Modelling How the Brain Represents Visual Information

Project Report: Advanced Machine Learning (Autumn 2023)

Course Code: KSADMAL1KU

Aleky Gałkowski
agac@itu.dk

Emily Krüger
ekry@itu.dk

Frederik Bredgaard
frebr@itu.dk

Raghav Vacher
ragv@itu.dk

***Abstract*—In this study, we explore the challenging task of predicting brain activity in response to visual stimuli, aiming to deepen our understanding of how the visual cortex processes information. Employing a range of convolutional neural networks, our approach is grounded in feature extraction from images, followed by a regression model to predict cortical responses. However, our findings highlight the complexities inherent in this task. The models, despite their varied architectures, consistently underperformed with many failing to establish a meaningful correlation between predicted and actual fMRI signals. This outcome points to the need for more nuanced models and methods, potentially incorporating aspects like sequential image processing and alternative neural network architectures, to more accurately capture the intricate workings of the visual cortex.**

I. Introduction

This project is inspired by the Algonauts 2023 challenge, which seeks to predict fMRI brain activity from input-images [1]. In expansion to the challenge goal of building a well-predicting model, we also aim to compare the different layer activations to specific regions in the brain. This will allow us to explore, whether (or how) the model architecture corresponds to how the visual cortex represents and processes visual information.

A. Motivation

Aim of this project is to predict cortical surface visual area brain activity from visual stimuli as measured by fMRI. Through this, we seek to develop an understanding of how the brain responds to and models visual information and to develop a model that generalises to unseen images and to subjects it was not trained on. Currently, the most widely applied method for this type of task involves extracting features from the stimulus using a pretrained model and then mapping these features to fMRI activity using subject-specific linear regression models [2]. Our approach follows this framework while enabling the application of the model on other datasets.

II. Data

The challenge data consists of occipital lobe cortical surface BOLD responses from 8 subjects in response to around 9000 unique and 1000 shared images from the COCO dataset [3]. Subjects saw each image three times for a total of around 30,000 trials per subject, and fMRI activity for an image represents the average activity across these three trials. Each subject was scheduled to complete 40 sessions but not all subjects completed all sessions, leading to some missing data. The BOLD response data is split into separate 2D arrays for the left and right hemisphere for each subject. Each array has the number of images as rows and cortical surface vertices as columns. For all subjects except 6 and 8, the numbers of vertices recorded in the left and right hemisphere are 19,004 and 20,544, respectively. For subjects 6 and 8, some data is missing, resulting in only 18,978 left hemisphere and 20,220 right hemisphere vertices being recorded for subject 6 and 18,981 and 20,530 being available for subject 8. Descriptive statistics for each subject are reported in Table II in the appendix.

In brain imaging, noise is invariably a serious concern. Each subject viewing every image several times might on the one hand reduce noise because activity unrelated to the image becomes less salient. However, the high resolution of MRI imaging makes aligning voxel-to-voxel between scanning sessions near-impossible, introducing a different source of noise.

One notable feature of this dataset stems from its collection method, namely that both stimulus presentation and the rest period between trials are short. This means that each trial likely depends at least in part on its predecessor. This feature was exploited by the winner of the Algonauts challenge to encode memory of several images preceding the currently predicted trial activity [4]. While this approach adds to the understanding of visual- and working memory, we decided against encoding memory of previously seen images as our objective was to predict visual cortex activity from a single stimulus image.

We split the data into 80% train 10% validation and 10% test split. To ensure that all participants are roughly equally represented in each split we sampled these percentages from each subject. To prevent deliberately sampling from a particular part of the sequence of image trials per subject (e.g. the beginning or end of the sequence), we randomly sampled throughout the sequence.

A. Preprocessing

The fMRI challenge data is preprocessed by averaging activity for a given subject's three trials per stimulus image and z-socring within each scanning session. Images come preprocessed only by cropping from their original, varying, sizes to a fixed 425x425 pixels. Before feeding a given image to our model we perform additional preprocessing by resizing to 224x224 pixels and normalising pixel values according to values recommended in torch documentation [5] (mean = [0.485, 0.456, 0.406], std = [0.229, 0.224, 0.225]).

The missing data for subjects 6 and 8 somewhat complicate the prediction process as it would force our model to adapt its predictions to different output sizes. To avoid this issue, we preprocess the fMRI data using principal component models fit to each subject to reduce the roughly 40,000 dimensions to 100 principal components which our model will predict. Our experiments show that marginal explained variance begins to drastically decrease beyond roughly this number of components (see example plot in Appendix Figure 5). These 100 components explain about 65 to 70% of the variance (depending on the subject) in the fMRI data when they represent both hemispheres. Our experimentation showed that the same number of components explain a greater portion of total variance when the hemispheres are treated as one brain rather than separately. This is likely due to a lack of hemispheric specialisation, particularly in lower level visual areas leading to bilateral associations.

III. Model architecture

The base concept underlying our model is to extract features from the presented image and then linearly process them to predict activation levels for each of the 100 principal components representing the true signal. These predicted components can then be transformed to the same dimension as the true signal.

The majority of the forward pass through the model consists of extracting features. We have employed various versions of the ResNet [6] model as well as AlexNet [7] in our experiments. Regardless of the specific model applied, the function applied is one which takes the input from its preprocessed shape of 224x224x3 through a number of feature-extracting filters and convolutions ultimately ending up with a much smaller resolution feature map represented in more channels. Finally, the resulting feature tensors are flattened, allowing them to pass through the appropriate linear layers for regression.

The regression module of our model is inspired by the Algonauts challenge submission by BlobGPT [8] and aims to strike a balance between generalisability and performance. Namely, we wanted to take advantage of the fact that the training data came from relatively few subjects to let the model detect subject-specific patterns while maintaining an applicable model for the hypothetical $9^{th}+$ subject. We do this by passing the extracted features through a generalised sub-model as well as a subject-specific model and treating outputs from both models equally by averaging them before the final regression layers. This method lets the model reserve parameters to adjust for subject-specific nuances but also allow our model to make predictions when it does not know which subject it is predicting by simply bypassing the subject-specific layers. Both the shared and subject-specific sub-models consist of four linear blocks of three layers with ReLU activation and batch normalization at the end of each block.

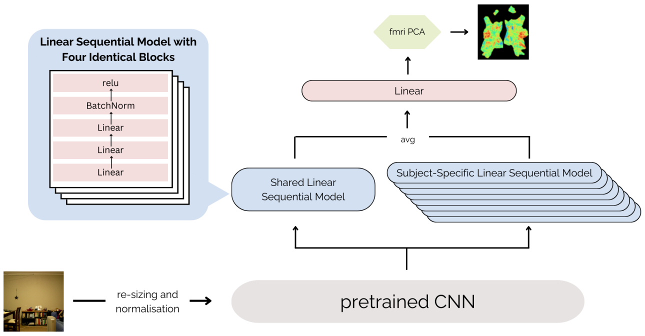

Linear Sequential Model with Four Identical Blocks
relu
BatchNorm
Linear
Linear
Linear
fmri PCA
Linear
avg
Shared Linear Sequential Model
Subject-Specific Linear Sequential Model
re-sizing and normalisation
pretrained CNN

Fig. 1. Model overview

During training we calculated loss training as the mean of squared errors rather than the challenge metric. This allows us to calculate a meaningful and somewhat interpretable loss value on the PCA components that our model predicts and saves us the step of converting back to the full dimension of the original data at each training step. The full loss function used during training is described in Equation (1).

A. Experiments

*1) **Training process:*** Our experimental workflow was executed on High-Performance Computing (HPC) clusters, which allowed for efficient processing and evaluation of various neural network configurations. Utilizing the computing power of HPC and customised bash scripts, we implemented a training regimen with continuous monitoring, generating plots at the end of each epoch, and an early stopping mechanism. This real-time feedback enabled us to obtain preliminary indications of overfitting and model performance (see Figure 2).

*2) **Picking the backbone model:*** The BlobGPT submission [8] uses a pre-trained trunk model to extract features of the image before passing them on to the next layers. We diverged from this approach by opting for Convolutional Neural Networks (CNNs), selecting the ResNet [6] and AlexNet [7] architectures for their proven capabilities in feature extraction that might parallel the human visual cortex's processing. We began with ResNet18 for its simplicity and quick training times, which facilitated early experimentation. As our research evolved, we included more complex models like larger ResNets and AlexNet for their unique architectural benefits—ResNet's ability to learn through added depth and

AlexNet's foundational success in image recognition tasks.

B. Fine-tuning

*1) **Principle Component Analysis**:* During the fine-tuning, we tested several configurations of the models with both PCA with 100 components, and without using PCA. We found that the models using PCA generally tend to overfit, however surprisingly resulting in a similar loss on the test set as the model without PCA.

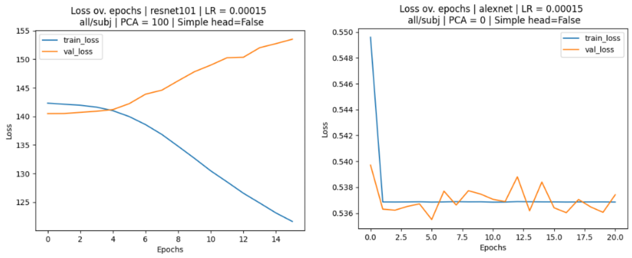


Fig. 2. Example loss plots

When using PCA, we compiled a separate PCA model for each subject to account for the different lengths of voxel tensors between them. One strong limitation in our approach was the way we standardise varying tensor sizes between the subjects, when not using PCA. Because of time restraints, we opted for an easy approach to end-trim all subjects to match the lengths of the shortest hemisphere arrays, therefore losing information from a few hundred voxels per trial for subjects 6 and 8.

*2) **Regularization**:* As a result of our testing we applied two forms of regularization; L2 regularization and dropout. By gradually increasing from 25% dropout while monitoring performance we eventually worked our way to 50% dropout for nodes in the sub-model that construct both the shared and subject-specific layers of the regression model.

While several of our models struggled to fit to the data in transferable ways, we found that adding L2 regularization alleviated the problem. The L2 regularization term adds a penalty to the loss function that is proportional to the squared values of parameters. This pushes parameters to be smaller, letting the model rely less on any specific parameters. We implemented the L2 parameter $\lambda = 0.01$ such that

$$L = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 + 0.01\sum_{j=1}^{p}(w_j + b_j)^2 \tag{1}$$

where L is the total loss, p is the number of parameters, w_j are the weights of the model, and b_j are the biases of the model.

These regularization techniques make it less likely for the model to overfit to the training data. This is particularly important given the noisy nature of fMRI data.

*3) **Simple vs sequential head**:* After many attempts that still resulted in overfitting, we considered the risk of the model head being overly complex, and not learning effectively. Therefore, we considered a simple head in addition to the previously used sequential one. The simple head consisted of a single linear layer, as opposed to the sequential head, which was a multi-layer architecture involving several blocks of linear layers with batch normalization and ReLU activations, followed by dropout for regularization.

IV. Results

A. Metrics

1) Challenge Metric: The metric used in the Algonauts challenge is mean noise-normalized Pearson correlation (MNNPC) which is computed as

$$\text{MNNPC} = \frac{1}{n}\sum_{v=1}^{n}\frac{R_v^2}{NC_v} \times 100 \tag{2}$$

where n is the number of voxels, NC is the noise ceiling and Rv is the Pearson correlation between ground truth and predicted fMRI data, G and P, respectively, calculated as

$$R_v = \frac{\sum_t(G_{v,t} - \overline{G}_v)(P_{v,t} - \overline{P}_v)}{\sqrt{\sum_t(G_{v,t} - \overline{G}_v)^2(P_{v,t} - \overline{P}_v)^2}} \tag{3}$$

where v is the index of vertices across subjects and hemispheres, t is the index of test stimuli images, and $\overline{G}$ and $\overline{P}$ are the ground truth and predicted fMRI activities averaged across test images.

According to the original publication of the dataset [9], the noise ceiling is calculated using the individual trials for each image. Unfortunately, these are not accessible to us, making us unable to calculate the MNNPC. As such, we will resort to reporting the mean Pearson correlation and the MSE.

*2) **Reported Metrics**:*

a) Mean Normalized Pearson Correlation: We report the mean Pearson correlation converted to the range [0, 1], following the example of other submission (e.g. [8]). Correlation for a given vertex is computed as in Equation (3) and normalized as

$$C_v = 1 - \frac{R_v + 1}{2} \tag{4}$$

Now, we can take the mean of all voxel-wise correlations to obtain the final score in mean normalized Pearson correlation (MNPC):

$$MNPC = \frac{1}{n}\sum_{v=1}^{n} C_v \tag{5}$$

b) Mean Squared Error (MSE): In addition to the MNPC, we also report the MSE. The mean of squared errors between predicted voxel values and actual voxel values is calculated as

$$MSE = \frac{1}{n} \sum_{v=1}^{n} (G_v - P_v)^2 \tag{6}$$

This metric can add nuance to the comparison of model results on the basis of MNPC by adding greater emphasis to those predictions that are very wrong.

B. Evaluation

1) ***Performance Metrics:*** The two performance metrics as shown in table I indicate, that all models perform very similarly, with only slight differences between AlexNet or the different ResNet models or a simple head in comparison to the sequential models. Additionally, none of the training parameters, such as learning rate or batch size seem to lead to a tendency towards a better or worse performance.

Notably, a MNPC score of 0.5 indicates, that there is no correlation between the prediction and the ground truth activation. The majority of the models yielded NaN values. Upon further inspection we found, that those models predict the exact same activation values for all input images for some subjects. These constant values lead to NaN correlation scores. To further understand the high similarity of MSE values, we inspected exemplary predictions of different models. When not constant across pictures, predicted values are very close to zero. Interestingly, even a model that predicted only 0 would lead to an MSE of 0.5. We must therefore conclude, that none of our models trained well on the given data and yielded satisfactory performance.

2) ***ROI-specific performance:*** Despite all models performing similarly poorly, we further inspected the performance for specific Regions of Interest (ROI) to see, if the model performed exceptionally bad in some ROI, yet better in others. Visualizing the predicted and the ground truth performance for a randomly picked image alongside the squared error shows, that this seems to indeed be the case (Figure 6).

While the squared error is close to 0 in most areas, there is one central region, where our model does predict poorly. However, this area does not seem to lie within one single ROI as defined by the Algonauts challenge. When visualizing MNPC for each ROI averaged across subjects (see figure 4), we see that values are very similar across all ROI. The only exception are those ROI that yielded NaN correlation scores, indicating that the model predicted constant activations across images for these regions.

3) ***Layer-Specific Performance:*** To gain insights about how the model architecture might correspond to the structure of the visual cortex, we extracted predictions for all images from each of the convolutional and fully-connected layers of the pretrained AlexNet model. As the visualisation in the appendix (figures 7- 13) show, there are no strong differences in how well each of the layers predict each of the ROI. While there are marginal differences between the correlation scores for each of the layers, they do not seem to differ systematically: Neither of the convolutional or fully-connected layers stand out with a significantly higher or lower correlation score for one ROI than the other layers. We therefore cannot use our model to make inferences about the structure of the visual cortex. Considering the overall poor performance of the model, this is unsurprising.

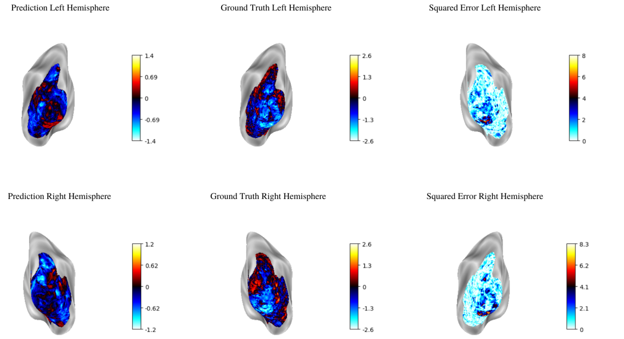

Prediction Left Hemisphere
1.4
0.69
0
-0.69
-1.4
Ground Truth Left Hemisphere
2.6
1.3
0
-1.3
-2.6
Squared Error Left Hemisphere
8
6
4
2
0
Prediction Right Hemisphere
1.2
0.62
0
-0.62
-1.2
Ground Truth Right Hemisphere
2.6
1.3
0
-1.3
-2.6
Squared Error Right Hemisphere
8.3
6.2
4.1
2.1
0

Fig. 3. example for predicted vs. ground truth activation and squared error (test set: subject 3, image 370)

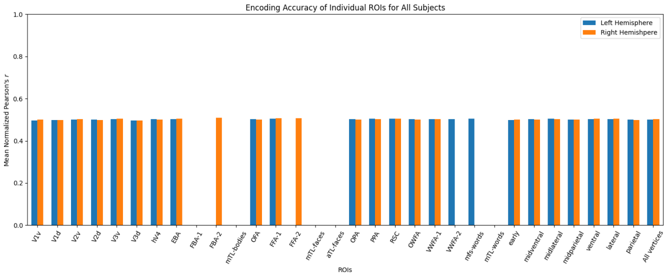


Fig. 4. MNPC Scores per ROI

V. Discussion: Shortcomings and Future Work

The remarkably similar results of various models gives reason for pause. MNPC of 0.5 corresponds to no detectable relationship between model predictions and observed fMRI signal. While the noisy nature of brain activity might ease the severity of this damning performance metric, additional perspective does not – in particular, an infinitely less computationally demanding algorithm simply predicting the mean activity, 0, would obtain the same MNPC of 0.5 and a very similar MSE to most models of roughly 0.5.

A. How Predictable Is This Dataset?

While images display different objects and varied scenes, they likely share some fundamental features. Not least, the fact that they are all displayed as two-dimensional representations

TABLE I
MODEL ARCHITECTURE COMPARISON

backbone	submodel	LR	batchsize	PCA	MNPC	MSE
AlexNet	sequential	0.0003	64	100	0.50	0.9
ResNet152	simple	0.00015	64	-	0.499	0.54
ResNet18	simple	0.00001	32	-	NaN	0.54
AlexNet	sequential	0.00015	64	-	NaN	0.54
ResNet18	sequential	0.00015	128	100	NaN	0.54
ResNet34	sequential	0.00015	32	-	NaN	0.54
ResNet101	sequential	0.00015	64	100	NaN	0.54
ResNet101	sequential	0.0003	64	-	NaN	0.54
ResNet152	sequential	0.0003	64	-	NaN	0.54
ResNet152	sequential	0.00015	64	-	NaN	0.54

of three-dimensional scenes and presented in the same part of the visual field. This, along with the fact that trial runs during collection were so numerous and had such short rests between them, might mean that an even greater amount of the variance in fMRI activity is caused by factors external to the current image. Such include lingering activity from previous trials, a lack of focus due to faltering vigilance during long and strenuous scanning sessions, or activity entirely unrelated to any of the presented stimulus images. While all of the fMRI data provided concerns areas specialized in visual processing, no part of the brain is so specialized that it is entirely unaffected by other factors.

Perhaps previous images, i.e. lingering activity or 'memory' of similar yet distinct stimuli has an immense yet unaccounted-for effect on the total MRI image, making it extraordinarily difficult to extract the activity relying only on the currently displayed image as input. This suggests that future work might follow the example set by the top submission [4] and treat the images and corresponding fMRI activity as a sequence, where the next fMRI image is to be predicted based in part on the previous stimulus images.

B. Feature Extractor

A considerable potential for future work may lie in improving the feature-extracting part of the model. While several of the top submissions opted for vision transformers (e.g. [4], [8], [10]), we have applied a convolutional neural network in its place. This decision was made not only due to suggestions to use a CNN in the exam-case description, it is also based on the assumption that this architecture would more closely mirror the processing layers of the visual cortex.

We ended up not using skip-connections to extract the intermediate CNN layers, like some of the submissions did, which arguably could have made the model closer to being bio-inspired, as it would account for independent connections between the vision processing layers and other parts of a human brain. However, the poor predictive power of these intermediate layers suggests that it would not improve the performance of our chosen models. Furthermore, it warrants giving even more attention to the backbone model as a potential cause of bad model performance, and experimenting with vision transformers and intermediate layer feature extraction, taking inspiration from the top submissions.

C. Training Methodology

Our model distinguishes between subjects by the subject ID passed to it along with the input image. This implementation means that the IDs in the batches that the model trained on had to be homogeneous to pass the entire batch through the same layers. We accomplished this by building a custom sampler, which ensures that images are sampled randomly but always from the same subject. One derived effect of this is that each time the trainer back-propagates through the model to update its parameters, it does so on the basis of only one subject's predictions. Even with a low learning rate, this could make it difficult for the model to adjust the subject-specific models to the patterns that are unique to the given subject and push those adjustments to the shared model. This can hinder the model's ability to learn the more generally applicable patterns. One potential solution might be to work with the subject IDs as an embedding attached to each image, allowing us to serve the model with batches containing data from several subjects.

An alternative approach to dealing with missing data for subjects 6 and 8 might be found in imputation techniques such as atlas-based interpolation or padding, or by simply limiting the analysis to ROIs in which all data is available. While zero-padding would be the most straightforward and efficient way of padding the fMRI images, as it corresponds to inserting the mean activity value and therefore does not drastically influence the overall dataset statistics, there is an issue with padding in this case: We do not know to which areas - i.e. corresponding to which voxels - the missing data would belong. As such, padding the end of the array would risk mapping voxels to incorrect positions in the brain.

Atlas-based interpolation involves registering the fMRI data into a standardized space. Here, the transformed coordinates can, in theory, be used to infer more meaningful voxel values via k-nearest neighbours or similar methods.

The more complex version of our model includes batch normalization layers at the end of each block to prevent vanishing gradients and improve the model's ability to learn. Unfortunately, we have discovered near the deadline of this project, that the default PyTorch implementation of L2 regularization (specified as weight decay in the optimizer) does not discriminate between parameter types in its application of the penalty [11], [12]. This means that the γ (scale) and β (bias) values of each BatchNorm layer are also subject to the penalty. This limits the intended effect of both scaling and normalizing layer outputs, which may have led to such problems as unstable activation distributions throughout the model and vanishing gradients - a theory further supported by our predicted values being very close or equal to zero, which could indicate weights also being forced towards zero.

D. Regression Model

In considering potential modifications to our model's architecture, two intriguing questions arise. Firstly, what would be the implications of omitting subject-specific layers from the model? This change could potentially streamline the model's learning process, allowing it to focus more on general patterns rather than individual subject nuances. Secondly, the possibility of simplifying the model's output processing by directly feeding the output of the convolutional neural network¡ (CNN) into a single or sequential block — essentially, a straightforward, single head — merits exploration. Such a modification could lead to a more efficient and less complex model, potentially enhancing its performance by focusing on core features extracted by the CNN.

In exploring alternatives to our current model, we could consider a shift towards simpler approaches like linear regression, as in the Algonauts tutorial [13] which utilized the straightforward linear regression model from sci-kit learn [14]. Additionally, the potential of ensemble models such as random forests or gradient boosting offers another solution, promising potentially improved performance.

VI. Conclusion

Our study revealed that the models employed showed no significant correlation between predictions and observed fMRI signal, indicating a lack of detectable relationship. Despite various model architectures and training methodologies, all models performed similarly poorly, with many predicting constant activation values across images. The analysis suggests that other factors, such as memory of previous stimuli and external influences during trial runs, might play a significant role in fMRI activity, complicating the prediction process. The feature extraction approach, mainly relying on CNNs, may require reevaluation. Future improvements should further focus on incorporating sequential image processing to account for the influence of preceding stimuli. Additionally, refining the feature extraction process to more closely mirror the visual cortex's processing and employing techniques like intermediate layer extraction could yield more promising results. Finally, improvements to the training methodology to address potential problems of vanishing gradients and subject-specific batches might enhance performance.

Our project highlights the intricate nature of brain activity and the need for more sophisticated models and methodologies to better capture and understand the nuances of the visual cortex's response to stimuli.

References

[1] A. T. Gifford, B. Lahner, S. Saba-Sadiya, M. G. Vilas, A. Lascelles, A. Oliva, K. Kay, G. Roig, and R. M. Cichy, "The Algonauts Project 2023 Challenge: How the Human Brain Makes Sense of Natural Scenes," *arXiv.org*, 2023. [Online]. Available: https://arxiv.org/abs/2301.03198

[2] T. Naselaris, K. N. Kay, S. Nishimoto, and J. L. Gallant, "Encoding and decoding in fmri," *NeuroImage*, vol. 56, no. 2, pp. 400–410, 2011.

[3] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dollar, "Microsoft COCO: Common Objects in Contex," *arXiv.org*, 2015. [Online]. Available: https://arxiv.org/abs/1405.0312

[4] H. Yang, J. Gee, and J. Shi, "Memory encoding model," *arXiv.org*, 2023.

[5] PyTorch Team, "PyTorch Vision Models," https://pytorch.org/vision/stable/models.html.

[6] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2015. [Online]. Available: https://api.semanticscholar.org/CorpusID:206594692

[7] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, pp. 84 – 90, 2012. [Online]. Available: https://api.semanticscholar.org/CorpusID:195908774

[8] C. Lane and G. Kiar, "A Parameter-efficient Multi-subject Model for Predicting fMRI Activity," *arXiv.org*, 2023. [Online]. Available: https://arxiv.org/abs/2308.02351

[9] E. J. Allen, G. St-Yves, Y. Wu, J. L. Breedlove, J. S. Prince, L. T. Dowdle, M. Nau, B. Caron, F. Pestilli, I. Charest, J. B. Hutchinson, T. Naselaris, and K. Kay, "A massive 7T fMRI dataset to bridge cognitive neuroscience and artificial intelligence," *Nature Neuroscience*, vol. 25, no. 1, pp. 116–126, 2022.

[10] H. Adeli, S. Minni, and N. Kriegeskorte, "Predicting brain activity using transformers," *bioRxiv*, 2023.

[11] "Weight Decay in the Optimizers is a Bad Idea, Especially with BatchNorm," https://discuss.pytorch.org/t/weight-decay-in-the-optimizers-is-a-bad-idea-especially-with-batchnorm/16994/9, accessed: 06-01-2024.

[12] PyTorch Team, "BatchNormalization (bn) layer," in *PyTorch Documentation*, 2023. [Online]. Available: ⟨https://pytorch.org/docs/stable/generated/torch.nn.BatchNorm1d.html⟩

[13] A. T. Gifford, B. Lahner, S. Saba-Sadiya, M. G. Vilas, A. Lascelles, A. Oliva, K. Kay, G. Roig, and R. M. Cichy, "Algonauts Tutorial Google Colaboratory — colab.research.google.com," https://colab.research.google.com/drive/1bLJGP3bAo_hAOwZPHpiSHKlt97X9xsUw?usp=share_link#scrollTo=T0hxrG5hysjb, 2023, [Accessed 06-01-2024].

[14] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[15] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," *arXiv preprint arXiv:1912.01703*, 2019.

VII. Appendix

A. Tables

TABLE II
FMRI descriptive statistics per subject

	subject 1	subject 2	subject 3	subject 4	subject 5	subject 6	subject 7	subject 8
mean	0.002182	0.001639	0.002815	0.009924	0.002840	0.001174	-0.000325	0.004358
median	0.006040	0.000896	-0.003185	0.000993	-0.009892	-0.002173	-0.000429	-0.006079
std	0.709001	0.716021	0.757755	0.768562	0.722640	0.760016	0.679135	0.754423
min value	-6.224722	-7.106630	-9.143569	-10.779788	-7.615682	-9.317543	-10.895111	-11.138769
max value	6.395816	7.303041	7.551446	14.346183	7.400172	9.996536	8.991561	10.573892

B. Figures

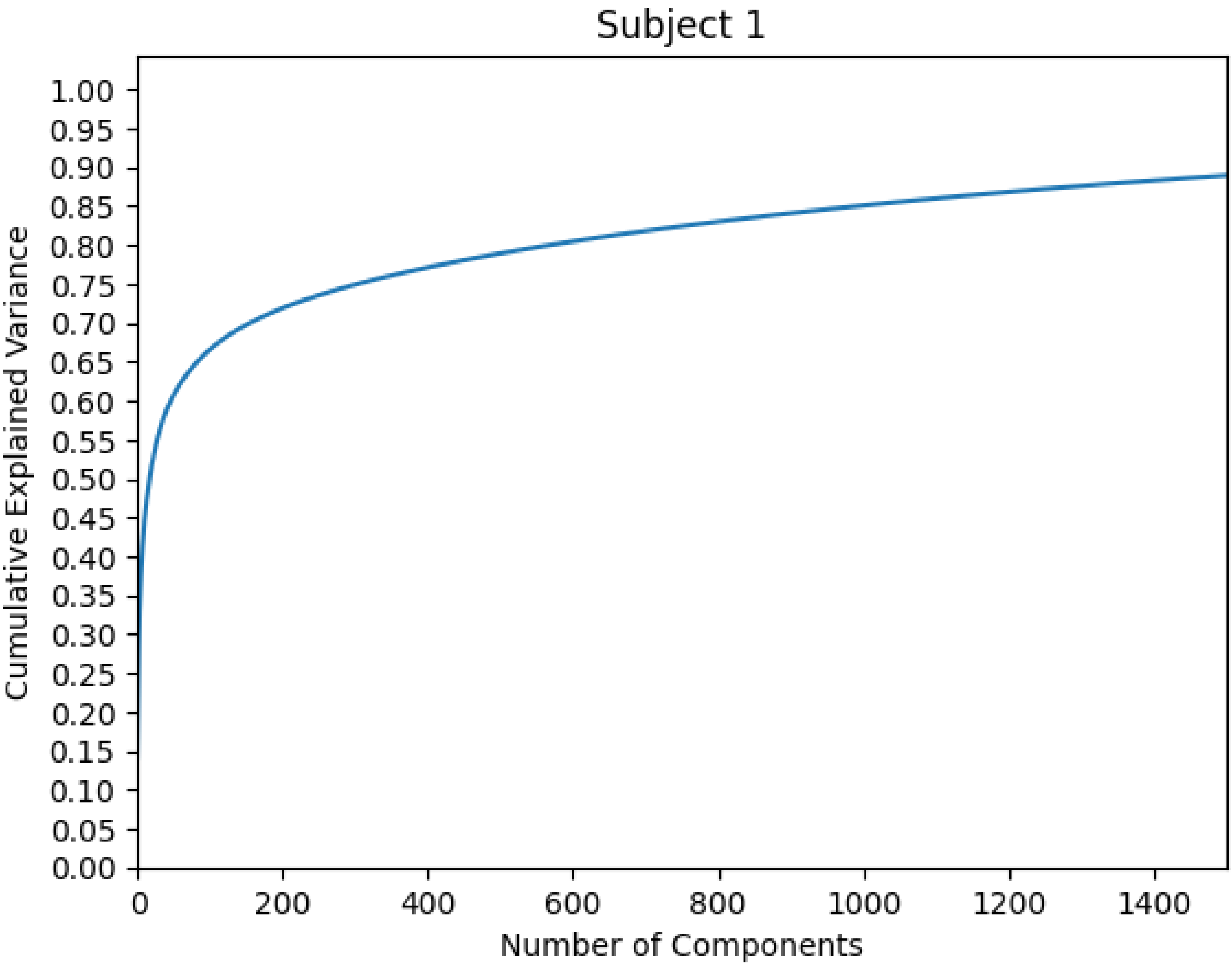
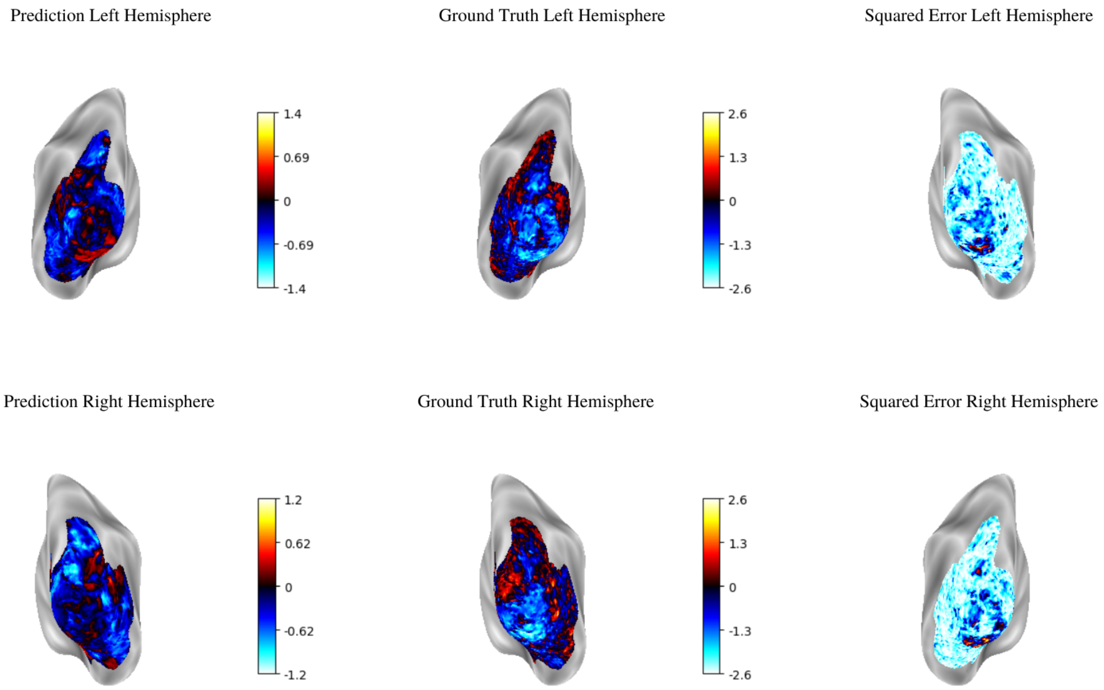


Fig. 5. Cumulative Explained Variance per Component - Subject 1


Prediction Left Hemisphere

Ground Truth Left Hemisphere

Squared Error Left Hemisphere

Prediction Right Hemisphere

Ground Truth Right Hemisphere

Squared Error Right Hemisphere

Fig. 6. example for predicted vs. ground truth activation and squared error (test set: subject 3, image 370)

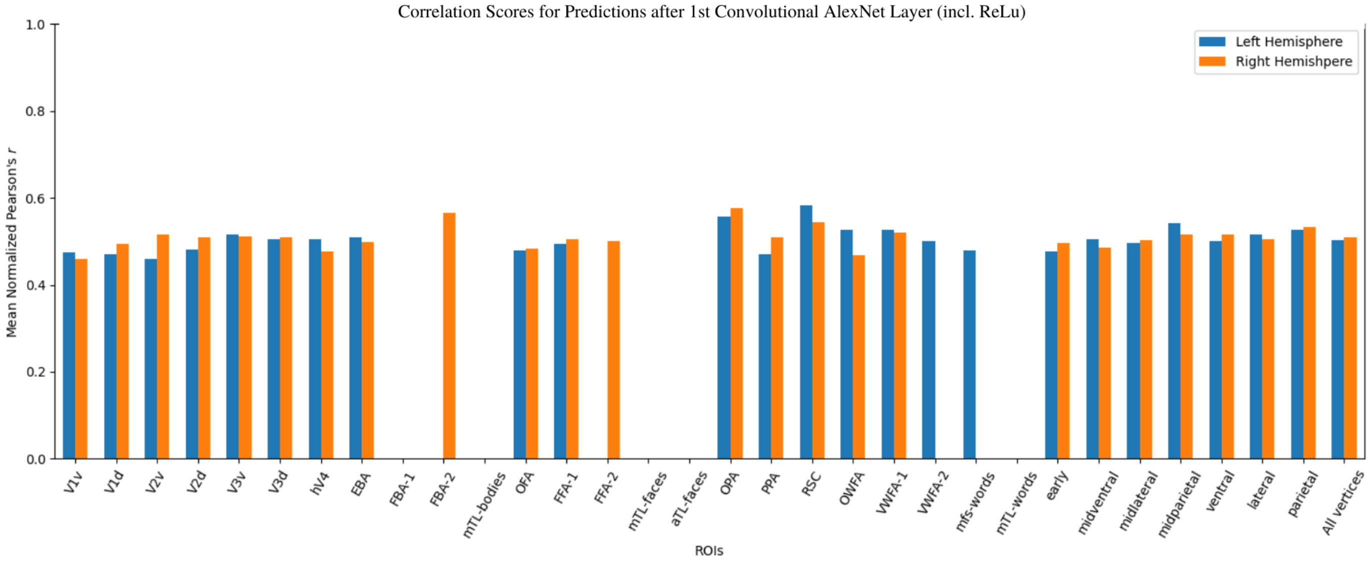

Correlation Scores for Predictions after 1st Convolutional AlexNet Layer (incl. ReLu)

Fig. 7. Correlation Scores 1st Convolutional Layer

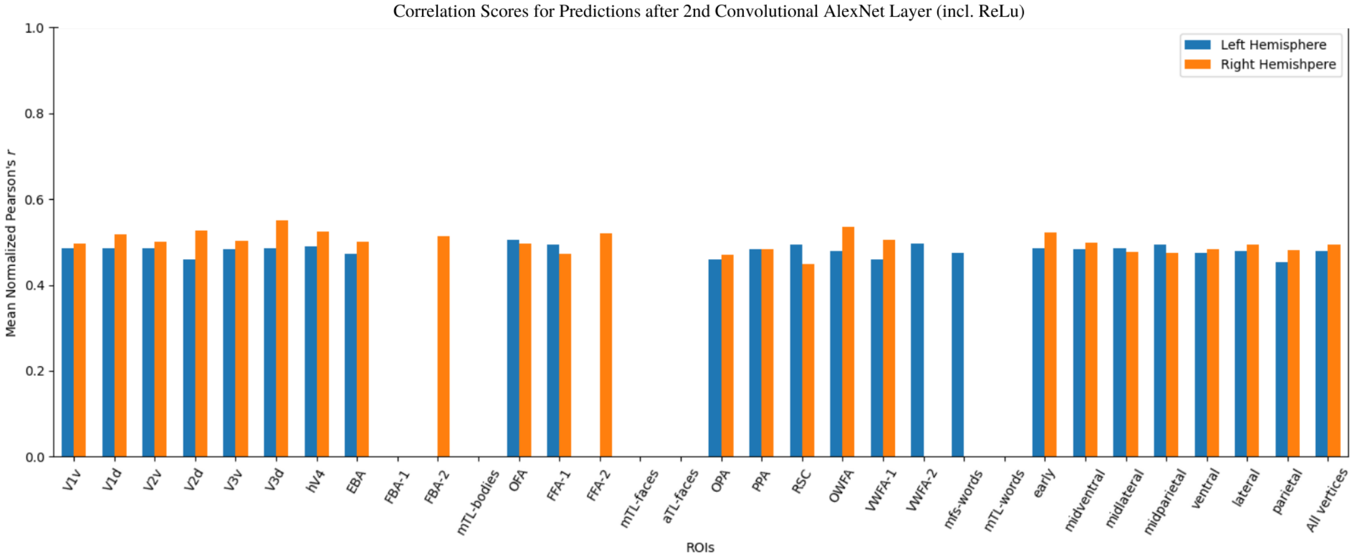


Fig. 8. Correlation Scores 2nd Convolutional Layer

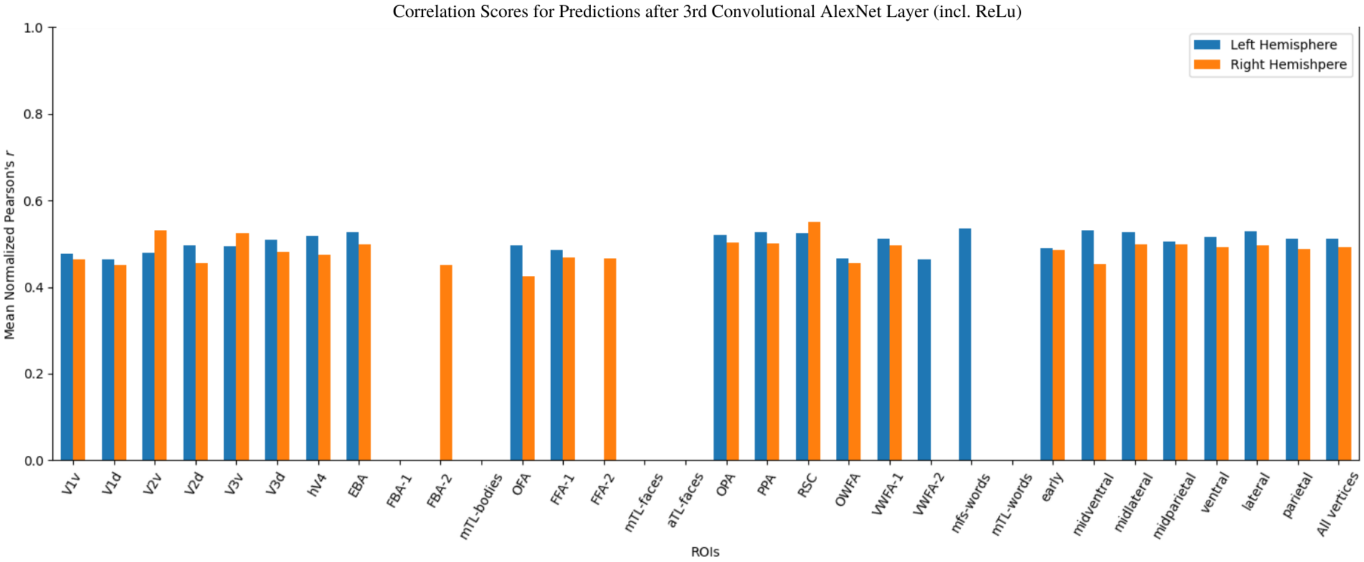


Fig. 9. Correlation Scores 3rd Convolutional Layer

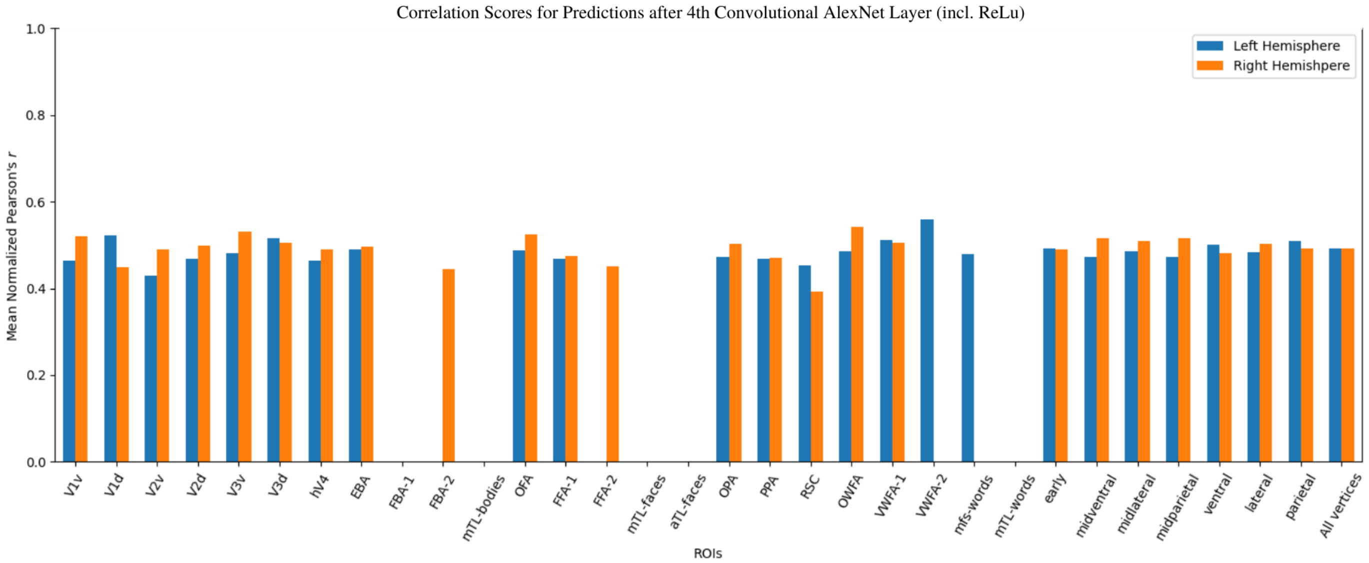


Fig. 10. Correlation Scores 4th Convolutional Layer

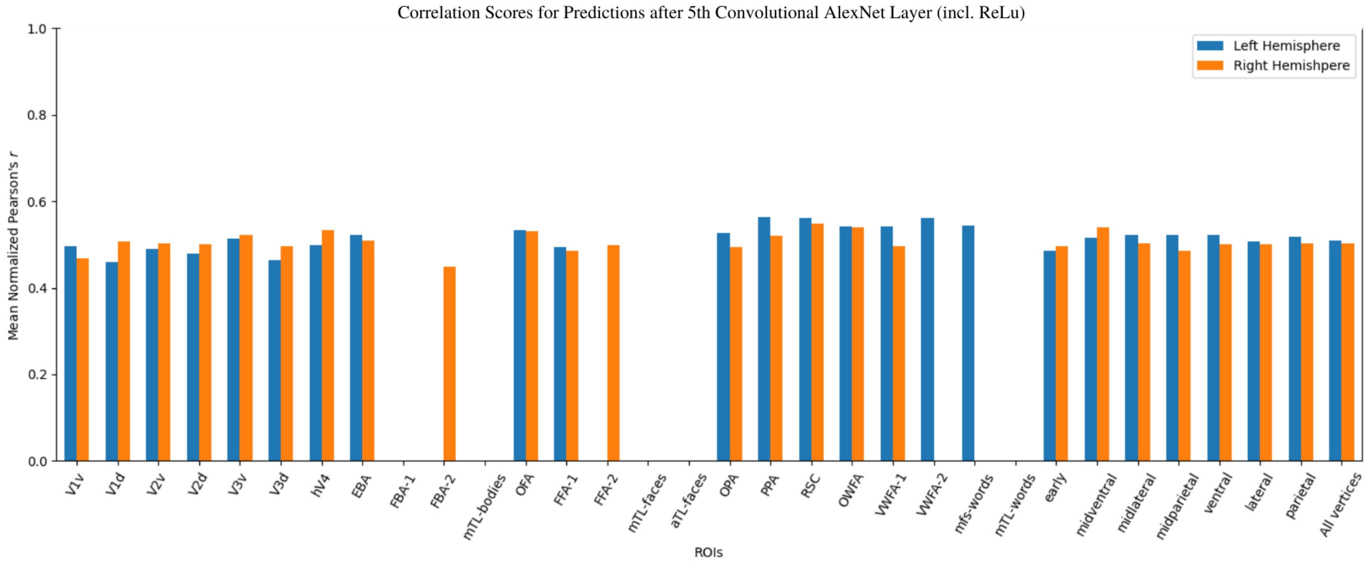


Fig. 11. Correlation Scores 5th Convolutional Layer

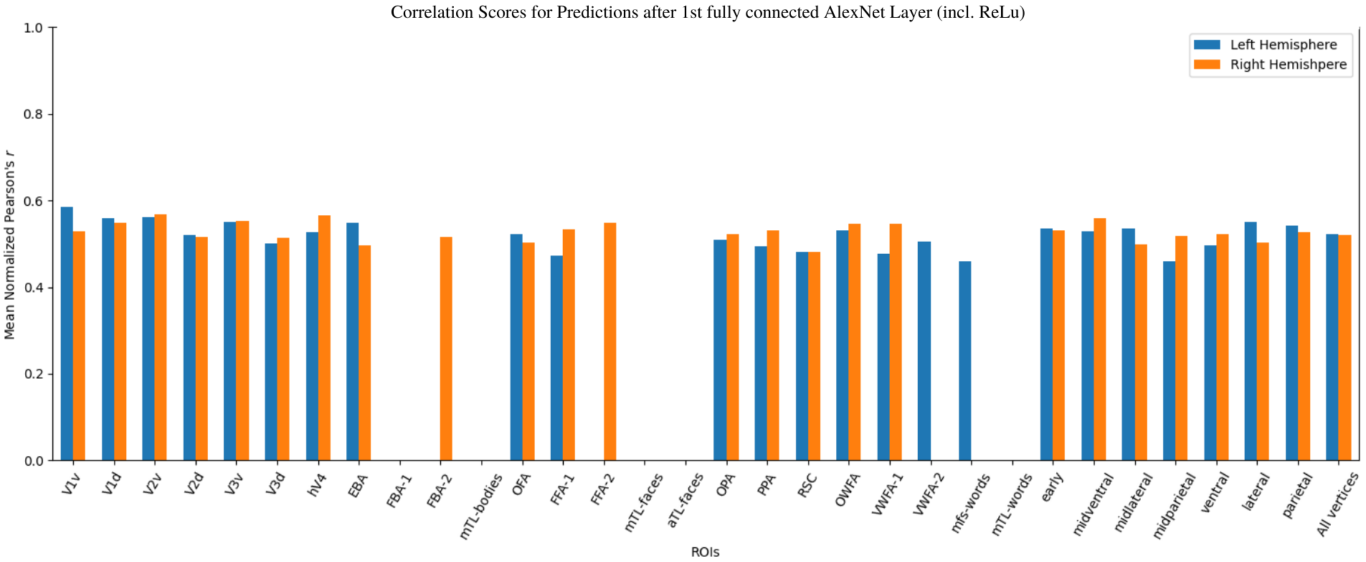


Fig. 12. Correlation Scores 1st fully connected Layer

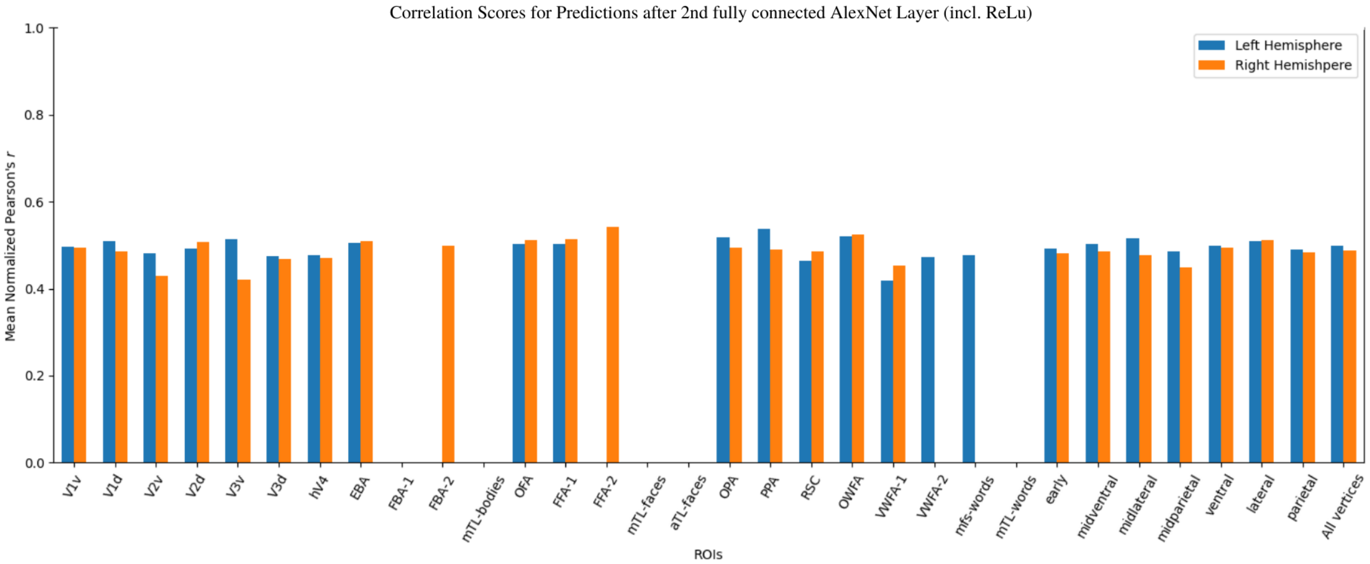


Fig. 13. Correlation Scores 2nd fully connected Layer