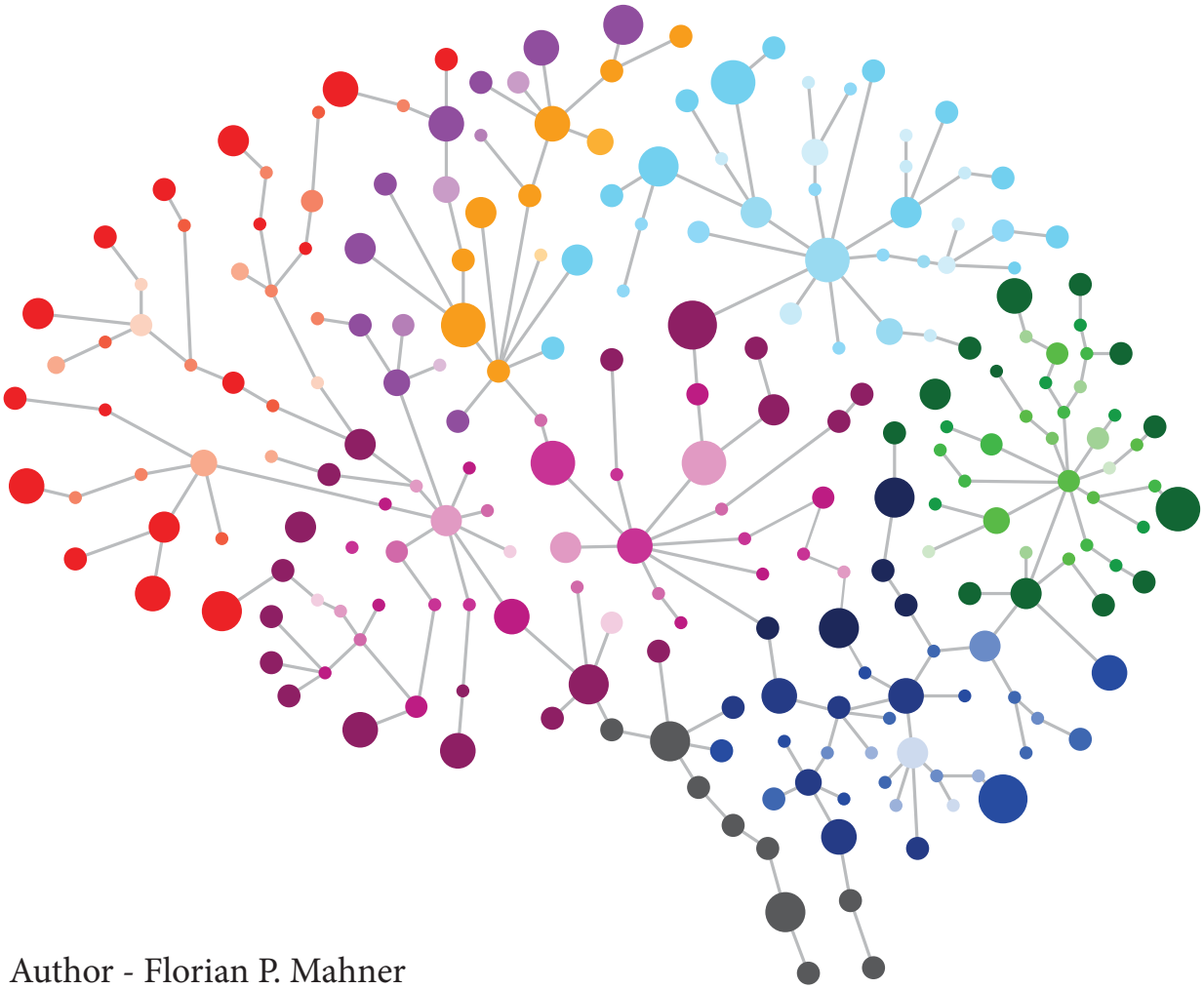
Radboud University
Master Thesis Artificial Intelligence
s1026530

Neural Decoding with Normalizing Flows



Author - Florian P. Mahner

Supervisor - Dr. Umut Güçlü
Second Reader - Dr. Yağmur Güçlütürk

Faculty of Socal Sciences
Department of Artificial Cognitive Systems
May 2021

ABSTRACT

The pattern of light that reaches the retina is projected onto patterns of neural activity via a cascade of reflexive computations alongside six interconnected brain areas in the occipital lobe called the visual ventral stream. The complex hierarchical architecture of this visual stream allows for efficient processing of visual information in the human brain. Machine learning has provided a powerful tool for encoding and decoding between brain responses and perceptual content. This thesis probes the use of generative flows as models of neural patterns from the perspective of decoding. We introduce a Neural Flow model that can bijectively map between a topographical interpretation of brain activity and the original input stimulus for a large-scale fMRI dataset of the BBC series *Doctor Who*. This approach is unique, as it inherently allows to combine neural encoding and decoding in a single neural network model. Our results show that it is possible to use this model to reconstruct simulated brain activity. For this, we applied the receptive field estimators directly on the input frames. This imitates a noise-free representation of brain data and mimics the sparsity created within the encoding scheme. Through this, we show that the model successfully achieves to extrapolate from missing spatial information. We further find through several ablation experiments on the simulation data that the model is robust to sparse training data sizes and to sparse receptive field information. Across all ablation experiments, we find strong positive correlations for the reconstructed images. The reconstructions on actual brain data found in this thesis match the previous benchmark results obtained on the same dataset. We achieve this, while at the same time including activity from only single regions of interest of brain activity, with leading performances for early visual areas V1 and V2. Overall, we find that our normalizing flow model succesfully allows to reconstruct brain activity, while contributing a unified approach to neural encoding and decoding.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

> If real is what you can feel, smell, taste and see, then real is simply electrical signals interpreted by your brain.
>
> Wachowski, A. and Wachowski, L.
> The Matrix, Warner Bros. 1999

Sensory neuroscience is closely linked to an interpretation of the brain as an information processing system. The brain represents and transforms sensory information in different visual areas to mediate cognitive function and behavior. This mechanistic interpretation of the brain questions its representational form (Decharms and Zador, 2000). How is visual information represented and stored in the brain; what information is represented; and how do different processing stages along the visual stream transform information?

1.1 BACKGROUND

The visual system consists of anatomically distinct, highly recurrent and interconnected areas (Felleman and Van Essen, 1991; Malach et al., 2002). Each processing stage involves simple neural operations, like weighted linear sums, filtering, piecewise non-linearities and response normalization (Carandini et al., 2005). These non-linear transformations strive to untangle the input space, where ecologically distinct visual stimuli can be closely entangled in pixel space (DiCarlo and Cox, 2007; DiCarlo et al., 2012). Early work by Hubel and Wiesel (1959) has shown that the brain creates an invariant representation along a series of hierarchically organized cortical areas. The cascade of simple non-linear transformations gives rise to the complex hierarchical processing of visual information, where receptive field properties increase in complexity towards downstream regions. That is, receptive fields in striate cortex are well defined, restricted to small regions in the input space and, for simple cells, highly structured, responding to narrow ranges in stimulus orientation (Hubel and Wiesel, 1962). Conversely, in inferior temporal cortex as the final stage of visual processing, receptive fields are large and respond selectively to complex visual stimuli, such as faces or objects (Bruce et al., 1981; Fujita et al., 1992). Understanding the hierarchical processing of the visual cortex therefore requires a quantitative model that can capture these complex non-linear dynamics created within the encoding scheme. Functional magnetic resonance imaging (fMRI) has provided a powerful tool for addressing the question of representation and probing models as neural information processing systems. By estimating local differences in blood flow as proxy for local neural processing, it provides the means to record large populations of neurons. Its high spatial resolution has made it possible to measure activity from many voxels (i.e. volumetric pixels) of the brain. Using these recordings, one can approach representational questions using computational models of the visual system (Churchland and Sejnowski, 1994). Current machine learning has advanced the computational methods to model visual processing. Ultimately, these

models strive to capture the information processing mechanism from sensory input received and projected by the retina to measured neural responses.

1.2 NEURAL ENCODING AND DECODING

Computational modeling of visual processing can be generally subdivided into encoding and decoding approaches (Naselaris et al., 2011). An encoding model seeks to explain how a stimulus modulates the activity of a population of neurons and how the measured activity is affected by the data recording. Inversely, a decoding model aims to predict the stimulus from the measured response as to probe the representational content of the measured activity of neural populations. Handcrafted feature detectors provided the base for early approaches in modeling neural representations. Following this, it has for instance been shown that receptive fields of visual cortical neurons that receive direct input from the thalamus can be modeled with Gabor functions (Jones and Palmer, 1987). Since the brain is nevertheless a deep recurrent neural network with complex non-linear multi-stage dynamics, these approaches have been limited by not being able to generalize above such low-level tuning properties. The recent advent of deep neural networks (DNNs) and especially convolutional neural networks (CNNs) has revolutionized many domains of AI (LeCun et al., 2015) and leveraged the success of encoding and decoding models by learning feature-detectors automatically from training data instead of hand-engineering them. CNNs are inspired by biological brains, albeit they simplify and neglect many implementational details. They are composed of simple artificial units that compute weighted linear combinations of the input passed through static non-linearities, like a sigmoid function, for instance. Stacking many of these units in different layers, multiple stages of non-linear computations occur. Analogous to the visual ventral stream, CNNs process different regions of the input locally through convolutions, where different convolutional filters act as feature detectors across different spatial locations. This form of processing of visual input has revealed a homology between the hierarchical representations that evolve in the brain and the ones of the CNN (see; Kietzmann et al., 2018; Kriegeskorte, 2015; Yamins and DiCarlo, 2016).

Decoding models

Neural decoding is generally composed of three different problems: classification, identification, and reconstruction that can be defined formally. Let $\{\boldsymbol{x}\}_{i=1}^{N}$ represent N different stimuli and l be a function that maps stimuli to categorical labels. Furthermore, let $\boldsymbol{q}_i$ denote the activity pattern evoked by a stimulus $\boldsymbol{x}_i$ on a specific trial. *Classification* means for a certain activity pattern $\boldsymbol{p}_i$, determine the label $l(\boldsymbol{x}_i)$. *Identification* asks for a pattern $\boldsymbol{p}_i$ to determine $\boldsymbol{x}_i$ out of a finite set of stimuli, where $\boldsymbol{x}_i$ is part of the set. *Reconstruction* is the most general task and challenges for a pattern $\boldsymbol{p}_i$ to reconstruct the full input $\boldsymbol{x}_i$. Whereas decoding categorical information has been shown to work sufficiently well (Carlson et al., 2003; Haxby et al., 2001; Kamitani and Tong, 2005; Kay et al., 2008; Mitchell et al., 2008), initial attempts on reconstructing the full input domain have proven to be much more challenging. The difficulty of reconstruction is in part due to the one-to-many mapping from brain activity to perceptual content (St-Yves and Naselaris, 2018), where during encoding the brain incorporates prior information into a sparse representation of

neural activity. Since the space of possible outputs in pixel space is thus much richer than the brain input, the decoder has to reveal more than the encoded information. Having accurate reconstructions then suggests that the model has captured something about the underlying information processing mechanism. Thirion et al. (2006) pioneered reconstructions from fMRI activity by modeling voxel activity in early visual areas with rotating Gabor wavelets that reveal the location and size of the spatial receptive field for each voxel. By inverting the receptive field model, the observed wavelets have successfully been reconstructed for perceived and imagined stimuli. In a similar study, Miyawaki et al. (2008) used a linearizing decoding model to reconstruct flashing geometric patterns for binary 10 × 10 images. Naselaris et al. (2009) combined a structural encoding model for early visual areas with a semantic model for more anterior regions and a prior on natural images. They then used a Bayesian framework and selected the image with the largest posterior probability out of the large set of natural images for the reconstruction. They showed that the quality of prior information has a substantial impact on the quality of the reconstructions. These approaches were further developed, for instance using independent component analysis (Güçlü and van Gerven, 2013) or a linear decoding model with stimulus domain prior (Schoenmakers et al., 2013). Most advancements have nevertheless been achieved while exploiting CNNs for brain reconstruction (Güçlütürk et al., 2017; Han et al., 2019; Horikawa and Kamitani, 2017; Horikawa et al., 2013; Kok et al., 2012; Seeliger et al., 2018b; Van Gerven et al., 2010). CNNs generalize from learning Gabor-like feature detectors in the earliest layer that are similar to the response properties of neurons in striate cortex, to also capturing increasingly complex stimulus features of the visual hierarchy in deeper layers (Zeiler and Fergus, 2014).

Whereas a large body of research has focused on static stimuli, comparable results on reconstructing dynamical input are still missing. Reconstructing dynamical input is much more challenging, since one has to additionally model the spatiotemporal dynamics of visually coherent and semantically dependent frames. Even though fMRI has a high spatial resolution (Logothetis, 2008), its sparse temporal information, high signal-to-noise ratio (SNR) and temporal delay (Friston et al., 1994) consequences that the encoded information contains more information than its measured responses. This largely complicates the reconstruction of dynamical input. Nishimoto et al. (2011) were one of the first to achieve the reconstuction of movie sequences using a motion energy encoding model combined with a large natural movie prior. To date, it has mostly been possible to only reconstruct low-level properties and silhouettes with limited natural motion characteristics (Le et al., 2021; Wen et al., 2018).

Encoding models

Brain encoding strives to estimate large-scale models of neural data. Similar to neural decoding, most successes have so far been achieved with CNNs along the visual ventral stream (Cadena et al., 2019; Cichy et al., 2016; Güçlü and van Gerven, 2014, 2015; Seeliger et al., 2018a, 2019a), even though some studies have also focussed on the dorsal (Eickenberg et al., 2017; Güçlü and van Gerven, 2017) and auditory stream (Güçlü et al., 2016). Trained to recognise objects, it has been shown that CNNs develop V1-like receptive fields in early layers of the network (Güçlü and van Gerven, 2015), while also being able to predict higher order feature representations in, for

instance, single-cell recordings in macaque IT (Cadieu et al., 2014; Khaligh-Razavi and Kriegeskorte, 2014; Yamins et al., 2014). The underlying assumption is that if, with sufficient data, the model and the brain compute similar features by capturing the non-linear transformation inside the network, then linear combinations of the model features should be sufficient to predict neural responses (Naselaris et al., 2011). This approach has revealed hierarchical correspondences between features evolving in neural network layers and visual regions of interest (ROIs) (Güçlü and van Gerven, 2015, 2017). In particular, this means that lower neural network layers better predict activity in lower level visual representations and higher layers better predict activity in more anterior cortical areas.

1.3 REPRESENTATIONS OF THE VISUAL SYSTEM

Neural encoding and decoding hence seeks to answer representational questions either by predicting the activity of neurons in different ROIs or by revealing perceptual content based on neural activity only. Among others, this raises the question on how to interpret measured brain activity to be most suitable for computational modeling. One key observation has thereby been that in the processing hierarchy of the visual system retinal input is preserved several times in the cortex, starting in the primary visual cortex V1 and once in each of the other visual areas, like V2, V3, V3a, V4 (Henschen, 1893; Holmes, 1918; Hubel and Livingstone, 1987; Hubel and Wiesel, 1959; Inouye, 1909; Teuber et al., 1960; Zeki, 1978). These separate visual representations are topographically organized and are broadly arranged in a hierarchy (DiCarlo et al., 2012), albeit with strong feedback connections. One possibility is to try to utilize the retinotopy in the visual cortex for neural encoding and decoding. The retinotopy associates specific visual field locations with cortical or voxel locations and provides an orderly connection from the visual field to cortical voxels (Engel et al., 1994; Sereno et al., 1995).

Exploiting this topological and hierarchical organization, it has recently been shown that brain decoding can be successfully cast into a form of image-to-image translation problem (Le et al., 2021). One thereby interprets the retinotopic information as a forward mapping from the visual field onto the visual cortex that maintains the topography. One further assumes that there exists a transfer function that maps the visual stimulation patterns onto the primary visual cortex and from there onwards to higher visual areas. The goal is then to model this transfer function. Since the entanglement of visual input through the transfer function is highly nonlinear, the untangling process must also be highly nonlinear. Moreover, because CNNs are homeomorphisms (i.e. the topology of the input is preserved across the transformations between neural network layers), they have been shown to be especially suited for image-to-image translations (see; Gatys et al., 2016; Isola et al., 2017; Johnson et al., 2016). One therefore further assumes that these models can also be utilized for a topographical interpretation of brain activity.

Entangling prior information

Having a representation of brain activity and a computational modeling idea available, this still leaves open on what basis one probes representational questions. Of primary importance to all the previous encoding and decoding research is there-

fore the data sample collected. Most approaches have usually been limited by the availability of sufficient fMRI data for explaining brain responses. Despite their success in computer vision and AI, CNNs usually require training on large datasets and are highly-parametrized (with several tens or hundreds of millions of parameters). Training these models directly on limited neural observations has shown to be difficult to achieve generalization of the parameters beyond the training set, which are usually multiple magnitudes larger than the data sample collected. CNNs have therefore often been trained to optimize external task objectives (goal-driven) instead of being derived from neural data directly (task-driven). However, even reaching human-level performance on these tasks does not yield insights about similarities to underlying neural processing mechanisms (Ritter et al., 2017). To gain this insight, one has to test these models on neural measurements. This can be done, for instance, by relating the features that evolve in neural network layers to brain activity from different visual areas. Notably, the similarities between for instance model trained using object categorization and response properties of the visual ventral stream that have been shown previously do not imply that the objective of training is sufficient for explaining neural responses (Van Gerven, 2017). This is because the receptive field properties that emerge are not necessarily a primer for a functional characterization (Churchland and Sejnowski, 1990). To that extent, even though DNN features and brain responses might share a significant amount of correspondence, it is questionable if these approaches will allow to provide genuine models of neural information processing in the brain. Conversely, task-driven approaches try to show correspondence solely based on neural observations, which is for neural encoding, often referred to as neural system identification (Stanley, 2005; Wu et al., 2006). Recent developments in large-scale fMRI databases have successfully allowed to train encoding models using a task-driven approach (Klindt et al., 2017; Seeliger et al., 2019a).

All of these limitations from the encoding perspective from above also cause difficulties for the reconstruction problem, where high-quality reconstructions are similarly limited by fMRI data availability. Much work has therefore focussed on using encoding models together with a Bayesian framework, where reconstruction can be formulated as an inference problem. Out of the box, it is not possible to invert the neural response function of the encoding scheme, due to the stochastic dynamics of the neural processes and hence also the stochastic dynamics of the encoding model (Dayan and Abbott, 2001). In a Bayesian setting, an encoding model strives to predict neural responses $\boldsymbol{y}_t$ for a certain stimulus $\boldsymbol{x}_t$, for different measurement time points t. We can capture the relationship between stimulus and response inside the conditional probability distribution $p(\boldsymbol{y}_t \mid \boldsymbol{x}_t)$. Decoding can be accomplished via inversion of the encoding model:

$$p(\boldsymbol{x}_t \mid \boldsymbol{y}_t) \propto p(\boldsymbol{y}_t \mid \boldsymbol{x}_t) p(\boldsymbol{x}_t) \,, \tag{1.1}$$

where the right-hand side follows directly from Bayes' theorem and $p(\boldsymbol{x}_t)$ is a stimulus prior that acts as a statistical model of the environment. Using this, much work has successfully useds CNNs for neural reconstruction via a forward encoding model (Güçlü and van Gerven, 2013; Güçlütürk et al., 2017; Han et al., 2019; Naselaris et al., 2009; Schoenmakers et al., 2013, 2015; Van Gerven et al., 2010). One key problem of this approach is that adding prior knowledge (e.g. a generative model of natural images) distorts the reconstructed input to not only reflect information

based on the neural population, but also entangles the prior representation (Glaser et al., 2020; Kriegeskorte and Douglas, 2019). A complex prior then complicates the qualitative analysis of reconstruction results, since the semantic detail is not necessarily encoded in the brain region of interest directly. Training models on neural measurements only partially overcomes this difficulty. Due to limited availability of fMRI data, obtaining high quality reconstructions is usually difficult without using a complex prior in combination with an encoding model. Training models end-to-end on neural measurements has nevertheless recently also been fruitful to reconstruct brain activity (Le et al., 2021; Shen et al., 2019).

While good encoding and decoding models have been proposed, all of the previous approaches leave open on how to connect both in a single unified model that omits entangling prior representations through Bayesian inference. Recent developments of flow-based neural network models (Dinh et al., 2014; Kingma and Dhariwal, 2018; Rezende and Mohamed, 2015) have shown that it is possible to design network architectures with invertible transformations. These transformations allow to bijectively map between two topolological spaces in analytically exact ways. In this case, one can thus think to use these models to map between the topological space of the visual input onto the topological space of recorded brain activity. This invertibility requirement on the network architecture therefore potentially allows to combine encoding and decoding in a single model without a Bayesian prior. This makes it particularly interesting to explore representational questions. This project aims to explore using flow-based models as to probe neural representations from the decoding persepective. Here, we will explictly focus on the reconstruction problem for a topographical interpretation of brain activity. We design a novel flow model that can translate between a topographical representation of brain activity in 2d space and the original input through invertible transformations. We test and question whether this model can give accurate reconstructions on simulation and on brain data. Using previous measures for brain reconstructions established in the literature, this question becomes easily quantifiable. The overall aim of the thesis is therefore to build a decoding model using flow-based models as to provide accurate reconstructions for brain activity.

1.4 STRUCTURE OF THE THESIS

This thesis is split into two parts that introduce the methodological approach taken (Chapters 2, 3) and the experimental results with brain and simulated data (Chapters 4, 5). It is thereby organized as follows:

- In Chapter 2, we conceptually introduce the methodology used for this thesis and give an overview of generative modeling and normalizing flows.
- In Chapter 3, we describe how we represent brain activity, and formally introduce a generative flow model andtt its architecture. We thereby relate flow-based modeling to the neural decoding problem of this thesis.
- In Chapter 4, we show the main results of this thesis based on simulation experiments and the topographical interpretation of brain activity.
- In Chapter 5, we discuss and summarize the main findings, give an outlook on future research ideas and conclude the thesis.

Part I

NEURAL FLOW FRAMEWORK

2 NORMALIZING FLOWS

Outline
This chapter gives an overview of the background methodology used in this thesis. In Section 2.1, we will give a general introduction into probability distributions and the maximum likelihood framework. In Section 2.2, we will then relate maximum likelihood estimation to generative flows as flexible tool for modeling rich distributions.

2.1 PROBABILITY DISTRIBUTIONS

In machine learning (ML) and statistics one is usually interested in estimating probability densities of random variables. We will focus on generative modelling, where we collect a dataset of independent and identically distributed (i.i.d) training samples $\boldsymbol{X} = \{\boldsymbol{x}^1, \boldsymbol{x}^2, \dots, \boldsymbol{x}^N\}$ living in a space $\mathcal{X} = \mathbb{R}^D$. We assume that the data originates from an unknown distribution $\boldsymbol{x} \sim p^*$. The goal of any generative model is then to approximate this data distribution through the collection of samples $\boldsymbol{X}$, which is usually done using parametric approximations. Each parametric probability distribution has a unique vector of parameters associated with it $\boldsymbol{\theta} = [\theta_1, \theta_2, \dots, \theta_k]$ that allow to evaluate a family of distributions $\{p(\cdot\,; \boldsymbol{\theta}) \mid \boldsymbol{\theta} \in \Theta\}$ within a euclidean parameter space Θ. We wish to find parameters, such that our model distribution minimizes some notion of difference D to the actual data distribution:

$$\min_{\boldsymbol{\theta} \in \Theta} D(p_{\boldsymbol{\theta}}, p^*) \,. \tag{2.1}$$

One method for finding suitable parameters describing a probability distribution is maximum likelihood estimation (MLE). Since the sampled data points are i.i.d, the likelihood of the training dataset $\boldsymbol{X}$ under $p_{\boldsymbol{\theta}}$ equals the product of the univariate density functions:

$$p(\boldsymbol{X}; \boldsymbol{\theta}) = \frac{1}{N} \prod_{\boldsymbol{x} \in \boldsymbol{X}} p(\boldsymbol{x}; \boldsymbol{\theta}) \,. \tag{2.2}$$

The goal of MLE is to find a point estimator of the parameters $\hat{\boldsymbol{\theta}}$ that maximizes the likelihood function:

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta} \in \Theta} p(\boldsymbol{X}; \boldsymbol{\theta}) \,. \tag{2.3}$$

Instead of taking the product over all training samples, it is convenient to construct the problem in logarithmic space. Taking the logarithm has the property of not changing the extrema of the likelihood function and transforms the product into a sum:

$$\log p(\boldsymbol{X}; \boldsymbol{\theta}) = \frac{1}{N} \sum_{\boldsymbol{x} \in \boldsymbol{X}} \log p(\boldsymbol{x}; \boldsymbol{\theta}) \,. \tag{2.4}$$

Ultimately, our goal of modelling is to find the set of parameters that minimizes our difference D, such that $p(\boldsymbol{x};\boldsymbol{\theta}) \approx p^*(\boldsymbol{x})$. We can generally characterize the gap between the two probability distributions in terms of the Kullback-Leibler (KL) divergence (Kullback and Leibler, 1951):

$$D_{KL}\left[p^*(\boldsymbol{x}) \| p(\boldsymbol{x};\boldsymbol{\theta})\right] = \int_{-\infty}^{\infty} p^*(\boldsymbol{x}) \log \frac{p^*(\boldsymbol{x})}{p(\boldsymbol{x};\boldsymbol{\theta})} d\boldsymbol{x} \tag{2.5}$$

$$= \mathbb{E}_{\boldsymbol{x}\sim p^*}\left[\log \frac{p^*(\boldsymbol{x})}{p(\boldsymbol{x};\boldsymbol{\theta})}\right] \tag{2.6}$$

$$= \mathbb{E}_{\boldsymbol{x}\sim p^*}\left[\log p^*(\boldsymbol{x})\right] - \mathbb{E}_{\boldsymbol{x}\sim p^*}\left[\log p(\boldsymbol{x};\boldsymbol{\theta})\right] . \tag{2.7}$$

It is straightforward to observe that obtaining the exact KL divergence is not possible, since it requires knowledge of the underlying distribution p^*, which is what we are trying to approximate. We can nevertheless further see that the left term of Equation (2.7) is the entropy $\mathbb{H}\left(p^*\right)$ of the true distribution, which is constant, as it does not depend on the parameters $\boldsymbol{\theta}$. By the law of large numbers we further know that:

$$\lim_{N\to\infty} -\frac{1}{N}\sum_{i}^{N} \log p\left(\boldsymbol{x}^{(i)};\boldsymbol{\theta}\right) = -\mathbb{E}_{\boldsymbol{x}\sim p^*}\left[\log p(\boldsymbol{x};\boldsymbol{\theta})\right] , \tag{2.8}$$

which says that with increasing sample size the empirical mean converges to the expected value. Combining the above, we obtain:

$$D_{KL}\left[p^*(\boldsymbol{x}) \| p(\boldsymbol{x};\boldsymbol{\theta})\right] \approx -\log p(\boldsymbol{X};\boldsymbol{\theta}) + \mathbb{H}\left(p^*\right) . \tag{2.9}$$

What we see is that minimizing the KL-Divergence between our estimate and real data distribution is the same as minimizing the negative log-likelihood (NLL), a criterion often used in ML. In other words, it is equivalent to maximizing the likelihood function from Equation 2.2. This gives us a proxy for estimating the underlying distribution through MLE without having direct access. Importantly, we often use this log-likelihood as a quantitative measure for generative models. Nevertheless, for discrete data (e.g. 8-bit images), the continuous (differential) entropy is negative infinity and $p(\boldsymbol{x};\boldsymbol{\theta})$ alone does not have an interpretation as a probability density. Obtaining probabilities requires integrating over some subregion $\boldsymbol{\Omega} : P(\boldsymbol{\Omega}) = \int_{\boldsymbol{\Omega}} p(\boldsymbol{x};\boldsymbol{\theta}) d\boldsymbol{x}$. Adding a fixed noise $\tilde{\boldsymbol{x}} = \boldsymbol{x} + \boldsymbol{\delta}$ term dequantizes the data and can simulate this integration step (Theis et al., 2015):

$$\log \int_{-\infty}^{\infty} p\left(\tilde{\boldsymbol{x}};\boldsymbol{\theta}\right) p(\boldsymbol{\delta}) d\boldsymbol{\delta} \geq \mathbb{E}_{\boldsymbol{\delta}}\left[\log p\left(\tilde{\boldsymbol{x}};\boldsymbol{\theta}\right)\right] \approx \log p\left(\tilde{\boldsymbol{x}};\boldsymbol{\theta}\right) , \tag{2.10}$$

where $\boldsymbol{\delta}$ is drawn from the noise distribution $p_{\boldsymbol{\delta}}$, in our case $\boldsymbol{\delta} \sim \mathcal{U}(0, a)$ with a being the discretization level of the data. In the case of continuous data, we then minimize the following negative log-likelihood:

$$-\log p(\boldsymbol{X};\boldsymbol{\theta}) = -\frac{1}{N}\sum_{\tilde{\boldsymbol{x}}\in\boldsymbol{X}} \log p(\tilde{\boldsymbol{x}};\boldsymbol{\theta}) + c , \tag{2.11}$$

where $c = -M \cdot \log a$ and M is the dimensionality of $\tilde{\boldsymbol{x}}$.

2.2 GENERATIVE FLOWS

One intuition in statistics is describing the set of observed i.i.d samples as part of a generative process. This has in part been reasoned by the fact that the observed high dimensional data is usually concentrated on a much lower-dimensional manifold with intrinsic coordinates $\boldsymbol{z}$ in a space $\mathcal{Z} \in \mathbb{R}^d$ that are unobserved. We can characterize these so-called latent variable models through the joint distribution:

$$p(\boldsymbol{x}, \boldsymbol{z}; \boldsymbol{\theta}) = p(\boldsymbol{x}|\boldsymbol{z}; \boldsymbol{\theta}) p(\boldsymbol{z}; \boldsymbol{\phi}) \, . \tag{2.12}$$

where $\boldsymbol{\phi}$ are the parameters describing the prior distribution. The generative modelling procedure is then the following:

$$\boldsymbol{z} \sim p(\boldsymbol{z}; \boldsymbol{\phi}) \tag{2.13}$$
$$\boldsymbol{x} \sim p(\boldsymbol{x}|\boldsymbol{z}; \boldsymbol{\theta}) \, . \tag{2.14}$$

In this case, the joint distribution is characterized by the product space $\mathcal{X} \times \mathcal{Z}$. However, obtaining the marginal likelihood $p(\boldsymbol{x}; \boldsymbol{\theta}) = \int_{\boldsymbol{z}} p(\boldsymbol{x}, \boldsymbol{z}; \boldsymbol{\theta}) \, d\boldsymbol{z}$ quickly becomes intractable for higher dimensionalities, since it involves integrating over all source of variation of $\boldsymbol{z}$. Alternatively, we can define $p(\boldsymbol{x})$ using a much simpler distribution $p(\boldsymbol{z})$ with the change of variables formula. Let $f : \mathcal{X} \to \mathcal{Z}$ denote the diffeomorphism from the data space onto the latent space. We can then compute integrals over $\boldsymbol{z}$ as an integral over $\boldsymbol{x}$ in the following way:

$$\int_{\boldsymbol{z}} p_{\boldsymbol{Z}}(\boldsymbol{z}) d\boldsymbol{z} = \int_{\boldsymbol{x}} p_{\boldsymbol{Z}}(\boldsymbol{f}(\boldsymbol{x})) \left| \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}} \right| d\boldsymbol{x} \tag{2.15}$$
$$= \int_{\boldsymbol{x}} p_{\boldsymbol{X}}(\boldsymbol{x}) d\boldsymbol{x} \tag{2.16}$$
$$= \int_{\boldsymbol{z}} p_{\boldsymbol{X}}(\boldsymbol{f}^{-1}(\boldsymbol{z})) \left| \frac{\partial \boldsymbol{f}^{-1}}{\partial \boldsymbol{z}} \right| d\boldsymbol{z} \, . \tag{2.17}$$

Normalizing flows (Rezende and Mohamed, 2015) explore this change of densities as a flexible tool for generating rich distributions. Let us first generally consider the transformation of densities. Let $f : \mathbb{R}^d \to \mathbb{R}^d$ denote a bijective, also called invertible, mapping between two probability densities, where $f^{-1} = g$. The approach is to start with a simple, tractable density $\boldsymbol{z} \sim p(\boldsymbol{z}; \boldsymbol{\phi})$, like an isotropic normal distribution $\boldsymbol{z} \sim \mathcal{N}(\mathbf{o}, \boldsymbol{I})$, for instance. For probability densities, we typically want to do two distinct operations: sampling and scoring. Sampling is trivial. We first sample from our tractable density $\boldsymbol{z}$ and then transform this random variable using our mapping $\boldsymbol{x} = f(\boldsymbol{z})$, with $\boldsymbol{z} = g(\boldsymbol{x}) = f^{-1}(\boldsymbol{x})$. Scoring typically means evaluating the density for $p_{\boldsymbol{X}}(\boldsymbol{x})$, where $p_{\boldsymbol{Z}}(\boldsymbol{z})$ is transformed using the change of variables formula and allows us to obtain the marginal likelihood for $p(\boldsymbol{x})$:

$$p_{\boldsymbol{X}}(\boldsymbol{x}) = p_{\boldsymbol{Z}}(\boldsymbol{z}) \left| \det \left(\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{z}} \right) \right| \, , \tag{2.18}$$

where $\left| \det \left(\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{z}} \right) \right|$ is the absolute determinant of the Jacobian matrix of partial derivatives. We can compose arbitrarily complex densities, by stacking a chain of K simple transformations $f = f_1 \circ f_2 \circ \ldots \circ f_K$:

$$\boldsymbol{z} \overset{\boldsymbol{f}_1}{\longleftrightarrow} \boldsymbol{h}_1 \overset{\boldsymbol{f}_2}{\longleftrightarrow} \boldsymbol{h}_2 \cdots \overset{\boldsymbol{f}_K}{\longleftrightarrow} \boldsymbol{x} \, , \tag{2.19}$$

and repeatedly applying (2.18) for each intermediate transformation. The probability density then becomes:

$$\log p_X(\boldsymbol{x}) = \log p_Z(\boldsymbol{z}) + \log \left| \det \left(\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{z}} \right) \right| \tag{2.20}$$

$$= \log p_Z(\boldsymbol{z}) + \sum_{t=1}^{K} \log \left| \det \left(\frac{\partial \boldsymbol{h}_t}{\partial \boldsymbol{h}_{t-1}} \right) \right| . \tag{2.21}$$

Notably, this only holds for specific kinds of transformation with known Jacobian determinant. The term *normalizing flow* thereby describes two distinct properties of the transformation. *Normalizing* means that with the change of variables the probability density after applying the invertible transformation is also normalized. *Flow* here means that we can arbitrarily stack invertible transformations and that their composition results in a more complex and invertible transformation. By suitable design of transformations, both likelihood evaluation and sampling can be performed efficiently. Consequently, a flow model can be trained efficiently to maximize the likelihood of the observed dataset (i.e. MLE). In the next chapter, we describe how we represent our brain data and thereby also define the model and the distinct transformations with tractable Jacobian determinant in closer detail.

3 NEURAL FLOW MODEL

Outline
In this chapter we define how we represent the brain data used in this thesis and provide the architectural details of the normalizing flow model described previously. In Section 3.1, we describe the topographical representation of brain activity in the 2d plane. In Section 3.3, we define each layer of the model and describe how each transformation accounts for the change in probability density through the Jacobian determinant. In Section 3.2, we introduce the fMRI data used and in Section 3.4 we provide the exact training specifics for the model.

3.1 TOPOGRAPHICAL REPRESENTATIONS OF THE VISUAL SYSTEM

The goal of this work is to exploit the topographical organization of the visual stream, where the topology of retinal input is preserved within distinct visual areas (Henschen, 1893; Inouye, 1909). Topology hereby means that the orderly array of retinal locations is preserved in the projection from the retina onto the striate cortex. The underlying idea of the approach used in this thesis is to redefine neural reconstruction as an image-to-image problem, which has been first introduced in Le et al. (2021). This definition is especially useful, since the ML literature for paired image-to-image translation is vast (see; Gatys et al., 2016; Isola et al., 2017; Johnson et al., 2016) and high-quality translations between any two domains of interest are possible. We can generally interpret the brain as 3d topological spheres of distinct visual areas (e.g. V1, V2, V3, MT, FFA), where the topology is maintained in each area. Out of the box, there is no direct way of applying a convolutional architecture on the individual brain volumes directly, since the representation of the retina is distorted by the geometry of the cortex and visual input is non-uniformly sampled, through for instance a selective sampling of the binocular visual field.

Formally, we therefore want to find a mapping $f : \mathbb{R}^3 \rightarrow \mathbb{R}^2$ from a 3d coordinate (v_1, v_2, v_3) of voxels of measured brain activity onto its topographical representation in 2d cartesian plane (x, y). This map defines the coordinates that drive the highest response in a voxel and describes a spatial correspondence between a visual input projected on the sensory surface and its measured brain activity. Due to the topographical organization, there exists a point in the 2d plane that drives the highest response of a voxel:

$$\mathrm{RF}\left(v_1, v_2, v_3\right) = \left(x, y\right), \tag{3.1}$$

where RF is the receptive field map from the visual cortex onto the image space. This definition is limited to point-like receptive fields, where each location in the image maps onto a single voxel. Importantly, this transformation is a homeomorphism (i.e. a continuous, bijective mapping between two topological spaces), since visual areas are topographically organized. This means that with the transformation the topology of brain responses is preserved in the 2d plane. Since the central part of the

Figure 3.1: Sample images from the training set. Each original ground truth (GT) input frame has a corresponding topographical projection of voxel activity in pixel space (Input) that is used for reconstruction.

retina, which includes the fovea, is substantially overrepresented in the topographic map of the retina in the striate cortex, RF distorts the overall structure of the input image. With fMRI we are nevertheless not able to measure direct responses, but only differences in the magnetic properties of oxygenated and deoxygenated blood. Let $g(\cdot)$ therefore denote the mapping from neural responses onto the measured BOLD responses and $N(x, y)$ the total number of voxels that map from the 3d to the 2d plane. We can then account for multiple voxels having the same or a similar receptive field in the input plane by simply summing the activity:

$$g(x, y) = \frac{1}{N(x, y)} \sum_{v_1, v_2, v_3; \mathrm{RF}(v_1, v_2, v_3) = (x, y)} g(v_1, v_2, v_3) \,, \tag{3.2}$$

and can generally express the summation inside a weight matrix $\boldsymbol{W}$:

$$g\left(x, y\right) = \sum_{v_1, v_2, v_3} \boldsymbol{W}^{x,y}_{v_1, v_2, v_3} g\left(v_1, v_2, v_3\right) . \tag{3.3}$$

There are two-principled ways to determine $\boldsymbol{W}$. One can either use fixed receptive field locations obtained from a classical ring and wedge retinotopy session, or learn the receptive field maps together with the network. Since only slight improvements have been shown in Le et al. (2021) when learning the weights, we use in this thesis fixed receptive field estimators. Importantly, the linear transportation map from Equation 3.1 does not allow for accurate reconstructions, since the mapping from visual inputs onto brain responses is highly non-linear. Since spatial correspondences are nevertheless maintained within the non-linear mapping onto the cortex (which is a direct property of homeomorphic spaces), we can transform the linear maps in 2d pixel space back onto the image space using any non-linear image-to-image translation model. The task of the model is then to learn to abstract from the low-level topography onto generating rich global feature with meaningful visual semantics of natural images.

3.2 FMRI DATASET

We use the fMRI data recorded from a single-participant watching 30 episodes from the BBC series *Doctor Who* that was published in (Seeliger et al., 2019b) (see Figure 3.1 for a collection of frames and their topographical representation). Detailed experimental procedures are described extensively in the original study. In brief, 3 T whole volume brain data was recorded (TR = 0.7 s, voxel size = $2.4 \times 2.4 \times 2.4$ mm^3, 64 transversal slices) with the subject fixating the center of the screen. fMRI recordings were repeated for several runs of the training and test set. The train set was repeatedly

shown over 121 runs and the test set over 7 runs. Each trial of the test set was averaged across 10 repetitions for model evaluation. In total, this resulted in 118.417 whole-brain volumes for the training set and 1.034 volumes for the test set.

Receptive field estimation

Receptive fields for all regions of interests (ROIs) where estimated end-to-end using Neural Information Flow (Seeliger et al., 2019a). The underlying idea is to estimate neural models from neural activity only. The approach models observable signals as 3d convolutions that map visual inputs onto encoded responses, while at the same time accounting for the causal interactions between neural populations in distinct brain regions. Each layer of the convolutional architecture is coupled through low-rank observation models to the measured responses in a specific brain region. The model combines convolutional architectures that describe the topographical processing of the visual stream with observation models that couple intermediate convolutional activities to observed responses. The observation model is thereby factorized into spatial, temporal and feature receptive field maps for individual neuronal populations. The receptive field location (x, y) in cartesian coordinates was then estimated as the highest response in the low-rank decomposition of the spatial receptive field maps. Processing of visual information from retinal ganglion cells and the LGN is accounted with a linear $(3 \times 3 \times 1)$ convolutional filter that learns a purely spatial transformation prior to its afferent connection to V1.

Preprocessing

We standardize the brain data for the train and test set to have zero mean and unit variance across time based on the statistics of the training set. We downsample the video frames from the *Doctor Who* dataset both spatially ($96 \times 96 \times 3$) and temporally. We downsample temporally such that the frames in the training set match the temporal resolution of the fMRI recordings, with a rate of 0.7 seconds per frame. For each ROI (e.g. V1, V2, V3), we then map the activity from 3d space onto its topographical representation in cartesian plane based on the retinotopic map. For each ROI, we stack three different time channels (TRs). This results for each region in a tensor of shape $96 \times 96 \times 3$. We warp each frame using a fish eye transformation that mimics the spatial sampling properties of the retina (Bashivan et al., 2019), where visual acuity is sharpest at the central part of the fovea.

3.3 MODEL DEFINITION AND INVERTIBLE TRANSFORMATIONS

We modify the architecture of the Glow model from Kingma and Dhariwal (2018) that builds upon previous work on normalizing flows proposed in (Dinh et al., 2014, 2016). The architecture of the generative flow network consists of separate building blocks, where each transformation has a known Jacobian determinant to account for the change in volume density. The architecture is based on distinct building blocks that are combined into a multi-scale architecture.

The Glow model uses a data dependent normalization, called *activation normalization* (actnorm). Each actnorm layer has a trainable scale and bias parameter that is initialized such that in the first forward pass the post-norm activations of the

corresponding batch have zero mean and unit variance per channel. Afterwards, they are treated as regular parameters of the model. Actnorm has been introduced by (Kingma and Dhariwal, 2018) and works similar to Batch Normalization (Ioffe and Szegedy, 2015), but works also for a mini-batch size of 1, hence also allows to train highly parametrized networks.

The actnorm layer is followed by a 1×1 *convolution* operation that replaces discrete permutation operations and to reverse the ordering of channels. This allows the ordering of all channels to alter all the data dimensions throughout the network. The weight matrix of the convolution is initialized as a random rotation matrix, which results in the following log-determinant for a $h \times w \times c$ tensor of hidden layer outputs $\boldsymbol{h}$, $c \times c$ weight matrix $\boldsymbol{W}$ and a 2d convolution operation O:

$$\log \left| \det \left(\frac{dO(\boldsymbol{h}; \boldsymbol{W})}{d\boldsymbol{h}} \right) \right| = h \cdot w \times \log |\det(\boldsymbol{W})| \,. \tag{3.4}$$

Inverting the convolution requires to compute the inverse $\boldsymbol{W}^{-1}$ of the rotation matrix. For most considerations of the number of channels computing the inverse is still computationally feasible. One can further decrease the cost of computing $\det(\boldsymbol{W})$ by parametrizing $\boldsymbol{W}$ in its LU decomposition. We refer the interested reader to the original paper for additional details.

A powerful bijective transformations is the *affine coupling layer* that has been introduced in (Dinh et al., 2014). Affine coupling layers transform any input $\boldsymbol{x}$ of dimension D based on translation and scaling operations. Let $\boldsymbol{x} = (\boldsymbol{z}_1, \boldsymbol{z}_2)$ denote two disjoint subset of $\boldsymbol{x}$ with $\boldsymbol{z}_1 \in \mathbb{R}^d$ and $\boldsymbol{z}_2 \in \mathbb{R}^{D-d}$, $d < D$. The affine coupling layer then takes the following form:

$$\boldsymbol{y}_1 = \boldsymbol{z}_1 \tag{3.5}$$

$$\boldsymbol{y}_2 = \left[\exp\{s(\boldsymbol{z}_1; \boldsymbol{\theta}_s)\} \odot \boldsymbol{z}_2 + t(\boldsymbol{z}_1; \boldsymbol{\theta}_t) \right] \,, \tag{3.6}$$

where $\odot$ denotes the Hadamard product. The first subset is fed into two arbitary (non-linear) transformations with outputs $s(\boldsymbol{z}_1; \boldsymbol{\theta}_s)$ and $t(\boldsymbol{z}_1; \boldsymbol{\theta}_t)$. These are then scaled by $\exp(\cdot)$ and shifted by $(\cdot) + t(\boldsymbol{z}_1; \boldsymbol{\theta}_t)$. Both are functions from $\mathbb{R}^d \to \mathbb{R}^{D-d}$. One required property of the transformation for normalizing flows is invertibility, which is easily done for the coupling layer:

$$\boldsymbol{z}_1 = \boldsymbol{y}_1 \tag{3.7}$$

$$\boldsymbol{z}_2 = (\boldsymbol{y}_2 - t(\boldsymbol{y}_1; \boldsymbol{\theta}_t)) \odot \exp\{-s(\boldsymbol{y}_1; \boldsymbol{\theta}_s)\} \,. \tag{3.8}$$

The Jacobian of this transformation is a lower triangular matrix:

$$\boldsymbol{J} = \begin{bmatrix} \mathbb{I}_d & \mathbf{0} \\ \frac{\partial \boldsymbol{y}_2}{\partial \boldsymbol{z}_1} & \operatorname{diag}\left(\exp\{s\left(\boldsymbol{z}_1; \boldsymbol{\theta}_s\right)\}\right) \end{bmatrix} , \tag{3.9}$$

and the determinant is therefore easy to compute through the product along the diagonal:

$$\det(\boldsymbol{J}) = \prod_{j=1}^{D-d} \exp\{s\left(\boldsymbol{z}_1; \boldsymbol{\theta}_s\right)\}_j = \exp\left\{ \sum_{j=1}^{D-d} s\left(\boldsymbol{z}_1; \boldsymbol{\theta}_s\right)_j \right\} \,. \tag{3.10}$$

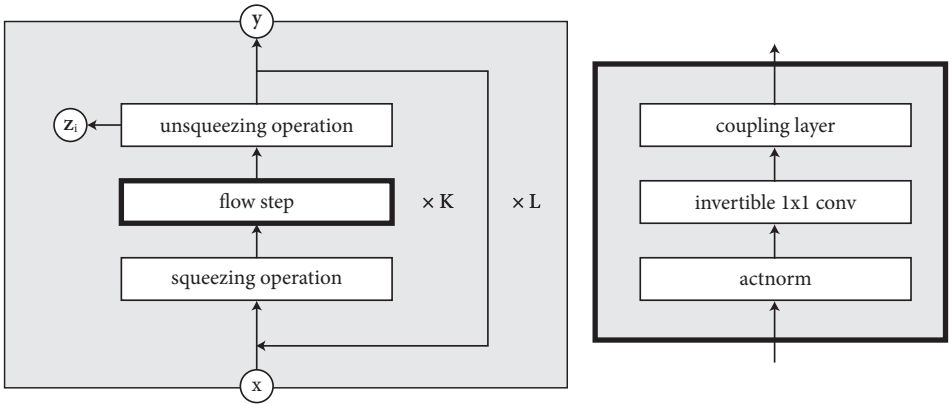


Figure 3.2: Multi-scale architecture of the Neural Flow model. K flow steps are stacked in L flow blocks and intermediate representations $\boldsymbol{z}_i$ are gaussianized.

Interestingly, computing the Jacobian determinant of this transformation does not involve computing the Jacobian determinant of the transformations s or t. These functions can therefore be made arbitrarily complex. One suitable way is making them deep neural networks.

These three components (*actnorm*, 1×1 *convolution*, *affine coupling*) describe one step of flow. Each flow step is preceded by a squeezing operation that trades spatial size for feature channels. The squeezing operation transforms an $s \times s \times c$ tensor into a $\frac{s}{f} \times \frac{s}{f} \times f^2 c$ tensor, where f is a squeezing factor. After a block of flow steps, the original Glow model factors out half of the channel dimensions (splitting operation). Each splitting operation gaussanizes the intermediate units $\boldsymbol{z}_i$ that are not factored out. For MLE based on a gaussian prior distribution at the last layer output ($\boldsymbol{y}$), gaussianizing intermediate representations helps distributing the loss throughout the network (Dinh et al., 2016; Lee et al., 2015). We will now describe how we modify the original architecture such that we can use it for reconstruction.

Modeling topographical representations of brain activity

One property of the original Glow model is that after each block consisting of a squeezing operation, a flow step and a splitting operation the amount of feature channels is doubled and the spatial dimensions are halved. For an RGB image with three color channels, this creates an output consisting of six channels after each block. When estimating a low-dimensional latent representation this is useful, as it decreases computational complexity by trading spatial dimensions for feature maps, hence reducing the number of convolution operations. Since we want to use the model to translate between our two image domains, this definition does not allow to do so for single frames. We therefore remodel the architecture of the Glow network for our purpose. Since we use this model for representations of brain activity, we name it Neural Flow, though in principle it can be used to translate between any two image domains. The important difference of the Neural Flow model is that after each flow step, we unsqueeze the input to restore the full spatial output again (see Figure 3.2). This has multiple advantages. By trading spatial dimensions for the number of channels, we effectively reduce the complexity of computation within the flow step and throughout the coupling layer. In our experiments, we test the effect of

different squeezing factors (see Section 4.3). After each unsqueezing operation, we still gaussianize the corresponding intermediate representation, but omit factoring out half of the channel dimensions to maintain the full input shape, hence omit using the splitting operation. Hence, this still supports maximum likelihood training by distributing the loss through the network. One of these forward passes describes one block of our Neural Flow model that is stacked L times, where each block contains K flow steps. With this model definition, we can stack an arbitrarily large number of flow steps and flow blocks, while maintaining the dimensionality of the input.

Flow models trained with maximum likelihood estimation still require a tractable density as prior distribution (e.g. a normal distribution in our case). This is required to evaluate the model likelihoods under the change of variables formula. Here, the latent distribution is not on a much lower dimensional manifold than our input brain responses, but an actual translated image. We still describe each pixel using a normal distribution with trainable scale and bias parameters. We enforce different additional loss components besides log-likelihood to force the model to learn to reconstruct visual semantics. The different components are explained in detail in the next section.

3.4 TRAINING PROCEDURE

We train the entire model using the Adam optimizer (Kingma and Ba, 2014) with parameters $\eta = 1e{-}4, \beta_1 = 0.9, \beta_2 = 0.999$. The primary objective of the Glow network is to maximize the likelihood function or to minimize the negative log-likelihood. We train using a combination of the log-likelihood scaled by the number of pixels (bits-per-dimension), pixel wise and feature loss to ensure that the reconstructions match the targets on a low and high feature level. The bits-per-dimension are defined as:

$$\mathcal{L}_{\text{bpd}} = \frac{1}{C} - \log p(\boldsymbol{X}; \boldsymbol{\theta}) \,, \tag{3.11}$$

where $C = \log_2 h \cdot w \cdot c$. The resulting values can be interpreted as the number of bits that a compression scheme based on this model would need to compress every RGB color value in the reconstruction. The pixel loss is simply the $\mathcal{L}_2$ loss weighted by λ_{pix}:

$$\mathcal{L}_{\text{pix}} = \lambda_{\text{pix}} \, \mathbb{E}\left[\| \boldsymbol{y} - f(\boldsymbol{x}) \|^2 \right] \,. \tag{3.12}$$

The loss to match higher level semantics in the reconstruction is a combination of a weighted feature and temporal loss:

$$\mathcal{L}_{\text{feat}} = \lambda_{\text{vgg}} \underbrace{\mathbb{E}\left[\| \xi(\boldsymbol{y}) - \xi\left(f(\boldsymbol{x})\right) \|^2 \right]}_{\mathcal{L}_{\text{vgg}}} , \tag{3.13}$$

where $\xi(\cdot)$ is the layer 15 and layer 22 output of the pre-trained VGG-16 network (Simonyan and Zisserman, 2014) trained on ImageNet for object categorization. The total training objective is then simply the sum of the three losses $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{bpd}} + \mathcal{L}_{\text{pix}} + \mathcal{L}_{\text{feat}}$. For all experiments, we empirically tested different weighting factors and finally set $\lambda_{\text{pix}} = 200$ and $\lambda_{\text{vgg}} = 200$ for each frame.

Problem statement

Having now described the brain data used in this thesis and the normalizing flow model, we can now formally summarize the methodological target of this work. Let $\boldsymbol{X} = \{\boldsymbol{x}^1, \ldots, \boldsymbol{x}^N \mid \boldsymbol{x}^i \in \mathbb{R}^{h \times w \times c}\}$ be a set of sensory inputs and $\boldsymbol{Y} = \{\boldsymbol{y}^1, \ldots, \boldsymbol{y}^N \mid \boldsymbol{y}^i \in \mathbb{R}^q\}$ a set of measured brain responses. During encoding, we want to make sense of a brain responses given its sensory inputs, which is fully be characterized by the conditional probability distribution $p(\boldsymbol{y}|\boldsymbol{x};\boldsymbol{\theta})$. Using Bayes' Theorem we could generally characterize decoding as the inversion of the encoding model:

$$p(\boldsymbol{x}|\boldsymbol{y};\boldsymbol{\theta}) \propto p(\boldsymbol{y}|\boldsymbol{x};\boldsymbol{\theta})p(\boldsymbol{x};\boldsymbol{\phi}) \,, \tag{3.14}$$

but we want to omit entangling a prior representation $p(\boldsymbol{x};\boldsymbol{\phi})$. Interestingly, when assuming a uniform prior distribution, the mode of the posterior and the MLE are the same, since under this prior structure the posterior and likelihood distribution are proportional. Here, we therefore use MLE to learn an invertible transformation from sensory inputs onto brain responses and vice versa:

$$\boldsymbol{y} = f_{\boldsymbol{\theta}}(\boldsymbol{x}) \tag{3.15}$$

$$\boldsymbol{x} = f_{\boldsymbol{\theta}}^{-1}(\boldsymbol{y}) \,, \tag{3.16}$$

where $f_{\boldsymbol{\theta}} : \mathbb{R}^{h \times w \times c} \to \mathbb{R}^q$ is a neural network and in our case the Neural Flow model described in the previous section. We first transform our sensory inputs into pixel space using a receptive field mapping and obtain a set of voxel images $\boldsymbol{V} = \{\boldsymbol{v}^1, \ldots, \boldsymbol{v}^N \mid \boldsymbol{y} \in \mathbb{R}^{h \times w \times c}\}$, where h and w are the size of our brain image in pixel space and c is the time range of fMRI data included. We then want to learn the sequence of transformations that can transform the probability density of brain response in pixel space $p_{\boldsymbol{\theta}}(\boldsymbol{v})$ into the density of sensory inputs $p_{\boldsymbol{\theta}}(\boldsymbol{x})$. Our model definition transforms the input tensor $\boldsymbol{v}$ back into pixel space by enforcing different loss components on the reconstruction. Our model thereby achieves to maintain the full spatial information within the transformation.

Part II

NEURAL DECODING

4

RESULTS

Outline
This chapter shows the main results of this thesis. In Section 4.1, we give results for simulated brain recordings based on receptive field estimators applied directlty on the original input frames. In Section 4.2, we test this model across several ablation experiment. In Section 4.3, we look into the effect of different squeezing factors on the reconstruction performances on the simulation data. In Section 4.4, we then show the main reconstruction results from brain data for different regions of interest.

4.1 SIMULATING BRAIN RECORDINGS

The brain recordings based on fMRI activity can be seen as a sparse representation of the visual input received and projected by the retina. Reconstructing frames or entire video sequences from this lower-dimensional manifold of data therefore needs to partially reconstruct the encoding scheme of the brain. Le et al. (2021) has provided a proof-of-concept that a topographical interpretation of brain activity can give meaningful reconstructions. We therefore first tested whether our normalizing flow model can be generally used to translate between a representation of brain activity in pixel space onto the original frame. To do this, we simulated brain data by sampling a noise free representation of brain activity and applied the receptive field estimators directly on the input frames. The procedure of sampling these masks from the original targets is identical to the mapping from brain responses onto visual space described in the previous chapter. The key difference of the simulation is that instead of constructing the voxel images in pixel space based on neural measurements, we base the representation on the actual target images to be reconstructed. The exact procedures was as follows. For each 96×96 image in RGB, we only used the 879 receptive field locations to mask the activity for an individual frame. All pixels that do not have a receptive field location in the input (i.e. the visual field) are zeroed out. This form of simulation creates, in analogy to the encoding scheme of the brain, a sparser representation of the input space, where spatial information from only $\approx 9\%$ ($\frac{879}{96^2}$) of the pixels is available. This enforces the model to extrapolate missing pixel information for the reconstruction. Each forward pass of the model then reconstructs from a sparse simulated voxel representation onto the original input frame. We used the frame rate that matches with the temporal resolution of the fMRI recordings (e.g. 0.7 seconds per frame, resulting in a total of 118.417 samples). We use this procedure to mask the entire dataset and train a Neural Flow model with $L = 10$ flow steps and $K = 2$ blocks. Corresponding results are then shown based on the average performances on the hold-out test set.

We can qualitatively see that the Neural Flow model can clearly reconstruct missing spatial information beyond the training set (see Figure 4.1). Due to the overrepresentation of the fovea, the reconstructions are most accurate at the central part of the image and become more blurry towards the sides. These results visually

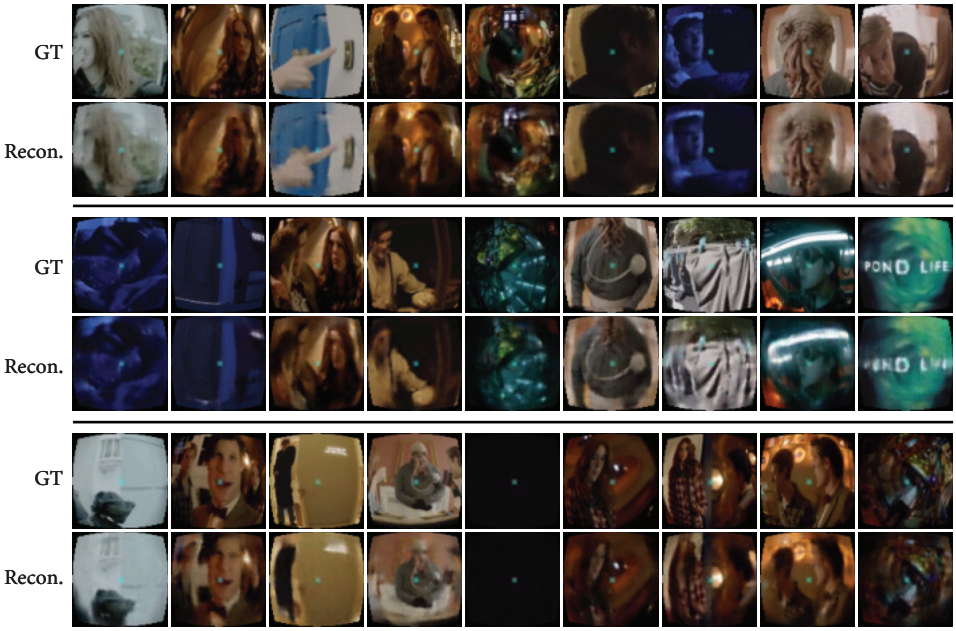


Figure 4.1: Reconstructions from the Neural Flow model trained on simulated brain data and the entire training volume. The results show different random samples drawn from the test set (GT) and their matching reconstructions (Recon.). Overall, we can see that the model successfully extrapolates from missing spatial information to reconstruct individual frames accurately. We can observe larger density of receptive field locations of an image around the fovea drives better reconstruction for central region. Moving towards the warped borders of the image, the reconstructions become slightly less sharp.

suggest that the receptive field locations carry enough spatial information for the model to generate accurate reconstructions and that the architecture allows for an invertible image-to-image translation. We analyze the reconstructions quantitatively through high- and low-level feature representations of different neural network models. In particular, we analyze the reconstructions based on correlations from features extracted from the pre-trained AlexNet model (Krizhevsky et al., 2012) trained to recognize objects and the C3D model (Tran et al., 2015) for action regnition. While the former does 2d convolutions on the input, we inspect spatio-temporal features through 3d convolutions in the latter. Specifically, we take intermediate layers from each model, and then report the average pixel-wise Pearson correlation coefficient r and a mean similarity coefficient corresponding to the inverse of the $\mathcal{L}_2$-norm given by $(\mathcal{L}_2 + 1)^{-1}$ (see Eq. 3.12; $\lambda_{\text{pix}} = 1$). The similarity coefficient is inversely proportional to the euclidean distance, but bounded to the range $[0, 1]$, where 1 shows perfect similarity. We report these metrics based on the entire test set. The quantitative results on the simulation data support the preliminary qualitative analysis through strong positive correlations between actual and reconstructed frames with high similarities (see Table 4.1; All samples).

4.2 ABLATION EXPERIMENTS

Due to excellent initial performances when training on the entire volume, we further simulated even sparser representations of brain data based on several ablation

Table 4.1: Performances on the test set for the simulation experiments including several ablations. The correlation and similarity values are based on intermediate feature layers from different pre-trained neural network models.

		Number of samples			
		All samples ($\approx 10^5$)	10^4	10^3	10^2
Correlation	AlexNet pool2	**0.890**	0.872	0.825	0.736
	AlexNet pool5	**0.803**	0.739	0.653	0.555
	AlexNet fc6	**0.850**	0.777	0.681	0.568
	C3D pool2	**0.828**	0.800	0.779	0.683
	C3D pool5	**0.619**	0.487	0.473	0.306
	C3D fc6	**0.575**	0.448	0.441	0.271
Similarity	AlexNet pool2	**0.909**	0.893	0.861	0.811
	AlexNet pool5	**0.885**	0.856	0.819	0.785
	AlexNet fc6	**0.793**	0.728	0.659	0.593
	C3D pool2	**0.900**	0.891	0.884	0.844
	C3D pool5	**0.578**	0.564	0.540	0.479
	C3D fc6	**0.800**	0.791	0.766	0.721

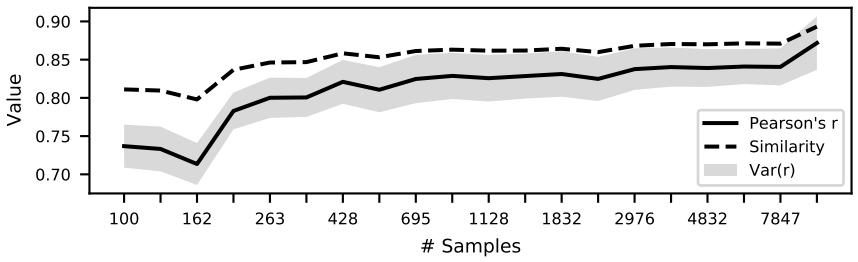


Figure 4.2: Ablation experiment based on datasize. The number of samples are drawn from logarithmic space to highlight the strength of the model for small sample sizes. The average correlation and similarity values on the test set are reported for the pool2d layer of the AlexNet model. The grey shade shows the variance of the Pearson correlation across all test set samples. The results show clear positive correlations with high similarities, also for very sparse datasizes.

experiments. We did this by (*i*) using a subset of the available training data and (*ii*) using a sparser representation of receptive fields. A priori, it is important to note that the size of the training set inherently influences the training duration and the convergence behavior of the model. For same model complexities (i.e. the number of neural network parameters), large datasizes are assumed to converge quicker as a function of the number of epochs. To compare the models with different training data sizes, we therefore train each model until the test loss has plateaued for 30 epochs, while enforcing each model to train for at least 50 epochs. The metrics reported for each ablation model are then based on the model state with the lowest average test loss. Since our baseline experiments showed strong positive correlations for only 10.000 training samples ($r > 0.7$ across all intermediate AlexNet features), we took this as the baseline for all additional ablations. This means that the model achieved prior to further ablation studies clear reconstructions based

Table 4.2: Performances on the test set for the simulation experiments including ablations of receptive field information. The correlation and distance values are based on intermediate feature layers from different pre-trained neural network models. All simulation data results are based on 10.000 training samples.

		Percentage			
		100%	70%	25%	10%
Correlation	AlexNet pool2	**0.872**	0.851	0.795	0.755
	AlexNet pool5	**0.739**	0.706	0.630	0.582
	AlexNet fc6	**0.777**	0.745	0.670	0.619
	C3D pool2	**0.800**	0.770	0.705	0.656
	C3D pool5	**0.487**	0.448	0.385	0.350
	C3D fc6	**0.448**	0.414	0.345	0.317
Similarity	AlexNet pool2	**0.893**	0.879	0.849	0.822
	AlexNet pool5	**0.856**	0.842	0.817	0.797
	AlexNet fc6	**0.728**	0.703	0.660	0.624
	C3D pool2	**0.891**	0.877	0.848	0.826
	C3D pool5	**0.564**	0.547	0.524	0.461
	C3D fc6	**0.791**	0.781	0.758	0.699

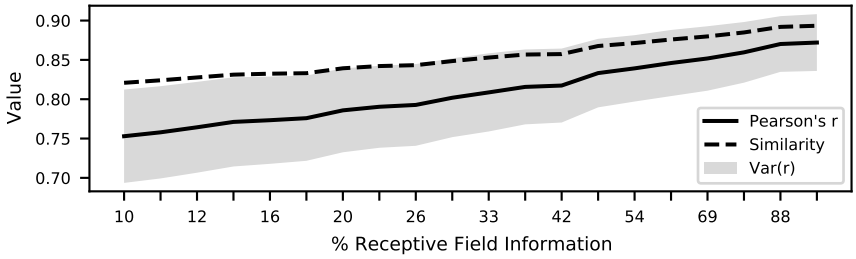


Figure 4.3: Ablation experiment as a function of the available receptive field information. The percentage of receptive field locations are drawn from logarithmic space to highlight the strength of the model for a small fractions. The average correlation and similarity values on the test set are reported for the pool2d layer of the AlexNet model. The grey shades show the variance of the Pearson correlation across all test set samples. The results show a positive linear trend for correlations and similarities and the percentage of receptive field locations. The more receptive field information, the smaller the variance of the correlations between all test set reconstructions.

on a subset of < 10% of the total training volume and sparse pixel information of 9% for each training sample. We similarly train all ablation models with $K = 10$ flow steps and $L = 2$ flow blocks and report the same metrics for the hold-out test set.

For the datasize ablations, we sampled the number of training samples from logarithmic space between 100 and 10.000 to test the strength of the model for small sizes. In Figure 4.2, we can see a positive linear trend on the reconstruction performances with increasing sample size. The model excels also for very small training volumes and sparse receptive field information, with very good reconstruction results across all experiments (see Table 4.1). This was also the case when including a very small

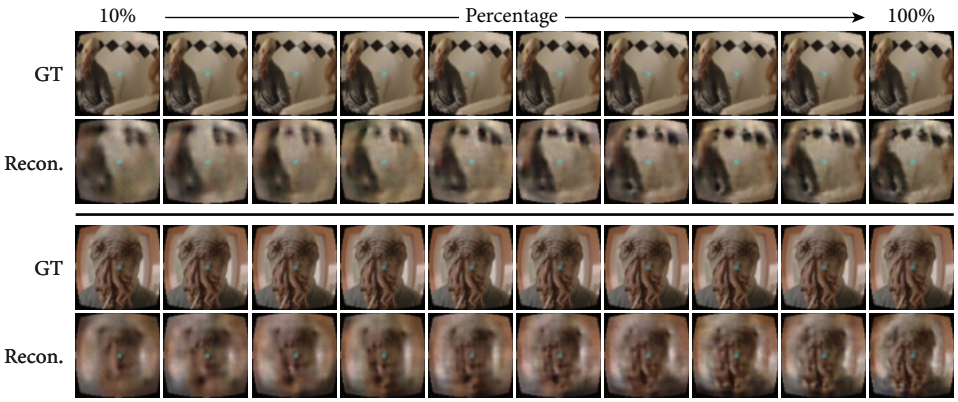


Figure 4.4: Exemplary visualizations for two samples from the test set with increasing percentage of receptive field information from left to right. The reconstructions are clear but more blurry for small fractions of receptive field locations and gain in sharpness, especially in the central part of an image, when increasing the number of receptive fields.

fraction of the training set of only 100 samples. This is especially remarkable, since this data size is an order of magnitude smaller compared to the entire test volume of 1034 samples. This highlights the strength of the Neural Flow model to also work for very limited training data sizes.

We further investigated the effect of sparser receptive field information for each frame on the reconstructions. For this, we looked at the quantitative and qualitative effect as a function of the percentage of receptive field information available from the original estimators. To do this, we sampled from each frame a subset of the 879 receptive field locations from the binary receptive field mask. We determine this subset by sampling at random from the number of active pixels at varying percentages. Similar to the ablations on datasize, we sample the percentage from logarithmic space between 10% and 90%. For the lowest percentage, this means that only ≈ 88 pixels contain spatial information for the reconstruction. For the 96×96 input image this equals $\approx 0.95\%$ ($\frac{88}{96^2}$) of all the pixels. Importantly, this sampling regime consequences to ignore the eccentricity of foveal representations, where the density of points is larger in the central part of an image. Sampling uniformly from the density of receptive field locations in the limit nevertheless also maintains the overrepresentation of the fovea. Across all receptive field ablations, we fix the number of training samples to 10.000. Our results show that even with limited receptive field and therefore limited spatial information information, it was possible to accurately reconstruct samples from the test set (see Table 4.2). We can see in Figure 4.3 a positive linear trend of the correlation and similarity values with an increase in the percentage. Interestingly, the variance in the test set predictions decreases with the linear trend. We can qualitatively see the effect of varying receptive field percentages in Figure 4.4. While overall color properties for surrounding regions are easily captured by the models, small receptive field sizes result in more blurry reconstructions. This becomes especially evident at the central part of an image, where the density of foveal locations is also higher the more receptive fields are sampled.

Table 4.3: Effect of three squeezing factors on the reconstructions. The results are averaged over five independent runs with different network initialization trained on 10.000 samples. We can observe a small improvement for a small squeezing factor for earlier layers, where the effect becomes larger for more intermediate representations. This improvement trades at a much larger compuational cost and training duration until convergence.

		Factor		
		2	4	8
Correlation	AlexNet pool2	**0.877**	0.874	0.865
	AlexNet pool5	**0.773**	0.752	0.719
	AlexNet fc6	**0.817**	0.792	0.751
	C3D pool2	**0.806**	0.805	0.797
	C3D pool5	**0.565**	0.532	0.478
	C3D fc6	**0.528**	0.483	0.426
Similarity	AlexNet pool2	**0.898**	0.895	0.888
	AlexNet pool5	**0.871**	0.861	0.847
	AlexNet fc6	**0.761**	0.740	0.709
	C3D pool2	**0.891**	0.891	0.887
	C3D pool5	**0.556**	0.540	0.516
	C3D fc6	**0.787**	0.773	0.756

In sum, all of the above simulation and ablation experiments show that the model successfully works for paired image-to-image translations and extrapolates from missing spatial information. This suggests that it might be suited to reconstruct brain activity, which we test in Section 4.4.

4.3 SQUEEZING INTERMEDIATE REPRESENTATIONS

One significant contribution of the Neural Flow model introduced in this thesis is that it can be used for any paired image-to-image translation problem. The full spatial input is maintained by reverting the squeezing operation after each flow block. The squeezing factor trades channel for spatial dimensions prior to the forward pass of this block and therefore significantly impacts the computational requirements. That is, a large squeezing factor reduces the spatial dimensions to the fraction of that factor and therefore also reduces the number of convolution operations. Intuitively, we assumed that this would come at a cost of creating a larger pixel noise in the reconstruction when increasing the squeezing factor. This intuition was based on assumed similarities to the checkerboard patterns found in transposed convolutions (Odena et al., 2016). We tested this by training a simple Neural Flow model with $K = 10$ flow steps and $L = 2$ blocks for three different squeezing factors $f = \{2, 4, 8\}$. We run each model and each squeezing factor factor five times with different random network initializations. This reduces the impact of reporting quantitative results based on chance. The results in Table 4.3 are based on average performances based on these five runs. Our results show that reconstructions are similarly accurate for different squeezing factors, with small difference in performances, especially for

Table 4.4: Performances on the test set for the brain reconstruction results when including different ROIs. We report results from the Brain2Pix paper (Le et al., 2021) with and without adversarial training. Across all correlations, the Neural Flow model trained on single ROIs was not able to surpass the Brain2Pix model trained on activity from V1-V3. Our model nevertheless outperforms significantly compared to the results without adversarial training.

		Neural Flow			Brain2Pix	
		V1	V2	V3	V1-V3	w/o Adversarial
Correlation	AlexNet pool2	0.425	0.423	0.333	**0.461**	0.162
	AlexNet pool5	0.316	0.319	0.262	**0.350**	0.130
	AlexNet fc6	0.349	0.359	0.322	**0.460**	0.156
	C3D pool2	**0.516**	0.510	0.417	0.486	0.162
	C3D pool5	0.165	0.171	0.156	**0.242**	0.039
	C3D fc6	0.176	0.182	0.160	**0.251**	0.049
Similarity	AlexNet pool2	**0.189**	0.188	0.150	0.177	0.070
	AlexNet pool5	**0.469**	0.470	0.437	0.440	0.385
	AlexNet fc6	0.409	0.413	0.387	**0.917**	0.317
	C3D pool2	**0.464**	0.462	0.409	0.419	0.117
	C3D pool5	**0.224**	0.223	0.190	0.224	0.151
	C3D fc6	0.500	0.496	**0.517**	0.456	0.267

early feature layers. For more intermediate feature representations the difference in correlations and similarity increases, with slightly inferior results for a larger squeezing factor. We found an additional difference of training duration until each model converged. A small factor did not only increase the overall training time for each batch, but also took more epochs until the model has converged. This shows that the compression of spatial dimensions before each flow block increases not only the computational speed, but at the same time also improves convergence speed.

In theory and from a practical neural network view point it would be more intuitive to learn the downsampling of channel dimensions also inside the network using for instance 1×1 convolutions. This operation nevertheless lacks the invertibility requirement of the network design since the weight matrix is non-square and therefore cannot be utilized in normalizing flow models. We elaborate on this in Chapter 5.

4.4 RECONSTRUCTING STIMULI FROM BRAIN ACTIVITY

To reconstruct brain activity, we train our model for several ROIs separately and thereby convert each brain sample with three TRs in 2d pixels space into an actual RGB image. In theory, it would also be possible to use a smaller time frame of, for example, one TR and to stack three ROIs on the time dimension. This would nevertheless increase the chance of missing the response as shifted by the haemodynamic delay. Since the brain incorporates complex multi-stage dynamics, a neural network model needs to be similarly complex for clear reconstructions. We therefore

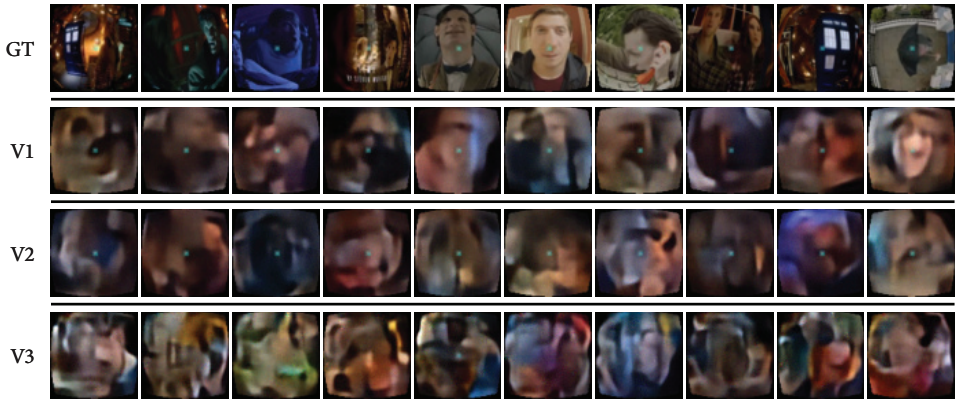


Figure 4.5: Brain reconstructions from the Neural Flow model. The results show different random samples drawn from the test set (Recon.) and the reconstructions (Recon.) for V1, V2 and V3. Overall, we can observe different levels of abstractions in the reconstructions, where V3 as a later processing stage of visual processing seems to encode more abstract stimulus properties.

Figure 4.6: Brain reconstructions from the Neural Flow model. The results show different samples drawn from the test set (GT) in sequential order of consecutive frames and the reconstructions (Recon.) for V1, V2 and V3. We can here see that the model achieves slight temporal coherence for reconstrucions, shown through little changes between consecutive frames.

increased the numbers of the parameters of the network and train our Neural Flow model with $K = 16$ flow steps and $L = 3$ blocks for each ROI. This model has ≈ 78 million parameters. We tested reconstruction performances for different model complexities, ablations on training data sizes and other model architectures, but found that these gave the best results. A comparison of the reconstruction performances for each ROI can be seen in Table 4.4 and in Figure 4.6. We thereby directly relate the results for each ROI to the previous benchmark results from Le et al. (2021) trained on combined information from V1 to V3 with and without adversarial training.

The Neural Flow model performed quantitatively similar for V1 and V2 and nearly matches the previous benchmark results trained on the combined activity from V1 to V3. V1 as the earliest stage of visual processing provided the best reconstruction results, although with little difference compared to and sometimes surpassed by V2. The model was able to capture visual coherence for individual frames, with natural motion and color characteristics. Differences in brightness were captured slightly

Table 4.5: Performances on the test set for the brain reconstruction results for different benchmarks. We report results from the Neural Flow model for V1, the Brain2Pix paper (Le et al., 2021) and from (Nishimoto et al., 2011; Shen et al., 2019).

		Neural Flow	Brain2Pix	Nishimoto et al.	Shen et al.
Correlation	AlexNet pool2	0.425	**0.461**	0.252	0.414
	AlexNet pool5	0.316	**0.350**	0.231	0.326
	AlexNet fc6	0.349	**0.460**	0.201	0.419
	C3D pool2	**0.516**	0.486	0.366	0.421
	C3D pool5	0.165	**0.242**	0.048	0.203
	C3D fc6	0.176	**0.251**	0.041	0.218
Similarity	AlexNet pool2	**0.189**	0.177	0.153	0.157
	AlexNet pool5	**0.469**	0.440	0.445	0.443
	AlexNet fc6	0.409	**0.917**	0.366	0.402
	C3D pool2	**0.464**	0.419	0.412	0.416
	C3D pool5	**0.224**	0.224	0.273	0.204
	C3D fc6	0.500	0.517	**0.536**	0.457

for V1 and V2, but not for V3. The model generally had difficulties to reconstruct anything above abstract shapes. While the V1 reconstructions contain less color, they visually suggests the most accurate results. Qualitatively, we can further see that the reconstructions seem to contain more abstract stimulus features for brain activity from more downstream regions (i.e. higher levels of abstraction for V3 compared to V1). Yet, this increase in feature complexity observed visually is not directly supported by the quantitative results, where we would assume that this would translate to higher correlations and similarities for deeper feature layers of the AlexNet and C3D model. In our experiments, the V3 model resulted in inferior reconstruction performances compared to the other brain regions. Visually, V3 gave the most colorful reconstructions with abstract shapes. In direct comparison to previous benchmark results from Le et al. (2021), we nevertheless found that the results surpassed the corresponding model without adversarial training across all feature correlations and similarities. As can be seen from Table 4.4; Brain2Pix, adversarial training had a significant impact on the results, which suggests that this loss is a driving factor for accurate reconstructions. Opposite to their findings, our model achieved good reconstruction results also for the early visual area V1. Visually, these quantitative differences are less clear. While not improving upon Le et al. (2021), our results for V1 surpass the reconstruction model from Nishimoto et al. (2011) and match the performances froms Shen et al. (2019) done in baseline experiments in their study (see Table 4.5).

5

DISCUSSION AND CONCLUSION

This thesis introduced a Neural Flow model that can translate between a topographical interpretation of brain activity and the original input stimulus space for the large-scale fMRI dataset of the BBC series *Doctor Who*. Our results showed that it is possible to use this model to reconstruct simulated brain activity. For this, we applied the receptive field estimators directly on the input frames. This simulates a noise-free representation of brain data and mimics the sparsity created within the encoding scheme. The simulation results demonstrated that the model successfully extrapolates from missing spatial information. We further illustrated through several ablation experiments on the simulation data that the model is robust to sparse training data sizes and to sparse receptive field information. Across all ablations, we showed strong positive correlations with high similarities between the reconstructed and target frames. Moreover, the results indicated that the receptive field locations carry enough spatial information to reconstruct single frames. By squeezing intermediate representations to a small spatial factor, we have further highlighted that we can develop a generative flow model that reduces the computational requirements at little cost in reconstruction performance. This makes the model also particularly suited to the training of large images, entire video sequences, or possibly other image-to-image translation problems.

Our findings do not translate to the same extent on actual brain data using an identical receptive field mapping. This has been the case for different regions of interest as model input. A direct comparison is nevertheless not meaningful, as the accuracy of the neural reconstruction is bounded by the proportion of the variance of the brain response. A lot of this variance is related to measurement noise and not to the actual stimuli. While acknowledging this, we observe that the reconstructions still capture natural motion characteristics, color constancy in larger regions of a frame and temporal coherence between frames. These qualitative findings are supported by correlations and similarities that nearly match the previous benchmark on the same dataset.

There are several possible limitations of the present work that could improve the reconstruction. One limitation is inherently given by our model definition and the invertibility requirement. Our Neural Flow model can only translate between matching input and output channels. Since one cannot invert non-square matrices, the invertibility requirement prohibits a downsampling of channel dimensions, as could for instance be done using $1x1$ convolutions. We were therefore only able to include single ROIs with a TR of three to reconstruct a single frame in RGB. In theory, it is likely to be beneficial to include larger time ranges for the reconstruction. This increase in time range would increase the probability of capturing the shifted response as given by the haemodynamic delay. Moreover, it would be advantageous to include brain data from multiple ROIs, as each region accounts for different feature attributes in the processing hierarchy. This suggests that including multiple ROIs would also increase the visual complexity of the reconstruction, as

has previously also been shown through best performances in Le et al. (2021) when trained on combined activity from V1 to V3. One possibility to achieve this for our model would be to create the topographical images in 2d pixel space based on a larger ROI that combines the voxel activity from V1 to V3. For this, one would not treat the visual areas separately, but rather find locations in the image that drive the highest response for the combined volume. This would inherently include the activity from multiple ROIs, while at the same time maintaining the number of channel dimensions. Similarly, it would thus also be interesting to include even higher visual areas in the temporal and parietal cortex, like IT or MT. These areas have been show to respond to more abstract and higher-level stimulus properties, which might also translate to greater visual semantics in the reconstructions.

It remains unclear to what extent the findings are limited by the model architecture of the generative flow. One general difficulty of the approach is that our generative flow explicitly models the probability density per pixel in the reconstruction through maximum likelihood training. To achieve this, the invertibility requirement enforces many layers to contain parameters that are learned beyond the weight and bias terms of convolutional blocks. A flow model that is too complex might not therefore converge in estimating the likelihood, whereas if too simple, it might not accurately estimate the input distribution. We observed that the model quickly focused on the MLE and the bits-per-dimension error term and less on the loss components that enforce visual semantics, like the feature and pixel loss. One indication for this was that the model overall saturated very quickly on the test set performance after little training. This was also the case while experimenting with different weighting factors for each loss term. Following this, it has also been shown that the brain likely also optimizes several objectives across space and time (Marblestone et al., 2016). Neural responses may also include this multitude of objectives simultaneously (DiCarlo and Cox, 2007). This might explain the extent to which different loss objectives during training relate to more accurate reconstructions and why in Le et al. (2021) a large fraction of the reconstruction performance was due to adversarial training. In their benchmark results, the adversarial loss is a driving factor in the quantitative reconstruction results. Without this training, their correlations dropped below $r < 0.2$ for all intermediate feature representations on a model trained on brain data from V1 to V3. Here, we achieve reconstruction results that are similar to their benchmark, without adversarial training and including only a single ROI. For each of these ROIs, the difference in reconstruction performance was not large, and the results for V1 were significantly better than the ones from Le et al. (2021). This finding suggest that the main difference between both models is due to the adversarial training. An adversarial loss enforces learning higher level feature representations for the model and therefore might limit the reconstruction of low-level visual properties as computed by V1, which shows through the inferior reconstruction performances for this visual area in their model. Since an adversarial loss has been shown to work well with generative adversarial networks on many image-to-image translation problems, it would in the future be interesting to also explore upon adversarial training and incorporate such a loss for our model.

If successful, the advantages of having generative flow model to study brain representations are nevertheless vast. A good reconstruction model is a necessary baseline in order to use the model from an encoding perspective. If so, then it

becomes possible to directly relate neural encoding and to probe as to where good visual semantics of reconstructions actually translate to similar encoding results. In this line, one could then additionally also show receptive field properties for individual voxels. Instead of giving actual brain data for the reconstruction, one could for such a case activate only a single voxel in pixel space and then reconstruct this voxel image. One can then interpret the reconstruction as the receptive field visualization of that voxel in image space. Additionally, one could experiment not inputting all brain areas at the very start of the model, but instead feeding all ROIs to all layers. The idea is to then probe whether the hierarchy that evolves in the model is similar to the visual hierarchy of the brain. Additionally, the definition of our Neural Flow model would allow to inclusion of temporal dependencies in the reconstruction. Since we can squeeze spatial dimensions largely in the forward pass, the model is especially suited for training larger images or sequences of many frames. While we experimented with including multiple TRs from multiple ROIs at once to reconstruct a sequence of frames, this did not improve the results. We accounted for this by adding temporal losses to enforce temporal coherence, but with little success. Future work could also continue exploring temporal dependencies. Interestingly, the ablation results suggest that the model can possibly also be used for smaller dataset sizes than the one used in this thesis. This is usually an advantage for neural decoding, since fMRI datasets are mostly of small training volume.

However, one should be careful when intepreting decoding results, as they do not say anything about the underlying processing mechanism of the encoding scheme, only of its product (Kriegeskorte and Douglas, 2019). Combining this with an encoding model therefore makes it additionally interesting to test to what extent decoding results give information on the type of processing and format done in a specific brain region. In this line, it would therefore also be interesting to probe to what degree different ROIs encode visual information. If one could, for instance, use only V1 to reconstruct an image precisely, then this would suggest that V1 maintains full input information, which has never been shown to date. In contrast, an encoding model predicts the representational space of the brain, so as to explain how activities in neural populations are changed in response to sensory stimuli, motor responses and cognitive processes (Diedrichsen and Kriegeskorte, 2017). Aiming for such explanations using flow-based models is an interesting avenue for future research.

5.1 CONCLUSION

This thesis explored the use of generative flows to probe neural representations. Its main contribution is a normalizing flow model that can be used to translate between two image domains. With maximum likelihood training and invertible transformations, the probability density from one image domain is transformed into the other by accounting for the difference in volume density with the change of variables formula. This makes the approach particularly interesting for neural encoding and decoding. This work exploits their inverse relationship in a single neural network model that can bijectively map between image spaces and brain responses. This makes it possible to combine neural encoding and decoding in a single model, without entangling prior representations.

Overall, this thesis questioned the extent to which it is possible to design a generative flow model to be used for neural reconstruction. For this, we developed a normalizing flow model to reconstruct brain activity. One can use the same neural network model used for decoding and transform it into an encoding model. We tested our Neural Flow model to reconstruct simulated and actual brain recordings from fMRI. The simulation results showed that the model can succesfully be used for paired image-to-image translation and excels for sparse training volumes and sparse receptive field information. The reconstructions on actual brain data found in this thesis nearly almost match the previous benchmark results obtained on the same dataset. We achieve this, while at the same time including activity from only single regions of interest of brain activity. We found leading performances for early visual areas V1 and V2. V3 had qualitatively and quantitatively inferior performances in comparison. Overall, we therefore conclude that our normalizing flow model succesfully allows to reconstruct brain activity. while contributing a unified approach to neural encoding and decoding.

While the results in this thesis come close to the previous benchmark results, the reconstructions still stayed mostly abstract. This thesis did therefore not cover the encoding perspective. Encoding performance of the corresponding Neural Flow model by inverting the forward pass thereby directly relates to the quantitative reconstruction performances. If so, then it becomes possible to directly relate neural encoding and to probe as to where good visual semantics of reconstructions actually translate to similar encoding results. Combining encoding and decoding in a single model is the strength of the proposed approach. Combining both should still be an aim for future work building on this thesis. Instead of starting from the perspective of neural decoding, one could alternatively also start with an encoding model. Upon good encoding results, this model could then also be used for reconstruction.

Outlook

The whole field of neural encoding and decoding is thereby generally fast-moving. Much research has focused on using fMRI data because of its high spatial resolution. Due to its known challenges with revealing temporal information, multivariate pattern analysis from different data modalities such as magnetoencephalography (MEG) and electroencephalography (EEG) could improve upon some limitations to reveal more about the rapid neural dynamics. Yet, having reconstructions with the same semantic detail compared to using data from fMRI has not been possible. If successful, neural decoding from these data modalities could advance the use of brain computer interfaces (BCIs). These systems rely on fast processing of information and are not limited by the haemodynamic delay induced by BOLD responses. This could then be used to reveal covert mental activity to control devices through a communication interface, or for neural prostheses (Van Gerven et al., 2009).

6

ACKNOWLEDGMENTS

First, I would like to acknowledge for what I had little influence throughout my life, but what had a significant impact on the road I have been following until today. I have been consistently able to follow my interests and heart, without being disadvantaged by my racial, ethnic, and gender identity, political and religious opinions, or financial burdens, to name a few. Being highly priviliged, I feel lucky to be where I am right now and much of the work done in this thesis has been made possible simply because the barriers were less.

Despite this, I want to wholeheartedly thank Dr. Umut Güçlü for supervising my project and giving me the opportunity. His immense pool of ideas has been the driving factor for many results discovered in this thesis and his fascination to do science influences me largely. I further would like to thank him for providing me with such valuable input for many academic and non-academic questions and for continuously believing in the success of the project.

I further want to thank Prof. Marcel A.J. van Gerven, who initially gave me the opportunity to intern at the Donders Institute, which introduced me into the whole deep learning and neural coding field and motivated to select the corresponding Masters program at Radboud. Equal thanks goes to Dr. Katja Seeliger, whose ideas, work attitude and interests have been a major drive in my personal and academic development. Without her meticulous collection of the dataset, this thesis would not have been possible. Throughout the project, I have also consistently been supported by Lynn Le, who helped me with handling the data and previous scripts and motivated me to continue exploring at times when results were stagnating.

Besides this, there are many people in my everyday life that continuously support me and accept me for who I am. My deepest gratitude goes to my closest friends and my family who have guided me and been at my side throughout this time.

7

REFERENCES

Bashivan, P., Kar, K., and DiCarlo, J. J. Neural population control via deep image synthesis. *Science*, 364(6439), 2019.

Bruce, C., Desimone, R., and Gross, C. G. Visual properties of neurons in a polysensory area in superior temporal sulcus of the macaque. *Journal of neurophysiology*, 46(2):369–384, 1981.

Cadena, S. A., Denfield, G. H., Walker, E. Y., Gatys, L. A., Tolias, A. S., Bethge, M., and Ecker, A. S. Deep convolutional models improve predictions of macaque v1 responses to natural images. *PLoS computational biology*, 15(4):e1006897, 2019.

Cadieu, C. F., Hong, H., Yamins, D. L., Pinto, N., Ardila, D., Solomon, E. A., Majaj, N. J., and DiCarlo, J. J. Deep neural networks rival the representation of primate it cortex for core visual object recognition. *PLoS Comput Biol*, 10(12):e1003963, 2014.

Carandini, M., Demb, J. B., Mante, V., Tolhurst, D. J., Dan, Y., Olshausen, B. A., Gallant, J. L., and Rust, N. C. Do we know what the early visual system does? *Journal of Neuroscience*, 25(46):10577–10597, 2005.

Carlson, T. A., Schrater, P., and He, S. Patterns of activity in the categorical representations of objects. *Journal of cognitive neuroscience*, 15(5):704–717, 2003.

Churchland, P. S. and Sejnowski, T. J. Neural representation and neural computation. *Philosophical Perspectives*, 4:343–382, 1990.

Churchland, P. S. and Sejnowski, T. J. *The computational brain*. MIT press, 1994.

Cichy, R. M., Khosla, A., Pantazis, D., Torralba, A., and Oliva, A. Comparison of deep neural networks to spatio-temporal cortical dynamics of human visual object recognition reveals hierarchical correspondence. *Scientific reports*, 6:27755, 2016.

Dayan, P. and Abbott, L. F. *Theoretical neuroscience: computational and mathematical modeling of neural systems*. Computational Neuroscience Series, 2001.

Decharms, R. C. and Zador, A. Neural representation and the cortical code. *Annual review of neuroscience*, 23(1):613–647, 2000.

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

DiCarlo, J. J., Zoccolan, D., and Rust, N. C. How does the brain solve visual object recognition? *Neuron*, 73(3):415–434, 2012.

Diedrichsen, J. and Kriegeskorte, N. Representational models: A common framework for understanding encoding, pattern-component, and representational-similarity analysis. *PLoS computational biology*, 13(4):e1005508, 2017.

Dinh, L., Krueger, D., and Bengio, Y. Nice: Non-linear independent components estimation. *arXiv preprint arXiv:1410.8516*, 2014.

Dinh, L., Sohl-Dickstein, J., and Bengio, S. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803*, 2016.

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

Engel, S. A., Rumelhart, D. E., Wandell, B. A., Lee, A. T., Glover, G. H., Chichilnisky, E.-J., and Shadlen, M. N. fmri of human visual cortex. *Nature*, 1994.

Felleman, D. J. and Van Essen, D. C. Distributed hierarchical processing in the primate cerebral cortex. *Cerebral cortex (New York, NY: 1991)*, 1(1):1–47, 1991.

Friston, K. J., Jezzard, P., and Turner, R. Analysis of functional mri time-series. *Human brain mapping*, 1(2):153–171, 1994.

Fujita, I., Tanaka, K., Ito, M., and Cheng, K. Columns for visual features of objects in monkey inferotemporal cortex. *Nature*, 360(6402):343–346, 1992.

Gatys, L. A., Ecker, A. S., and Bethge, M. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2414–2423, 2016.

Glaser, J. I., Benjamin, A. S., Chowdhury, R. H., Perich, M. G., Miller, L. E., and Kording, K. P. Machine learning for neural decoding. *Eneuro*, 7(4), 2020.

Güçlü, U., Thielen, J., Hanke, M., and van Gerven, M. Brains on beats. In Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016.

Güçlü, U. and van Gerven, M. Unsupervised learning of features for bayesian decoding in functional magnetic resonance imaging. In *Belgian-Dutch Conference on Machine Learning*, 2013.

Güçlü, U. and van Gerven, M. A. Unsupervised feature learning improves prediction of human brain activity in response to natural images. *PLoS Comput Biol*, 10(8):e1003724, 2014.

Güçlü, U. and van Gerven, M. A. Deep neural networks reveal a gradient in the complexity of neural representations across the ventral stream. *Journal of Neuroscience*, 35(27): 10005–10014, 2015.

Güçlü, U. and van Gerven, M. A. Increasingly complex representations of natural movies across the dorsal stream are shared between subjects. *NeuroImage*, 145:329–336, 2017.

Güçlütürk, Y., Güçlü, U., Seeliger, K., Bosch, S., van Lier, R., and van Gerven, M. A. Reconstructing perceived faces from brain activations with deep adversarial neural decoding. In *Advances in Neural Information Processing Systems*, pages 4246–4257, 2017.

Han, K., Wen, H., Shi, J., Lu, K.-H., Zhang, Y., Fu, D., and Liu, Z. Variational autoencoder: An unsupervised model for encoding and decoding fmri activity in visual cortex. *NeuroImage*, 198:125–136, 2019.

Haxby, J. V., Gobbini, M. I., Furey, M. L., Ishai, A., Schouten, J. L., and Pietrini, P. Distributed and overlapping representations of faces and objects in ventral temporal cortex. *Science*, 293(5539):2425–2430, 2001.

Henschen, S. E. On the visual path and centre. *Brain*, 16(1-2):170–180, 1893.

Holmes, G. Disturbances of vision by cerebral lesions. *The British journal of ophthalmology*, 2(7):353, 1918.

Horikawa, T. and Kamitani, Y. Hierarchical neural representation of dreamed objects revealed by brain decoding with deep neural network features. *Frontiers in computational neuroscience*, 11:4, 2017.

Horikawa, T., Tamaki, M., Miyawaki, Y., and Kamitani, Y. Neural decoding of visual imagery during sleep. *Science*, 340(6132):639–642, 2013.

Hubel, D. H. and Livingstone, M. S. Segregation of form, color, and stereopsis in primate area 18. *Journal of neuroscience*, 7(11):3378–3415, 1987.

Hubel, D. H. and Wiesel, T. N. Receptive fields of single neurones in the cat's striate cortex. *The Journal of physiology*, 148(3):574–591, 1959.

Hubel, D. H. and Wiesel, T. N. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of physiology*, 160(1):106–154, 1962.

Inouye, T. *Die Sehstörungen bei Schussverletzungen der kortikalen Sehsphäre: nach Beobachtungen an Verwundeten der letzten japanischen Kriege*. Engelmann, 1909.

Ioffe, S. and Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

Isola, P., Zhu, J.-Y., Zhou, T., and Efros, A. A. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017.

Johnson, J., Alahi, A., and Fei-Fei, L. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pages 694–711. Springer, 2016.

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

Kamitani, Y. and Tong, F. Decoding the visual and subjective contents of the human brain. *Nature neuroscience*, 8(5):679, 2005.

Kay, K. N., Naselaris, T., Prenger, R. J., and Gallant, J. L. Identifying natural images from human brain activity. *Nature*, 452(7185):352, 2008.

Khaligh-Razavi, S.-M. and Kriegeskorte, N. Deep supervised, but not unsupervised, models may explain it cortical representation. *PLoS computational biology*, 10(11):e1003915, 2014.

Kietzmann, T. C., McClure, P., and Kriegeskorte, N. Deep neural networks in computational neuroscience. *BioRxiv*, page 133504, 2018.

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

Kingma, D. P. and Dhariwal, P. Glow: Generative flow with invertible 1x1 convolutions. In *Advances in neural information processing systems*, pages 10215–10224, 2018.

Klindt, D. A., Ecker, A. S., Euler, T., and Bethge, M. Neural system identification for large populations separating" what" and" where". *arXiv preprint arXiv:1711.02653*, 2017.

Kok, P., Jehee, J. F., and De Lange, F. P. Less is more: expectation sharpens representations in the primary visual cortex. *Neuron*, 75(2):265–270, 2012.

Kriegeskorte, N. Deep neural networks: a new framework for modeling biological vision and brain information processing. *Annual review of vision science*, 1:417–446, 2015.

Kriegeskorte, N. and Douglas, P. K. Interpreting encoding and decoding models. *Current opinion in neurobiology*, 55:167–179, 2019.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

Kullback, S. and Leibler, R. A. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86, 1951.

Le, L., Ambrogioni, L., Seeliger, K., Güçlütürk, Y., van Gerven, M., and Güçlü, U. Brain2pix: Fully convolutional naturalistic video reconstruction from brain activity. *bioRxiv*, 2021.

LeCun, Y., Bengio, Y., and Hinton, G. Deep learning. *nature*, 521(7553):436–444, 2015.

Lee, C.-Y., Xie, S., Gallagher, P., Zhang, Z., and Tu, Z. Deeply-supervised nets. In *Artificial intelligence and statistics*, pages 562–570. PMLR, 2015.

Logothetis, N. K. What we can do and what we cannot do with fmri. *Nature*, 453(7197): 869–878, 2008.

Malach, R., Levy, I., and Hasson, U. The topography of high-order human object areas. *Trends in cognitive sciences*, 6(4):176–184, 2002.

Marblestone, A. H., Wayne, G., and Kording, K. P. Toward an integration of deep learning and neuroscience. *Frontiers in computational neuroscience*, 10:94, 2016.

Mitchell, T. M., Shinkareva, S. V., Carlson, A., Chang, K.-M., Malave, V. L., Mason, R. A., and Just, M. A. Predicting human brain activity associated with the meanings of nouns. *Science*, 320(5880):1191–1195, 2008.

Miyawaki, Y., Uchida, H., Yamashita, O., Sato, M.-a., Morito, Y., Tanabe, H. C., Sadato, N., and Kamitani, Y. Visual image reconstruction from human brain activity using a combination of multiscale local image decoders. *Neuron*, 60(5):915–929, 2008.

Naselaris, T., Prenger, R. J., Kay, K. N., Oliver, M., and Gallant, J. L. Bayesian reconstruction of natural images from human brain activity. *Neuron*, 63(6):902–915, 2009.

Naselaris, T., Kay, K. N., Nishimoto, S., and Gallant, J. L. Encoding and decoding in fMRI. *Neuroimage*, 56(2):400–410, 2011.

Nishimoto, S., Vu, A. T., Naselaris, T., Benjamini, Y., Yu, B., and Gallant, J. L. Reconstructing visual experiences from brain activity evoked by natural movies. *Current Biology*, 21(19): 1641–1646, 2011.

Odena, A., Dumoulin, V., and Olah, C. Deconvolution and checkerboard artifacts. *Distill*, 1 (10):e3, 2016.

Rezende, D. J. and Mohamed, S. Variational inference with normalizing flows. *arXiv preprint arXiv:1505.05770*, 2015.

Ritter, S., Barrett, D. G., Santoro, A., and Botvinick, M. M. Cognitive psychology for deep neural networks: A shape bias case study. In *International conference on machine learning*, pages 2940–2949. PMLR, 2017.

Schoenmakers, S., Barth, M., Heskes, T., and Van Gerven, M. Linear reconstruction of perceived images from human brain activity. *NeuroImage*, 83:951–961, 2013.

Schoenmakers, S., Güçlü, U., Van Gerven, M., and Heskes, T. Gaussian mixture models and semantic gating improve reconstructions from human brain activity. *Frontiers in computational neuroscience*, 8:173, 2015.

Seeliger, K., Fritsche, M., Güçlü, U., Schoenmakers, S., Schoffelen, J.-M., Bosch, S., and Van Gerven, M. Convolutional neural network-based encoding and decoding of visual object recognition in space and time. *NeuroImage*, 180:253–266, 2018a.

Seeliger, K., Güçlü, U., Ambrogioni, L., Güçlütürk, Y., and van Gerven, M. A. Generative adversarial networks for reconstructing natural images from brain activity. *NeuroImage*, 181:775–785, 2018b.

Seeliger, K., Ambrogioni, L., Güçlütürk, Y., Güçlü, U., and van Gerven, M. A. J. Neural system identification with neural information flow. *bioRxiv*, 2019a. doi: 10.1101/553255.

Seeliger, K., Sommers, R., Güçlü, U., Bosch, S., and van Gerven, M. A large single-participant fmri dataset for probing brain responses to naturalistic stimuli in space and time. *bioRxiv*, page 687681, 2019b.

Sereno, M. I., Dale, A., Reppas, J., Kwong, K., Belliveau, J., Brady, T., Rosen, B., and Tootell, R. Borders of multiple visual areas in humans revealed by functional magnetic resonance imaging. *Science*, 268(5212):889–893, 1995.

Shen, G., Dwivedi, K., Majima, K., Horikawa, T., and Kamitani, Y. End-to-end deep image reconstruction from human brain activity. *Frontiers in Computational Neuroscience*, 13, 2019.

Simonyan, K. and Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

St-Yves, G. and Naselaris, T. Generative adversarial networks conditioned on brain activity reconstruct seen images. In *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 1054–1061. IEEE, 2018.

Stanley, G. B. Neural system identification. In *Neural Engineering*, pages 367–388. Springer, 2005.

Teuber, H.-L., Battersby, W. S., and Bender, M. B. Visual field defects after penetrating missile wounds of the brain. 1960.

Theis, L., Oord, A. v. d., and Bethge, M. A note on the evaluation of generative models. *arXiv preprint arXiv:1511.01844*, 2015.

Thirion, B., Duchesnay, E., Hubbard, E., Dubois, J., Poline, J.-B., Lebihan, D., and Dehaene, S. Inverse retinotopy: inferring the visual content of images from brain activation patterns. *Neuroimage*, 33(4):1104–1116, 2006.

Tran, D., Bourdev, L., Fergus, R., Torresani, L., and Paluri, M. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 4489–4497, 2015.

Van Gerven, M., Farquhar, J., Schaefer, R., Vlek, R., Geuze, J., Nijholt, A., Ramsey, N., Haselager, P., Vuurpijl, L., Gielen, S., et al. The brain–computer interface cycle. *Journal of neural engineering*, 6(4):041001, 2009.

Van Gerven, M. A. A primer on encoding models in sensory neuroscience. *Journal of Mathematical Psychology*, 76:172–183, 2017.

Van Gerven, M. A., De Lange, F. P., and Heskes, T. Neural decoding with hierarchical generative models. *Neural computation*, 22(12):3127–3142, 2010.

Wachowski, A. and Wachowski, L. The Matrix, Warner Bros. 1999.

Wen, H., Shi, J., Zhang, Y., Lu, K.-H., Cao, J., and Liu, Z. Neural encoding and decoding with deep learning for dynamic natural vision. *Cerebral Cortex*, 28(12):4136–4160, 2018.

Wu, M. C.-K., David, S. V., and Gallant, J. L. Complete functional characterization of sensory neurons by system identification. *Annu. Rev. Neurosci.*, 29:477–505, 2006.

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

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

Zeiler, M. D. and Fergus, R. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.

Zeki, S. M. Uniformity and diversity of structure and function in rhesus monkey prestriate visual cortex. *The Journal of Physiology*, 277(1):273–290, 1978.

A

EXTENDED SAMPLE OF BRAIN RECONSTRUCTIONS

V1

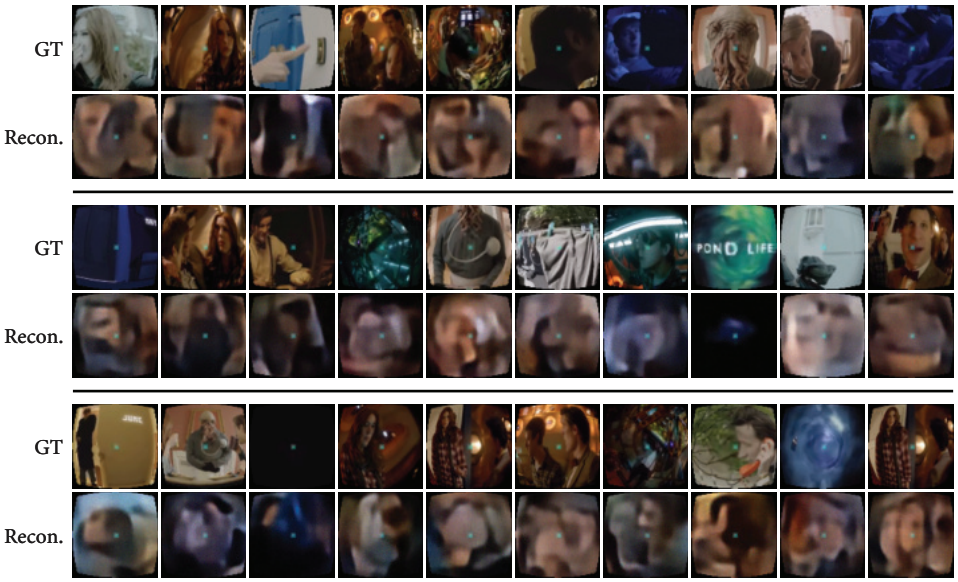


Figure A.1: Larger collection of random samples (GT) and their reconstructions (Recon.) for V1.

V2

Figure A.2: Larger collection of random samples (GT) and their reconstructions (Recon.) for V2.

V3

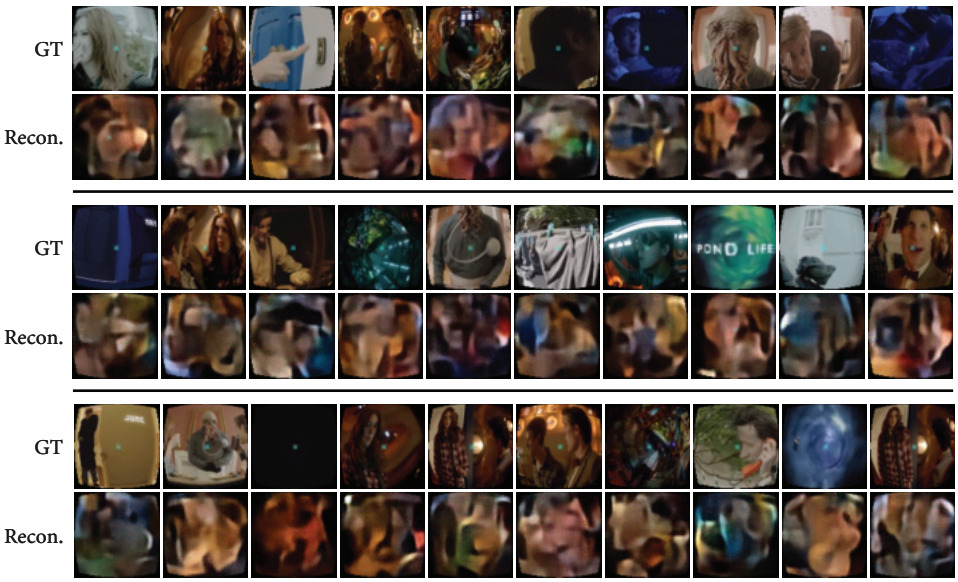


Figure A.3: Larger collection of random samples (GT) and their reconstructions (Recon.) for V3.