# COMPUTER GRAPHICS FROM A NEUROSCIENTIST'S PERSPECTIVE

**Shreya Kapoor & Bernhard Egger**
Department of Computer Science
Friedrich-Alexander-Universität Erlangen-Nürnberg
{shreya.kapoor, bernhard.egger}@fau.de

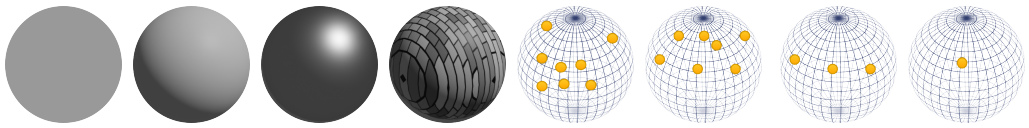
## ABSTRACT

A hallmark of human vision is to recognize objects in complex naturalistic scenes. However, the exact mechanism behind the representations of a three-dimensional scene remains obscure. This study proposes a tool to investigate human perception by using a computer graphics approach. We use three-dimensional object meshes to render synthetic scenes and try to study how these scenes will be represented in the brain. We render a collection of datasets with different appearance and pose variations by changing exactly one property at a time. A model is trained on each of these datasets for a classification task and is then evaluated using alignment metrics; deviations in metrics such as Centered Kernel Alignment (CKA) and Representational Similarity Analysis (RSA) indicate the importance of a particular brain region in representing a particular property. In conclusion, we propose a promising method to study the brain using computer graphics to provide valuable insights into human vision.

## 1 INTRODUCTION

Human vision can easily process complex scenes and perform object recognition, allowing us to perform day-to-day tasks such as scene understanding and spatial navigation (Wardle & Baker, 2020). Getting an idea about the neural underpinnings of these mechanisms is of interest to a variety of disciplines such as computational neuroscience and artificial intelligence. In this work, we propose a tool for studying the mechanism of scene representation in the brain by using a computer graphics approach.

As humans, we can recognize objects in three dimensions from two-dimensional images; there are several important components of 3D perception, including shape and depth cues. For example, we can estimate shape from shading (Bruckstein, 1988), specularities (Savarese, 2014), and texture (Todd & Thaler, 2010) variations. As depicted in Figure 1, subtle variations in appearance change the perception of objects in 3D. This highlights the complex interaction between visual cues and cognitive processes that allow us to infer depth and shape (Spröte et al., 2016). Another interesting phenomenon in human vision is the change in the perception of depth due to viewpoint changes (Deng et al., 2024). Figures 1e to 1h highlight the sampling of camera angles along different perspectives or viewpoints. Hence, for the scientific community, it is interesting to study the factors that affect human visual perception by making viewpoint changes.



(a) (b) (c) (d) (e) (f) (g) (h)

Figure 1: (a) a sphere with no shading (b) a sphere with shading (c) a sphere with specular reflection (d) a textured sphere (e) camera viewpoints with views sampled along the sphere (f) camera viewpoints sampled along the upper hemisphere (g) camera viewpoints along the z-axis (h) constant camera viewpoint

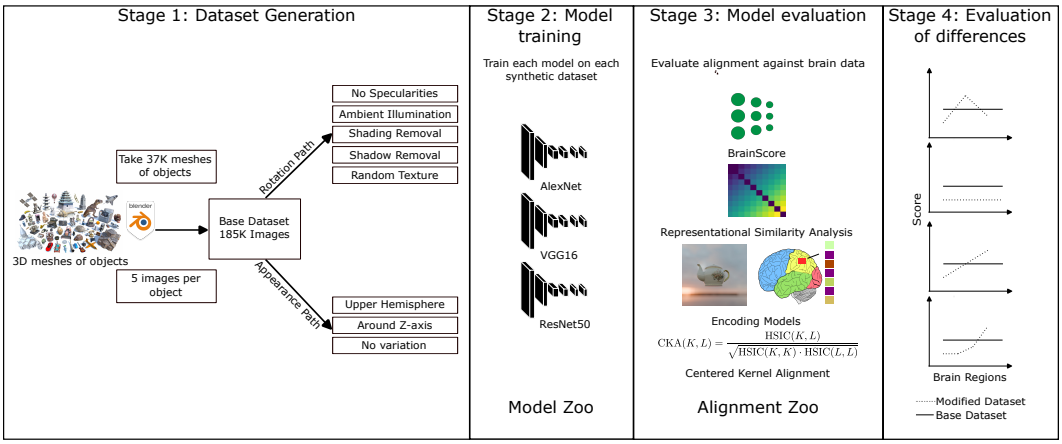



Figure 2: Visualization of the main pipeline of our work. The process begins with three-dimensional meshes of objects, which are rendered by varying the camera position and appearance. Using each of these configurations we train a zoo of models. Variations in input datasets lead to changes in how brain regions are represented within the model. After this, we use alignment metrics to evaluate the alignment of neural network models to the brain. Finally, we evaluate the alignment of the models trained based on these datasets for comparison.

The traditional way to understand the nuances of human vision is through experimental studies involving neuroimaging or behavioral data. Many studies such as Majaj et al. (2015) and Rajalingham et al. (2018) have attempted to understand the brain using methods such as electrode arrays and behavioral experiments. Most of these studies are limited by the amount of scanning time and the number of images that can be shown to subjects. Furthermore, stimuli are tailored to a specific hypothesis and are not designed for tasks such as ours where we want to use computer graphics to understand human vision. We cannot decompose brain activations from ImageNet-like images into activations from shadows, shading, textures, or other appearance components. To understand how each of these properties is individually processed, we propose the following pipeline in Figure 2.

In the first step of our pipeline, we adopt a data-centric approach. We generate synthetic data by controlling the appearance and rotation factors in the scene using Blender (Community, 2018) as our rendering engine. At first, we generate a dataset with maximal variation of appearance and rotation effects and name it the *Base* dataset. After this, we vary one factor (such as shadows, shading, textures, and rotation variation) at a time to generate a set of *modified* datasets. We hypothesize that we can individually study factors that affect perception in human vision at different stages of the visual ventral stream using synthetically rendered datasets.

In the second step (as depicted in Figure 2), we expose deep learning models to a variety of synthetic datasets to ensure that each model instance learns different types of representations. For this, we take inspiration from previous work such as Dwivedi et al. (2020); Yildirim et al. (2020) which point out that there are three ways to accomplish this task. Namely, changing the architecture, changing the loss function, or changing the training data. We choose to change the training data because by modifying the training data we can directly influence the features that the model learns. We use this technique following the lines of contrastive learning and domain adversarial training used in computer vision with reference from (Adhikarla et al., 2023).

After we have trained the neural networks, we need a good model of the human visual system. We consider Convolutional Neural Networks (CNNs) as such and hypothesize that they learn representations similar to the human brain. To compare the neural network activations to brain responses, we make use of alignment metrics such as RSA (Kriegeskorte, 2015) and Centered Kernel Alignment (CKA) (Murphy et al., 2024) which have shown to be successful in cross-domain comparisons (Cui et al., 2022). Each of these metrics is then used to evaluate the difference between the set of synthetic datasets we have rendered. The direction, significance, and magnitude of these differences can highlight the importance of a particular brain region in representing a particular property. Figure 3 highlights these trends and provides a way to interpret the direction magnitude and difference. We

take the scores from the *Base* and subtract the scores of the modified from it. Then we try to evaluate the trend of these scores down the visual hierarchy.

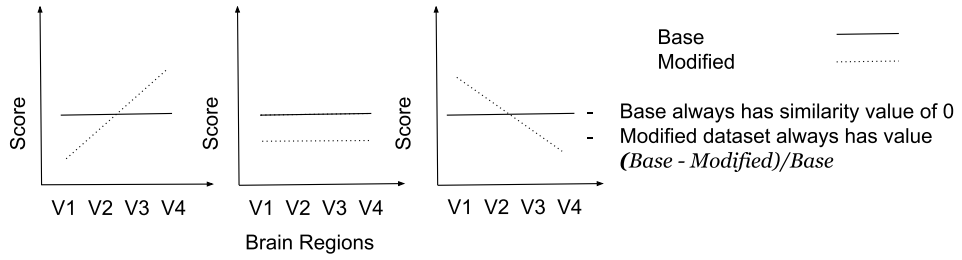



Figure 3: This figure illustrates the various types of trends observed in the differences of the metrics for the modified and *Base* datasets. Different types of trends can be seen over the visual hierarchy. First, the score difference is increasing, which means that the feature we removed from the *Base* dataset to create the modified dataset has a representation in the later visual areas. Second, the score difference does not change. This means that this feature is less represented in the visual areas. Third, the score difference decreases down the visual hierarchy, this means that the removed feature is more represented in the early visual areas as compared to the later visual areas. A higher score means that the match to the brain is worse for the modified dataset as compared to the *Base* dataset and hence the feature that was removed in the modified dataset will be represented in that brain region.

In summary, our work proposes a tool for studying the processing of scene intrinsics in human vision. By utilizing CNNs and synthetic data, we can discover and offer insights into the complex relationships between scene attributes and neural responses. This research could contribute to cognitive neuroscience and the development of more advanced computer vision models.

## 2 Related Work

In recent years, the study of alignment between human and neural network representations has gained widespread attention. Particularly, studies such as Sucholutsky et al. (2023) provide the framework to study the relationships between representations in the human brain and neural networks. Additionally, studies such as Güçlü & Van Gerven (2015) show that shallow representations of CNNs are similar to early visual areas, while deep layers are similar to those of higher-level visual areas. Furthermore, the study from Ratan Murty et al. (2021) shows that Artificial Neural Networks (ANNs) can be used to predict responses in areas such as PPA and FFA effectively while Guclu & van Gerven (2015) have shown high accuracy in predicting responses in the IT cortex.

It is well known that the representations that a model can learn depend mainly on three factors, namely training data, network architecture, and loss function (Yildirim et al., 2020). For example, in the taskonomy dataset (Zamir et al., 2018) which is widely used in computer vision, the architecture remains constant, but the loss function is changed to perform different tasks. This ensures that the differences in the model representations arise due to the difference in task and not due to the difference in model architecture. Another widely used computer vision data set is the ModelNet data set (Qi et al., 2017) where the architecture is changed and the loss function remains constant. This helps evaluate the type of layers that help in achieving performance on a particular task. It shows a way how variation in the architecture leads to variation in representations keeping the cross-entropy loss constant. We chose to change the training data so that we could control the representations learned using the features in the image training set after being inspired by previous work by Dobs et al. (2022).

In addition to understanding how these factors influence model representations, it is important to evaluate how closely the model's visual representations align with the brain. This has been called for in the field of NeuroAI. There are a couple of methods that have been widely used for aligning human and neural network representations. RSA is one of the traditional methods to measure the similarity of these representations. The description of RSA is mentioned in Section 3.3.1. It has been used in various Neuroimaging studies involving fMRI, MEG, and EEG modalities (Dimsdale-Zucker

& Ranganath, 2018) (Kaneshiro et al., 2015). New metrics such as CKA and Linear Shape Metrics have also been introduced. Mostly, these metrics are not correlated with each other, as shown in Soni et al. (2024a).

## 3 METHODOLOGY

In summary, our approach is as follows: first, we generate different synthetic datasets by controlling the parameters (appearance and pose variation) of the synthetic scene (Section 3.1). Second, we used these synthetic datasets to train a variety of DNNs (Section 3.2). Third, as described in Section 3.3, we benchmark each of these networks on a subset of publicly available benchmarks of Coggan & Tong (2024) in *BrainScore* (Schrimpf et al., 2020), RSA, and CKA. Once we have a goodness of fit from each of these settings, we evaluate the significance of the differences in scores (using an independent samples t-test) to make inferences about where a particular scene property is represented in the brain.

### 3.1 DATASET GENERATION

The first step of our pipeline is to use a computer graphics approach to generate synthetic datasets. We make use of the Blender (Community, 2018) rendering engine to simulate a three-dimensional scene while having full control over all parameters of the scene whereas such control is not available when using most real-world datasets.

To generate our data set, we use meshes from the *Objaverse 1.0* (Deitke et al., 2023) dataset. This dataset was chosen because it is one of the biggest repositories of three-dimensional meshes available publicly. It comes with three-dimensional meshes, annotations, and captions of the images. This dataset (*version 1.0*) contains around $800K+$ objects out of which only $80K$ have *LVIS* annotations (Gupta et al., 2019), which partitions the dataset into $1156$ common categories instead of partitioning the dataset into a larger number of categories with less number of instances per class. Using *LVIS* only $80K$ meshes were suitable for our task, which is still a comparatively large number for multiclass recognition tasks. The main focus of our work is to evaluate downstream effects on brain predictivity based on changing the training data.

We select classes which have $n$ number of objects where $n \in [20, 120]$ and from the Objaverse dataset and then render five images per configuration (see Figure 2). We generated multiple data sets to train our models. The first variation of our dataset is referred to as the *Base* dataset. It contains all possible appearance and pose variations. Figure 4 shows one view per object to highlight illumination changes on the same object. Figure 5 shows five views per object to highlight the rotational differences. The *Base* dataset is rendered in the following manner by modifying the script at [1] :

(a) (b) (c) (d) (e) (f)

Figure 4: A teapot rendered under different illumination effects: (a) *Base* dataset with all the scene properties intact (b) random textures (c) ambient illumination where a white environment map is used to lighten the scene (d) without shading (e) without shadows and (f) without specular reflections. A plane is rendered at the bottom of the objects to visually show what happens when we remove shadows from the scene.

[1] GitHub: `https://github.com/allenai/objaverse-rendering/blob/main/scripts/blender_script.py`

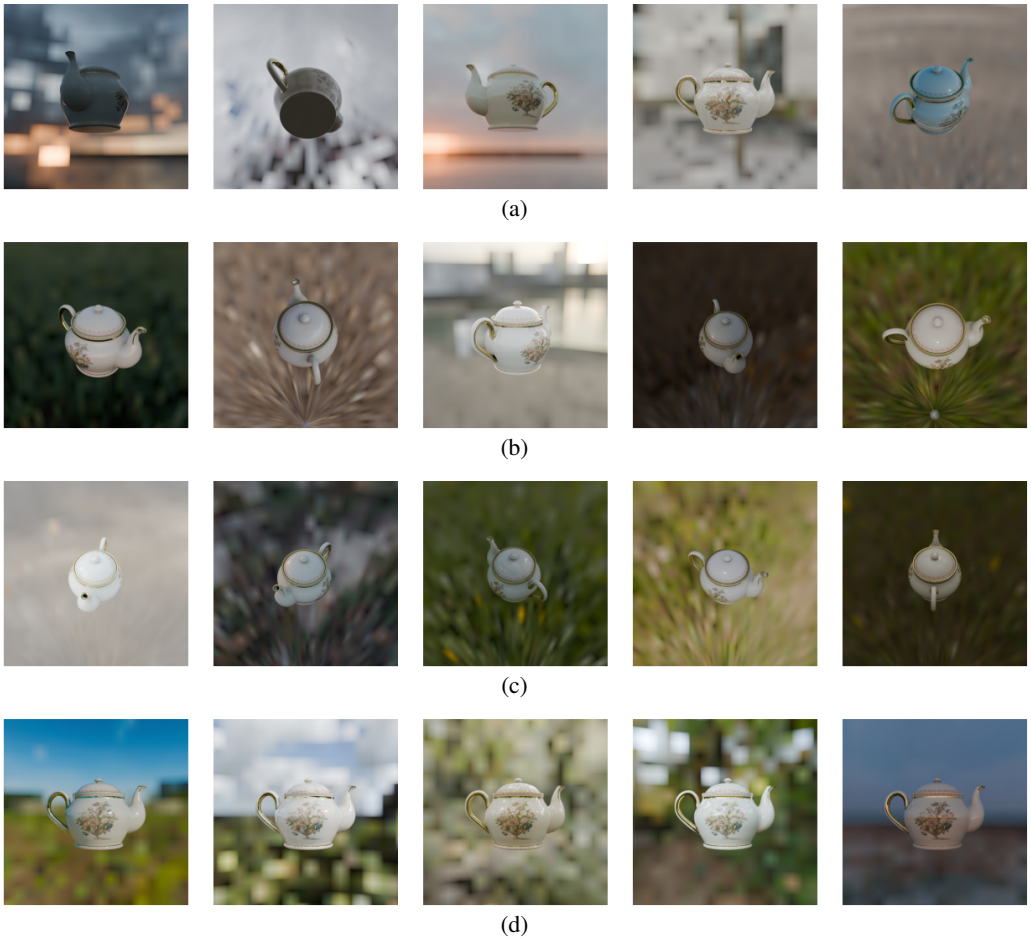



Figure 5: Datasets with rotation effects. (a) Base dataset with the variation of the camera angle along a full sphere, (b) Variation along a hemisphere, (c) Variation along the Z-axis only, (d) No variation of the camera angle.

As defined above, this results in a selection of $k = 775$ classes and $37K$ objects in total. To create a synthetic scene, a random environment map is taken from a set of 1673 spherical HDRI maps introduced in the training set by Gardner et al. (2022) that controls the background image and the illumination of the scene. The original properties of the scene such as shadows, shading, specularities, and textures remain intact. The camera viewpoint is varied along a sphere (see Figure 5) to get different viewpoints of the object. Five images per object are then rendered.

We then generated eight different versions of this dataset, leading to nine synthetic datasets in total; for each version, we changed exactly one scene parameter at a time. To do so, we group the scene parameters into two sets (which we call *paths*): the *appearance path* and the *pose variation path*. For the appearance path, the settings of the scene are changed using Blender (see Figure 4). The datasets generated using the *appearance path* are:

- **Random texture:** depicted in Figure 4b: a new random texture sampled from the describable textures dataset (*dtd*) (Cimpoi et al., 2014) and is added on top of the object. This changes the surface properties of the object such as the color, reflectivity, and transparency. The color is set according to the new image while the other properties are set to default values in blender.

- **Ambient illumination:** depicted in Figure 4c uses an ambient white environment map for lightning the scene and a random environment map sampled from Gardner et al. (2022)

dataset as a background image. This results in all parts of the image being illuminated uniformly and remove any shading or sharp illumination effects.

- **Shading removal:** depicted in Figure 4d. We remove all the shading (local and global illumination effects) from the material. When we remove the shading, it means we are now using only the material texture to model how the light interacts with the surface of the object. Particularly, in Blender it means that we are no longer using the principled bsdf shader which combines multiple shaders into one.
- **Shadow removal:** depicted in Figure 4e removes shadows such as cast shadows and self-shadows. The cast shadow can be seen with the help of the plane at the bottom of this object.
- **Specularity removal:** depicted in Figure 4f removes properties such as shininess, glossiness, and metallic properties from the image (Fleming et al., 2004).

For the *rotation* path, we vary the position of the camera (see Figures 1e-1h & 5 for the distribution of camera angles). Five images per object are rendered by sampling camera positions. We make the use of polar coordinates to define the position, $(x, y, z)$. The general equation for the camera point is:

$$x = r\sin(\phi)\cos(\theta), \quad y = r\sin(\phi)\sin(\theta), \quad z = r\cos(\phi) \tag{1}$$

where $\theta$ is the azimuthal and $\phi$ the polar angle and $r$ is the camera distance, we use $r = 1.2$.

- **Base Dataset**: depicted in Figure 5a. The angles vary as $\theta \in [0, 2\pi]$ and $\phi \in [0, \pi]$.
- **Less Variation**: depicted in Figure 5b vary in the upper hemisphere $\phi \in [0, \pi/2]$ and constant $\theta = \pi/4$
- **Z-Axis**: depicted in Figure 5c, rotation around the z-axis with $\theta \in [0, 2\pi]$ and $\phi = \pi/2$.
- **No Variation**: depicted in Figure 5d with constant angles $\theta = \pi/4$ and $\phi = \pi/2$.

### 3.2 Model Training

Once the data was generated, a neural network zoo was trained for subsequent analysis. We chose to train Convolutional Neural Networks (CNNs) due to their widespread use in the field of NeuroAI. The CNNs were implemented using PyTorch (Paszke et al., 2017).

In particular, the models AlexNet (Krizhevsky et al., 2012), VGG16 (Simonyan & Zisserman, 2015), and ResNet50 (He et al., 2016) were trained. These models were used to perform multiclass classification tasks on each of the synthetic datasets derived from (Deitke et al., 2023). This task aims to classify the object based on synthetic images with $n = 775$ classes. As portrayed in Figure 2, we train 3 models on each of the 9 datasets and perform 5 iterations of the same (five different train and test splits). This means that in total we train $m = 135$ model instances. We modified the number of neurons in the last layer of each model to match $n$ and then trained the networks.

The hyperparameters used to train the models are mentioned in Table 1. We make sure that for each type of model, the architecture and the loss function remain constant due to the reasons mentioned in section 2. The model weights were initialized using *Xavier initialization* (Glorot & Bengio, 2010). The instances of these models were trained until convergence if the model loss did not decrease after 15 epochs (early stopping). This convergence criterion ensures stability in the model training and prevents unnecessary training time and resource expenditure (Prechelt, 1998). We were able to achieve reasonable accuracy, as presented in Table 2 on almost all pairwise combinations of models and datasets. The models were trained from scratch with a set of parameters similar to the ones used for training such models on the ImageNet (Deng et al., 2009) dataset.

### 3.3 Aligning Neural Networks to the Brain

To evaluate the hypothesis of our study, it is important to study the alignment of representations in the neural networks to representations in the human brain. For this, we take inspiration from studies like Sucholutsky et al. (2023) that aim to measure the alignment between any two systems and show a general framework to compute a metric of alignment between two systems. In this section, we highlight the metrics used in this work to measure the alignment between networks trained on our synthetic datasets and the data from a variety of neuroimaging studies.

#### 3.3.1 REPRESENTATIONAL SIMILARITY ANALYSIS

RSA (Kriegeskorte, 2015), as mentioned in Section 1 is one of the popular methods to compute the similarity of representations in the brain and neural networks. In general, the mechanism is to compute pairwise correlations between the representational dissimilarity matrices (RDMs) of the neural networks and RDMs of the brain regions. We use the python Net2brain[1] toolbox and an offline instance of *BrainScore* (Schrimpf et al., 2020).

Using the Net2brain (Bersch et al., 2022) toolbox we get an average RDM (from the brain activity of 8 subjects in the NSD (Allen et al., 2022) dataset) for each brain region and one RDM for each NN layer and compute correlations (Pearson's r) as mentioned above. On the other hand, we used the publicly available benchmarks from (Coggan & Tong, 2024) available on BrainScore (Schrimpf et al., 2020) to benchmark our models. For each of the networks, instead of letting *BrainScore* choose the layers, we assigned the layers to each brain region according to recommendations on *BrainScore* website, to see only the effects of changing the training data. The layers were chosen according to Table 3 in the Appendix.

#### 3.3.2 CENTERED KERNEL ALIGNMENT

Centered kernel alignment has become a widely used metric for comparing activations between neural networks and the brain (Murphy et al., 2024). For our use case, it serves as an effective tool for comparing the representations in brain RDMs to neural network RDMs. We use the formulation mentioned in (Kornblith et al., 2019) to compare the matrices (obtained from the Net2brain (Bersch et al., 2022) package) in a pairwise manner. This means each brain region RDM is compared against each neural network layer RDM. The combination with the highest value of CKA is reported i.e. the layer which is most aligned to a particular brain region is reported in Figures 7 and 6.

#### 3.3.3 ENCODING MODELS

Encoding models are a class of models used to directly predict brain activity in response to stimuli. Wang et al. (2019) is one of the established sets of models used to predict localized brain responses. In our study, we make use of ROI-based feature-weighted receptive field model (fwrf-model from (Gifford AT), (Gifford et al., 2024)) to synthesize neural responses for images of our own choice. This approach provides a quick way to map how features of an image contribute to the activation patterns observed in the brain. This encoding model can be trained for each ROI has an explained variance score of 65.94% for V1, 59.71% for V2, 52.02% for V3 and 44.45% for V4 (Gifford et al., 2024). The encoding model uses Allen et al. (2022) as the base and comes with EEG and fMRI responses for *150,000* naturalistic images coming from multiple publicly available datasets such as ImageNet (Deng et al., 2009). This fact ensures that the synthetic responses are generated from an underlying distribution, allowing for a more accurate simulation of the neural responses and providing an alternative for conducting large-scale in vivo experiments.

We began by generating synthetic responses to a set of 500 images (5 images for each of the 100 objects) (see Figure 5). Next, we generated synthetic responses for each of the proposed dataset modifications in the same manner. We chose these objects visually so that they include more appearance effects as compared to the average object in the dataset. These responses are visualized using NiLearn [3]. We used an encoding model for each subject and each ROI. Using the MNI template (Collins et al., 1994), we averaged the subject-level responses for each ROI to create a group-level prediction for a particular ROI. The predictions at the group level were compared based on the dataset used to synthesize the neural responses. We then calculated the difference in the NIfTI images obtained from synthetic responses and finally visualized these differences using the Human Connectome Atlas (Van Essen et al., 2013) to enhance the visual interpretation of the results. With this set of experiments, our goal is to depict the sensitivity of encoding models to changes in the data used for synthesizing a response. In this way, we can compare it to the experiments from CKA and RSA.

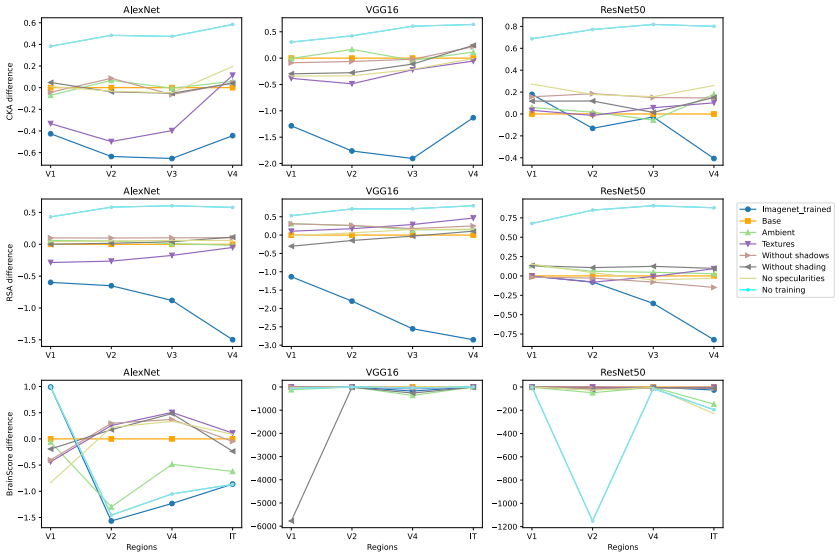



Figure 6: We highlight how variations in scene appearance affect the corresponding alignment metrics. Specifically, we compute the difference between the alignment score of each modified dataset and that of the base dataset, i.e., *Base – Modified*, and normalize this difference by the score of the Base dataset. Each line in the graph represents the normalized difference in alignment metrics between the modified and Base datasets. The Representational Similarity Analysis (RSA) and Centered Kernel Alignment (CKA) scores are computed on the dataset from Allen et al. (2022) using Bersch et al. (2022), while the BrainScore differences are calculated for a subset of benchmarks introduced in Coggan & Tong (2024).

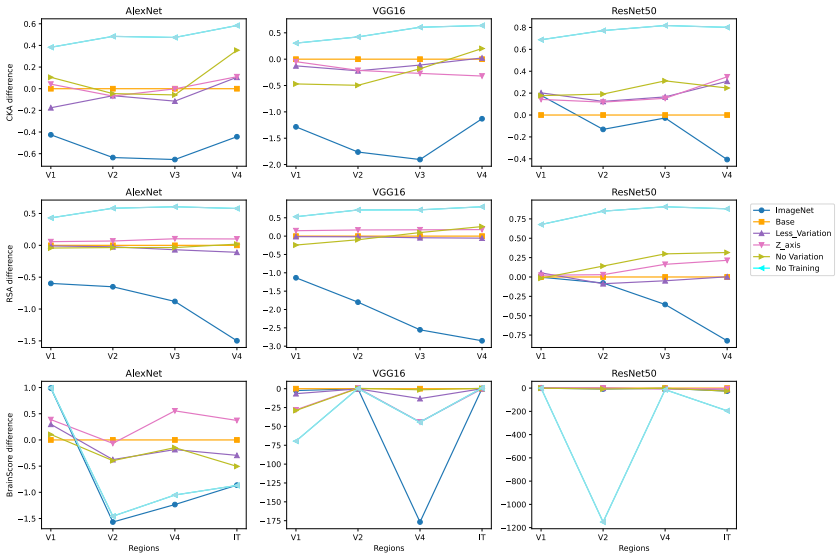



Figure 7: Differences in alignment metrics due to rotation variations. The plot shows normalized differences between the *Base* and *Modified* datasets.

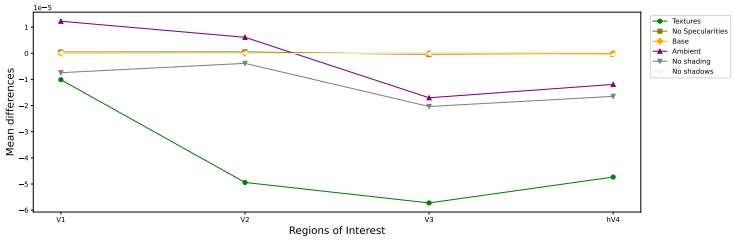



Figure 8: Encoding model (from Gifford AT) is used to predict the intensity in the brain regions highlighted on the x-axis. Each point on this plot shows the difference in average activity (mean of intensities) of the *Base* dataset as compared to that of the modified dataset. Using these line plots we can see a trend along the visual hierarchy.

## 4 Experiments and Results

Figures 7 and 6 highlight differences in alignment metrics of the modified datasets compared to the *Base* dataset across brain regions and different networks. It can be seen that for the same network, for example, AlexNet the three metrics RSA, CKA, and *BrainScore* do not always show the same trend. This phenomenon is expected and has been reported in previous literature such as (Soni et al., 2024b) which mentions that different alignment metrics might not be perfectly correlated to each other for the same set of measurements, but are nevertheless useful since we get to see the patterns innate in the data using different comparison metrics.

After computing the alignment metrics for each of the pairwise combinations of dataset and models, we compute an independent samples t-test between the metrics of five iterations of the *Base* dataset and *modified* datasets. The p-values of this analysis are represented in the Appendix in Tables 4, 5 and 6. Furthermore, we highlight all the p-values $p < 0.05$ to highlight the significance of the results. Analyzing these results, we see that CKA is one metric that significantly differs for different datasets across networks for a large chunk of combinations. RSA and *BrainScore* also show significant results but with a comparatively smaller number of observations having $p < 0.05$. Hence, we suggest that CKA is a more favorable metric as compared to RSA and *BrainScore* for our use case. Furthermore, a proof of concept is that ImageNet-trained networks and untrained networks are significantly different as compared to the *Base* dataset. The difference for ImageNet-trained networks is always negative since they are trained on more realistic scenes and match the brain better as compared to our *Base* dataset. The difference for the untrained networks is almost always more positive as compared to ImageNet-trained and our modified networks since the untrained networks are randomly initialized and match the brain worse as compared to the base dataset i.e. $(base - untrained/base) > 0$. For the rotational datasets, it can be seen in general that the difference (along the visual hierarchy) as compared to the *Base* dataset is more on the positive side as the rotation in the dataset decreases, in general. This means that probably complex rotation could be represented in higher visual areas. Such observations are in agreement with prior literature e.g. from Roe et al. (2012).

Using Figure 6 (alongside Tables 4, 5 and 6 in the Appendix) we can conclude that for the textured dataset, the difference is negative (but higher in magnitude) in the early visual areas and moves towards zero for V3 and V4. Both the RSA and CKA analyses are significant (p-values of score differences < $0.05$) for VGG16 and AlexNet according to Table 4. Additionally, CKA is significant across networks for the textured dataset (as highlighted by Table 4 to Table 6). This could suggest that there is a representation of texture in the early visual system that follows along the lines of Hucka & Kaplan (1995). The textured dataset has more variations in its feature space as compared to the *Base* dataset, as a result of this we cannot take the generalized interpretations from Figure 3. In this case, we are not removing a feature from the *Base* dataset but adding a new feature (texture).

The datasets without shadows, shading, and ambient illumination seem to have RSA and CKA scores significantly different from the *Base* dataset. This is highlighted by Table 4, Table 6 and Table 5 with early visual layers having p-values $p < 0.05$ in most brain regions for all the three networks. This

[2]GitHub: `https://github.com/cvai-roig-lab/Net2Brain.git`
[3]GitHub: `https://nilearn.github.io/stable/index.html`

suggests that illumination effects are being represented more in the early visual areas as compared to the later visual areas. This is in agreement with prior literature (Chen & Tarczy-Hornoch, 2006) that highlights the role of V1 in the processing of shadows. Our results are consistent with the literature, Murray et al. (2006) confirms this effect and has found the presence of brightness and transparency in cells V1 and V2. The paper Lamme et al. (1999) shows the role of V1 in depth perception which can be supported by appearance cues.

We conducted encoding model experiments for the appearance dataset only as it was hypothesized that there would be few effects for the rotational dataset since the response to a large number of images would be needed to see an effect (and this could not be determined how large of an effect would be needed). Figure 10 shows an example of results from the encoding model. The full-length results are presented in the Appendix with Figures 10 - 14, shows the visualizations as well as the direction of the differences, we can see the difference and direction using Figure 8. We can see which regions show higher or lower differences. However, these results are not all conclusive but provide a method to rank the regions for particular datasets. These can be checked for agreement with other alignment metrics such as CKA and RSA. In general, we can conclude that we can see a variety of trends for different effects.

## 5 Conclusion and Discussion

We demonstrate a novel technique that uses computer graphics as a tool to approximately study human vision. Generating different controlled training datasets by rendering synthetic scenes can potentially be useful in understanding the brain. Our experiments try to predict the trends of human perceptual responses to various synthetically controlled image features. Using different alignment metrics, we can rank the importance of a particular brain region in the representation of a particular feature. Even though not all alignment metrics agree with each other, a general trend along the ventral visual stream can be observed.

An important point to note in the results is that for the NSD (Allen et al., 2022) dataset, RDMs in Figures 7 and 6 are extracted for the regions V1, V2, V3 and V4. The NSD dataset contains masks for regions lhV4 and rhV4 (left and right hemisphere V4) which are relevant to studying the IT cortex but as such are not named as IT. Using a subset (from Coggan & Tong (2024)) of benchmarks *BrainScore*, we were able to benchmark in regions V1, V2, V4, and IT. Another point to note is that the (Allen et al., 2022) dataset is not perfectly suited for the task of RSA. Hence, we have used encoding models to add another layer of confidence to our experimental results. The encoding model does show trends in differences along the visual hierarchy, which can be compared against the three metrics proposed before.

In summary, we provide a pipeline that encompasses a tool to study the brain using computer graphics. Using different alignment metrics, we evaluate how different scene intrinsics are represented in the brain. Even though this method is approximate, it still gives a starting point before trying to study the brain by designing specialized fMRI experiments and it might guide studies into novel directions.

## 6 Acknowledgements

We thank Maximilian Weiherer for his valuable feedback and support towards finalizing the manuscript. This work was funded by the German Federal Ministry of Education and Research (BMBF), FKZ: 01IS22082 (IRRW). The authors of this publication are responsible for all its content. Additionally, the authors are grateful to the Erlangen National High Computing Center (NHR@FAU) of the Friedrich-Alexander-Universität Erlangen-Nüremberg which provided scientific support and HPC resources under the NHR project b112dc IRRW. NHR is funded by the federal and Bavarian state authorities. NHR@FAU hardware is partially funded by the German Research Foundation (DFG) – 440719683.

## References

Eashan Adhikarla, Kai Zhang, Jun Yu, Lichao Sun, John Nicholson, and Brian D. Davison. Robust computer vision in an ever-changing world: A survey of techniques for tackling distribution shifts, 2023. URL https://arxiv.org/abs/2312.01540.

Emily J. Allen, Ghislain St-Yves, Yihan Wu, Jesse L. Breedlove, Jacob S. Prince, Logan T. Dowdle, Matthias Nau, Brad Caron, Franco Pestilli, Ian Charest, J. Benjamin Hutchinson, Thomas Naselaris, and Kendrick Kay. A massive 7t fmri dataset to bridge cognitive neuroscience and artificial intelligence. *Nature Neuroscience*, 25:116–126, 2022. doi: 10.1038/s41593-021-00962-x. URL https://www.nature.com/articles/s41593-021-00962-x. Published: 16 December 2021.

Domenic Bersch, Kshitij Dwivedi, Martina Vilas, Radoslaw M. Cichy, and Gemma Roig. Net2brain: A toolbox to compare artificial vision models with human brain responses, 2022.

Alfred M. Bruckstein. On shape from shading. *Computer Vision, Graphics, and Image Processing*, 44(2):139–154, 1988. ISSN 0734-189X. doi: https://doi.org/10.1016/S0734-189X(88)80002-1. URL https://www.sciencedirect.com/science/article/pii/S0734189X88800021.

VJ Chen and K Tarczy-Hornoch. Interpolating light and shadow in v1. *Investigative Ophthalmology & Visual Science*, 47(13):5876–5876, 2006.

M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, , and A. Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2014.

David D. Coggan and Frank Tong. Evidence of strong amodal completion in both early and high-level visual cortices. *under review*, 2024.

D. L. Collins, P. Neelin, T. M. Peters, and A. C. Evans. Automatic 3d intersubject registration of mr scans of the human brain. *IEEE Transactions on Medical Imaging*, 13(4):743–756, 1994. doi: 10.1109/42.317252.

Blender Online Community. *Blender - a 3D modelling and rendering package*. Blender Foundation, Stichting Blender Foundation, Amsterdam, 2018. URL http://www.blender.org.

Tianyu Cui, Yogesh Kumar, Pekka Marttinen, and Samuel Kaski. Deconfounded representation similarity for comparison of neural networks. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2022. Curran Associates Inc. ISBN 9781713871088.

Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13142–13153, 2023.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Zhiqing Deng, Jie Gao, Toni Li, Yan Chen, BoYu Gao, Fang Fang, Jody C. Culham, and Juan Chen. Viewpoint adaptation revealed potential representational differences between 2d images and 3d objects. *Cognition*, 251:105903, 2024. ISSN 0010-0277. doi: https://doi.org/10.1016/j.cognition.2024.105903. URL https://www.sciencedirect.com/science/article/pii/S0010027724001896.

Halle R. Dimsdale-Zucker and Charan Ranganath. Chapter 27 - representational similarity analyses: A practical guide for functional mri applications, 2018. ISSN 1569-7339. URL https://www.sciencedirect.com/science/article/pii/B9780128120286000276.

Katharina Dobs, Julio Martinez, Alexander J. E. Kell, and Nancy Kanwisher. Brain-like functional specialization emerges spontaneously in deep neural networks. *Science Advances*, 8(11):eabl8913, 2022. doi: 10.1126/sciadv.abl8913. URL https://www.science.org/doi/abs/10.1126/sciadv.abl8913.

Kshitij Dwivedi, Michael F. Bonner, Radoslaw Martin Cichy, and Gemma Roig. Unveiling functions of the visual cortex using task-specific deep neural networks. *bioRxiv*, 2020. doi: 10.1101/2020.11.27.401380. URL https://pmc.ncbi.nlm.nih.gov/articles/PMC8407579/.

R.W. Fleming, A. Torralba, and E.H. Adelson. Specular highlights and the perception of shape. *Journal of Vision*, 4(9):798–820, 2004. doi: 10.1167/4.9.5. URL https://doi.org/10.1167/4.9.5.

James A. D. Gardner, Bernhard Egger, and William A. P. Smith. Rotation-equivariant conditional spherical neural fields for learning a natural illumination prior, 2022.

Alessandro T. Gifford, Maya A. Jastrzębowska, Johannes J. D. Singer, and Radoslaw M. Cichy. In silico discovery of representational relationships across visual cortex, 2024. URL https://arxiv.org/abs/2411.10872.

Cichy RM. 2024 Gifford AT. URL https://github.com/gifale95/NED.

Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In Yee Whye Teh and Mike Titterington (eds.), *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9 of *Proceedings of Machine Learning Research*, pp. 249–256, Chia Laguna Resort, Sardinia, Italy, 13–15 May 2010. PMLR. URL https://proceedings.mlr.press/v9/glorot10a.html.

U. Guclu and M. A. J. van Gerven. Deep neural networks reveal a gradient in the complexity of neural representations across the ventral stream. *Journal of Neuroscience*, 35(27):10005–10014, July 2015. ISSN 1529-2401. doi: 10.1523/jneurosci.5023-14.2015. URL http://dx.doi.org/10.1523/JNEUROSCI.5023-14.2015.

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

Agrim Gupta, Piotr Dollár, and Ross Girshick. Lvis: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5356–5364, 2019. doi: 10.1109/CVPR.2019.00549. URL https://openaccess.thecvf.com/content_CVPR_2019/papers/Gupta_LVIS_A_Dataset_for_Large_Vocabulary_Instance_Segmentation_CVPR_2019_paper.pdf.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016. doi: 10.1109/CVPR.2016.90.

Michael Hucka and Stephen Kaplan. Texture-based processing in early vision and a proposed role for coarse-scale segmentation. Technical report, University of Michigan, 1995. URL https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=2d3beb2505bdb61d1620a0a393b8214828cfda98.

Blair Kaneshiro, Marcos Guimaraes, Hyungsuk Kim, Anthony Norcia, and Patrick Suppes. A representational similarity analysis of the dynamics of object processing using single-trial eeg classification. *PloS one*, 10:e0135697, 08 2015. doi: 10.1371/journal.pone.0135697.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *International conference on machine learning*, pp. 3519–3529. PMLR, 2019.

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

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J. Burges, L. Bottou, and K.Q. Weinberger (eds.), *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012. URL https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.

Victor A. F. Lamme, Hans Supèr, Rogier Landman, Pieter R. Roelfsema, and Henk Spekreijse. The role of primary visual cortex (v1) in visual awareness. *Vision Research*, 39(10):1435–1448, 1999. doi: 10.1016/S0042-6989(99)00243-6.

Najib J. Majaj, Ha Hong, Ethan A. Solomon, and James J. DiCarlo. Simple learned weighted sums of inferior temporal neuronal firing rates accurately predict human core object recognition performance. *Journal of Neuroscience*, 35(39):13402–13418, 2015. ISSN 0270-6474. doi: 10.1523/JNEUROSCI.5181-14.2015. URL https://www.jneurosci.org/content/35/39/13402.

Alex Murphy, Joel Zylberberg, and Alona Fyshe. Correcting biased centered kernel alignment measures in biological and artificial neural networks, 2024. URL https://arxiv.org/abs/2405.01012.

S. O. Murray, S. He, K. Wray, R. Sekuler, and G. Mather. Brightness and transparency in the early visual cortex. *Journal of Vision*, 6(6):1064–1074, 2006. doi: 10.1167/6.6.1064.

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

Lutz Prechelt. Early stopping – but when? In C. J. C. Burges B. Schölkopf and A. J. Smola (eds.), *Neural Networks: Tricks of the Trade*, pp. 55–69. Springer, 1998.

Charles R. Qi, Li Yi, Hao Su, and Leonidas J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Advances in Neural Information Processing Systems*, volume 30, pp. 5099–5108, 2017. URL https://proceedings.neurips.cc/paper/2017/file/d8bf84be3800d12f74d8b05e9b89836f-Paper.pdf.

Rishi Rajalingham, Elias B. Issa, Pouya Bashivan, Kohitij Kar, Kailyn Schmidt, and James J. DiCarlo. Large-scale, high-resolution comparison of the core visual object recognition behavior of humans, monkeys, and state-of-the-art deep artificial neural networks. *Journal of Neuroscience*, 38(33):7255–7269, 2018. doi: 10.1523/JNEUROSCI.0388-18.2018. URL https://www.jneurosci.org/content/38/33/7255.

N. A. Ratan Murty, P. Bashivan, A. Abate, et al. Computational models of category-selective brain regions enable high-throughput tests of selectivity. *Nature Communications*, 12:5540, 2021. doi: 10.1038/s41467-021-25409-6. URL https://doi.org/10.1038/s41467-021-25409-6.

Anna W. Roe, Leonardo Chelazzi, Charles E. Connor, Bevil R. Conway, Ichiro Fujita, Jack L. Gallant, Haidong Lu, and Wim Vanduffel. Toward a unified theory of visual area v4. *Neuron*, 74(1):12–29, 2012. doi: 10.1016/j.neuron.2012.03.011. URL https://pmc.ncbi.nlm.nih.gov/articles/PMC4912377/.

Silvio Savarese. *Shape from Specularities*, pp. 726–729. Springer US, Boston, MA, 2014. ISBN 978-0-387-31439-6. doi: 10.1007/978-0-387-31439-6_261. URL https://doi.org/10.1007/978-0-387-31439-6_261.

Martin Schrimpf, Jonas Kubilius, Ha Hong, Najib J. Majaj, Rishi Rajalingham, Elias B. Issa, Kohitij Kar, Pouya Bashivan, Jonathan Prescott-Roy, Franziska Geiger, Kailyn Schmidt, Daniel L. K. Yamins, and James J. DiCarlo. Brain-score: Which artificial neural network for object recognition is most brain-like? *bioRxiv*, 2020. doi: 10.1101/407007. URL https://www.biorxiv.org/content/early/2020/01/02/407007.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 1–9, 2015.

Ansh Soni, Sudhanshu Srivastava, Konrad Kording, and Meenakshi Khosla. Conclusions about neural network to brain alignment are profoundly impacted by the similarity measure. *bioRxiv*, 2024a. doi: 10.1101/2024.08.07.607035. URL https://www.biorxiv.org/content/early/2024/08/09/2024.08.07.607035.

Ansh Soni et al. Conclusions about neural network to brain alignment are profoundly impacted by the similarity measure, 2024b.

Pascal Spröte, Filipp Schmidt, and Roland Fleming. Visual perception of shape altered by inferred causal history. *Scientific Reports*, 6:36245, 2016. doi: 10.1038/srep36245. URL https://doi.org/10.1038/srep36245.

Ilia Sucholutsky, Lukas Muttenthaler, Adrian Weller, Andi Peng, Andreea Bobu, Been Kim, Bradley C. Love, Erin Grant, Iris Groen, Jascha Achterberg, Joshua B. Tenenbaum, Katherine M. Collins, Katherine L. Hermann, Kerem Oktar, Klaus Greff, Martin N. Hebart, Nori Jacoby, Qiuyi Zhang, Raja Marjieh, Robert Geirhos, Sherol Chen, Simon Kornblith, Sunayana Rane, Talia Konkle, Thomas P. O'Connell, Thomas Unterthiner, Andrew K. Lampinen, Klaus-Robert Müller, Mariya Toneva, and Thomas L. Griffiths. Getting aligned on representational alignment, 2023. URL https://www.bifold.berlin/impact-transfer/publications/view/publication-detail/getting-aligned-on-representational-alignment.

James T. Todd and Lore Thaler. The perception of 3d shape from texture based on directional width gradients. *Journal of Vision*, 10(5):17–17, 05 2010. ISSN 1534-7362. doi: 10.1167/10.5.17. URL https://doi.org/10.1167/10.5.17.

David C Van Essen, Steven M Smith, Deanna M Barch, Timothy E J Behrens, Esmaeel Yacoub, and Kamil Ugurbil. The human connectome project: A data acquisition perspective. *NeuroImage*, 62 (4):2222–2231, 2013. doi: 10.1016/j.neuroimage.2012.02.018.

Aria Wang, Michael Tarr, and Leila Wehbe. Neural taskonomy: Inferring the similarity of task-derived representations from brain activity. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/f490c742cd8318b8ee6dca10af2a163f-Paper.pdf.

Susan G Wardle and Chris Baker. Recent advances in understanding object recognition in the human brain: deep neural networks, temporal dynamics, and context. *F1000Research*, 9:F1000 Faculty Rev–590, Jun 11 2020. doi: 10.12688/f1000research.22296.1. URL https://doi.org/10.12688/f1000research.22296.1.

I. Yildirim, M. Belledonne, W. Freiwald, and J. Tenenbaum. Efficient inverse graphics in biological face processing. *Science Advances*, 6(10):eaax5979, 2020. doi: 10.1126/sciadv.aax5979. URL https://doi.org/10.1126/sciadv.aax5979.

Amir R. Zamir, Alexander Sax, William B. Shen, Leonidas J. Guibas, Jitendra Malik, and Silvio Savarese. Taskonomy: Disentangling task transfer learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2018.

# 7 APPENDIX

| Model | Optimizer | Momentum | Weight Decay | Learning Rate | Scheduler | Factor | Patience | Initialization |
|---|---|---|---|---|---|---|---|---|
| **AlexNet** | Adam | 0.9 | 5e-4 | 0.01 | Reduce LR on Plateau | 0.1 | 10 | Xavier |
| **VGG16** | SGD | 0.9 | 5e-4 | 0.01 | Step LR | 0.1 | 30 | Xavier |
| **ResNet50** | SGD | 0.9 | 8e-4 | 0.1 | Step LR | 0.1 | 30 | Xavier |

Table 1: Model training hyperparameters, one of the special cases was that for training ResNet50 on the textured dataset we used a learning rate of 0.01.

| Model Name | Keyword | Training Accuracy | Validation Accuracy | Training Loss | Validation Loss |
|---|---|---|---|---|---|
| **AlexNet** | Ambient | 0.81 ± 0.10 | 0.59 ± 0.03 | 0.65 ± 0.43 | 2.37 ± 0.02 |
| | Less Variation | 0.76 ± 0.11 | 0.58 ± 0.02 | 0.89 ± 0.44 | 2.27 ± 0.03 |
| | Z axis | 0.90 ± 0.04 | 0.68 ± 0.01 | 0.39 ± 0.24 | 1.72 ± 0.04 |
| | No Variation | 0.81 ± 0.11 | 0.89 ± 0.04 | 0.69 ± 0.45 | 0.43 ± 0.19 |
| | Base | 0.66 ± 0.24 | 0.52 ± 0.08 | 1.32 ± 0.89 | 2.72 ± 0.22 |
| | Textures | 0.63 ± 0.10 | 0.47 ± 0.03 | 1.58 ± 0.49 | 3.11 ± 0.01 |
| | Without Shadows | 0.83 ± 0.03 | 0.56 ± 0.00 | 0.63 ± 0.05 | 2.61 ± 0.05 |
| | Without Shading | 0.81 ± 0.03 | 0.53 ± 0.00 | 0.74 ± 0.12 | 2.78 ± 0.02 |
| | No Specularities | 0.80 ± 0.04 | 0.58 ± 0.05 | 0.72 ± 0.16 | 2.54 ± 0.01 |
| **ResNet50** | Ambient | 0.80 ± 0.24 | 0.57 ± 0.18 | 1.03 ± 0.75 | 2.03 ± 0.92 |
| | Less Variation | 0.70 ± 0.28 | 0.50 ± 0.18 | 1.48 ± 1.11 | 2.26 ± 0.81 |
| | Z axis | 0.96 ± 0.06 | 0.74 ± 0.04 | 0.39 ± 0.41 | 1.18 ± 0.18 |
| | No Variation | 0.90 ± 0.08 | 0.79 ± 0.16 | 0.60 ± 0.35 | 0.91 ± 0.66 |
| | Base | 0.83 ± 0.19 | 0.59 ± 0.06 | 0.67 ± 0.73 | 1.95 ± 0.26 |
| | Textures | 0.93 ± 0.07 | 0.55 ± 0.01 | 0.42 ± 0.35 | 2.31 ± 0.04 |
| | Without Shadows | 0.99 ± 0.03 | 0.65 ± 0.03 | 0.08 ± 0.05 | 1.61 ± 0.01 |
| | Without Shading | 0.93 ± 0.15 | 0.60 ± 0.05 | 0.32 ± 0.60 | 1.86 ± 0.17 |
| | No Specularities | 0.74 ± 0.05 | 0.56 ± 0.01 | 1.08 ± 0.17 | 1.92 ± 0.04 |
| **VGG16** | Ambient | 0.78 ± 0.08 | 0.60 ± 0.01 | 0.93 ± 0.42 | 2.08 ± 0.01 |
| | Less Variation | 0.83 ± 0.06 | 0.62 ± 0.01 | 0.73 ± 0.10 | 2.00 ± 0.02 |
| | Z axis | 0.80 ± 0.04 | 0.73 ± 0.01 | 0.82 ± 0.15 | 1.33 ± 0.05 |
| | No Variation | 0.94 ± 0.03 | 0.97 ± 0.01 | 0.25 ± 0.14 | 0.10 ± 0.03 |
| | Base | 0.72 ± 0.08 | 0.57 ± 0.01 | 1.09 ± 0.34 | 2.30 ± 0.02 |
| | Textures | 0.66 ± 0.20 | 0.50 ± 0.07 | 1.50 ± 1.01 | 2.81 ± 0.28 |
| | Without Shading | 0.85 ± 0.14 | 0.18 ± 0.01 | 0.60 ± 0.14 | 4.50 ± 0.08 |
| | Without Shadows | 0.68 ± 0.13 | 0.20 ± 0.30 | 1.35 ± 0.65 | 4.40 ± 0.06 |
| | No Specularities | 0.75 ± 0.08 | 0.56 ± 0.01 | 0.99 ± 0.31 | 2.45 ± 0.02 |

Table 2: Performance metrics for different models and conditions

| Model | Layer Identifiers |
|---|---|
| **ResNet50** | `V1:layer1.0.conv1, V2:layer3.5.bn3, V4:layer3.0.conv1, IT:layer4.0.relu` [4] |
| **AlexNet** | `V1:features.2, V2:features.5, V4:features.7, IT:features.12` [5] |
| **VGG16** | `V1: features.16, V2: features.16, V4: features.16, IT: features.23` [6] |

Table 3: Layer identifiers assigned to different regions of the neural network for each model during benchmarking on *BrainScore*.

| Network | Keyword | Region | p-value (RSA) | p-value (CKA) | p-value (*BrainScore*) |
|---|---|---|---|---|---|
| **AlexNet** | ImageNet Trained | V1 | $\mathbf{5.11 \times 10^{-09}}$ | $\mathbf{1.30 \times 10^{-06}}$ | $\mathbf{6.13 \times 10^{-04}}$ |
| | | V2 | $\mathbf{2.11 \times 10^{-09}}$ | $\mathbf{6.98 \times 10^{-08}}$ | $\mathbf{1.52 \times 10^{-03}}$ |
| | | V3/IT | $\mathbf{5.95 \times 10^{-12}}$ | $\mathbf{3.11 \times 10^{-09}}$ | $\mathbf{6.88 \times 10^{-03}}$ |
| | | V4 | $\mathbf{2.65 \times 10^{-12}}$ | $\mathbf{3.74 \times 10^{-06}}$ | $\mathbf{2.56 \times 10^{-05}}$ |
| | Textures | V1 | $\mathbf{1.80 \times 10^{-05}}$ | $\mathbf{4.69 \times 10^{-04}}$ | $2.91 \times 10^{-01}$ |
| | | V2 | $\mathbf{3.55 \times 10^{-05}}$ | $\mathbf{2.36 \times 10^{-04}}$ | $2.04 \times 10^{-01}$ |
| | | V3/IT | $\mathbf{8.70 \times 10^{-05}}$ | $\mathbf{2.40 \times 10^{-04}}$ | $3.41 \times 10^{-01}$ |
| | | V4 | $1.40 \times 10^{-01}$ | $2.39 \times 10^{-01}$ | $\mathbf{2.69 \times 10^{-03}}$ |
| | Ambient | V1 | $\mathbf{4.86 \times 10^{-02}}$ | $2.90 \times 10^{-01}$ | $8.93 \times 10^{-01}$ |
| | | V2 | $5.65 \times 10^{-02}$ | $\mathbf{4.98 \times 10^{-02}}$ | $\mathbf{1.35 \times 10^{-02}}$ |
| | | V3/IT | $6.51 \times 10^{-01}$ | $\mathbf{1.90 \times 10^{-02}}$ | $\mathbf{3.92 \times 10^{-02}}$ |
| | | V4 | $7.08 \times 10^{-01}$ | $9.72 \times 10^{-01}$ | $\mathbf{1.17 \times 10^{-02}}$ |
| | Without shadows | V1 | $\mathbf{3.67 \times 10^{-03}}$ | $8.22 \times 10^{-01}$ | $6.81 \times 10^{-01}$ |
| | | V2 | $\mathbf{2.75 \times 10^{-03}}$ | $\mathbf{4.30 \times 10^{-03}}$ | $9.70 \times 10^{-02}$ |
| | | V3/IT | $\mathbf{1.99 \times 10^{-04}}$ | $\mathbf{2.19 \times 10^{-02}}$ | $6.90 \times 10^{-01}$ |
| | | V4 | $\mathbf{3.55 \times 10^{-03}}$ | $7.82 \times 10^{-01}$ | $\mathbf{2.95 \times 10^{-02}}$ |
| | Without Shading | V1 | $9.12 \times 10^{-01}$ | $8.38 \times 10^{-01}$ | $9.05 \times 10^{-01}$ |
| | | V2 | $5.40 \times 10^{-01}$ | $9.79 \times 10^{-01}$ | $2.33 \times 10^{-01}$ |
| | | V3/IT | $1.01 \times 10^{-01}$ | $6.00 \times 10^{-01}$ | $5.90 \times 10^{-01}$ |
| | | V4 | $\mathbf{3.53 \times 10^{-03}}$ | $1.62 \times 10^{-01}$ | $\mathbf{3.72 \times 10^{-03}}$ |
| | No specularities | V1 | $1.52 \times 10^{-01}$ | $9.62 \times 10^{-01}$ | $2.74 \times 10^{-01}$ |
| | | V2 | $8.73 \times 10^{-02}$ | $1.24 \times 10^{-01}$ | $1.63 \times 10^{-01}$ |
| | | V3/IT | $\mathbf{3.58 \times 10^{-02}}$ | $2.53 \times 10^{-01}$ | $3.73 \times 10^{-01}$ |
| | | V4 | $6.94 \times 10^{-02}$ | $8.78 \times 10^{-01}$ | $\mathbf{3.95 \times 10^{-02}}$ |
| | Less Variation | V1 | $7.99 \times 10^{-01}$ | $\mathbf{2.64 \times 10^{-02}}$ | $2.02 \times 10^{-01}$ |
| | | V2 | $5.07 \times 10^{-01}$ | $8.30 \times 10^{-01}$ | $3.72 \times 10^{-01}$ |
| | | V3/IT | $1.51 \times 10^{-01}$ | $3.33 \times 10^{-01}$ | $4.11 \times 10^{-01}$ |
| | | V4 | $1.12 \times 10^{-01}$ | $3.26 \times 10^{-01}$ | $5.78 \times 10^{-01}$ |
| | Z-axis | V1 | $8.50 \times 10^{-02}$ | $3.33 \times 10^{-01}$ | $\mathbf{3.92 \times 10^{-02}}$ |
| | | V2 | $5.52 \times 10^{-02}$ | $8.64 \times 10^{-01}$ | $8.26 \times 10^{-01}$ |
| | | V3 | $\mathbf{2.04 \times 10^{-03}}$ | $\mathbf{2.76 \times 10^{-02}}$ | $\mathbf{3.62 \times 10^{-02}}$ |
| | | V4 | $\mathbf{1.65 \times 10^{-02}}$ | $6.21 \times 10^{-01}$ | $\mathbf{1.16 \times 10^{-03}}$ |
| | No variation | V1 | $3.20 \times 10^{-01}$ | $1.39 \times 10^{-01}$ | $3.40 \times 10^{-01}$ |
| | | V2 | $3.96 \times 10^{-01}$ | $8.43 \times 10^{-01}$ | $3.83 \times 10^{-01}$ |
| | | V3/IT | $4.09 \times 10^{-01}$ | $8.39 \times 10^{-01}$ | $2.53 \times 10^{-01}$ |
| | | V4 | $7.31 \times 10^{-01}$ | $6.35 \times 10^{-02}$ | $6.39 \times 10^{-01}$ |
| | No training | V1 | $\mathbf{2.42 \times 10^{-06}}$ | $\mathbf{2.71 \times 10^{-04}}$ | $\mathbf{6.18 \times 10^{-04}}$ |
| | | V2 | $\mathbf{4.27 \times 10^{-07}}$ | $\mathbf{9.03 \times 10^{-06}}$ | $\mathbf{2.08 \times 10^{-03}}$ |
| | | V3/IT | $\mathbf{3.79 \times 10^{-07}}$ | $\mathbf{2.09 \times 10^{-07}}$ | $\mathbf{7.24 \times 10^{-03}}$ |
| | | V4 | $\mathbf{1.47 \times 10^{-06}}$ | $\mathbf{2.11 \times 10^{-04}}$ | $\mathbf{4.81 \times 10^{-05}}$ |

Table 4: P-values on independent samples t-test conducted on the alignment metrics computed on the *Base* dataset as well as the modified datasets for the model AlexNet. All significant values i.e. $p < 0.05$ are displayed in bold. The V3/IT marking shows that we have used V3 region for RSA and CKA but used the IT region for V4.

[4]GitHub: link
[5]GitHub: link
[6]GitHub: link

| Network | Keyword | Region | p-value (RSA) | p-value (CKA) | p-value (*BrainScore*) |
|---|---|---|---|---|---|
| **VGG16** | ImageNet Trained | V1 | $\mathbf{2.90 \times 10^{-13}}$ | $\mathbf{2.10 \times 10^{-10}}$ | $3.75 \times 10^{-01}$ |
| | | V2 | $\mathbf{2.38 \times 10^{-09}}$ | $\mathbf{1.17 \times 10^{-08}}$ | $9.26 \times 10^{-01}$ |
| | | V3/IT | $\mathbf{5.11 \times 10^{-08}}$ | $\mathbf{3.24 \times 10^{-06}}$ | $6.28 \times 10^{-01}$ |
| | | V4 | $\mathbf{2.61 \times 10^{-09}}$ | $\mathbf{1.95 \times 10^{-07}}$ | $4.50 \times 10^{-01}$ |
| | Textures | V1 | $1.06 \times 10^{-01}$ | $1.49 \times 10^{-01}$ | $3.54 \times 10^{-01}$ |
| | | V2 | $\mathbf{4.42 \times 10^{-02}}$ | $1.01 \times 10^{-01}$ | $\mathbf{1.14 \times 10^{-02}}$ |
| | | V3/IT | $\mathbf{3.90 \times 10^{-02}}$ | $1.79 \times 10^{-01}$ | $1.86 \times 10^{-01}$ |
| | | V4 | $\mathbf{1.55 \times 10^{-03}}$ | $\mathbf{1.85 \times 10^{-02}}$ | $7.79 \times 10^{-01}$ |
| | Ambient | V1 | $\mathbf{8.51 \times 10^{-07}}$ | $\mathbf{6.50 \times 10^{-03}}$ | $5.89 \times 10^{-01}$ |
| | | V2 | $\mathbf{2.82 \times 10^{-03}}$ | $\mathbf{7.13 \times 10^{-03}}$ | $\mathbf{2.58 \times 10^{-03}}$ |
| | | V3/IT | $1.83 \times 10^{-01}$ | $\mathbf{4.18 \times 10^{-02}}$ | $3.51 \times 10^{-01}$ |
| | | V4 | $1.26 \times 10^{-01}$ | $\mathbf{2.67 \times 10^{-03}}$ | $\mathbf{1.98 \times 10^{-04}}$ |
| | Without Shadows | V1 | $\mathbf{1.44 \times 10^{-05}}$ | $\mathbf{4.37 \times 10^{-03}}$ | $3.56 \times 10^{-01}$ |
| | | V2 | $\mathbf{2.95 \times 10^{-03}}$ | $\mathbf{1.89 \times 10^{-02}}$ | $2.20 \times 10^{-02}$ |
| | | V3 | $1.17 \times 10^{-01}$ | $\mathbf{1.74 \times 10^{-02}}$ | $2.20 \times 10^{-02}$ |
| | | V4 | $\mathbf{2.90 \times 10^{-02}}$ | $1.28 \times 10^{-04}$ | $1.64 \times 10^{-01}$ |
| | Without Shading | V1 | $\mathbf{1.45 \times 10^{-03}}$ | $\mathbf{4.58 \times 10^{-03}}$ | $4.64 \times 10^{-02}$ |
| | | V2 | $1.67 \times 10^{-01}$ | $4.01 \times 10^{-02}$ | $4.95 \times 10^{-02}$ |
| | | V3 | $8.10 \times 10^{-01}$ | $4.96 \times 10^{-01}$ | $\mathbf{2.03 \times 10^{-05}}$ |
| | | V4 | $2.14 \times 10^{-01}$ | $4.06 \times 10^{-04}$ | $4.59 \times 10^{-02}$ |
| | No Specular | V1 | $7.36 \times 10^{-01}$ | $5.72 \times 10^{-01}$ | $6.85 \times 10^{-01}$ |
| | | V2 | $4.03 \times 10^{-01}$ | $3.16 \times 10^{-01}$ | $8.96 \times 10^{-03}$ |
| | | V3 | $1.91 \times 10^{-01}$ | $2.78 \times 10^{-01}$ | $1.70 \times 10^{-02}$ |
| | Less Variation | V1 | $8.39 \times 10^{-01}$ | $4.70 \times 10^{-01}$ | $5.90 \times 10^{-01}$ |
| | | V2 | $8.55 \times 10^{-01}$ | $2.27 \times 10^{-01}$ | $6.29 \times 10^{-01}$ |
| | | V3/IT | $8.30 \times 10^{-01}$ | $8.33 \times 10^{-01}$ | $6.92 \times 10^{-01}$ |
| | | V4 | $7.19 \times 10^{-01}$ | $8.56 \times 10^{-02}$ | $2.38 \times 10^{-01}$ |
| | Z axis | V1 | $\mathbf{1.82 \times 10^{-02}}$ | $1.41 \times 10^{-01}$ | $5.08 \times 10^{-01}$ |
| | | V2 | $\mathbf{4.35 \times 10^{-02}}$ | $6.42 \times 10^{-01}$ | $\mathbf{3.42 \times 10^{-03}}$ |
| | | V3/IT | $1.32 \times 10^{-01}$ | $6.69 \times 10^{-01}$ | $1.00 \times 10^{+00}$ |
| | | V4 | $7.66 \times 10^{-02}$ | $9.55 \times 10^{-01}$ | $8.02 \times 10^{-01}$ |
| | No Variation | V1 | $\mathbf{3.39 \times 10^{-02}}$ | $\mathbf{4.97 \times 10^{-06}}$ | $6.24 \times 10^{-01}$ |
| | | V2 | $3.42 \times 10^{-01}$ | $\mathbf{7.49 \times 10^{-04}}$ | $1.41 \times 10^{-01}$ |
| | | V3/IT | $3.09 \times 10^{-01}$ | $3.90 \times 10^{-01}$ | $6.85 \times 10^{-02}$ |
| | | V4 | $\mathbf{2.48 \times 10^{-02}}$ | $\mathbf{5.44 \times 10^{-03}}$ | $7.87 \times 10^{-01}$ |
| | No Training | V1 | $\mathbf{9.60 \times 10^{-04}}$ | $\mathbf{6.94 \times 10^{-04}}$ | $9.51 \times 10^{-01}$ |
| | | V2 | $\mathbf{1.19 \times 10^{-04}}$ | $\mathbf{7.82 \times 10^{-05}}$ | $4.81 \times 10^{-01}$ |
| | | V3/IT | $\mathbf{8.81 \times 10^{-04}}$ | $\mathbf{3.73 \times 10^{-04}}$ | $\mathbf{1.74 \times 10^{-03}}$ |
| | | V4 | $\mathbf{6.96 \times 10^{-05}}$ | $\mathbf{2.25 \times 10^{-06}}$ | $1.15 \times 10^{-01}$ |

Table 5: P-values on independent samples t-test conducted on the alignment metrics computed on the *Base* dataset as well as the modified datasets for the model VGG16.All significant values i.e. $p < 0.05$ are displayed in bold. The V3/IT marking shows that we have used the V3 region for RSA and CKA but used the IT region for V4.

| Network | Keyword | Region | p-value (RSA) | p-value (CKA) | p-value (*BrainScore*) |
|---|---|---|---|---|---|
| **ResNet50** | ImageNet Trained | V1 | $7.80 \times 10^{-01}$ | $\mathbf{7.60 \times 10^{-47}}$ | $6.68 \times 10^{-01}$ |
| | | V2 | $4.07 \times 10^{-01}$ | $\mathbf{5.87 \times 10^{-48}}$ | $2.92 \times 10^{-01}$ |
| | | V3/IT | $\mathbf{8.87 \times 10^{-04}}$ | $\mathbf{3.70 \times 10^{-39}}$ | $6.54 \times 10^{-01}$ |
| | | V4 | $\mathbf{2.99 \times 10^{-07}}$ | $\mathbf{5.38 \times 10^{-47}}$ | $6.47 \times 10^{-01}$ |
| | Textures | V1 | $8.24 \times 10^{-01}$ | $\mathbf{0.00 \times 10^{+00}}$ | $2.93 \times 10^{-01}$ |
| | | V2 | $4.26 \times 10^{-01}$ | $\mathbf{0.00 \times 10^{+00}}$ | $3.38 \times 10^{-01}$ |
| | | V3/IT | $8.84 \times 10^{-01}$ | $\mathbf{2.52 \times 10^{-117}}$ | $3.72 \times 10^{-01}$ |
| | | V4 | $9.56 \times 10^{-02}$ | $\mathbf{0.00 \times 10^{+00}}$ | $5.80 \times 10^{-01}$ |
| | Ambient | V1 | $1.23 \times 10^{-01}$ | $\mathbf{0.00 \times 10^{+00}}$ | $\mathbf{9.54 \times 10^{-03}}$ |
| | | V2 | $4.01 \times 10^{-01}$ | $\mathbf{0.00 \times 10^{+00}}$ | $\mathbf{1.97 \times 10^{-02}}$ |
| | | V3/IT | $4.03 \times 10^{-01}$ | $\mathbf{0.00 \times 10^{+00}}$ | $6.70 \times 10^{-02}$ |
| | | V4 | $4.92 \times 10^{-01}$ | $\mathbf{0.00 \times 10^{+00}}$ | $7.15 \times 10^{-01}$ |
| | Without Shadows | V1 | $8.47 \times 10^{-01}$ | $\mathbf{1.69 \times 10^{-03}}$ | $3.81 \times 10^{-01}$ |
| | | V2 | $8.66 \times 10^{-01}$ | $\mathbf{2.59 \times 10^{-03}}$ | $3.02 \times 10^{-01}$ |
| | | V3/IT | $3.86 \times 10^{-01}$ | $\mathbf{2.95 \times 10^{-03}}$ | $9.44 \times 10^{-01}$ |
| | | V4 | $\mathbf{4.45 \times 10^{-02}}$ | $\mathbf{2.76 \times 10^{-07}}$ | $8.27 \times 10^{-01}$ |
| | Without shading | V1 | $1.54 \times 10^{-01}$ | $\mathbf{4.85 \times 10^{-04}}$ | $8.81 \times 10^{-01}$ |
| | | V2 | $1.79 \times 10^{-01}$ | $\mathbf{8.86 \times 10^{-05}}$ | $4.96 \times 10^{-01}$ |
| | | V3/IT | $8.85 \times 10^{-02}$ | $\mathbf{1.84 \times 10^{-02}}$ | $9.18 \times 10^{-01}$ |
| | | V4 | $1.10 \times 10^{-01}$ | $\mathbf{1.52 \times 10^{-04}}$ | $6.64 \times 10^{-02}$ |
| | No specularities | V1 | $1.11 \times 10^{-01}$ | $\mathbf{1.79 \times 10^{-05}}$ | $1.10 \times 10^{-01}$ |
| | | V2 | $4.06 \times 10^{-01}$ | $\mathbf{7.33 \times 10^{-10}}$ | $1.50 \times 10^{-01}$ |
| | | V3/IT | $7.68 \times 10^{-01}$ | $\mathbf{4.73 \times 10^{-06}}$ | $9.75 \times 10^{-02}$ |
| | | V4 | $9.13 \times 10^{-01}$ | $\mathbf{2.51 \times 10^{-05}}$ | $5.25 \times 10^{-01}$ |
| | Less Variation | V1 | $5.05 \times 10^{-01}$ | $2.45 \times 10^{-01}$ | $4.88 \times 10^{-01}$ |
| | | V2 | $4.25 \times 10^{-01}$ | $9.60 \times 10^{-01}$ | $9.25 \times 10^{-01}$ |
| | | V3/IT | $6.80 \times 10^{-01}$ | $4.84 \times 10^{-01}$ | $7.09 \times 10^{-01}$ |
| | | V4 | $9.07 \times 10^{-01}$ | $\mathbf{3.71 \times 10^{-03}}$ | $2.44 \times 10^{-01}$ |
| | Z axis | V1 | $7.00 \times 10^{-01}$ | $\mathbf{1.58 \times 10^{-03}}$ | $4.25 \times 10^{-01}$ |
| | | V2 | $5.80 \times 10^{-01}$ | $\mathbf{2.07 \times 10^{-02}}$ | $2.74 \times 10^{-01}$ |
| | | V3/IT | $\mathbf{3.67 \times 10^{-02}}$ | $\mathbf{1.20 \times 10^{-03}}$ | $4.54 \times 10^{-01}$ |
| | | V4 | $\mathbf{5.71 \times 10^{-03}}$ | $\mathbf{3.52 \times 10^{-07}}$ | $6.40 \times 10^{-02}$ |
| | No Variation | V1 | $8.65 \times 10^{-01}$ | $\mathbf{8.64 \times 10^{-04}}$ | $6.68 \times 10^{-01}$ |
| | | V2 | $1.80 \times 10^{-01}$ | $\mathbf{8.56 \times 10^{-03}}$ | $2.92 \times 10^{-01}$ |
| | | V3/IT | $\mathbf{8.43 \times 10^{-03}}$ | $\mathbf{3.14 \times 10^{-03}}$ | $6.54 \times 10^{-01}$ |
| | | V4 | $\mathbf{3.85 \times 10^{-03}}$ | $\mathbf{6.10 \times 10^{-03}}$ | $6.47 \times 10^{-01}$ |
| | No training | V1 | $\mathbf{4.42 \times 10^{-05}}$ | $\mathbf{1.33 \times 10^{-08}}$ | $7.43 \times 10^{-01}$ |
| | | V2 | $\mathbf{1.38 \times 10^{-06}}$ | $\mathbf{9.52 \times 10^{-12}}$ | $\mathbf{3.52 \times 10^{-02}}$ |
| | | V3/IT | $\mathbf{6.70 \times 10^{-07}}$ | $\mathbf{5.47 \times 10^{-12}}$ | $9.67 \times 10^{-02}$ |
| | | V4 | $\mathbf{1.70 \times 10^{-07}}$ | $\mathbf{1.23 \times 10^{-10}}$ | $1.49 \times 10^{-01}$ |

Table 6: P-values on independent samples t-test conducted on the alignment metrics computed on the *Base* dataset as well as the modified datasets for the model ResNet50. All significant values i.e. $p < 0.05$ are displayed in bold. The V3/IT marking shows that we have used the V3 region for RSA and CKA but used the IT region for V4.

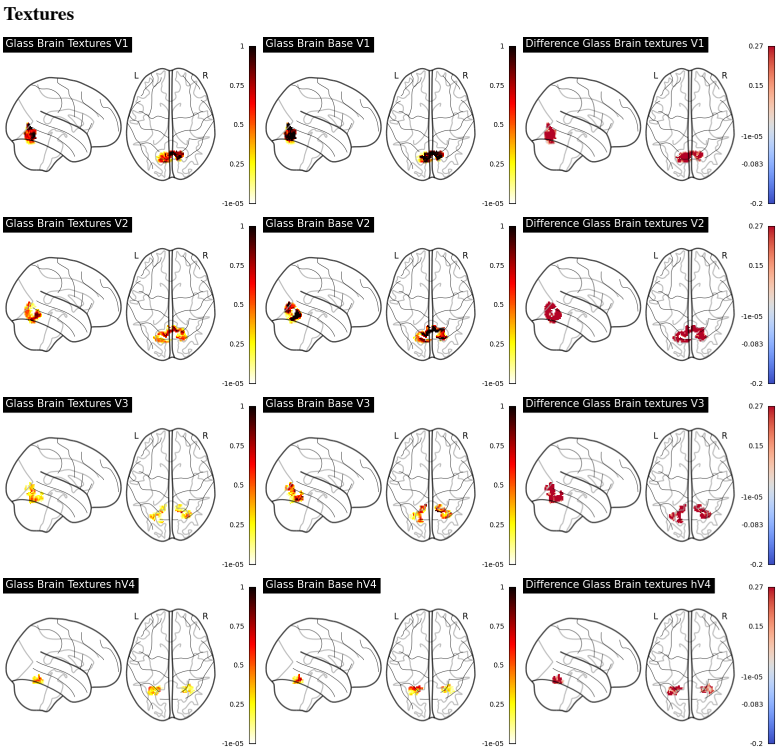



Figure 9: Visualization of responses to the encoding model specified in Regions of Interest(ROIs). The model's predictions on the synthetic dataset are shown using Nilearn's `plot_glass_brain` function. The Base dataset responses are visualized next, followed by a difference map computed from the two datasets. The difference is rendered using a hot colormap to highlight both the direction and intensity of changes across brain regions.

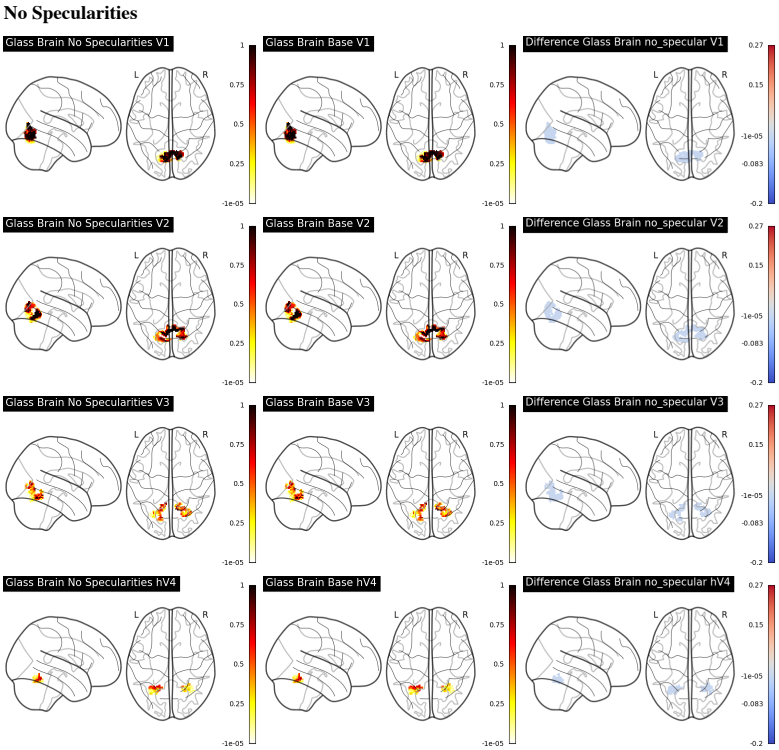



Figure 10: Visualization of responses to the encoding model specified in Regions of Interest (ROIs). The model's predictions on the synthetic dataset are shown using Nilearn's `plot_glass_brain` function. The Base dataset responses are visualized next, followed by a difference map computed from the two datasets. The difference is rendered using a hot colormap to highlight both the direction and intensity of changes across brain regions.

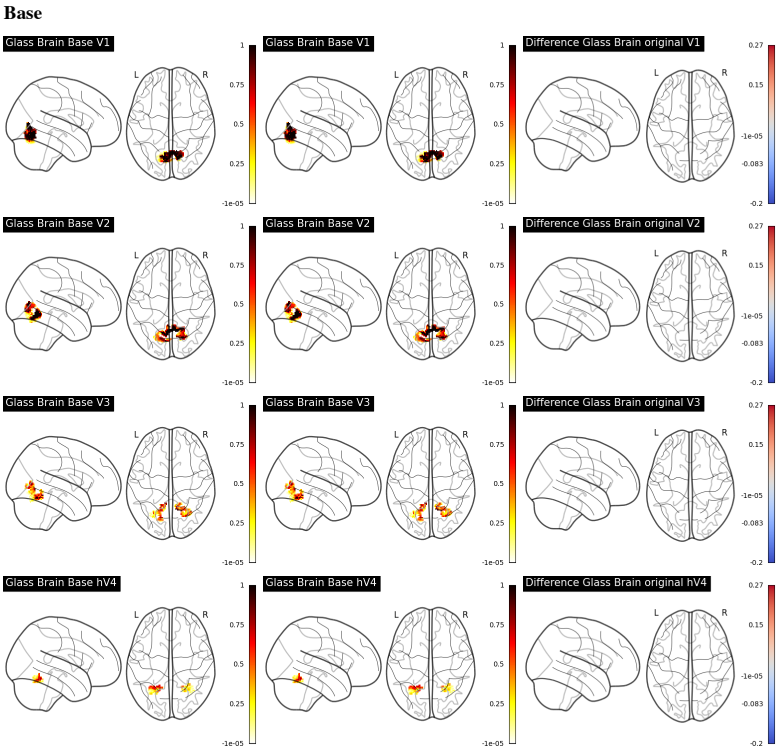



Figure 11: Visualization of responses to the encoding model specified in Regions of Interest (ROIs). The model's predictions on the synthetic dataset are shown using Nilearn's `plot_glass_brain` function. The Base dataset responses are visualized next, followed by a difference map computed from the two datasets. The difference is rendered using a hot colormap to highlight both the direction and intensity of changes across brain regions.

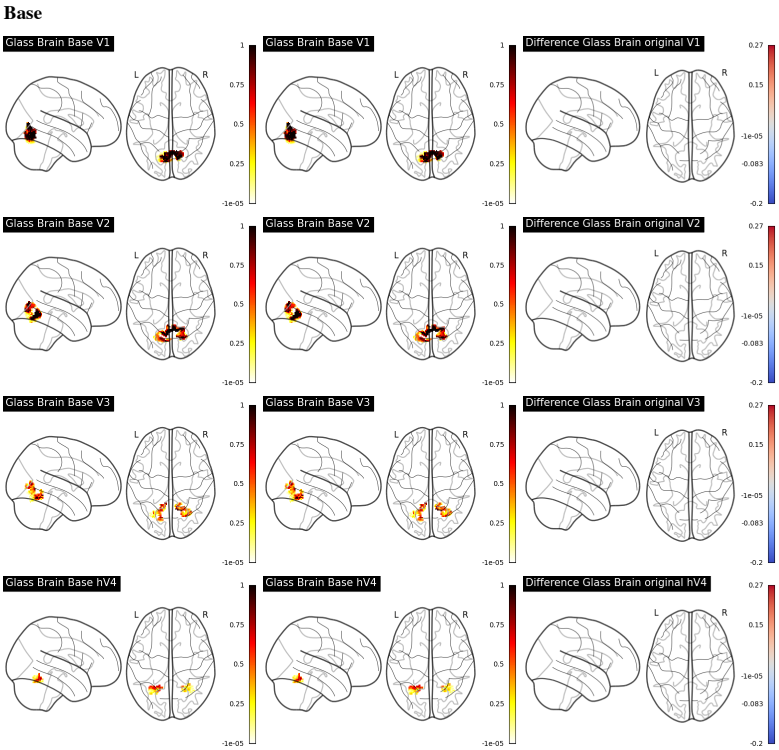



Figure 12: Visualization of responses to the encoding model specified in Regions of Interest (ROIs). The model's predictions on the synthetic dataset are shown using Nilearn's `plot_glass_brain` function. The Base dataset responses are visualized next, followed by a difference map computed from the two datasets. The difference is rendered using a hot colormap to highlight both the direction and intensity of changes across brain regions.

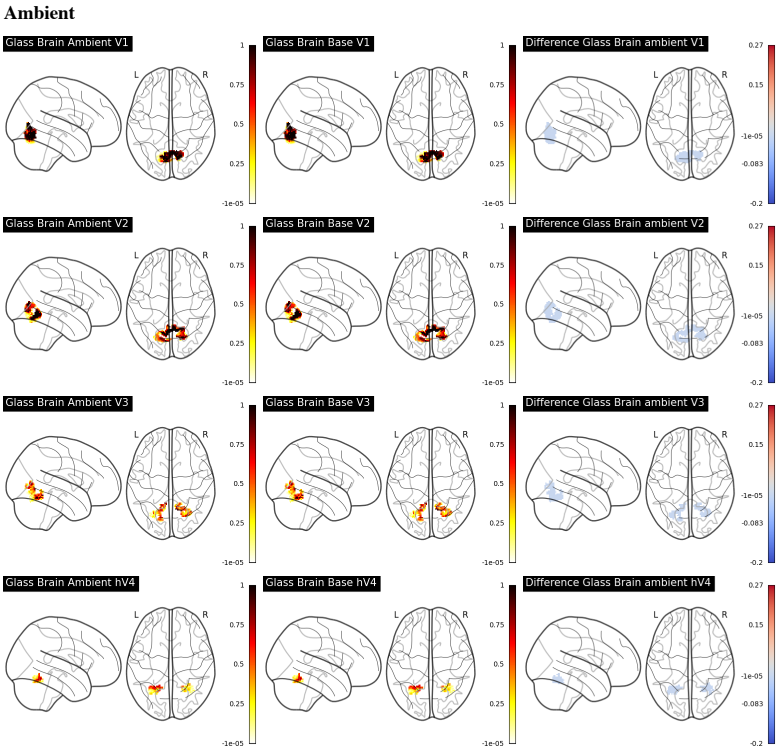



Figure 13: Visualization of responses to the encoding model specified in Regions of Interest (ROIs). The model's predictions on the synthetic dataset are shown using Nilearn's `plot_glass_brain` function. The Base dataset responses are visualized next, followed by a difference map computed from the two datasets. The difference is rendered using a hot colormap to highlight both the direction and intensity of changes across brain regions.

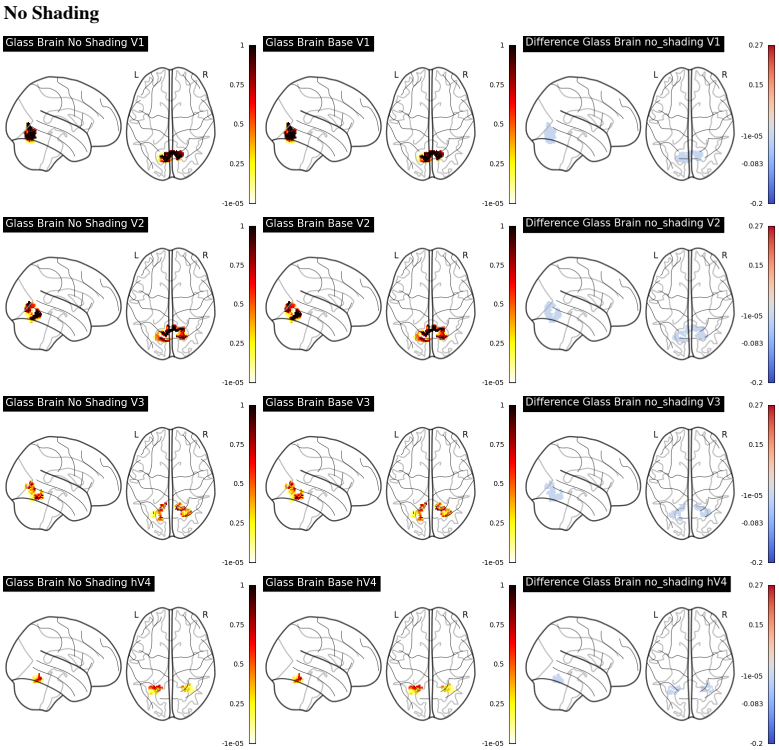



Figure 14: Visualization of responses to the encoding model specified in Regions of Interest (ROIs). The model's predictions on the synthetic dataset are shown using Nilearn's `plot_glass_brain` function. The Base dataset responses are visualized next, followed by a difference map computed from the two datasets. The difference is rendered using a hot colormap to highlight both the direction and intensity of changes across brain regions.