasks that go beyond simple visual inputs.

In our experiments with the MNIST-fMRI dataset, we note that the CNN baseline from Beliy et al. [30] performed poorly compared to the Large-CNN despite performing well on the Imagenet-fMRI dataset. We attribute this to the recently proposed *double descent* phenomenon [50][51] and hypothesize that the simpler nature of the MNIST-fMRI dataset allows the encoding models to operate in what is referred to as the *modern regime*. In the modern regime, largely overparametrized models well beyond the interpolation threshold, such as the Large-CNN and Capsule encoders, generalize better. However, the Imagenet-fMRI dataset is significantly more complex, and therefore the encoders are likely not sufficiently overparametrized in order to operate in this modern regime. Instead, they operate in the *classical regime* where underparametrized models underfit the data and overparametrized models overfit the data. In support of this hypothesis, we note that the Large-CNN performed poorly on the Imagenet-fMRI dataset despite having more parameters than the beliy encoder. Since the number of parameters of each model was different for each dataset due to differences in image resolution and number of voxels, we outline them in Table A.5. One can see that the capsule encoder has a larger number of parameters than the beliy encoder. We note that in our experiments with different capsule encoder architectures (See Section A.1), those with the same number of parameters as the beliy encoder did not perform significantly better than the one used in this work. Lastly,

although we use the encoder from Beliy et al. as our state-of-the-art baseline, we note that the same authors have since expanded upon this model in a subsequent work (Gaziv et al. [19]) which combines aspects of their previous encoder with the feature-weighted receptive field model of St. Yves et al. We assume this to be the current state-of-the-art since it builds upon the work of St. Yves et al and outperforms the encoder from Beliy et al. However, the models from St. Yves et al and Gaziv et al. have never been directly compared.

5.2 Characterizing Encoder Behaviour

Whilst experimenting with the various models it became evident that existing methods for evaluating and comparing fMRI-image encoders were insufficient. Voxelwise correlation is the only metric used in most existing work (See Section 3.4.2) including the state-of-the-art methods from Gaziv et al. [19] and St. Yves et al.[13]. Yet it does little to characterize model behaviour and is easily dampened by poorly predicted voxels making it difficult to compare models. Furthermore, due to the large number of voxels, the individual voxelwise correlations provide little additional insight beyond relative predictability between coarse anatomical regions (eg. V1, V2, V3, etc.) often displayed using cortical surface visualizations (see Figure A.1) such as in Gaziv et al. [19], Guclu et al. [10], Wen et al. [8] and Allen et al. [14]. One aspect of model behaviour that is often overlooked in existing work is the high number of voxels which are, with a large magnitude for many, negatively correlated. Many works such as Wen et al. [8], St Yves et al. [13], Eickenberg et al. [9], and Guclu et al. [10][33] use a threshold, often derived from extremely small significance thresholds (0.1% down to 0.000005%), to omit these voxels. However, none of the results prior to thresholding nor the percentage of voxels above this significance threshold are reported. This obscures the presence of these negatively correlated voxels and the true model behaviour.

The state-of-the-art work from Gaziv et al. also omits negatively correlated voxels but does not address how they are removed. Their presence is only revealed in a supplementary figure comparing the voxelwise correlation of voxels to their estimated SNR. In our experiments, we observed that despite being negatively correlated with the validation set, these voxels are positively correlated with the training set (See Figure A.7). This indicates that the model has overfit these voxels, likely to the noise in the fMRI data. This overfitting is not reflected in the validation loss, however, since the loss is a combination of error signals generated by all voxels, only a subset of which were overfit by the model. Notably, one can see from Figure A.7 that these negatively correlated voxels are not differentiable from those with significant positive voxelwise correlations, bringing into question the integrity of voxelwise correlation as a metric. Therefore, we sought to test the robustness of voxelwise correlation and whether the encoders were primarily learning to correlate with the true BOLD signal or just simply the noise. We did this by separating the negatively correlated voxels from those that are positively correlated, and training encoding models on these subsets separately. Our results showed (see Figure 4.2) that for the negatively correlated voxels, voxelwise correlation was not a robust metric. Their new voxelwise correlations formed an approximately normal distribution centered on zero indicating the model had not learned anything *meaningful* about the signal of these voxels. Voxels with positive correlations however were, on average, significantly more robust, closely following the X=Y line of Figure 4.2. One can extrapolate the statistical behaviour of the negatively correlated voxels and conclude that they are part of a subset of voxels for which the model was unable to learn anything *meaningful*. The rest of this subset would have randomly achieved positive voxelwise correlations for the original model and hence be part of the positive voxel subset. This explains why voxels with small positive correlations (on the y-axis) were less likely to be robust. One can therefore think of the encoding models as having two superimposed distributions in the voxel-

wise correlation dimension. The first, which we define as *unintentional*, is a random normal distribution with a mean of zero consisting of poorly predicted voxels that are overfit on noise in the training data and randomly correlated with noise in the validation data. The second, which we define as *intentional*, is a non-random distribution with a positive mean consisting of voxels for which the model has, to some extent, accurately mapped. We will refer to the voxels that form these distributions as the *unintentional* and *intentional* voxels respectively.

5.3 Novel Methods Describing Encoder Behaviour

Naturally, it is of interest to characterize the intentional and unintentional distributions and to subsequently separate them. Due to the significant overlap between these distributions, however, this is difficult using only a single dimension (voxelwise correlation). We therefore identified voxel response sensitivity as a metric that could act as a meaningful additional dimension. We show in Figure 4.3 that voxel response sensitivity can be used to discriminate between voxels that are positively correlated intentionally (ROI 1) and unintentionally (ROI 2). This is because, as seen in Figures 4.4 and 4.5, negatively correlated voxels consistently have lower voxel response sensitivity. We can therefore extrapolate that all unintentional voxels have similarly low response sensitivity. We show this explicitly using the naive model, which predicts all voxels in an *unintentional manner*. Further comparison shows that the encoders behave similarly to the naive model when predicting unintentional voxels. This is because predicting the mean voxel activation from the training set reduces MSE which is a component of the loss function. If the encoders are unable to predict the BOLD signal, then they resort to this naive approach in order to reduce the loss and then, using very small magnitude variations, overfit to the noise in the data. This overfitting is likely driven by the cosine similarity term of the loss function which,

similar to correlation, does not take into account the magnitude of predictions. Figure A.6 shows further evidence of voxel response sensitivity being predictive of voxel robustness. We can also observe other interesting properties of model behaviour by using both voxelwise correlation and voxel response sensitivity simultaneously (see Figures 4.4 and 4.5). For example, one can see that the unintentional distribution has a larger variance along the voxelwise correlation dimension for the MNIST-fMRI dataset. This can be explained by the fact that the MNIST-fMRI dataset has fewer validation samples and therefore the voxelwise vectors are shorter than those of the Imagenet-fMRI dataset. An increased number of samples reduces the probability that the predicted and ground truth voxelwise vectors will correlate strongly with one another by pure chance, therefore, decreasing the variance of voxelwise correlation for a model's unintentional distribution. Additionally, for the MNIST-fMRI dataset, we also observe that the larger models (Capsule and Large CNN) exhibit an increased response sensitivity, even for the negatively correlated voxels. It is possible that this is due to the larger overfitting capability of the higher parameter models.

In order to condense the information present in Figures 4.4 and 4.5 as well as more effectively separate the unintentional and intentional voxel distributions, we defined a new metric. This was done by combining voxelwise correlation and voxel response sensitivity into a single metric which we refer to as response weighted correlation (RWC). Figure 4.6 shows that our RWC thresholding approach can separate voxels into intentional and unintentional clusters intuitively while remaining robust to changes in the unintentional distribution. It does this by using the negative voxelwise correlations, which we assume to be unintentional, and using them to extrapolate the behaviour of unintentional voxels with positive voxelwise correlations. We also accompany RWC with the RWC Skew metric, which characterizes model behaviour by quantifying the model's *skew* towards either voxelwise correlation or voxel response sensitivity. The only other approach used to separate what we refer to as the inten-

tional and unintentional clusters is to threshold voxelwise correlations using a p-value significance threshold. This is the approach used by many existing works such as Wen et al. [8], St Yves et al. [13], Eickenberg et al. [9], Guclu et al. [10][33] and Naselaris et al. [32]. The primary limitation of this approach is that one must use an extremely high significance threshold to ensure all the negatively correlated voxels are classified as insignificant as seen in Figure A.5. Even with a p-value of 0.001, which is used by Wen et al. [8] and St Yves et al. [13], we would still expect a few voxels to surpass this threshold purely by chance due to the large number of total voxels in the Imagenet-fMRI dataset. Furthermore, a high significance threshold is likely to disregard many intentional voxels as insignificant. Figures 4.7 and 4.8 show how our RWC thresholding approach compares to p-value thresholding. One can see that RWC thresholding includes significantly more voxels in the intentional cluster (above the threshold), all of which were relatively robust and maintained a similar voxelwise correlation despite changes to the number of voxels during training or SNR of the validation set respectively. Furthermore, it achieves this without letting any non-robust voxels through. In statistical terms, the results show that our approach increases the number of true positives without increasing the number of false positives.

To summarize, we introduced a new metric called voxel response sensitivity, which takes into account the magnitude of predictions, and showed that it provides additional information not present in voxelwise correlation. We then combined these metrics into new metrics called response weighted correlation (RWC) and RWC skew, which we showed can be used to identify intentional and unintention voxels more effectively than traditional p-value significance thresholding.

5.4 Evaluating Encoders with Improved Methods

The RWC, RWC skew, and voxel response sensitivity metrics, as well as our improved thresholding approach for separating intentional and unintentional voxels, allow us to better evaluate, compare and characterize encoding models. We demonstrate this using improved visualizations as seen in Figures 4.4 and 4.5 as well as by calculating a new set of scalar metrics. The new metrics refer to those shown in Tables 4.3 and 4.4 which were computed using only voxels from the union of intentional distributions as estimated using our RWC thresholding approach. We note that when comparing encoders using scalar metrics, Voxel RWC should be used as the primary performance metric with Above Thresh % and RWC Thresh being important characterizing metrics that describe the size and shape of the unintentional distribution respectively. RWC Skew, Voxel Corr, and Voxel RS can be used for further characterization when seeking more detail. We separate the latter metrics from the former in Tables 4.3 and 4.4 to clarify this. For the Imagenet-fMRI dataset, one can see that our approach also shows that the Beliy encoder outperforms the capsule encoder. However, now we can see that it achieves this primarily by being able to *intentionally* predict a larger percentage of voxels as reflected by the significantly higher above threshold percentage. This is not evident using existing approaches such as calculating the average voxelwise correlation (see Table 4.1) or visualizing them on the cortical surface, which were the approaches taken by Allen et al. [14], Gaziv et al. [19], Wen et al. [8], Guclu et al. [10][33] and Naselaris et al. [32]. St. Yves et al. [13] compared models using what they called *accuracy-advantage plots* (by accuracy they are referring to voxelwise correlation) which we show for our models in Figure A.2. One can see that using this approach, it is not even clear that the capsule encoder has learned anything meaningful at all. However, the results from Table 4.3 and Figure 4.4b show that this is not the case. Our evaluation is also able to better characterize and compare the encoders trained on the MNIST-fMRI dataset. Namely, we see that all three models have similar

above threshold percentages and therefore intentionally predict similar amounts of voxels. However, the Capsule and Large CNN encoders are able to more closely map these voxels to the ground truth. This is shown primarily by their increased Voxel RWC. This metric also shows that the capsule encoder significantly outperforms the large CNN encoder, which is not clear using existing approaches such as the metrics shown in Table 4.2 or the accuracy-advantage plot from St. Yves et al. [13] shown in Figure A.2b. By looking at the other characterizing metrics we see that the capsule encoder achieves this primarily by predicting intentional voxels with a higher response sensitivity. Additionally, the RWC threshold value itself reflects the visual observation from Figure 4.5 that the larger models have unintentional distributions with more variance along the voxel response sensitivity dimension. Despite this, the capsule network is still able to achieve the highest above-threshold percentage indicating that it intentionally predicts the largest number of voxels. Lastly, we note that Tables 4.1 and 4.2 show that voxels which are unintentionally predicted *damp* evaluation metrics by reducing the metric averages across all voxels. By omitting many of these unintentional voxels, the differences between the encoders become clearer making them easier to characterize and compare which is reflected in Tables 4.3 and 4.4.

5.5 Characterizing Effects of Validation Set SNR

We also decided to explore how the SNR of the validation set used for evaluation would affect the performance of an encoder. Interestingly, our results from Figure 4.8 showed that many of the voxels correlated more strongly with the higher SNR validation set. We note that this is primarily the case for voxels that were positively correlated. Voxels that were negatively correlated did not exhibit the same increase in voxelwise correlation when evaluated with higher SNR data. We note that there are some outliers that were negatively correlated with the low-SNR validation set,

but strongly positively correlated with the high-SNR validation set as seen in the upper left quadrant of Figure 4.8. Given that there are no corresponding outliers in the lower right or lower left quadrants, we believe the model did in fact learn meaningful mappings for these voxels, but there was just too much noise in the low-SNR validation set to accurately deduce this. We also show the results for this experiment for each of the ROI's provided in the original dataset seperately in Figure A.9; one can see that voxels in the early/lower visual cortex (V1, V2, V3, LVC) were more strongly affected by the increased SNR of the validation set. The fact that the positively correlated voxels consistently obtain a higher voxelwise correlation when evaluated with the higher SNR validation set indicates that the model has predicted the BOLD signal with a higher voxelwise correlation than the low-SNR validation set is capable of accurately measuring. Importantly, the model is trained with data that is the same SNR as the low-SNR validation set and all stimulus images in the training set are unique. A common assumption in existing work is that, given a training set consisting of samples with unique visual stimuli, an encoder cannot surpass the consistency between repeated measurements in response to the same stimulus. This limit is often referred to as the noise ceiling, and is estimated using various different approaches as outlined by Lage-Castellanos et al. [52]. However, our results indicate that for many voxels, noise in the validation set was causing their voxelwise correlation to be consistently underestimated, suggesting that it is the validation data that imposes a ceiling due to noise, not the training data. We provide further evidence of this in Figure A.8 which shows that the predicted BOLD response of many voxels correlates more strongly with the measured BOLD response than repeated measurements do with each other. We propose that this is a result of what we refer to as *partial averaging*. It has been shown that averaging repeated fMRI measurements in response to the same stimulus image can increase the SNR of the BOLD signal and consequentially increase the estimated noise ceiling. This

is explicitly derived by Allen et al. [14] in the natural scenes dataset manual. If a supervised encoding model were to be given a training set that contained these repeated trials as separate samples, then presumably this could indirectly increase the noise ceiling in a similar fashion. One could say that the supervised model has effectively learned to do the trial averaging on its own. Partial averaging takes this one step further by proposing that unique visual stimuli from the training set are not entirely independent. Different images may share a subset of their features with one another in feature space, and it is these features that the model uses to predict voxel activity. Therefore, the model may treat these images as repeats of one another when predicting voxels that attend only to the shared features. For example, suppose there is a voxel that activates only in the presence of faces. If there are multiple distinct training images with faces in them, then from the perspective of this voxel, those images are effectively repeated trials of one another. The network can then average out the noise in the voxel's response to those images and obtain an estimate that has a higher SNR than a single non-averaged *ground truth* measurement. Therefore, the model has learned to predict the BOLD activity for this voxel more accurately than it was measured for a single stimulus. The voxelwise correlation would still be limited by the SNR of the validation data. However, if we obtained higher SNR validation data (effectively raising its *noise ceiling*), it would correlate more strongly for this voxel. This is what is observed in Figure 4.8. Although this is a simplified binary example, one can see how the lack of independence between training samples could allow a supervised network to denoise its predictions. What this means is that noise ceilings should be interpreted as a limitation of our evaluation methods, not as a measure of maximum achievable model performance given a certain training dataset. Works such as Gaziv et al. [19] and Allen et al. [14] show that their encoder mostly saturates the noise ceiling estimate and therefore deduce that there is little room for model improvement given the training set that was used. However, in reality, this may

not be the case. Not only could the model's performance have been better than what was estimated using the validation set, but the maximum achievable performance for the model could be above the noise ceiling estimate, which assumes a model can not surpass consistency between trials. This demonstrates the importance of using high-quality validation data to evaluate one's model, otherwise, you impose a *measurement noise ceiling* and underestimate model performance. We also note that using a low-SNR validation set would bias in favor of models that predict a large number of voxels with low voxelwise correlations in comparison to models that predict a smaller number of voxels with very high voxelwise correlations.

Chapter 6

Conclusion

6.1 Summary

In this work, we introduced a novel capsule encoder for fMRI, the first deep learning alternative to CNNs trained directly on fMRI data for modelling the human visual pathway. We tested this model using two distinct datasets, one with complex, high-resolution images of naturalistic scenes and one with simple, low-resolution images of handwritten digits. We showed that although the capsule encoder could effectively predict brain activity in response to images for both datasets, its performance did not scale well in response to increased image complexity and resolution when compared to CNN's. Our results indicated that capsule networks, in their current form, have more difficulty extracting features from naturalistic images. This hypothesis has since been acknowledged in recent work by Sabour et al. [49]. However, by introducing new improved metrics and analyses which take into account not only the pattern of predictions but the magnitude as well, we showed that our capsule encoder outperforms the state-of-the-art CNN based model from Beliy et al. [30] for visual stimuli that are relatively simplistic. This suggests that CNNs may not be the only DNN approach effective at modelling the visual pathway, warranting further exploration of

networks that share functional properties with the brain, such as capsule networks, for image-fMRI encoding. We also identified various overlooked limitations in existing methods for evaluating encoding models. We addressed these limitations in our new evaluation approach which we showed improves upon existing methods. Namely, we introduced new metrics such as voxel response sensitivity and response weighted correlation. Using these metrics we identified two types of predictive behaviours in visual encoders which we refer to as *intentional* and *unintentional*. We also introduce a thresholding approach for seperating intentional and unintentional voxels and show that it improves upon seperating voxels by p-value significance as done in many previous works such as Wen et al. [8], St Yves et al. [13], Eickenberg et al. [9], Guclu et al. [10][33] and Naselaris et al. [32]. Additionally, another limitation we identified was the inaccuracy in estimated encoder performance as a result of noise in the validation set. Although increasing the SNR of the validation set reduced this inaccuracy, it also revealed that the lower SNR validation set was consistently underestimating the voxelwise correlation of positively correlated voxels. This suggested that, for certain voxels, the predicted activity was more highly correlated with the ground truth than what was measured in the lower SNR validation set. This is despite the fact that during training the model had only seen data with the same SNR as the lower SNR validation set. To explain this, we introduce our *Partial Averaging* hypothesis which states that although samples may be independent in image space, they are not independent in feature space, allowing deep supervised models to learn to average out noise in a way that allows them to surpass consistency between repeated trials. We therefore believe that noise ceiling estimates should be interpreted as the maximum *measurable* performance as a result of one's validation set, not the maximum *achievable* performance as a result of one's training set.

6.2 Future Work

This work introduces the first deep learning alternative to CNNs for image-to-fMRI encoding. Although we found that early implementations of capsule networks are limited by image complexity, results on capsule encoding for simpler images show promise. This indicates that moving towards networks that are functionally more brain-like could improve our capacity to model the visual pathway. Future work should explore how to incorporate these functional properties into networks without introducing new limitations. Of particular interest would be viewpoint equivariance as CNNs are only translationally equivariant. Additionally, incorporating the network with a better understanding of the object-part hierarchy present within visual scenes could reveal relationships between voxels. An alternative approach would be to explore ways to address the existing limitations of capsule networks whilst preserving their brain-like functional properties. Various works have already attempted to do so [43][53][44][49][54] and hence exploring these newer capsule network alternatives could result in improved encoding performance. We also note that common techniques such as data augmentation, batch normalization, and dropout were found to be essential to the performance of the CNN encoders. However, in our limited exploration, we found that they did not seem to have a significant effect on capsule-based encoders. Future work should explore the effect of various different types of data augmentation and regularization techniques for image-to-fMRI encoding in more depth. For example, it would be interesting to see how adding noise to the stimulus images affects encoder predictions and whether this could be used as a form of data augmentation.

This work also introduces various new metrics and visualizations for evaluating encoding models. These new metrics can generally be utilized for any encoding models in fMRI. One such metric was the *Above Threshold Percentage* describing the percentage of voxels predicted above our novel RWC thresholding approach. This can also be thought of as our estimate of the percentage of voxels intentionally pre-

dicted by the encoders. One interesting analysis that could expand upon this would be to look at the overlap between the sets of intentional voxels for each model as a method of comparison. This would be an interesting way to concisely reveal whether different models are effective for the same populations of voxels and could be used as a measure of functional similarity between models.

This work also explored how the properties of the validation set affected model evaluation. One of our observations was that training with the MNIST-fMRI dataset, which had a smaller validation set, resulted in a larger variance in voxelwise correlation for unintentional voxels. Therefore, we hypothesize that larger validation sets reduce this variance, hence reducing the overlap between the intentional and unintentional distributions and allowing for more accurate model evaluation. Future work should attempt to further quantify how the number of validation samples affects encoder evaluation. However, we also showed that the SNR of the validation set can contribute to inaccuracies in model evaluation as well. Noise in the validation data imposes a ceiling on the maximum measurable voxelwise correlation as well as introduces error in our estimates of correlation. Future work should aim to quantify the relationship between the SNR of the validation set, the associated *measurement ceiling*, and the error in evaluation metrics. Additionally, since a common way to increase the SNR of fMRI data is to average repeated trials, given a limited number of stimulus presentations, this creates a trade-off between the number of unique visual stimuli and the number of repeats in the validation set. Therefore, future work should also aim to explore this trade-off and find the optimal ratio between the number of samples and repeats that results in the most effective validation set.

Lastly, we also note that in all existing work, the fMRI space has been effectively represented as a 1-dimensional vector. That is to say, that information on the spatial positions of the voxels relative to one another has been removed. Various other computer vision tasks, such as image segmentation, have shown the spatial informa-

tion of the output can be incredibly important. Models that use this information, such as U-Net based architectures, resulted in significant improvements. Future work should look to use the spatial dimensions of the fMRI data when designing encoding models. We note that there are various challenges associated with using this spatial information for image-to-fMRI encoding. Firstly, the input and output have different numbers of spatial dimensions (images are 2D and fMRI volumes are 3D). Future work could attempt to match the number of spatial dimensions to reduce the complexity of the mapping. For example, one could reduce the dimensionality of fMRI by mapping it to the cortical surface and flattening it, although this transformation is likely to result in a large amount of information loss. Alternatively one could try to increase the dimensionality of the images by extrapolating a 3rd spatial dimension or introducing depth information. However, it is possible that these enhancements of the image data would not be accurate enough to improve encoding performance or would result in computational constraints.

Appendix A

Supplementary Materials

A.1 Capsule Encoder Experiments

Learning Rate	Epochs	Batch Size	Loss	Accuracy
0.001	50	100	0.019	0.984

Table A.1: Our implementation of the original capsule network from Sabour et al. [26] on MNIST classification without data augmentation. The network was optimized using the Adam optimizer with its default TensorFlow settings. No learning rate schedule was used

Model	Input Conv ($N/K/S/A$)	Primary Caps ($N/K/S/C/A$)	Conv Caps ($L/N/K/S/C/A$)	Output Caps ($N/C/A$)	FC Layers ($N_1/N_2/.../N_n$)	Regularizers
1	256/9/1/relu	32/9/2/8/sq	0/-/-/-/-/-	10/16/sq	512/1024/N_v	-
2	pretrained	32/9/2/4x4/sq	1/32/3/2/4x4/d	32/4x4/sq	512/1024/N_v	-
3	pretrained	32/9/2/4x4/d	2/32/3/2/4x4/d	32/4x4/sq	512/1024/N_v	-
4	pretrained	32/9/2/4x4/d	2/32/3/2/4x4/d	32/4x4/sq	512/1024/N_v	L1, L2
5	pretrained	32/9/2/4x4/d	2/32/3/2/4x4/d	32/4x4/sq	512/1024/N_v	L1, L2, BatchNorm
6	256/9/2/relu	32/9/2/4x4/d	2/32/3/2/4x4/d	32/4x4/sq	32/128/256/100/N_v	L1, L2, BatchNorm
7	32/9/2/relu	16/9/2/4x4/d/	1/16/3/2/4x4/d	16/4x4/sq	1024/N_v	-
8	256/9/1/relu	32/9/2/8/sq	1/32/3/1/8/sq	2/16/n	512/1024/N_v	-
9	128/9/1/relu	8/9/2/8/sq	0/-/-/-/-/-	10/16/sq	512/1024/N_v	-
10	128/9/2/relu	32/9/2/4x4/sq	1/32/3/2/4x4/sq	32/4x4/sq	512/1024/N_v	-
11	256/9/1/relu	32/9/2/8/sq/	0/-/-/-/-/-	2/16/n	512/1024/N_v	-
12	256/9/1/relu	32/9/2/8/sq/	0/-/-/-/-/-	2/16/sq	512/1024/N_v	-

Table A.2: Detailed overview of various capsule network architectures. Variables are as follows: L-Number of layers, N-Number of units or channels, K-Kernel Size, S-Stride, C-Capsule Dimensionality, A-Capsule Activation (sq-Squash Function, d-Discriminatively Learned, n-Norm Function)

Model	Routing	Notes	Val MSE	Val Cosine Similarity
1	dynamic	200 Epochs, LR Milestone at epoch 20, Random shifting up to 5 pixels	0.0845	0.221
1	dynamic	40 Epochs, Random shifting up to 5 pixels	0.0852	0.210
1	dynamic	25 Epochs, MSE-Voxel Corr Loss, Random shifting up to 5 pixels	0.0802	0.249
1	dynamic	100 Epochs, MSE-Voxel Corr Loss, LR Milestone at epoch 20, Random shifting up to 5 pixels	0.0805	0.238
1	dynamic	25 Epochs, Random shifting up to 5 pixels	0.0820	0.192
11	dynamic	80 Epochs, LR Milestones at Epochs 20, 35, 45, 50	0.0821	0.200
11	dynamic	40 Epochs, LR Decay $1.25e^{-3}$, Grad Clip Value 0.5	0.0843	0.198
2	EM	10 Epochs, Adam Optimizer, Batch Size 75, LR $1e^{-3}$	0.0835	0.185
2	EM	20 Epochs, Batch Size 75, LR Decay $1e^{-6}$, Grad Clip Value 0.5	0.0838	0.184
2	EM	20 Epochs. Batch Size 75, Grad Clip Value 1.0	0.0836	0.185
2	EM	20 Epochs, Batch Size 75, LR Decay $1e^{-6}$, Grad Clip Value 0.5, MSE Loss	0.0838	0.164
2	EM	10 Epochs, Batch Size 75, LR Decay $1e^{-6}$, Grad Clip Value 0.5, tanh activation for last dense layer	0.0839	0.185
2	EM	10 Epochs, Batch Size 75, LR Decay $1e^{-6}$, Grad Clip Value 0.5	0.0838	0.184
3	EM	20 Epochs, Batch Size 75, Grad Clip Value 2.0	0.0883	0.124
3	EM	10 Epochs, Batch Size 75, LR Decay $1e^{-2}$, Grad Clip Value 0.5, MSLE Loss	0.0854	0.032
4	EM	10 Epochs, Batch Size 75, Grad Clip Value 2.0	0.0880	0.167
4	EM	20 Epochs, Batch Size 75, LR Decay $5e^{-3}$, Grad CliP Value 0.5	0.0837	0.185
4	EM	20 Epochs, Batch Size 75, LR Decay $1e^{-6}$, Grad Clip Value 0.5	0.0839	0.183
5	EM	20 Epochs, Batch Size 75, LR Decay $5e^{-3}$, Grad Clip Value 0.5	0.0835	0.185
6	EM	10 Epochs, Batch Size 75, LR Decay $1e^{-2}$, Grad Clip Value 0.5	0.0835	0.185
6	EM	10 Epochs, Batch Size 75, LR Decay $1e^{-2}$, Grad Clip Value 0.5, Used ELU in place of ReLU	0.0839	0.183
6	EM	10 Epochs, Batch Size 75, LR Decay $1e^{-2}$, Grad Clip Value 0.5, MSLE Loss	0.0851	0.000
6	EM	10 Epochs, Batch Size 75, LR Decay $1e^{-2}$, Grad Clip Value 0.5, MSE Loss	0.0847	0.163
6	EM	10 Epochs, Batch Size 75, LR Decay $1e^{-2}$, Grad Clip Value 0.5, MSE Loss, tanh activation for last dense layer	0.0845	0.141
7	EM	40 Epochs, Batch Size 75, LR Decay $2.5e^{-3}$, Grad Clip Value 0.5	0.0836	0.185
7	EM	20 Epochs, Batch Size 75, LR Decay $5e^{-3}$, Grad Clip Value 0.5, Random shifting up to 5 pixels	0.0837	0.185
7	dynamic	30 Epochs, Random shifting up to 5 pixels	0.0831	0.218
9	dynamic	50 Epochs, Random shifting up to 5 pixels	0.0857	0.217
9	dynamic	200 Epochs, LR Milestone at epoch 25, Random shifting up to 5 pixels	0.0840	0.223

Table A.3: Capsule Network Experiments on Imagenet-fMRI. Note that this is not an exhaustive list. Unless otherwise stated, models were trained using the SGD optimizer with a learning rate of 0.1, momentum of 0.9, batch size of 64, and the mse-cosine loss from [30]. All LR milestones had a gamma value of 0.1 (LR was decreased by a factor of 10), if no LR milestones are listed then there was no learning rate schedule

Model	Routing	Notes	Val MSE	Val Cosine Similarity
11	dynamic	400 Epochs, LR Milestones at epochs 10 and 20	1.010	0.188
11	dynamic	20 Epochs, LR Milestone at epoch 25, Random shifting up to 2 pixels	1.027	0.225
11	dynamic	100 Epochs, LR Milestones at epochs 10 and 20	1.038	0.195
11	dynamic	850 Epochs, LR Milestones at epochs 10 and 20	1.023	0.238
11	dynamic	400 Epochs, LR Milestones at epochs 10 and 20	1.016	0.246
11	dynamic	40 Epochs, Batch Size 40	0.958	0.232
7	dynamic	600 Epochs	1.084	0.017
7	dynamic	600 Epochs, LR Milestones at epochs 10 and 20	1.0838	0.017
7	EM	100 Epochs	1.084	0.005
7	EM	100 Epochs, LR $1e^{-4}$	1.084	0.018
7	EM	100 Epochs, SGD Optimizer	1.138	-0.018
7	EM	100 Epochs, LR Milestone at epoch 20	1.084	0.026
7	EM	150 Epochs, SGD Optimizer, Momentum 0.9	1.083	0.045
7	EM	400 Epochs, SGD Optimizer, Momentum 0.9, LR Milestone at epoch 150	1.084	0.018
7	EM	100 Epochs, Random shifting up to 5 pixels	1.084	0.015
7	EM	100 Epochs, LR Milestones at epochs 10 and 20	1.084	0.017
10	dynamic	600 Epochs, LR Milestones at epochs 10, 100 and 150	1.084	0.014
10	dynamic	150 Epochs	1.048	0.198
10	dynamic	200 Epochs	1.039	0.222
10	dynamic	100 Epochs	1.026	0.222
10	dynamic	200 Epochs	1.084	-0.006
10	dynamic	100 Epochs, SGD Optimizer, Momentum 0.9	1.083	0.041
10	dynamic	100 Epochs, SGD Optimizer, Momentum 0.9, LR Milestone at epoch 5	1.084	0.018
10	dynamic	100 Epochs, SGD Optimizer, Momentum 0.9, LR Milestones at epochs 5 and 15	1.084	0.018
10	dynamic	200 Epochs, SGD Optimizer, LR $1e^{-4}$, Momentum 0.9	1.393	-0.019
10	dynamic	100 Epochs, SGD Optimizer, LR $5e^{-4}$, Momentum 0.9	1.084	0.013
10	dynamic	100 Epochs, LR $5e^{-4}$, LR Milestone at epoch 40	1.022	0.225
10	dynamic	200 Epochs, LR $5e^{-4}$, LR Milestones at epochs 40 and 80	1.037	0.205
8	dynamic	100 Epochs, LR Milestone at epoch 10	1.047	0.222
8	dynamic	100 Epochs, LR Milestones at epochs 10 and 20	1.043	0.180
8	dynamic	400 Epochs, LR Milestones at epochs 10 adn 20	1.019	0.228
11	EM	200 Epochs, LR Milestones at epochs 10 and 20	1.071	0.120
11	EM	450 Epochs, LR Milestones at epochs 10 and 20	1.024	0.223
11	dynamic	25 Epochs, Batch Size 40	1.011	0.238
11	dynamic	150 Epochs, SGD Optimizer, Momentum 0.9, Batch Size 40, LR Decay $6.6e^{-4}$	1.055	0.223
11	dynamic	150 Epochs, Batch Size 40	1.065	0.171
11	dynamic	600 Epochs, LR Milestones at epochs 10 and 20	1.017	0.247
12	dynamic	400 Epochs, LR Milestones at epochs 10 and 20	1.011	0.247

Table A.4: Capsule Network Experiments on MNIST-fMRI. Note that this is not an exhaustive list. Unless otherwise stated, models were trained using the Adam optimizer with it's default tensorflow settings, a batch size of 8 and the mse-cosine loss from [30]. Additionally, batches were augmented by randomly shifting the images by up to 5 pixels in either direction. All LR milestones had a gamma value of 0.1 (LR was decreased by a factor of 10), if no LR milestones are listed then there was no learning rate schedule

A.2 Existing Methods for Encoder Evaluation

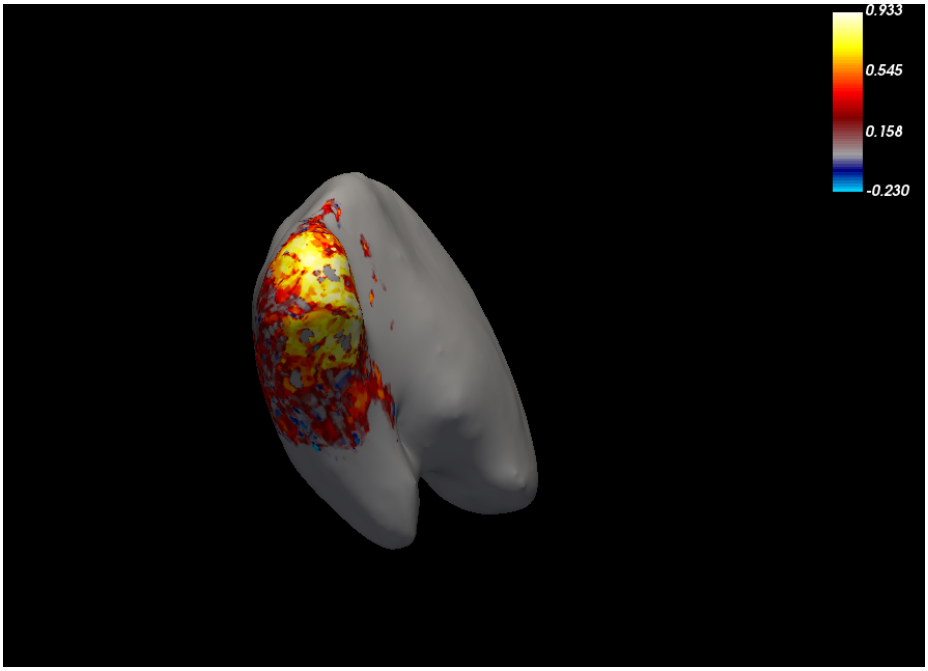
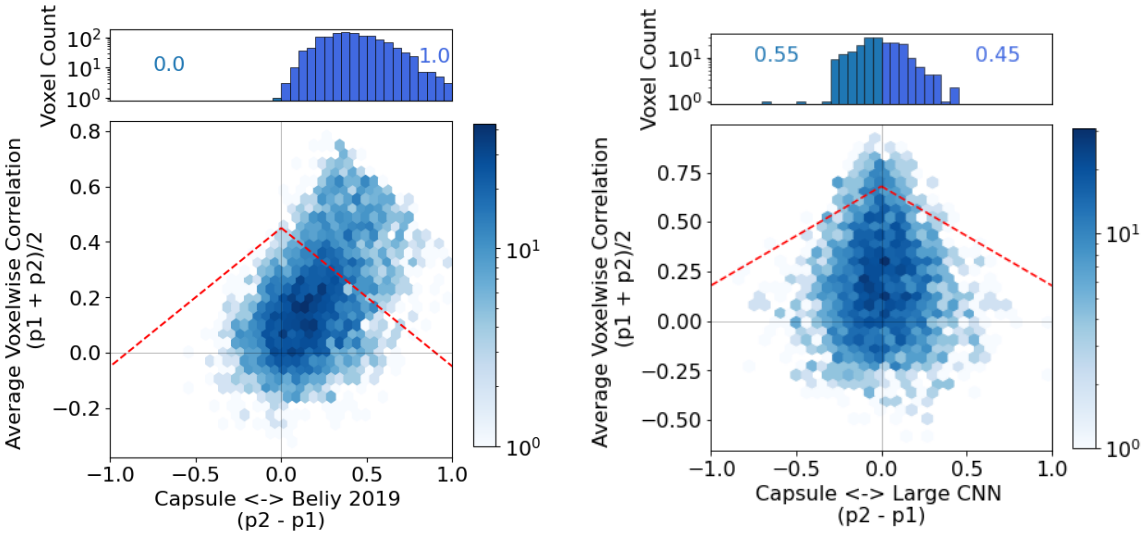


Figure A.1: Voxelwise Correlation for Beliy encoder on Imagenet-fMRI dataset displayed on cortical surface using freesurfer. Only the left hemisphere is shown.



(a) Imagenet-fMRI

(b) MNIST-fMRI

Figure A.2: Voxelwise-Correlation/Advantage plots from St Yves et al. [13] comparing the capsule encoders to CNN-based encoders on both datasets. The y-axis represents the average voxelwise correlation between the two models under comparison. The x-axis represents the difference in voxelwise correlation between the two models with shifts to left representing an advantage for the capsule encoders and shifts to the right representing an advantage for the CNN-based encoders. $\rho 1$ and $\rho 2$ are the voxelwise correlations for the CNN-based and capsule encoders respectively. The histograms show the distribution of the x-axis for voxels with a voxelwise correlation above a significance threshold of $p < 0.001$ for at least one of the two models. This corresponds to voxels above the red dashed line

A.3 Further Model Characterization

Model	Datsets	
	MNIST-fMRI	**Imagenet-fMRI**
Capsule	9.3M	105.0M
Beliy 2019	1.6M	29.2M
Large CNN	10M	175.2

Table A.5: Table outlining the sizes of each model architecture when trained on each dataset. The values are the number of parameters in the model in millions

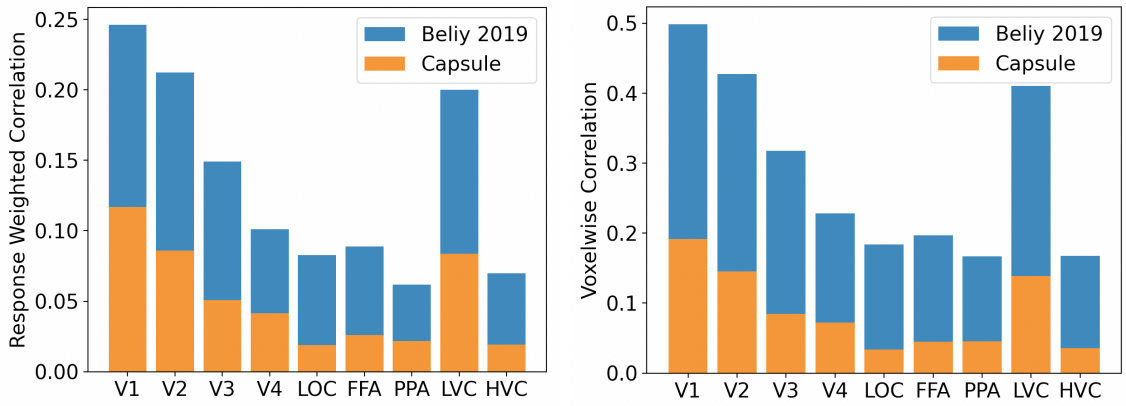


Figure A.3: Bar plots showing the response weighted correlation (RWC) and voxelwise correlations achieved by both the capsule and beliy [30] encoders on the ImageNet-fMRI dataset for various parts of the visual pathway. The ROI masks which select which voxels belong to which region of interest were provided in the dataset by the original authors [36]

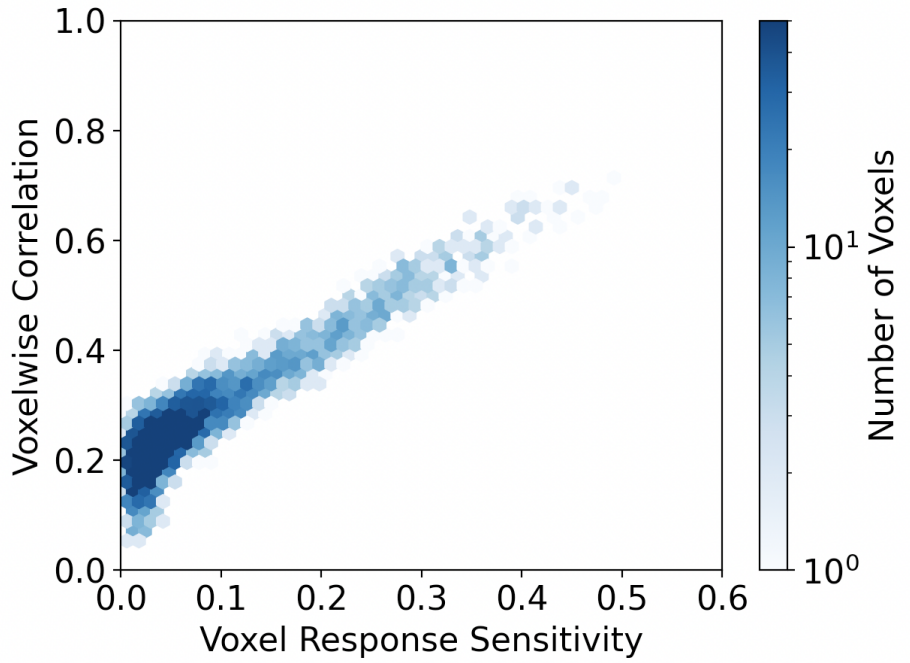


Figure A.4: Hexbin plot showing the relationship between voxel response sensitivity and the voxelwise correlation calculated using the training set for the Beliy encoder on Imagenet-fMRI

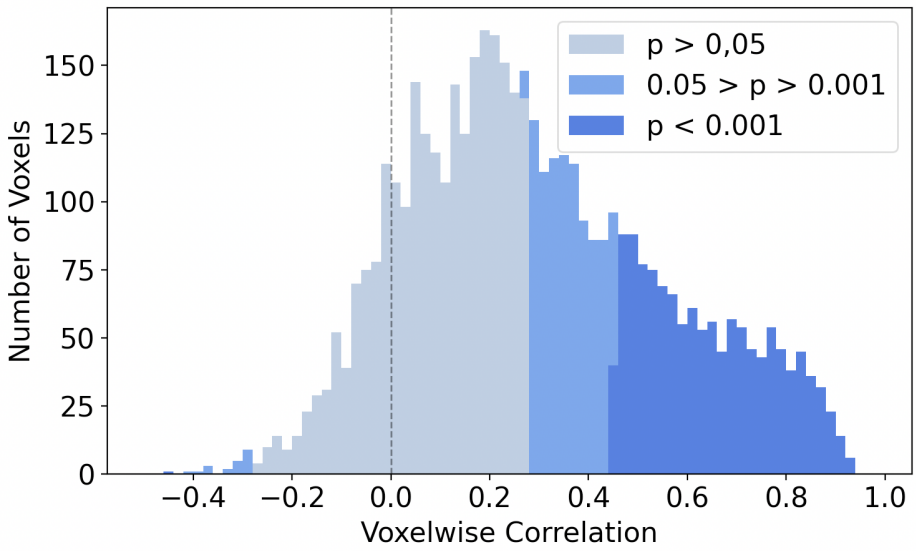
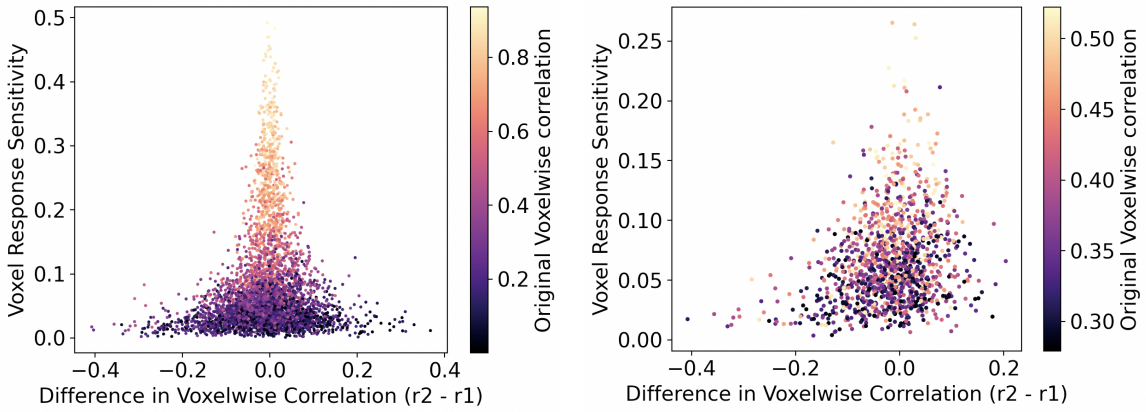


Figure A.5: Histogram of Voxelwise Correlations for the Beliy encoder on Imagenet-fMRI categorized by p-value



(a) Entire voxel Subset

(b) Voxels of Interest only

Figure A.6: Plot showing the relationship between voxel response sensitivity and the robustness of the voxels. The x-axis is the difference in voxelwise correlation between the Beliy encoder trained with only the subset of positive voxels (r2) and all the voxels (r1). The y-axis is the voxel response sensitivity of the Beliy encoder trained with all the voxels. The color of each point indicates the voxel's original correlation (r1) from the Beliy encoder trained with all the voxels. Figure A.6a includes points for all voxels in the positive subset and Figure A.6b focuses only on voxels that also had a significance between 5% and 0.1% ($0.05 > p < 0.001$)

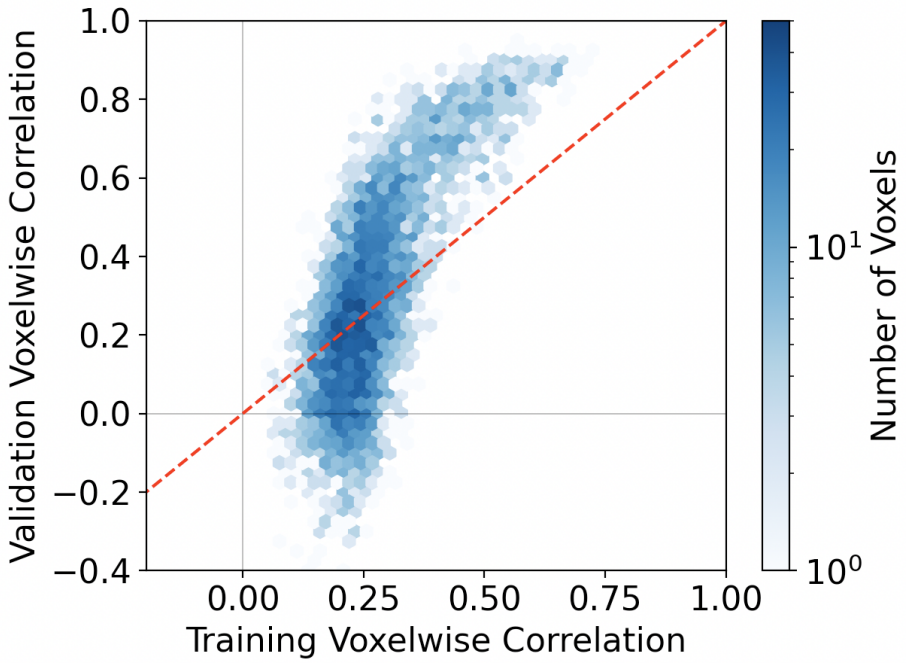


Figure A.7: Hex plot comparing the voxelwise correlations computed using the training and validation data for the Beliy model trained on Imagenet-fMRI. X=Y is denoted by the red dashed line

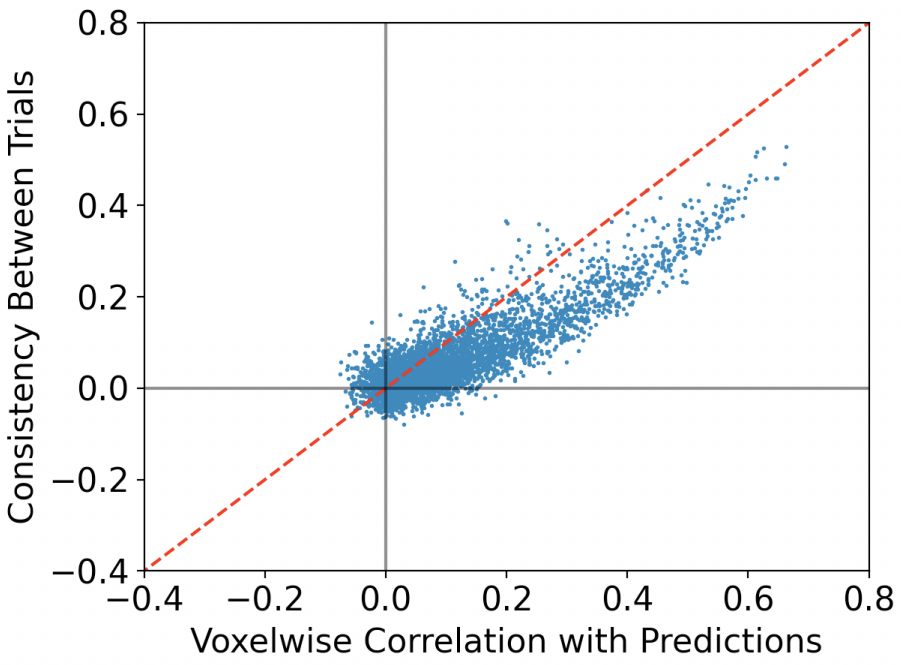


Figure A.8: Scatter plot showing the voxelwise correlation between repeated fMRI measurements in response to the same visual stimuli (y-axis) versus their voxelwise correlation with the fMRI activations predicted by the Beliy encoder [30] on Imagenet-fMRI. For the x-axis, the predicted fMRI response was correlated in a voxelwise manner with each of the 35 repeats, and then the voxelwise correlation for each voxel was computed by averaging the 35 resultant correlations. For the y-axis, each of the 35 repeated fMRI measurements were correlated in a voxelwise manner with another randomly chosen repeat other than themselves. The consistency between trials for each voxel was then calculated by averaging the 35 resultant voxelwise correlations. X=Y is denoted by the red dashed line. Points below this line represent voxels for which the predicted BOLD response was (on average) a better estimation than the measured BOLD response

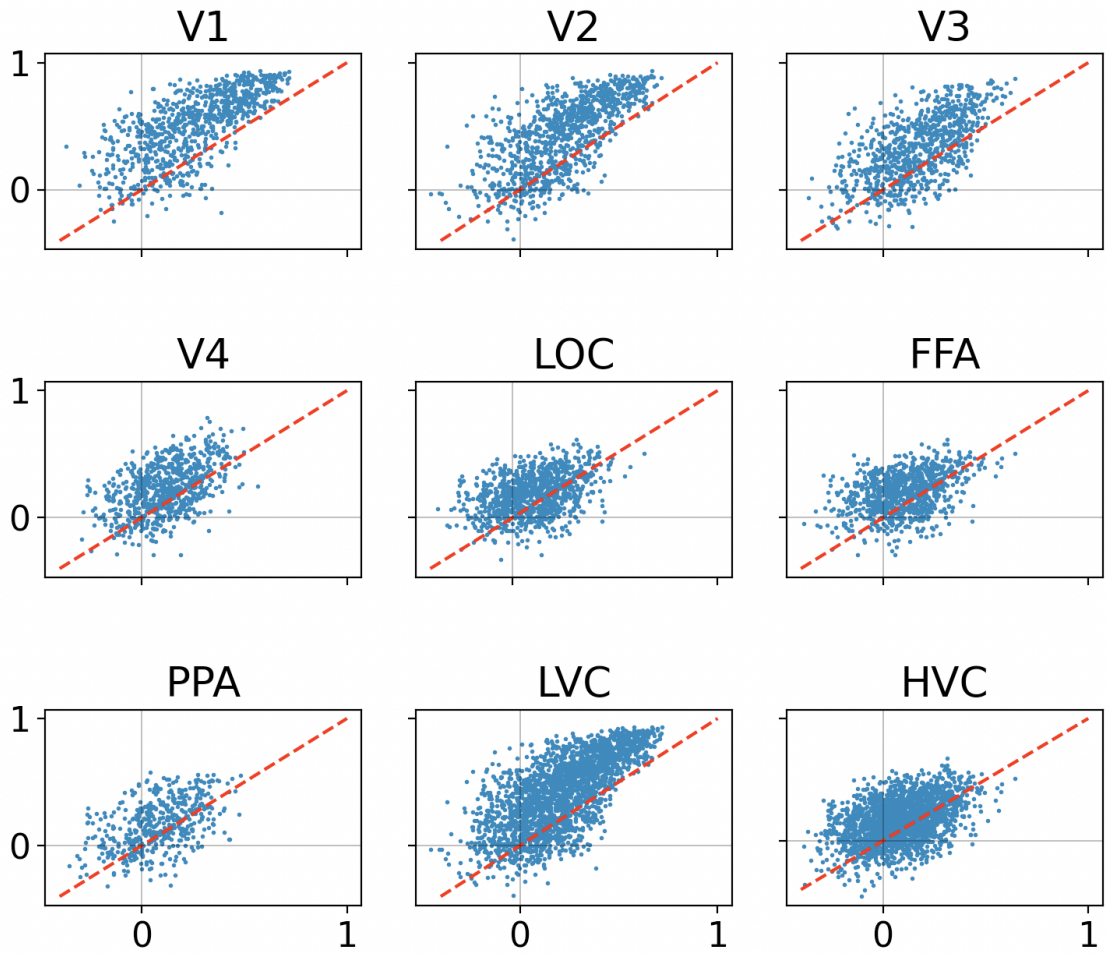


Figure A.9: Scatter plots showing how the SNR of the validation set affects the voxelwise correlations obtained by the Beliy 2019 model trained on the Imagenet-fMRI dataset for different regions of interest (ROI's). The y-axes use the original high-SNR validation set, obtained by averaging 35 repeated fMRI measurements for each stimulus image, to compute the voxelwise correlations for each voxel. The x-axes use only a single repeat for each image as the ground truth to compute voxelwise correlation (low-SNR validation set). X=Y is denoted by the red dashed line

Appendix B

List of Abbreviations

AI ∼ Artificial Intelligence

BCI ∼ Brain Computer Interface

BOLD ∼ Blood Oxygenation Level Dependent

CNN ∼ Convolutional Neural Network

dHb ∼ Deoxyhemoglobin or Deoxygenated Hemoglobin

DNN ∼ Deep Neural Network

EEG ∼ Electroencephalography

EM ∼ Expectation-Maximization

FFA ∼ Fusiform Face Area

fMRI ∼ Functional Magnetic Resonance Imaging

FOV ∼ Field of View

fwRF ∼ Feature Weighted Receptive Field

GWP ∼ Gaussian Wavelet Pyramid

Hb ∼ Hemoglobin

HVC ∼ Higher Visual Cortex

LGN ∼ Lateral Geniculate Nucleus

LO/LOC ∼ Lateral Occipital Cortex

LR ∼ Learning Rate

LVC ∼ Lower Visual Cortex

ML ∼ Machine Learning

MRI ∼ Magnetic Resonance Imaging

MSE ∼ Mean Squared Error

MT/MST Medial Superior Temporal Area

NMR ∼ Nuclear Magnetic Resonance

PPA ∼ Parahippocampal Place Area

ReLU ∼ Rectified Linear Unit

RF ∼ Radio-Frequency

ROI ∼ Region Of Interest

RS ∼ Response Sensitivity

RSA ∼ Representational Similarity Analysis

RWC ∼ Response Weighted Correlation

SGD ∼ Stochastic Gradient Descent

SNR ∼ Signal-to-Noise Ratio

***N*D** $\sim$ N-Dimensional

Appendix C

Glossary

Activations ~ Scalar values representing brain activity/level of neuronal firing within a specified anatomical region such as a voxel

Agreement ~ Used by capsule networks to dynamically weight features. Represents the degree to which a specific child capsule conforms to what the other child capsules predict the properties of a parent capsule to be. Child capsules can be said to *agree* with one another when they predict the properties of the parent capsule to be the same. *Agreement* is a concept rather than an absolute measurable property and therefore has no fixed equation.

Bagging Aggregation ~ A method in which multiple bootstrap replicates of some dataset are used in order to improve the robustness of estimates derived from the dataset by aggregating the estimates from each of the bootstrap replicates. Within the context of model training, refers to using bootstrap replicates of the training set to repeatedly train a model and aggregating the predictions of those models. Within the context of model evaluation refers to using bootstrap replicates of the validation set to repeatedly evaluate a single model and aggregating the evaluation metrics obtained from each replicate.

Bootstrap Replicate ∼ A *copy* of some dataset created by randomly selecting samples from the original dataset *with replacement* (samples are not removed from the original dataset after being selected and therefore may be selected multiple times to appear more than once in the bootstrap replicate).

Decoding ∼ Within the context of computational modelling, this refers to predicting physical stimuli (such as images) from measurements of brain activity (such as with fMRI). Models that perform this task are referred to as decoders.

Encoding ∼ Converting physical stimuli into neural activations. Within the context of computational modelling this refers to predicting measurements of brain activity (such as with fMRI) in response to physical stimuli (such as images). Models that perform this task are referred to as encoders

Features ∼ Information of interest derived from some input that more efficiently describes that input. For example, for image inputs, simple features could include things such as edges, shapes, color, etc. and more complex features could include things such as the position and location of objects within the image.

Higher Order/Level ∼ Refers to an increased complexity relative to the rest. Within the context of image features, this refers to features that are derived from many simpler features and themselves are more complex such as faces, dogs, etc. Within the context of the brain this is generally used to describe brain regions known for processing higher-level features and being further downstream the visual pathway (neuroscientifically referred to as the higher visual cortex). Within the context of DNNs, refers to deeper layers (further from the input) that are responsible for detecting higher level features.

Intentional ∼ Used to describe voxels for which an encoder has meaningfully

learned some signal with non-zero variance in response to visual stimuli. These voxels are said to be part of the *intentional distribution* and the model is said to have predicted them *intentionally*.

Kernels ∼ Values used in convolution to extract features from the input. Within the context of CNNs which use 2D convolutions, these kernels are generally small matrices learned by the network. Each kernel extracts one specific feature from an input image or feature map and the various learned kernels form the weights of a convolutional layer.

Lower Order/Level ∼ Refers to a decreased complexity relative to the rest. Within the context of image features, this refers to simpler features such as lines, edges, shapes, etc. Within the context of the brain, this could be used to describe brain regions known for processing lower-level features and being earlier in the visual pathway (neuroscientifically referred to as the early visual cortex). Within the context of DNNs, refers to shallow layers (closer to the input) that are responsible for detecting lower level features.

Noise Ceiling ∼ In existing work is defined as the maximum achievable model performance (usually measured in voxelwise correlation) that can be obtained given the amount of noise in a specific training set. Generally estimated using the consistency between repeated trials.

Partial Averaging Effect ∼ Hypothesis that describes how complex supervised networks can learn to ignore some of the noise in training data by taking advantage of overlapping similarities between distinct inputs (eg. images) in a latent feature space

Samplewise ∼ To use the axis that represents samples when indexing. A samplewise vector contains all of the voxel activations in response to a specific

sample which together form a brain volume. When a metric is calculated using samplewise vectors it is said to be computed samplewise.

Unintentional ∼ Used to describe voxels for which an encoder has not meaningfully learned any signal in response to visual stimuli. The predictive behaviour for these voxels is not robust and does not contain information about its relationship in response to visual stimuli. These voxels are said to be part of the *unintentional distribution* and the model is said to have predicted them *unintentionally*.

Voxels ∼ A 3D box-shaped volume in space that forms part of a larger volume. Generally used to parcellate continuous 3D volumes such that they can be represented digitally.

Voxelwise ∼ To use the axis that represents voxels when indexing. A voxelwise vector contains all of the activations for a single voxel across a set of different samples (eg. visual stimuli). When a metric is calculated using voxelwise vectors it is said to be computed voxelwise.

Bibliography

[1] M. Chen, J. Han, X. Hu, X. Jiang, L. Guo, and T. Liu, "Survey of Encoding and Decoding of Visual Stimulus via FMRI: An Image Analysis Perspective," *Brain imaging and behavior*, vol. 8, pp. 7–23, Mar. 2014.

[2] T. P. Trappenberg, *Fundamentals of computational neuroscience*. Oxford ; New York: Oxford University Press, 2nd ed ed., 2010. OCLC: ocn444383805.

[3] K. N. Kay, "Principles for models of neural information processing," *NeuroImage*, vol. 180, pp. 101–109, Oct. 2018.

[4] M. N. Hebart and C. I. Baker, "Deconstructing multivariate decoding for the study of brain function," *NeuroImage*, vol. 180, pp. 4–18, Oct. 2018.

[5] T. Naselaris, C. A. Olman, D. E. Stansbury, K. Ugurbil, and J. L. Gallant, "A voxel-wise encoding model for early visual areas decodes mental images of remembered scenes," *NeuroImage*, vol. 105, pp. 215–228, Jan. 2015.

[6] G. Shen, T. Horikawa, K. Majima, and Y. Kamitani, "Deep image reconstruction from human brain activity," *PLOS Computational Biology*, vol. 15, p. e1006633, Jan. 2019. Publisher: Public Library of Science.

[7] G. Shen, K. Dwivedi, K. Majima, T. Horikawa, and Y. Kamitani, "End-to-End Deep Image Reconstruction From Human Brain Activity," *Frontiers in Computational Neuroscience*, vol. 13, 2019. Publisher: Frontiers.

[8] H. Wen, J. Shi, Y. Zhang, K.-H. Lu, J. Cao, and Z. Liu, "Neural Encoding and Decoding with Deep Learning for Dynamic Natural Vision," *Cerebral Cortex*, vol. 28, pp. 4136–4160, Dec. 2018.

[9] M. Eickenberg, A. Gramfort, G. Varoquaux, and B. Thirion, "Seeing it all: Convolutional network layers map the function of the human visual system," *NeuroImage*, vol. 152, pp. 184–194, May 2017.

[10] U. Guclu and M. A. J. van Gerven, "Deep Neural Networks Reveal a Gradient in the Complexity of Neural Representations across the Ventral Stream," *Journal of Neuroscience*, vol. 35, pp. 10005–10014, July 2015.

[11] M. M. Monti, A. Vanhaudenhuyse, M. R. Coleman, M. Boly, J. D. Pickard, L. Tshibanda, A. M. Owen, and S. Laureys, "Willful Modulation of Brain Activity in Disorders of Consciousness," *New England Journal of Medicine*, vol. 362, pp. 579–589, Feb. 2010. Publisher: Massachusetts Medical Society _eprint: https://doi.org/10.1056/NEJMoa0905370.

[12] A. M. Owen, M. R. Coleman, M. Boly, M. H. Davis, S. Laureys, and J. D. Pickard, "Detecting Awareness in the Vegetative State," *Science*, vol. 313, pp. 1402–1402, Sept. 2006. Publisher: American Association for the Advancement of Science.

[13] G. St-Yves and T. Naselaris, "The feature-weighted receptive field: an interpretable encoding model for complex feature spaces," *NeuroImage*, vol. 180, pp. 188–202, Oct. 2018.

[14] E. J. Allen, G. St-Yves, Y. Wu, J. L. Breedlove, J. S. Prince, L. T. Dowdle, M. Nau, B. Caron, F. Pestilli, I. Charest, J. B. Hutchinson, T. Naselaris, and K. Kay, "A massive 7T fMRI dataset to bridge cognitive neuroscience and artifi-

cial intelligence," *Nature Neuroscience*, vol. 25, pp. 116–126, Jan. 2022. Number: 1 Publisher: Nature Publishing Group.

[15] D. D. Himabindu and S. P. Kumar, "A Survey on Computer Vision Architectures for Large Scale Image Classification using Deep Learning," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 10, 2021.

[16] S. Minaee, Y. Boykov, F. Porikli, A. Plaza, N. Kehtarnavaz, and D. Terzopoulos, "Image Segmentation Using Deep Learning: A Survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, pp. 3523–3542, July 2022. Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence.

[17] T. Karras, S. Laine, and T. Aila, "A Style-Based Generator Architecture for Generative Adversarial Networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2020. Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence.

[18] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, "Analyzing and Improving the Image Quality of StyleGAN," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8107–8116, June 2020. ISSN: 2575-7075.

[19] G. Gaziv, R. Beliy, N. Granot, A. Hoogi, F. Strappini, T. Golan, and M. Irani, "Self-supervised Natural Image Reconstruction and Large-scale Semantic Classification from Brain Activity," *NeuroImage*, vol. 254, p. 119121, July 2022.

[20] Horizon2333, "Auto-Encoder trained on ImageNet," Dec. 2022. original-date: 2021-08-11T05:33:40Z.

[21] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, June 2009. ISSN: 1063-6919.

[22] S. Bartunov, A. Santoro, B. Richards, L. Marris, G. E. Hinton, and T. Lillicrap, "Assessing the Scalability of Biologically-Motivated Deep Learning Algorithms and Architectures," in *Advances in Neural Information Processing Systems*, vol. 31, Curran Associates, Inc., 2018.

[23] R. Pogodin, Y. Mehta, T. Lillicrap, and P. E. Latham, "Towards Biologically Plausible Convolutional Networks," in *Advances in Neural Information Processing Systems*, vol. 34, pp. 13924–13936, Curran Associates, Inc., 2021.

[24] D. H. Hubel and T. N. Wiesel, "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex," *The Journal of Physiology*, vol. 160, no. 1, pp. 106–154, 1962. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1113/jphysiol.1962.sp006837.

[25] D. H. Hubel and T. N. Wiesel, "Sequence regularity and geometry of orientation columns in the monkey striate cortex," *The Journal of Comparative Neurology*, vol. 158, no. 3, pp. 267–293, 1974. Publisher: The Wistar Institute of Anatomy and Biology.

[26] S. Sabour, N. Frosst, and G. E. Hinton, "Dynamic Routing Between Capsules," in *Advances in Neural Information Processing Systems 30* (I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, eds.), (Long Beach, California), pp. 3856–3866, Curran Associates, Inc., 2017.

[27] G. E. Hinton, S. Sabour, and N. Frosst, "Matrix capsules with EM routing," (Vancouver, Canada), Feb. 2018.

[28] V. Gliozzi, G. L. Pozzato, and A. Valese, "Combining neural and symbolic approaches to solve the Picasso problem: A first step," *Displays*, vol. 74, p. 102203, Sept. 2022.

[29] S. Prasad and S. L. Galetta, "Anatomy and physiology of the afferent visual system," in *Handbook of Clinical Neurology*, vol. 102, pp. 3–19, Elsevier, 2011.

[30] R. Beliy, G. Gaziv, A. Hoogi, F. Strappini, T. Golan, and M. Irani, "From voxels to pixels and back: Self-supervision in natural-image reconstruction from fMRI," in *Advances in Neural Information Processing Systems*, vol. 32, Curran Associates, Inc., 2019.

[31] K. N. Kay, T. Naselaris, R. J. Prenger, and J. L. Gallant, "Identifying natural images from human brain activity," *Nature*, vol. 452, pp. 352–355, Mar. 2008. Number: 7185 Publisher: Nature Publishing Group.

[32] T. Naselaris, R. J. Prenger, K. N. Kay, M. Oliver, and J. L. Gallant, "Bayesian Reconstruction of Natural Images from Human Brain Activity," *Neuron*, vol. 63, pp. 902–915, Sept. 2009. Publisher: Elsevier.

[33] U. Güçlü and M. A. J. v. Gerven, "Unsupervised Feature Learning Improves Prediction of Human Brain Activity in Response to Natural Images," *PLOS Computational Biology*, vol. 10, p. e1003724, Aug. 2014. Publisher: Public Library of Science.

[34] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks," Feb. 2014. arXiv:1312.6229 [cs].

[35] T. Fang, Y. Qi, and G. Pan, "Reconstructing Perceptive Images from Brain Activity by Shape-Semantic GAN," Tech. Rep. arXiv:2101.12083, arXiv, Jan. 2021. arXiv:2101.12083 [cs] type: article.

[36] T. Horikawa and Y. Kamitani, "Generic decoding of seen and imagined objects using hierarchical visual features," *Nature Communications*, vol. 8, p. 15037, May 2017. Number: 1 Publisher: Nature Publishing Group.

[37] R. VanRullen and L. Reddy, "Reconstructing faces from fMRI patterns using deep generative neural networks," *Communications Biology*, vol. 2, p. 193, Dec. 2019.

[38] Z. Ren, J. Li, X. Xue, X. Li, F. Yang, Z. Jiao, and X. Gao, "Reconstructing seen image from brain activity by visually-guided cognitive representation and adversarial learning," *NeuroImage*, vol. 228, p. 117602, Mar. 2021.

[39] K. Seeliger, U. Güçlü, L. Ambrogioni, Y. Güçlütürk, and M. A. J. van Gerven, "Generative adversarial networks for reconstructing natural images from brain activity," *NeuroImage*, vol. 181, pp. 775–785, Nov. 2018.

[40] M. Mozafari, L. Reddy, and R. VanRullen, "Reconstructing Natural Scenes from fMRI Patterns using BigBiGAN," in *2020 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, July 2020. ISSN: 2161-4407.

[41] K. Qiao, C. Zhang, L. Wang, J. Chen, L. Zeng, L. Tong, and B. Yan, "Accurate Reconstruction of Image Stimuli From Human Functional Magnetic Resonance Imaging Based on the Decoding Model With Capsule Network Architecture," *Frontiers in Neuroinformatics*, vol. 12, 2018. Publisher: Frontiers.

[42] D. Tadin, J. S. Lappin, R. Blake, and E. D. Grossman, "What constitutes an efficient reference frame for vision?," *Nature Neuroscience*, vol. 5, pp. 1010–1015, Oct. 2002. Number: 10 Publisher: Nature Publishing Group.

[43] V. Mazzia, F. Salvetti, and M. Chiaberge, "Efficient-CapsNet: capsule network with self-attention routing," *Scientific Reports*, vol. 11, p. 14634, July 2021.

Bandiera_abtest: a Cc_license_type: cc_by Cg_type: Nature Research Journals Number: 1 Primary_atype: Research Publisher: Nature Publishing Group Subject_term: Computer science;Information technology;Software Subject_term_id: computer-science;information-technology;software.

[44] A. Byerly, T. Kalganova, and I. Dear, "No routing needed between capsules," *Neurocomputing*, vol. 463, pp. 545–553, Nov. 2021.

[45] M. van Gerven, F. de Lange, and T. Heskes, "Handwritten digits in fMRI ('69' data set)," Oct. 2018. Version Number: 1.0 Type: dataset.

[46] L. Deng, "The MNIST Database of Handwritten Digit Images for Machine Learning Research [Best of the Web]," *IEEE Signal Processing Magazine*, vol. 29, pp. 141–142, Nov. 2012. Conference Name: IEEE Signal Processing Magazine.

[47] R. M. Cichy, G. Roig, A. Andonian, K. Dwivedi, B. Lahner, A. Lascelles, Y. Mohsenzadeh, K. Ramakrishnan, and A. Oliva, "The Algonauts Project: A Platform for Communication between the Sciences of Biological and Artificial Intelligence," May 2019. Number: arXiv:1905.05675 arXiv:1905.05675 [cs, q-bio].

[48] R. M. Cichy, K. Dwivedi, B. Lahner, A. Lascelles, P. Iamshchinina, M. Graumann, A. Andonian, N. A. R. Murty, K. Kay, G. Roig, and A. Oliva, "The Algonauts Project 2021 Challenge: How the Human Brain Makes Sense of a World in Motion," Tech. Rep. arXiv:2104.13714, arXiv, Apr. 2021. arXiv:2104.13714 [cs, q-bio] version: 1 type: article.

[49] S. Sabour, A. Tagliasacchi, S. Yazdani, G. Hinton, and D. J. Fleet, "Unsupervised Part Representation by Flow Capsules," in *Proceedings of the 38th International Conference on Machine Learning*, pp. 9213–9223, PMLR, July 2021. ISSN: 2640-3498.

[50] J. Bornschein, F. Visin, and S. Osindero, "Small Data, Big Decisions: Model Selection in the Small-Data Regime," in *Proceedings of the 37th International Conference on Machine Learning*, pp. 1035–1044, PMLR, Nov. 2020. ISSN: 2640-3498.

[51] P. Nakkiran, G. Kaplun, Y. Bansal, T. Yang, B. Barak, and I. Sutskever, "Deep double descent: where bigger models and more data hurt*," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2021, p. 124003, Dec. 2021.

[52] A. Lage-Castellanos, G. Valente, E. Formisano, and F. D. Martino, "Methods for computing the maximum performance of computational models of fMRI responses," p. 25.

[53] W. Sun, A. Tagliasacchi, B. Deng, S. Sabour, S. Yazdani, and G. Hinton, "Canonical Capsules: Self-Supervised Capsules in Canonical Pose," p. 17.

[54] Y. Xiong, G. Su, S. Ye, Y. Sun, and Y. Sun, "Deeper Capsule Network For Complex Data," in *2019 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, July 2019. ISSN: 2161-4407.

[55] M. Svanera, A. T. Morgan, L. S. Petro, and L. Muckli, "An Unsupervised Deep Neural Network for Image Completion Resembles Early Visual Cortex fMRI Activity Patterns for Occluded Scenes," *bioRxiv*, p. 2020.03.24.005132, Jan. 2020.

[56] M. Svanera, M. Savardi, S. Benini, A. Signoroni, G. Raz, T. Hendler, L. Muckli, R. Goebel, and G. Valente, "Transfer learning of deep neural network representations for fMRI decoding," *Journal of Neuroscience Methods*, vol. 328, p. 108319, Dec. 2019.

[57] C. Li, B. Liu, and J. Wei, "Reconstruction of natural images from evoked brain activity with a dictionary-based invertible encoding procedure," *Neurocomputing*, vol. 456, pp. 338–351, Oct. 2021.

[58] S. Huang, L. Sun, M. Yousefnezhad, M. Wang, and D. Zhang, "Temporal Information Guided Generative Adversarial Networks for Stimuli Image Reconstruction from Human Brain Activities," *IEEE Transactions on Cognitive and Developmental Systems*, pp. 1–1, 2021. Conference Name: IEEE Transactions on Cognitive and Developmental Systems.

[59] S. Huang, W. Shao, M.-L. Wang, and D.-Q. Zhang, "fMRI-based Decoding of Visual Information from Human Brain Activity: A Brief Review," *International Journal of Automation and Computing*, vol. 18, pp. 170–184, Apr. 2021.

[60] G. St-Yves and T. Naselaris, "Generative Adversarial Networks Conditioned on Brain Activity Reconstruct Seen Images," in *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 1054–1061, Oct. 2018. ISSN: 2577-1655.

[61] R. Mukhometzianov and J. Carrillo, "CapsNet comparative performance evaluation for image classification," *arXiv:1805.11195 [cs, stat]*, May 2018. arXiv: 1805.11195.

[62] A. Kosiorek, S. Sabour, Y. W. Teh, and G. E. Hinton, "Stacked Capsule Autoencoders," *Advances in Neural Information Processing Systems*, vol. 32, pp. 15512–15522, 2019.

[63] M. Kwabena Patrick, A. Felix Adekoya, A. Abra Mighty, and B. Y. Edward, "Capsule Networks – A survey," *Journal of King Saud University - Computer and Information Sciences*, p. S1319157819309322, Sept. 2019.

[64] P. Afshar, K. N. Plataniotis, and A. Mohammadi, "Capsule Networks for Brain Tumor Classification based on MRI Images and Course Tumor Boundaries," *arXiv:1811.00597 [cs]*, Nov. 2018. arXiv: 1811.00597.

[65] R. LaLonde and U. Bagci, "Capsules for Object Segmentation," in *arXiv:1804.04241 [cs, stat]*, (Amsterdam, Netherlands), Apr. 2018. arXiv: 1804.04241.

[66] Z. Rakhimberdina, Q. Jodelet, X. Liu, and T. Murata, "Natural Image Reconstruction From fMRI Using Deep Learning: A Survey," *Frontiers in Neuroscience*, vol. 15, p. 795488, Dec. 2021.

[67] M. Van Horn, "Using Deep Learning for Visual Decoding and Reconstruction from Brain Activity: A Review," *arXiv:2108.04169 [cs]*, Aug. 2021. arXiv: 2108.04169.

[68] Y. Xiong, G. Su, S. Ye, Y. Sun, and Y. Sun, "Deeper Capsule Network For Complex Data," in *2019 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, July 2019. ISSN: 2161-4407.