Introduction

Although the externalization of the internal representations of perceptual content in the brain has been a long and challenging goal in neuroscience, brain decoding using machine learning with fMRI activity can allow us to access these representations. The rendering of human brain activity into images has been a particularly sought after goal in brain decoding. Many studies that have attempted this only found success in pure reconstruction of stimuli on low-level image bases (e.g. pictures of alphabets) (Miyawaki et al., 2008). A recent study by Shen et al. 2019 outlines an approach “Deep Image Reconstruction” in which hidden layer activity of Deep Convolutional Neural Networks (DCNNs) is decoded from brain activity and these hidden layers are used to perform general reconstruction of visual stimuli by using iterative optimization techniques. This study showed that utilizing recent advances in deep learning can greatly aid in reconstructing visual stimuli from brain activity.

Previous reconstruction studies use fMRI data from the entire visual system to reconstruct visual stimuli. In this study, we show that one can reconstruct visual stimuli from just one specific region in the visual system, Inferotemporal Cortex (IT Cortex), by taking advantage of the internal representations of Deep Convolutional Neural Networks (CNNs). According to Efremova & Inui 2015, IT cortex is known to be involved in “core object recognition” (which means recognizing an object in an image regardless of its position or location in the image). Recent work by Yamins et al. 2014 has indicated that the activations of CNNs whose weights are optimized to perform well in core object recognition tasks can linearly predict neural responses in IT cortex, meaning that the internal representations of CNNs make good encoding models of
IT neuron recordings. In fact, Yamins et al. 2014 shows that goal-driven CNNs perform better at predicting IT neural responses than standard encoding models that are informed by actual brain data or by brain anatomy, which means that CNN representations are intrinsically predictive of IT neural responses. In this study, we take advantage of the close relationship between IT Cortex and performance-optimized Deep Convolutional Neural Networks by decoding CNN hidden layer activity from IT cortex neuron activity and using these decoded layers to reconstruct visual stimuli as Shen et al. 2019.

**Methods**

**Data**

We ran our analyses on the neural recordings dataset from Majaj et al. 2015. In this study, multielectrode arrays were used to record neuronal activity of *Rhesus Macaque* monkeys in two regions in the ventral visual stream: V4 and Inferotemporal Cortex (IT). In this study, we use the recordings from Inferotemporal Cortex (IT cortex). Each of the 2,560 stimulus images were presented to the subject for 100ms with 100ms of neutral gray background between the images. **Figure 1** displays a visualization of the data and the regions it was collected from.
Figure 1. From Majaj et al. 2015. (a). IT and V4 neural responses across multiple images. Note that although the original study had 5,760 stimuli, we use a subset of the data of responses to 2,560 images called “high variation” in which the poses/angles of a single objects across its repeated images is varied the most. (b). A visualization of the *Rhesus Macaque* visual system. We used recordings that capture IT cortex (shaded in blue).

The monkeys had to do an object recognition tasks of the various objects in the images. Within the stimulus set, there were eight categories: cars, planes, animals, boats, tables, chairs, and fruits which means there were 320 images per category. Their task was to do binary classification of the images (car/not car, plane/not plane, etc). Within each category, there are 64 different objects, where each object is represented in 5 different images with different pose/angles. Since IT cortex is implicated in core object recognition (Efremova & Inui 2015), this object recognition task (in which there are images of objects in a variety of categories in different poses/angles) was designed to maximize the responses of IT cortex.

There were, in total, 576 implanted electrodes (two monkeys, three arrays, 96 electrodes in each). In IT cortex, the 168 most visually-receptive regions were picked, which was measured by estimating the evoked visual response in each electrode timeseries in a separate independent
stimulus set and averaging the top 10% evoked image responses. See Majaj et al. 2015 for more details.

**Overview of Deep Neuronal Reconstruction Approach**

We now describe the approach we used to reconstruct the visual stimuli from their corresponding neural recordings. We use the approach pioneered in Shen et al. 2019. An overview of the approach can be found in Figure 2.

**Figure 2.** Overview of reconstruction approach. Adapted from Majaj et al. 2015 and Shen et al. 2019. Deep Image Reconstruction involves two Deep Neural Networks: A Deep Generator Network (DGN) and a Deep Neural Network (DNN). DNN activations are decoded from brain activity and an image is iteratively optimized such that the image’s DNN activations gets progressively closer to the DNN activations decoded from brain activity. The DGN is a generative neural network that constructs the images that get closer and closer to the stimulus images through optimization.
The approach is to take a fully-trained DNN (VGG-19) and extract all of its hidden layers from the images. A regression is calculated that decodes brain activity in the visual system evoked from seeing this image to the DNN representation of the same image (this regression is described in detail in a subsequent section “DNN Feature Decoding from Brain Activity”). Once one finds the predicted DNN representation, one can create a new image by iteratively optimizing an image until the DNN representation of the new image matches the predicted DNN representation decoded by brain activity. Specifically, Generative Adversarial Network trained to produce an image given a vector (Dosovitskiy & Brox. 2016). The Generative Adversarial Network is thought of as a “Natural Image Prior” because it constrains the distribution of reconstructed images to those of that the network produces. This vector is iteratively optimized using Gradient Descent with Momentum until the generated image from the vector has DNN representations matching the one decoded from brain activity. The resulting image is the final reconstructed image decoded from brain activity. They ran this reconstruction algorithm for all images in the testing set and found that there was a resemblance between reconstructed images decoded from brain activity evoked from seen images.

To put this rigorously, let \( \mathbf{v} \in \mathbb{R}^{224 \times 224 \times 3} \) be an image and let \( \phi^l \) be the feature extraction function for a Deep Neural Network that takes an image and returns the layer representation of the vector. Let \( y^l \) be the predicted layer representation of the image from brain activity. Let \( L \) be the set of all layers in a DNN representation. Let \( G: \mathbb{R}^{4096} \to \mathbb{R}^{224 \times 224 \times 3} \) represent a Deep Generative Network (computed by training a Generative Adversarial Network) that takes a vector and generates an image. We want to minimize the following loss function

\[
z^* = \arg\min_z \sum_{l \in L} \beta_l \| \phi^l(G(z)) - y^l \|_2^2
\]
Where $\beta_i$ is the expected norm of layer $l$ empirically estimated using 10,000 random images from the ImageNet database (the database used to train these deep learning models). And our final reconstruction would be $v^* = G(z^*)$. We minimized the above cost function using Gradient Descent with Momentum.

In Gradient Descent with Momentum, the $t$-th iteration is based on a weighted average of gradients from steps 0 to $t$. Specifically, the vector $v$ that goes into $G$ is updated by: $v_{t+1} = v_t + \mu_t$ where the weighted average gradient is updated by $\mu_{t+1} = m\mu_t - l \times g_t$ where $l$ is the learning rate and $g$ denotes the gradient calculated with backpropagation. $m$ is the momentum term used for accumulating the gradients. For all reconstructions, we used a learning rate of 2.0 and decayed it to 0 linearly over 200 iterations. We used a momentum of $m=0.9$.

**DNN Activation Decoding from Brain Activity**

In this section, we will describe how we regressed brain activity onto DNN hidden layer representations in order to perform Deep Image Reconstruction as described in the previous section “Deep Image Reconstruction Overview.” Note that we performed this regression to extract, for each test image, the 19 layers of VGG-19 and the 16 layers of VGG-scene (networks described in previous sections). We utilized the same approach in Shen et al. 2019 that was pioneered and invented by Horikawa and Kamitani. 2017. We describe this approach invented in Horikawa and Kamitani 2017 in this section.

We created multineuronal decoding models to decode the DNN feature vector of a seen image from the IT neural patterns trained on the training images and applied to the test images using a set of linear regression models. We used the Sparse Linear Regression model (SLR; Bishop, 2006), which can automatically select important neurons for decoding, by introducing
sparsity into weight estimation through Bayesian parameters estimation with the automatic relevance determination (ARD) prior.

Given a neuron vector sample \( x = \{x_1, x_2, x_3, \ldots, x_d\} \) consisting of \( d \) neurons as input, the regression function can be written as

\[
Y(X) = \sum_{i=1}^{d} w_i x_i + w_0
\]

where \( x_i \) is a number that denotes the amplitude of neuron \( i \) and \( w_i \) contains the weights of that voxel and \( w_0 \) is the bias.

We trained a single Sparse Linear Regression to take IT cortex activity and predict each unit of the DNN hidden layer. In a single regression for a DNN unit, we use 168 IT cortex neurons to train and apply the Sparse Linear Decoder from IT cortex to DNN representation (so \( d = 168 \)). For example, the first fully connected layer (fc6) of VGG-19 contains 4096 units. For extracting fc6 from brain activity, we trained 4096 independent Sparse Linear decoders to predict each of the 4096 unit’s activity for the test images.

Let \( l \) be a single DNN representation unit (from any arbitrary layer). We can model \( l \) as a target variable \( t_l \). Bayesian parameter estimation was performed with the automatic relevance determination prior to introduce sparsity into the weight estimation. Consider the estimation of the weight parameter \( w \) given the training data sets \( \{X, t_l\} \). We can assume a Gaussian distribution prior for the weights \( w \) and non-informative priors for the weight precision parameters. With these priors, we can estimate the joint distribution \( P(w, a \mid X, t_l) \) where \( a \) and \( B \) are the weight precisions (inverse of variance). The joint posterior in this case is actually analytically intractable, so it is approximated using Variational Bayesian Method (Sato et al. 2001).
Natural Image Prior

We use a natural image prior just as how Shen et al. 2019 used it. The natural image prior is a constraint on the reconstructed images during the optimization process. Instead of optimizing over the distribution of every possible image, we optimize over the distribution of natural images as dictated by a trained Generative Adversarial Network. The pre-trained Generative Adversarial Network is a network described in Dosovitskiy & Brox 2016 designed to produce natural images given a code in the form of a vector of size 4,096 units (see “DGN” in Figure 2).

Zero-shot Framework

To display the high generalizability of our reconstruction model, we performed training/testing the model in a zero-shot setting. In machine learning, zero-shot learning is when a model is trained on images of different categories it is tested on. In this study, we perform 8 different train/test splits where for each split, we test on images from a single category that is held-out from training. For example, in one of the runs, we train the model on all images not in the “car” category (which is 320*7=2240 images) and test on all the car images (which is 320 images). We expect our model to be able to reconstruct images of cars from brain activity even though its training data did not contain any cars.

Reconstruction Evaluation

We mainly evaluate the reconstructions using a metric that is typically used in other reconstruction studies (Shen et al. 2019; Cowen et al. 2014) called the “matching accuracy” (and also called the “identification rate”). In matching accuracy, each test image was paired with a ‘lure’ image, which was a different test image. This pairing was repeated such that each of the test images was paired with each of the ‘other’ images. The Euclidean distance between each reconstruction and its corresponding test image (target), as well as the distance between the
reconstruction and the corresponding lure image, was computed. For each of these pairings, if the reconstruction had less distance to the test (target) image than the lure image, the trial was scored as a ‘hit’ (i.e., the corresponding reconstruction was successfully ‘matched’); otherwise it was scored as a ‘miss.’ We then report the number of hits divided by total possible number of hits and report this as a percentage (theoretical chance levels would be 50%).

*Finding Category Specific Information in the Reconstructions*

Since these reconstructions come from activity in IT cortex and IT cortex’s main role is object recognition, we hope that the reconstructions preserve the relevant information for object recognition (in particular, category-specific information). To evaluate the amount of category-specific information stored in the reconstructions, we feed each reconstructed image to a separate Deep Neural Network (VGG-16) not used for creating the reconstructions and attempt to classify object category from each layer’s activations using a linear SVM. If we think of Convolutional Neural Networks as hierarchical feature aggregators, the SVM’s accuracy will tell us the how much category-specific information can be linearly read-out at each level of image features. For example, a high classification accuracy from the first convolutional layer will tell us that there is a lot of category-specific information in the low level image features in the reconstructions since the first convolutional layer is known to capture low-level image features such as vertical or horizontal edges. Since the dimensionality of the convolutional layer activations are very large compared to the fully connected layers, we do PCA on any layer that has greater than 1,000 units and use the first 1,000 principal components for classification. Since PCA is a linear-based technique, this will still tell us what information can be linearly read-out from the reconstructions. For training/testing the SVM’s, we did 8-fold Stratified Cross Validation (so there was an equal representation of each category in both the train and test set).
Results

Example Reconstructions

Figure 3 contains example reconstructions. As one can see, with these images, the reconstructions still resemble the original images despite holding out entire categories during model testing.

![Example Reconstructions](image)

Figure 3. Example reconstructions across categories (cars, planes, animals, boats, tables, chairs, and fruits). We show 1 sample stimulus image that the monkey saw and its reconstruction from brain activity per category. Each true image is directly to the right of its reconstruction and each reconstruction is directly to the left of its true image. In our study, each reconstruction was generated from training a model that has not “seen” any image in the same category.

Reconstruction Matching Accuracies
Figure 4 shows the matching accuracies across categories. The matching accuracies were all very much above chance levels. Although we calculated matching accuracies within each category, we also calculated an overall matching accuracy by concatenating all images across categories and found it to be \(~77\%\), which indicates that every image in the entire stimulus set were properly matched. Results indicate that a) reconstruction quality do not significantly differ across category and b) reconstructions are (on average) consistently more similar to their true images than any other image in the test set. To our knowledge, this is the first study in reconstructing visual stimuli from single neuron recordings, which means we have no benchmark matching accuracies to compare to. However, our matching accuracies are on par with those from Shen et al. 2019, which is currently considered the state-of-the-art results for reconstructing visual stimuli from fMRI activity.

Figure 4. Matching accuracies across categories. Each matching accuracy was above chance (50%).

Category-specific Information Across Layers

We did three different iterations of the analysis described in “Finding Category Specific Information in the Reconstructions” of the methods section: one in which we trained the SVM’s on real image activations and tested on real image activations, one in which we trained on reconstructed image activations and tested on reconstructed image activations, and one in which
we trained on real image activations and tested on reconstructed image activations. These are shown in Figure 5.

**Figure 5.** Results from SVM analysis described in Methods section. The classification accuracies tells us how much category specific information exists in the reconstruction (for the level of image features that a particular DNN layer encodes). Error bars indicate Standard Errors.

For training/testing on real images, we see the expected result that the level of category-specific information gets higher as you go deeper into the network (which makes sense, because these networks are optimized to classify images into different categories using the last layer). Note that the fc8 accuracy is lower, because the categories that correspond to Imagenet (the dataset VGG is trained on) do not perfectly map onto the categories we have in these images (also, the stimulus images are vastly different from Imagenet images). For training/testing on reconstructed images, a positive result we see is that there is category specific information in the reconstructions since the accuracies across layers are all above chance. However, the pattern across layers is uniform rather than gradually increasing in the real image analysis. There is higher category-specific information in lower-level image features in the reconstructions than in the real images, which could be because we are reconstructing these images from IT cortex, which picks up more on categorical information due to its role in core object recognition. There is lower category-specific information in higher-level image features in the reconstructions than
in the real images, which may be due to the reconstructions not quite having enough categorical information as the real images. The third SVM analysis (train on real images and test on reconstructions) had accuracies at chance levels across all layers except the very last ones, which indicates that the specific categorical information in the reconstructions is actually different than that of the real images for all image features except for higher-level image features encoded by the fully connected layer.

**Discussion**

In this study, we were able to successfully use a machine learning approach to reconstruct complex visual stimuli from neural recordings taken from *Rhesus Macaque* IT cortex by taking advantage of the internal representations of the hidden layers in Deep Convolutional Neural Networks. In particular, we train linear decoding models to decode CNN activations from IT neural recordings and used an optimization procedure in order to optimize an image that matches the same layer activations in the CNN as the ones we decoded from brain activity. In the optimization procedure, we utilize a Generative Adversarial Network (GAN) to constrain the space of our generated images to the space of natural images that the GAN can produce. Our reconstruction model was trained within a zero-shot context, which means we held out entire categories during training. For example, our model was able to reconstruct pictures of planes from brain activity despite all planes being held out from the training set.

One may argue that in order to properly reconstruct visual stimuli from brain activity, one would need activity from the entire Visual System. Most previous studies in reconstruction such as Shen et al. 2019 use activity from the entire visual system (usually in the form of fMRI) to do these reconstructions. However, this study shows that one can get quite far in reconstructing visual stimuli from just using brain activity localized in IT cortex by taking advantage of the
representations in fully-trained Deep Convolutional Neural Networks. This may be because Deep Convolutional Neural Networks is giving us enough prior information to be able to properly reconstruct visual stimuli using just the information encoded in IT cortex.

The success of the reconstructions was indicated by the matching accuracies (the standardized way to evaluate reconstructions from brain activity) seen in Figure 4, which were all way above chance. To our knowledge, this is the first study in reconstructing visual stimuli from single neuron recordings, which means we have no benchmark matching accuracies to compare to. However, our matching accuracies are on par with those from Shen et al. 2019, which is currently considered the state-of-the-art results for reconstructing visual stimuli from fMRI activity. Although we achieved good matching accuracies, our categorical information analysis indicated that, although there is indeed categorical-level information within the reconstructed images, it is actually different from the categorical information in the true images. From our results in Figure 5, it may be possible that reconstructing images from IT cortex activity, the brain region known to be heavily involved in core object recognition, will result in images that uniformly spreads categorical information through low-level to high-level image features (instead of it gradually increasing it from low to high level image features). However, ceiling effects from the reconstructions may also be a reason in why categorical information was uniformly spread across levels of image features. Performing our study on other brain regions besides IT cortex (such as lower level visual areas in V1-V4) would help us more concretely determine which hypothesis is more likely.

Overall, our model was able to successfully take advantage of internal CNN representations to reconstruct visual stimuli belonging to previously unseen categories from neuronal activity localized within Rhesus Macaque IT cortex. Reconstructing visual content from
brain activity remains a highly exciting problem that can get us closer to the future of both better understanding our incredibly efficient and complicated visual system as well as developing more robust brain computer interfaces. Utilizing advances in deep learning has greatly aided progress in the reconstruction problem and will remain an exciting line of work.

References