

# RECONSTRUCTION OF VISUAL STIMULI FROM NEURAL ACTIVITY

A THESIS SUBMITTED FOR THE COMPLETION OF  
REQUIREMENTS FOR THE DEGREE OF

BACHELOR OF SCIENCE  
(RESEARCH)

BY

SUDHANSHU BHARADWAJ  
UNDERGRADUATE PROGRAMME  
INDIAN INSTITUTE OF SCIENCE



UNDER THE SUPERVISION OF

PROF. S.P. ARUN  
CENTRE FOR NEUROSCIENCE, INDIAN INSTITUTE OF SCIENCE



# Certificate

This is to certify that the thesis titled "Reconstruction of Visual Stimulus from Neural Activity" is the outcome of original work carried out by Sudhanshu Bharadwaj for the degree of Bachelor of Science (Research) with a Major in Biology at the Indian Institute of Science, Bangalore, India.

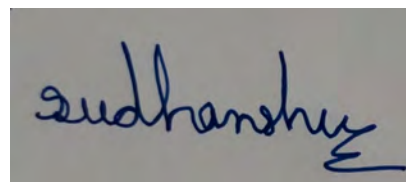


S.P. Arun  
Professor  
Centre for Neuroscience  
Indian Institute of Science  
Bangalore 560012

Date: 21 Apr 2024

# Declaration

This is to certify that the work presented in this thesis is my own and has been done under the guidance of Prof. S.P. Arun. This thesis has not formed the basis for the award of any degree, diploma, membership, or similar title of any university or institution.

A handwritten signature in blue ink, appearing to read 'Sudhanshu', is shown within a rectangular frame.

Sudhanshu Bharadwaj,  
Undergraduate,  
Indian Institute of Science.

# Acknowledgements

This thesis would not have been possible without the help and support of several people. I would like to thank Dr. S.P. Arun for his guidance and support throughout the project, and for giving me so much freedom while working in his lab. This incredible opportunity for me to design, collect and analyze high-quality data would not have been possible without his support, and him giving so much time for discussions and meetings. I would also like to thank everyone who had worked on setting up "series4", without which I couldn't have had access to much data. I'd also like to thank Georgin, my collaborator, without whom it would have been impossible to even start this project. His suggestions through all stages of the project were invaluable. I also had an amazing collaborative experience with Jason, with whom I trained monkeys to play video games but sadly it didn't make it into the thesis due to time constraints).

This thesis is also a culmination of my Bachelor's degree at IISc. I'm grateful to all my friends who have made my life at IISc much more enjoyable. Lastly, I would like to thank my parents, who have always given me their unconditional support.

*Sudhanshu*

# Contents

<b>1 Abstract</b>	<b>2</b>
Keywords/Notation . . . . .	3
<b>2 Introduction</b>	<b>4</b>
1. Inferior Temporal Cortex: High level representation of visual information . . . . .	4
2. Latent representations of Images . . . . .	5
3. Generative Image Models . . . . .	6
Generative Adversarial Networks . . . . .	6
4. Models used in this study . . . . .	7
A. ResNet Logits Inverted GANs . . . . .	7
B. BigBiGAN . . . . .	8
C. StyleGAN-XL . . . . .	8
5. Metric to evaluate reconstruction . . . . .	9
6. Earlier work in Reconstruction . . . . .	9
<b>3 Experimental Design</b>	<b>11</b>
Stimuli . . . . .	11
Out-of-natural-distribution Images . . . . .	12
Task . . . . .	13
<b>4 Results</b>	<b>14</b>
Sample Responses . . . . .	14
Representation of images in IT and different models . . . . .	16
Inter-day alignment . . . . .	16
Prediction of Neural response using different models . . . . .	17
Reconstruction of Natural Images . . . . .	18
Category specific Reconstruction . . . . .	20
Reconstruction of Out-of-natural-distribution Images . . . . .	21
<b>5 Discussion</b>	<b>26</b>
Future Work . . . . .	27
<b>6 Methods</b>	<b>28</b>
Data Collection . . . . .	28
Data Pre-Processing . . . . .	28
Neural data analysis . . . . .	28

Reconstruction Pipeline . . . . .	29
Quantification of Reconstruction . . . . .	31
Implementation Details . . . . .	32
 <b>I   Appendix</b>	 <b>33</b>
Setup . . . . .	34
More reconstructions . . . . .	34
Latent reconstructions . . . . .	35
Out-of-natural-distribution Images . . . . .	35

# Reconstruction of Visual Stimuli from Neural Activity



# Abstract

Visual areas in the brain are known to represent a wide variety of visual stimuli, ranging from natural images to abstract patterns. Different regions encode representations of the image at different scales and exhibit various levels of invariance. Reconstruction of visual stimuli from neural activity has been a long-standing goal in neuroscience, as it provides insights into the underlying neural representation of visual information. In this study <sup>1</sup> we reconstruct images from their neural activity in IT neurons, using three GAN's (each of which have a different latent representations of the images) and compare these reconstructions qualitatively and quantitatively. Through this we also aim to set a benchmarks for the reconstruction using different GAN's. We also look at reconstruction of several out-of-distribution classes of images which might help us how the brain represents different classes of images and what invariances are present in the neural representation.

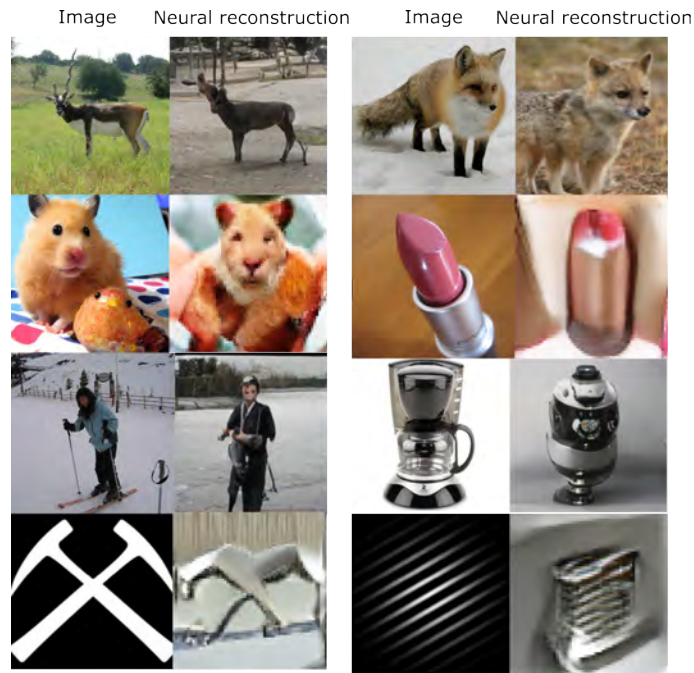


Figure 1.1: Example neural reconstructions for a few stimuli

---

<sup>1</sup>This project is a collaborative effort between me, Dr. S.P. Arun and Dr. Georgin Jacob

## Keywords/Notation

- MUA: Multi-Unit Activity
- IT: Inferior Temporal Cortex. The region from which neural data is recorded in this study.
- GAN: Generative Adversarial Network
- CNN: Convolutional Neural Network
- $\mathcal{I}$ : Image pixel space (dimensionality  $\approx 224 \times 224 \times 3$ )
- $\mathcal{N}$ : Neural space (dimensionality  $\approx n$  neurons=128)
- $\mathcal{L}$ : Latent space (dimensionality  $\approx d=500$ )
- $G$ : Generator in GAN: function that maps a latent vector to an image ( $\mathcal{L}$  to  $\mathcal{I}$ )
- $E$ : Encoder in GAN: function that maps images to features (latents) ( $\mathcal{I}$  to  $\mathcal{L}$ )
- Latent Reconstruction: Reconstruction of images from latents  $= G(E(Img))$
- Neural Reconstruction: Reconstruction of images from neural activity  $= G(D(NeuralData))$
- VGG-16 and ResNet : Popular CNN architectures used for feature extraction/classification
- Out-of-distribution images: In this study, images that are different from rectangular natural object images, and are often poorly represented in deep neural networks.

# Introduction

The brain is unparalleled in its ability to understand a huge variety of visual scenes and objects. Vision and its perception require a nuanced neural representation that captures the invariances and statistical regularities in the visual world. Understanding and decoding the underlying ‘*neural code*’ for such visual information has been a long-standing goal in neuroscience. In the past decade, the rise of deep neural networks (DNNs) has provided a new framework to understand such representations by allowing for much better decoding [1]. Visual decoding can be of several types, from stimuli category classification, to prediction of visual features, or even complete reconstruction of visual stimuli [2].

The goal of visual reconstruction is to completely reconstruct the visual stimulus from neural activity. Most importantly it offers a direct window to look at the visual representation and interpret decoding results. However, this is a challenging task. One of the key problems is the high dimensionality of the visual pixel space  $\mathcal{I}$  (e.g.  $224 \times 224 \times 3$  for an RGB image). One of the ways to tackle this problem is to utilize a low dimensional (*latent*) representation of images instead the pixel space  $\mathcal{I}$ . The choice of this latent representation is crucial, and best reconstruction can only be achieved when the latent representation captures the structure and invariances in the neural regions that are being recorded from.

## Inferior Temporal Cortex: High level representation of visual information

The ventral visual pathway in primates is known to represent visual information in a hierarchical fashion. Early visual areas like V1 and V2 are known to represent low level features like edges, and corners. Neurons in these areas have relatively small receptive fields. As one goes up the hierarchy to V4 and IT, the receptive fields increase in size, and the neural representation becomes more complex, coding for features like textures, shapes, contours or even objects and faces.[3]

The inferior temporal cortex (IT) is the final stage of this pathway and is known to encode high-level categorical visual information. IT neurons are known to be selective to complex shapes, objects, and faces [4]. The representation in IT is also known to be invariant to various perturbations like viewpoint, scale and illumination. [5] [6] This invariance is crucial to be able to recognize objects across different conditions, given the huge variability in retinal or pixel space  $\mathcal{I}$  for such perturbations. Another way of looking at this is that the IT carries disentangled representation [7] of visual information. What this means that representations of different objects are easily separable, which is not the case in retinal or pixel space  $\mathcal{I}$ . Therefore, the IT plays a crucial role in the recognition of objects and scenes, and being able to reconstruct visual stimuli from IT activity can provide insights into the neural representation of visual information. Several deep neural networks also capture such high level or disentangled representations, and are going to play a crucial role in this study.

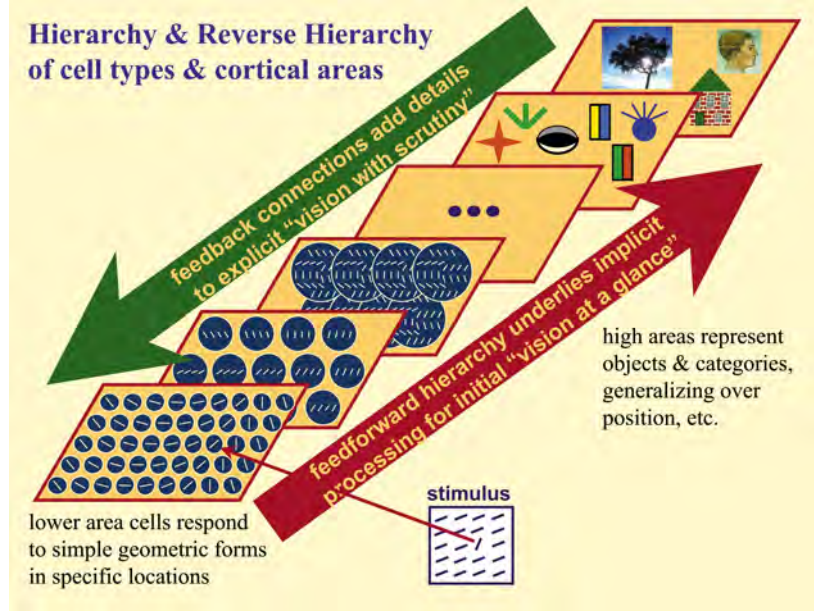


Figure 2.1: Hierarchical representation of visual information in the ventral pathway. (Figure adapted from [3])

## Latent representations of Images: A bridge between neural and image space

The goal of reconstruction is to find a **mapping from neural space to image pixel space**  $f : \mathcal{N} \rightarrow \mathcal{I}$ . As mentioned before, the high dimensionality of the image pixel space  $I$  makes this a challenging task. More importantly, pixel values in natural images are highly correlated and follow a certain structure, which makes independent reconstruction of each pixels a poor choice. However, this high pixel-wise correlation implies that the image space  $\mathcal{I}$  is highly redundant, and natural images lie on a low dimensional manifold in this space.[8] For example, all Gabor images in  $\mathcal{I}$  can be summarized by 3 "hidden" parameters - orientation, frequency and phase. Therefore, they can all be represented in a 3-dimensional latent space  $\mathcal{L}$ . Although natural images are much more complex, studies show that the *intrinsic dimension* of natural image datasets like Imagenet[9] (which has millions of images and thousands of categories) is around 40. [10] Therefore, it must be possible to find a low dimensional representation of images,  $\mathcal{L}$ . To effectively represent images in  $\mathcal{I}$ , dimensions in  $\mathcal{L}$  encode higher level visual features, making the mapping  $f : \mathcal{N} \rightarrow \mathcal{L}$  easier to learn. Further, the dimensions in  $\mathcal{L}$  are more independent, which allows independent decoding of each dimension of  $\mathcal{L}$ . In addition, due to limitations in data size, it would be desirable if latent and neural spaces are related linearly to allow linear decoding. Therefore, the originally ill-posed problem of reconstruction can be broken down into two parts:

1. Finding a low dimensional representation of images  $\mathcal{L}$ , with a generator  $G : \mathcal{L} \rightarrow I$  that can *generate* pixel-space images from this representation. It should also be possible to encode images in latent space with an encoder  $E : I \rightarrow \mathcal{L}$ . Importantly  $\mathcal{L}, G$  and  $E$  can be learnt independently of the neural data. This allows us to harness powerful pre-trained generative models trained on large datasets like *ImageNet*[9].
2. Finding a linear transformation from neural space to latent space. This can be done use  $\dim(\mathcal{L})$

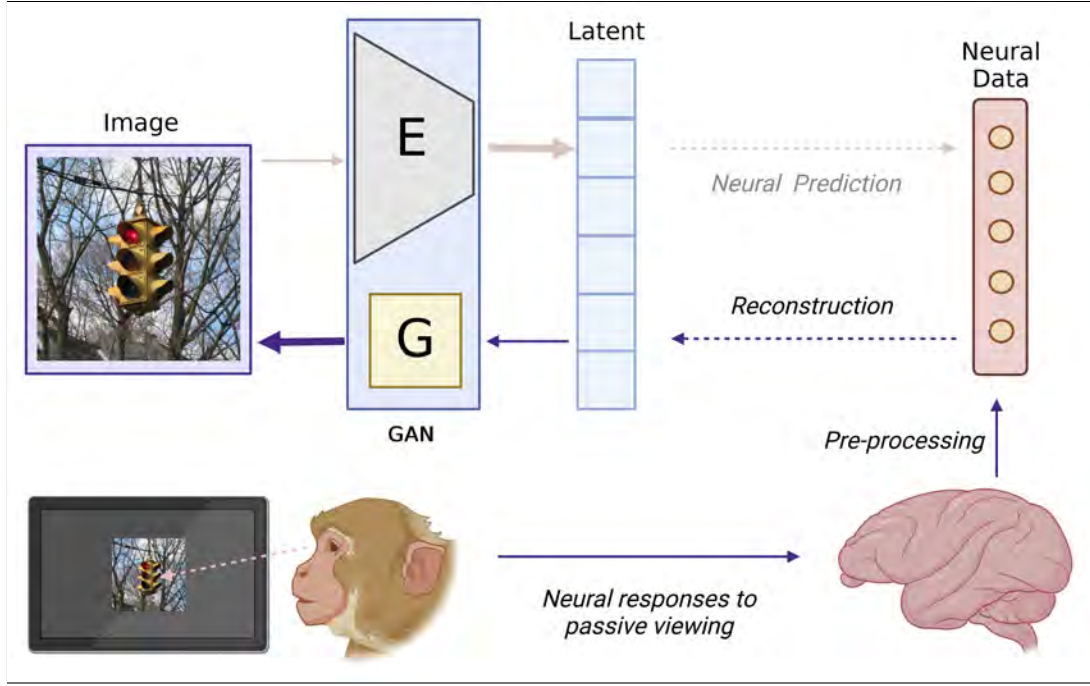


Figure 2.2: Overview of the reconstruction pipeline. Further details about the pipeline are in [methods 6.1](#)

separate linear decoders  $D_i : \mathcal{N} \rightarrow \mathcal{L}_i$  that can predict each dimension of the latent space  $\mathcal{L}$  from neural activity.

## Generative Image Models

Following the discussion in the previous section, the generative model chosen should have disentangled low-dimensional image representation close to that of IT. This restricts our choice of models. Both Diffusion and VAEs have image like latent-representation (of size (x,y,n-features)) with correlations between different values. In addition the latent space of Diffusion models are often high dimensional (32\*32\*4 for stable diffusion), and Variational AutoEncoders's are poor at image generation. GANs on the other had, are a perfect choice for generating a latent representation.

### Generative Adversarial Networks(GAN)

GANs are a class of generative models that aim learn the underlying distribution of the data (like car images) and generate new samples from this distribution. GANs consist of two networks, a generator  $G$  and a discriminator  $D$ . The generator takes random noise vectors  $z$  (usually generated from a multivariate gaussian distribution) and generates an image  $G(z)$ . The discriminator is trained to distinguish between real images and generated images,  $G(z)$ . This leads to adversarial training, where the generator tries to generate images that are indistinguishable from real images, while the discriminator tries to distinguish between real and generated images. The generator thus learns the underlying distribution of the images conditioned on  $z$ . These  $z$  vectors capture the underlying structure of the images in  $\mathcal{I}$  into a low dimensional latent space  $\mathcal{L}$ .

The precise nature of the latent space learnt is dependent on training objectives and model architecture. It is also possible to train conditional GANs[11], where the generator is conditioned on additional information  $c$  (usually class-labels). Thus, the generator  $G(z, c)$  can *generate image conditioned on categorical level information*. This is quite useful in our case, as IT neurons contain category specific information. This allows us to utilize the latent space of any image-encoding model (like a CNN) with the GAN framework.

## GAN Models used in this study

As potential candidates for the generator in our reconstruction task, we consider different GAN models trained on imagenet [9], each of which have a unique latent representations of images. Since the latent space is of lower dimensionality, the GAN’s latent space doesn’t encode images perfectly. That is to say  $G(E(Img)) \approx Img$ , the model itself has a reconstruction loss, and doesn’t hold all the information of the image in the latent space. We consider three different models with interesting latent spaces, and compare the quality of reconstructions from each of them.  $G(E(Img)) \approx Img$  for the different models have been visualized in fig6, to give an idea of the quality of reconstructions from each model.

Table 2.1: GAN Models used in this study. Direct latent reconstructions are shown in the appendix fig6

Model	Latent space dimensionality	Property of Latent space favorable for reconstruction
ResNet logits inverted GANs	1000 (to $64 \times 64 \times 3$ )	Features encoded are very similar to IT (In our study we demonstrate that this model has the highest predictive power in predicting neural data). Also has a <i>robust</i> variant, which is trained with adversarial Images
BigBiGAN	128 (to $128 \times 128 \times 3$ )	Very Low Dimensional Latent representation.
StyleGAN-XL	512 (to $128 \times 128 \times 3$ in this study; can generate images with resolution upto $1024 \times 1024 \times 3$ )	Disentangled representation, high quality reconstructions possible, which avoid any bottleneck due to model choice

### A. ResNet Logits Inverted GANs

CNNs(Convolutional Neural Network) are the most common neural networks used to categorize objects. They have a hierarchical representation of visual information, with early layers representing low level features and later layers representing high level features. The representation of images in these final layers of a CNN are low dimensional are similar to that of IT.[12]

ResNet-152 is popular CNN architecture trained on Imagenet, that excels at classification, and thus capturing high-level visual features. *Logits* are its final layer (before normalizing the probabilities) in the classification task. For ResNet-152, the logits are a 1000-dimensional vector, corresponding to the 1000 image categories. This high-level representation still captures a lot of information about the image, and can be used to generate images.



We particularly use the models described in [13] that "invert" these logits/features to generate images. The models are conditional GANs (described in the previous section), which means that they use the CNN features  $y$  along with the noise vector  $z$  to generate images  $G(z, y)$ . The encoder for this model is the CNN itself, which is used to extract the logits from the image. We have chosen two variants of the ResNet GAN. One is based on a normal ResNet-152 model, while the other is based a ResNet-152 model trained on noisy images. Such adversarial training is known to improve the robustness of the encoding, allowing it to better generalize to different types of images, which is crucial for our goal of reconstructing out-of-natural-distribution images. However, since the latent space was designed for classification not reconstruction, the reconstruction quality is pretty poor. The current architecture we use outputs images with a resolution of only  $64 \times 64 \times$  pixels.

## B. BigBiGAN

The BiGAN is a novel model trained to learn a bidirectional mapping between images and latents.[14] The discriminator  $D$  in this model, doesn't merely discriminate (real vs generated) the Image, i.e.  $G(z)$  vs  $Img \in \mathcal{I}$  like most GANs. Rather it discriminates both the latent and the Image  $(z, G(z))$  vs  $E((Img), Img) \in (\mathcal{L}, \mathcal{I})$ . This allows the encoder and generator to be trained simultaneously, allowing for a much more condensed latent space  $\mathcal{L}$ . Using merely a 128-dimensional latent space, BigBiGAN is able to generate images of resolution  $128 \times 128 \times 3$  pixels, on a large scale dataset like ImageNet.[15]

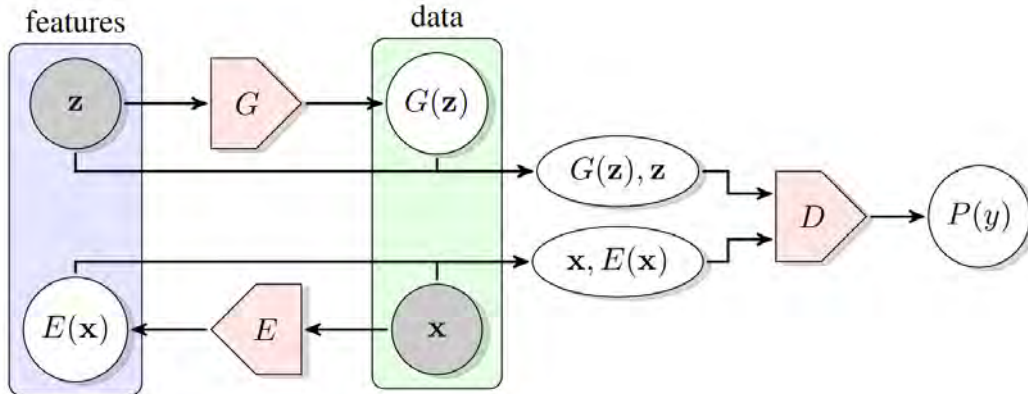


Figure 2.3: The structure of Bidirectional Generative Adversarial Networks (BiGAN). Figure adapted from [14]

## C. StyleGAN-XL

StyleGANs are a class of GANs that have been shown to capture a disentangled representation of images. They map latent vectors  $z \in \mathcal{L}$  to a style vector  $w \in \mathcal{L}_w$ . The  $w$ -latent is then used by a generator to generate an image. The  $w$ -latent is meant to be a more disentangled representation to allow for more control and interpolation in generation. Thus,  $w$ -latents encode different independent features of the image, making them a good choice for our reconstruction task. StyleGAN-XL scales this original model to large Imagenet like datasets. This model also uses supervised training to achieve this scaling. StyleGAN-XL has two generator networks  $G_m$  and  $G_s$ .  $G_m$  learns the mapping from  $z$  to  $w$  conditioned on class labels  $c$ , while  $G_s$  learns the mapping from  $w$  to image space. Thus, these 512-dimensional  $w$ -latents are disentangled,

while also having class specific information. This along with other improvements in the architecture and training, allows StyleGAN-XL to almost perfectly represent images in its latent space. This greatly increases the upper bound on the quality of reconstructions.

StyleGAN-XL, however doesn't have an encoder, to map images from pixel space  $\mathcal{I}$  to latent space  $\mathcal{L}_w$ . This problem of having to invert GANs is sidestepped by using latent optimization techniques, where the latent vector  $z$  is optimized to minimize the difference between the generated image and the original image. Further details on this are provided in the methods section.

## Metric to evaluate reconstruction

A great advantage of reconstruction is that one can just inspect the reconstructions and compare them to the original images to get a sense of the quality of the reconstructions. However, it is very important to quantify the quality of reconstruction. A simple measure is the pixel-wise mean squared error (MSE) between the original image and the reconstructed image. However, this measure is misleading, as it doesn't take into account the perceptual similarities between the images. For example, small changes like position-shifts and rotations in the image can lead to a large MSE, even though the image looks almost the same. A better metric is SSIM (Structural Similarity Index Measure) which takes into account the structural similarity between the images.

However, none of these metrics align well with perceptual reports of image similarity. With the popularity of CNN models, it has been shown that the features in the later layers of these models are more aligned with human perception. Therefore, it is possible to use the features of these models to evaluate the quality of reconstructions [16]. We choose the VGG-16 model to extract features (Although our results show that ResNet features are better aligned with IT. We avoid using ResNet, as one of the GANs we use is trained on Resnet features, which would lead to a bias). Here we use *cosine similarity* between the features of the original and reconstructed images in the different layers of the VGG-16 model (VGG-SIM). The early layers (VGG-1 sim) give a measure of how similar the low level features (the colours and contours) are, while the later layers (VGG-5 sim) give a measure of how similar the high level features are (like the objects in the image). This metric has been used in other reconstruction studies,[17] and allows us to compare results with other studies.

## Earlier work in Reconstruction

Several previous studies have shown that it is possible to reconstruct images from neural activity in the brain. Most of these early reconstruction studies use EEG and FMRI responses [18]. Since large-scale GANs and Diffusion models have only recently been developed, these studies couldn't use pre-trained generators. Studies were primarily focussed on using other feature extraction models to create a latent space, and train small GANs/VAEs on these features. Reconstruction using neural recordings from IT [19] was also attempted, but the quality of reconstructions was poor. The quality of the reconstructions in all these studies was limited by the quality of the latent space.

The development of generative models trained on large scale datasets, opened up a lot of new possibilities



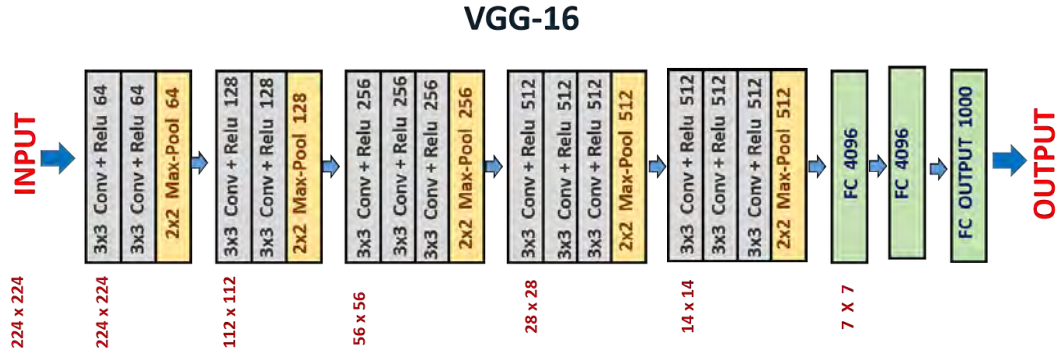


Figure 2.4: The VGG-16 model architecture. The model has 5 convolutional blocks, with each block having 2-3 convolutional layers followed by a pooling layer. We use features from the pooling layers of each block to calculate the VGG-SIM, and in other comparisons between CNNs and IT representation for computational efficiency. Figure adapted from [website](#)

for reconstruction. BigBigan, one of the early GAN models was quickly adopted for reconstruction [20]. The development of diffusion models has made high quality fMRI reconstructions possible [21]. These pre-trained models can also be used in conjunction with each other, which allows one to exploit features of different latent spaces to get very impressive results [22]. Efforts to reconstruct using Neuronal recordings from IT have also benefited from using pre-trained latent spaces. StyleGAN-XL has also been to reconstruct images from a combination of V1, V4, IT [17]. Notably this study shows the advantage of having both low level and high level features in neural recordings.

Such works however have mainly focussed on natural images. The visual system however can also effectively represent a much larger class of non-natural images that would be adversarial to deep networks (like line drawings, faces or abstract patterns). In this study, we also look at the reconstruction of out-of-natural-distribution images, which might shed light on the underlying representation of these images in the visual system.

# Experimental Design

Since the goal of this study is to reconstruct images from neural activity, it is crucial to get neural responses to a wide variety of images, which isn't possible in a single session. Therefore, we planned to record neural activity from two monkeys over 13 sessions (each). Both these monkeys had 128 electrodes implanted in the IT region. Below is the plan for the experiment.

Table 3.1: Images shown on each day

Day	Images shown
Day 0	200 common images $\times$ 32 repetitions
Day 1-10	200 common images (8 repetitions)+ 1000 unique natural images (8 repetitions)
Day 11	200 common images (12 repetitions)+ 1000 out-of-natural-distribution images (12 repetitions)
Day 12-13	200 common images (8 repetitions)+ Videos+ Natural Objects +Sleep

Table 3.2: 200 Categories of Object Images show

Type of Category	categories
Manmade Objects	Accordion, Airplane, Ax, Baby bed, Backpack, Balance beam, Banjo, Baseball, Basketball, Bathing cap, Beaker, Bench, Bicycle, Binder, Bookcase, Bow, Bow tie, Bowl, Brassiere, Bus, Can opener, Car, Cart, Cello, Chain saw, Chair, Chime, Cocktail shaker, Coffee maker, Computer keyboard, Corkscrew, Cornet, Cowboy hat, Croquet ball, Crutch, Diaper, Digital clock, Dishwasher, Display, Drum, Dumbbell, Electric fan, File, Flute, French horn, Frying pan, Golf ball, Golfcart, Guitar, Hammer, Hand blower, Harmonica, Harp, Helmet, Horizontal bar, Ipod, Ladle, Lamp, Laptop, Lipstick, Maillot, Maraca, Microphone, Microwave, Milk can, Miniskirt, Mitten, Motorcycle, Mug, Nail, Neck brace, Oboe, Pencil box, Pencil sharpener, Perfume, Piano, Ping-pong ball, Pitcher, Plastic bag, Plate rack, Pot, Power drill, Printer, Puck, Punching bag, Purse, Racket, Refrigerator, Remote control, Rubber eraser, Rugby ball, Rule, Saltshaker, Sax, Screwdriver, Ski, Snowmobile, Snowplow, Soap dispenser, Soccer ball, Sofa, Spatula, Stethoscope, Stove, Strainer, Stretcher, Sunglasses, Swimming trunks, Syringe, Table, Tape player, Tennis ball, Toaster, Toilet tissue, Traffic light, Train, Trombone, Unicycle, Vacuum, Vessel, Violin, Volleyball, Waffle iron, Washer, Water bottle, Windsor tie, Wine bottle, Tree
Animals	Ant, Antelope, Armadillo, Bear, Beaver, Bee, Bird, Butterfly, Camel, Cattle, Centipede, Cockroach, Dog, Domestic cat, Dragonfly, Elephant, Fox, Frog, Giant panda, Goldfish, Hamster, Hippopotamus, Horse, Isopod, Jellyfish, Koala, Ladybug, Lesser panda, Lion, Lizard, Lobster, Monkey, Mouse, Otter, Person, Porcupine, Rabbit, Ray, Scorpion, Seal, Sheep, Skunk, Snail, Squirrel, Starfish, Swine, Tick, Tiger, Turtle, Whale, Zebra
Food	Apple, Artichoke, Bagel, Banana, Bell pepper, Burrito, Cauliflower, Cucumber, Fig, Hamburger, Head cabbage, Hotdog, Ice lolly, Lemon, Mushroom, Orange, Pineapple, Pizza, Pomegranate, Pretzel, Strawberry

## Stimuli

In order to map out neural responses to a lot of natural images, we decided to record responses to around 10,000 images belonging to **200 object categories**. We chose the 200 categories of images, mostly from tiny imagenet which are listed above. For each of these categories we sampled 51 representative images from Imagenet, for a total of 10200 images. Since we recorded over multiple days, we show one image from each category, which we term "common images" on every day, to allow for inter-day alignment. 5 new images from each category are displayed for each of the 10 days. (1000 images per day)

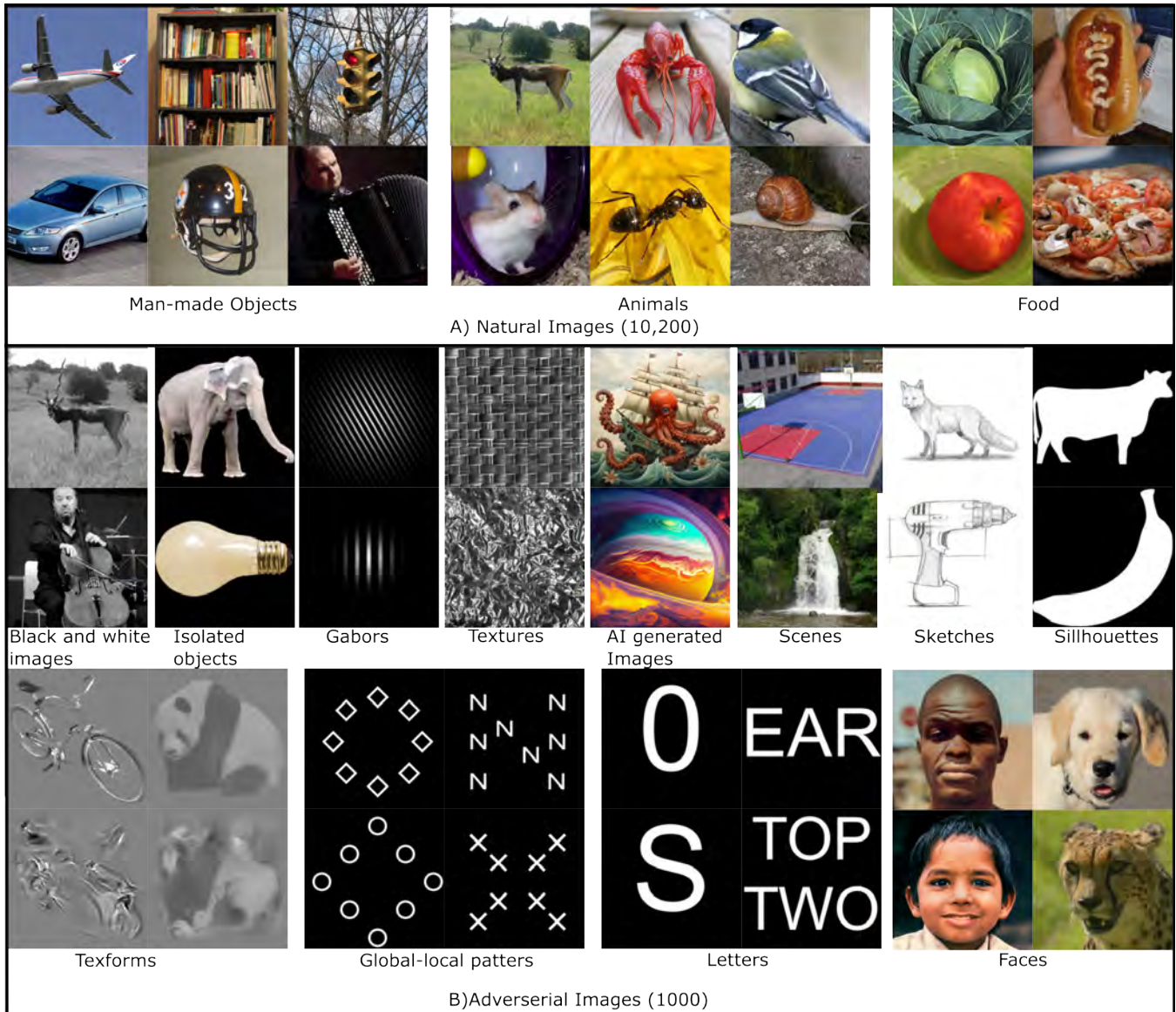


Figure 3.1: Stimuli used in the experiment. **A.** Natural images (200 object categories  $\times$  51 images). The first row has "common images", the second contains the other natural images. **B.** out-of-natural-distribution images (1000 images)

### Out-of-natural-distribution Images

To understand how the brain represents other classes of images, we also recorded responses to 1000 images that have different statistics than natural images. These are images that would in general, be adversarial to deep networks. The other classes of images that we chose were:

- Black and White images
- Isolated objects
- Gabors: Gabor filters of varying orientations and frequencies
- Textures (black and white)
- AI generated images (from [freepik.com](https://www.freepik.com))

- Scenes (from the Places dataset [23]): Images of scenes (Indoor, Outdoor manmade, Outdoor natural)
- Sketches
- Silhouettes
- Texforms: Images high and low frequencies removed [24]
- Global-Local Patterns of shapes:
- Letters, numbers and words
- Human and Animal faces

In addition, responses to videos, natural objects (not on screen) and sleep data was also recorded. However since that data has not been analyzed, it is not included in this thesis.

## Task

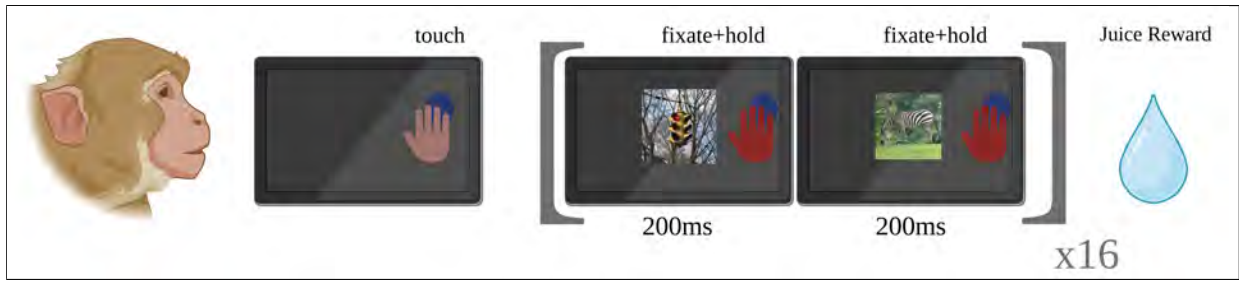


Figure 3.2: Passive fixation task. The monkey is required to touch the hold button to start the trial. Then the monkey has to fixate on images (16 images shown successively one after the other for 200ms each) while maintaining hold on the button. The monkey is rewarded for successfully maintaining fixation and holding on the image for the entire trial. Further details are in fig2

The Images(with resolution  $224 \times 224 \times 3$ ) were displayed in a random order in a passive fixation task. Each trial consisted of presenting 16 images. Image were displayed continually for 200ms one after another without any ISI. The monkey was required to fixate on the anywhere on the image. The image was 12 degrees visual angle. The monkey is rewarded for maintaining fixation anywhere on the image for the duration entire trial. We aimed to collect 8 correct repetitions each day. Therefore each, day consisted of 600 planned trials. If the monkey broke fixation, the trial was aborted and repeated at the end of the trial set. More information, about data collection can be found in the methods section 6 , and about the setup can be found in the appendix 2



# Results

Data was collected from 2 monkeys. Neural data has been analyzed for M1, and is summarized below.

## Sample Responses

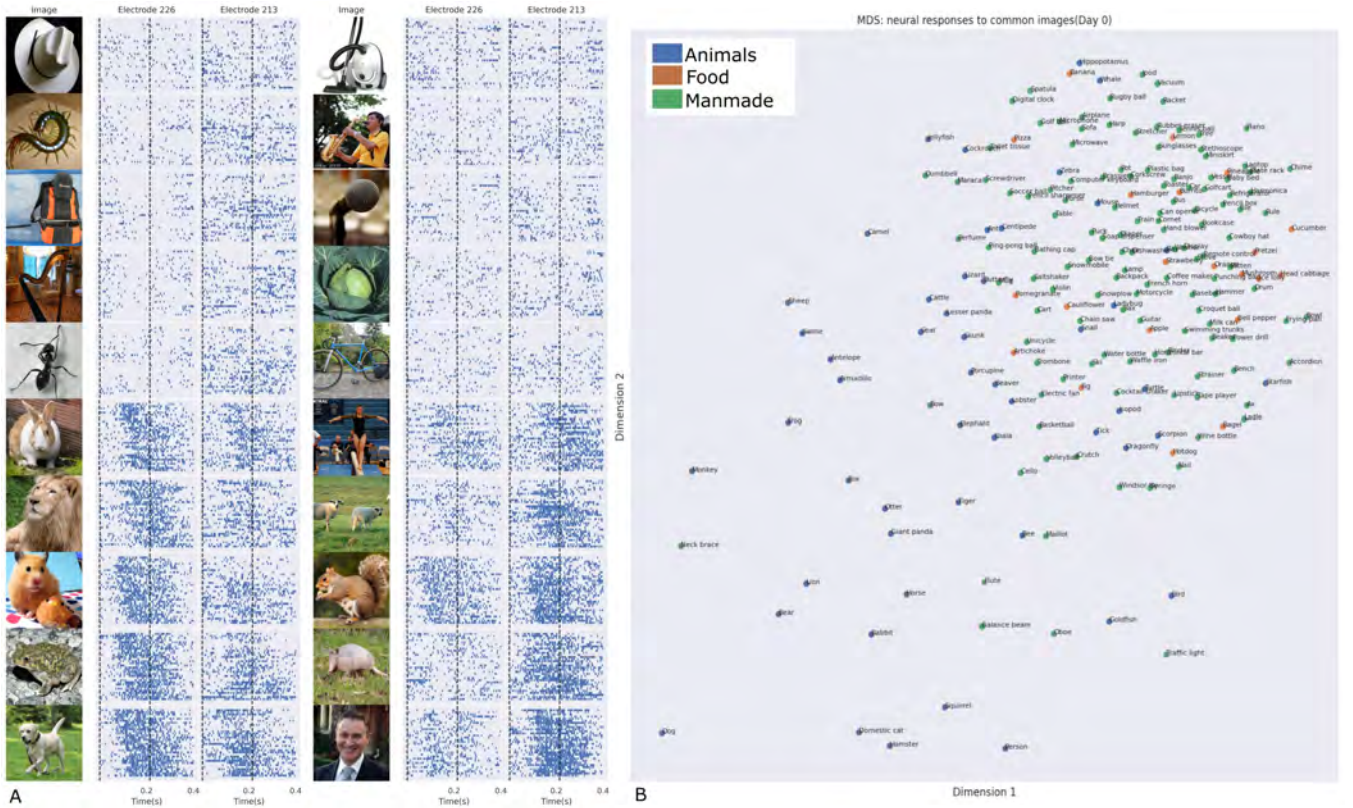


Figure 4.1: Sample responses to images in IT. **A.** Sample Raster-plots for two neurons. Each row represents a different trial for the image, with spike-times marked. The two vertical lines mark the times when stimulus is turned ON and OFF **B.** Population response to images in IT, visualized using MDS [6](#). The colour of the points indicates the category of the image (Animals, Food, Manmade Objects).

We first illustrate sample neural responses to the *common set of images on day 0*. The rasters for two neurons are plotted in [4.1A](#) (the images are top 5 images that elicit the highest responses and the the bottom 5 images that elicit the lowest response). The example electrodes show selectivity towards particular animals and poses, which alligns with our claims that IT encodes high-level visual features. To get a sense of the populations response, we performed MDS([methods 6](#)) on the population response - [4.1B](#). This also indicates the categorical nature of IT responses. Related categories , indicated by the same colour, cluster together.



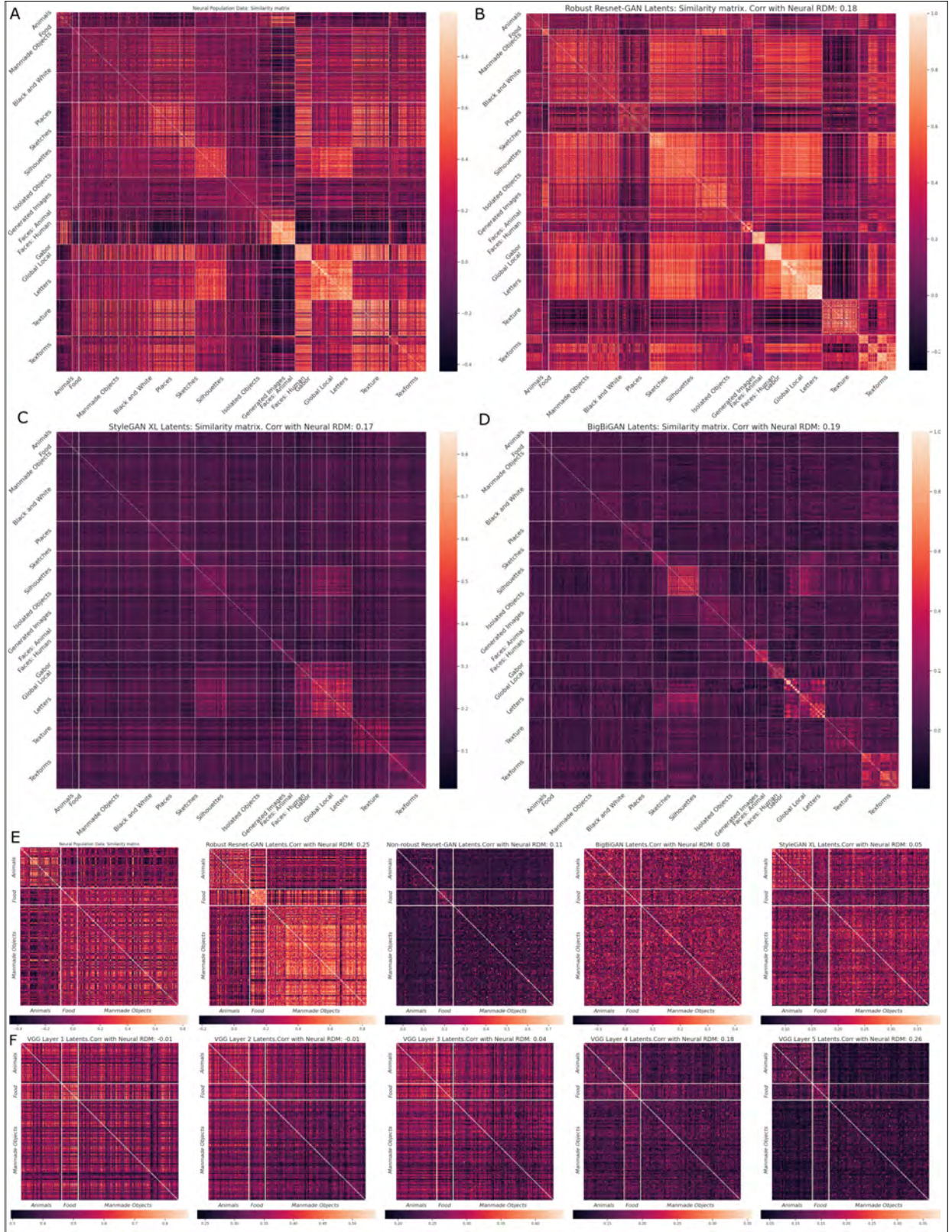


Figure 4.2: Similarity between responses to images in IT and latents of different models. White lines mark boundaries between different categories **A**. Neural Similarity matrix for natural and out-of-distribution images **B-D**. Similarity matrix for natural and out-of-distribution images for different GAN latents **E**. Similarity matrix for natural images for neural data and different GAN latents. Correlation between the pairwise distance matrices for each model with the IT responses is shown in title **F**. Same as E, but for different layers of the VGG model. (*methods 6*)

## Representation of images in IT and different models

To get a better sense of the representation encoded in the IT, we looked at the similarity between the responses to different images. Using the responses to the *common and out-of-natural-distribution images* recorded on day 11, we plotted *pairwise cosine similarity* as described in *methods 6*. From the neural similarity matrix, we see that out-of-natural-distribution classes like black and white images, isolated objects are quite similar to natural images. Faces, are extremely similar to each other and quite different from other categories. Out-of-natural-distribution categories with objects (like sketches and silhouettes) also demonstrate similarity with natural images. Other out-of-natural-distribution categories like texforms, textures, global-local patterns and symbols are quite different from natural images. (There is a high degree of similarity in within categories, but the similarity with other categories is not low, like it is for faces. This likely means that such images might be poorly represented/ inseparable in the IT). We also quantified the similarity between these representations by calculating the correlation between the pairwise distance matrices. However, this quantification isn't very informative, because of the wide range of images in the out-of-natural-distribution set, that skew the correlation.

Therefore, we have also plotted the similarity matrix only for natural images (neural data, GAN latents and different layers of the VGG model. Category specific-responses are also evident (Animals and non-animals have high-dissimilarity between each other). This is where we first see the similarity between CNN features and IT responses. The similarity to IT responses is highest for the Robust ResNet model and VGG last layer features. Early CNN layers (encoding low level features), are extremely dissimilar from IT responses (correlation of  $\approx 0$  between pairwise distances). It's also interesting to note that the robust ResNet model significantly outperforms the normal ResNet model, indicating the importance of adversarial training in capturing the IT representation.

## Inter-day alignment

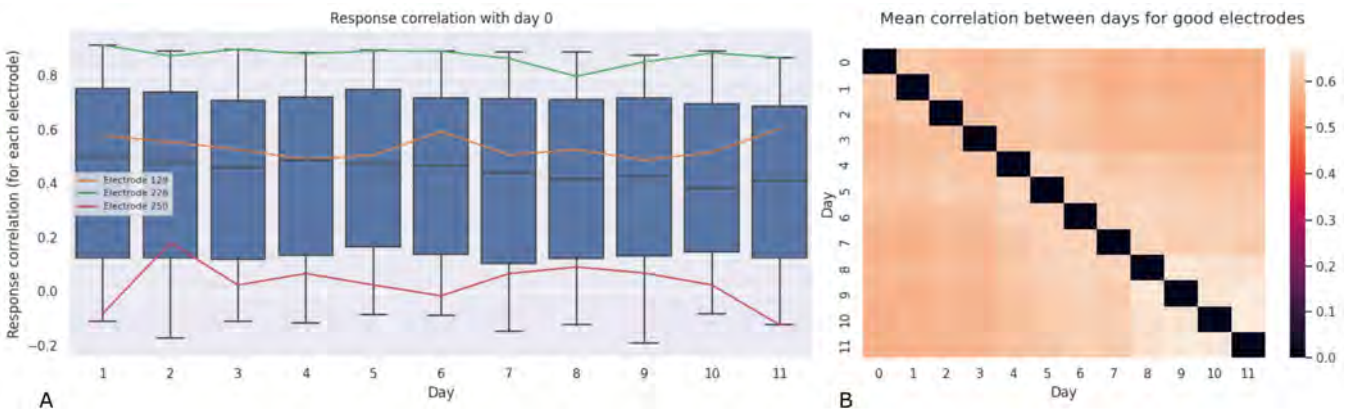


Figure 4.3: Comparison of responses on different days **A**. Distribution of correlation of responses to common images (for each IT electrode) on day 1-11 with responses on day 0. The central bar marks the median, and the thick blue bars mark the quartiles, and the error bars mark 99 and 1 percentile in the distribution of correlations. This is also illustrated for three sample electrodes using lineplots. **B**. Heatmap of the mean correlations to common images (for each good IT electrode) between different days. Good electrodes are selected using the procedure described in *methods 6*.



Before proceeding to use the entirety of the data, we needed to ensure that the responses are consistent across days. We had shown the common images shown on each day for this purpose of aligning the responses. We calculated the correlation between the responses to the common images **on day 0** with the responses to the same images on the **other days** in fig4.3. As seen in the figure, there isn't a significant drop in the mean correlation over days. We have also overlayed the correlation for 3 sample electrodes (with good, average and bad inter-day correlation), which illustrates that for a given electrode the correlation between days is consistent. The mean correlation between days for good electrodes is also shown on a heatmap, further confirming the consistency of the responses. However, correlations don't capture scale and offset differences between days (which are likely to change, based on the threshold of the MUA, which is done on a per-day basis). Therefore, we z-scored the responses on each day (for each electrode) using the mean and standard deviation of the responses to the common images on that day. This also ensured that the responses are comparable across days and neurons.

## Prediction of Neural response using different models

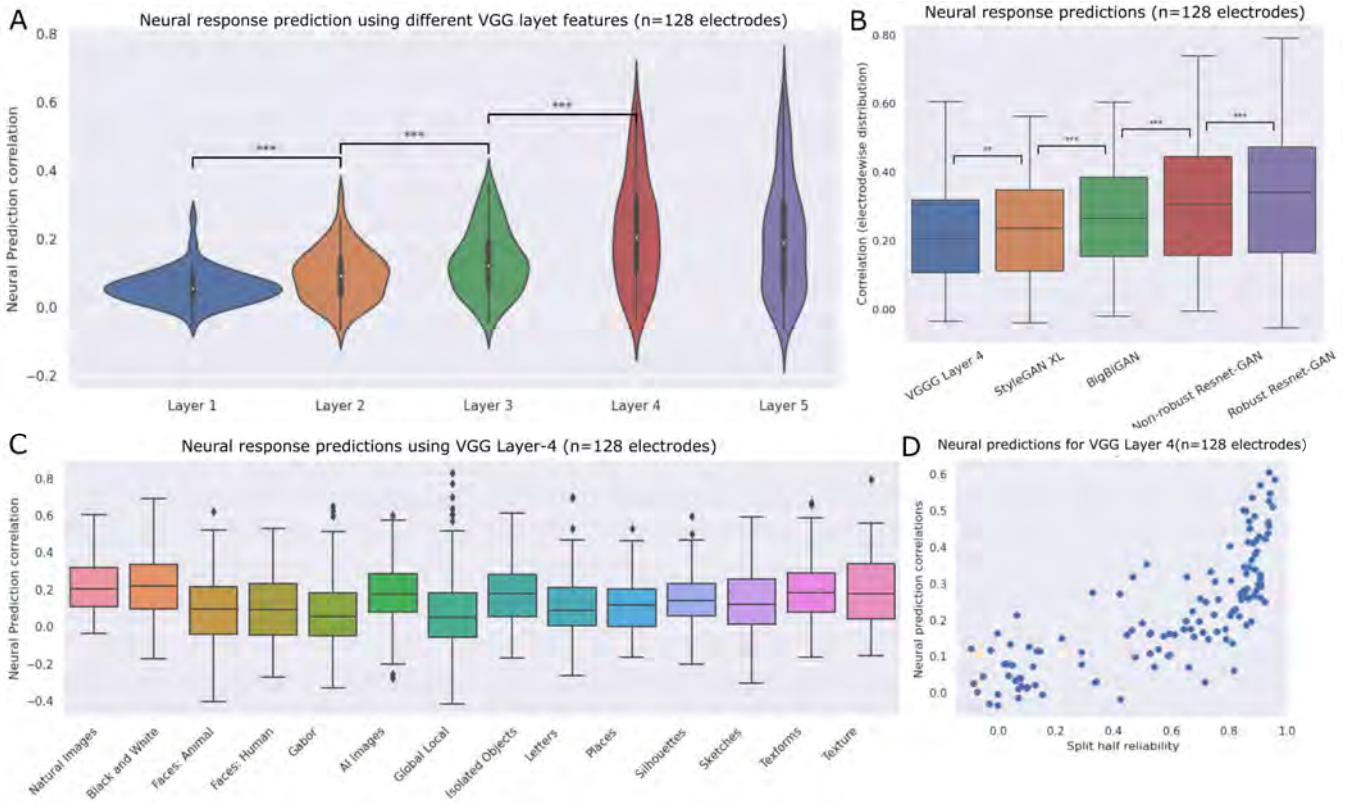


Figure 4.4: Prediction of neural responses using different GAN Latents and VGG features (*methods 6*). **A.** Correlation coefficients between predicted and actual neural responses (for each electrode) from different VGG layers (natural images). The distribution of these correlation coefficients across neurons is plotted on the violin plot **B.** Correlation coefficients between predicted and actual neural responses from different GAN latents (natural images) on a box plot, which indicates the quartiles and the 99 percentile intervals. We also performed a paired t-test between these correlations, and indicate the significant differences between neighbours on the plot. **C.** Test correlation coefficients between predicted and actual neural responses from different GAN latents for out-of-natural-distribution images. **D.** The correlation of neural response prediction against the split-half reliability. Each point on the scatterplot is an electrode (*methods 6*)



To further understand the representation of images in IT neurons and these models, we predicted the neural responses using the latents of these models (more details in methods 6). Briefly, we trained L2-regularized linear regression models to predict each neuron’s responses to different images from different latents. Later we tested the model on a held-out set of images. The correlation between the model predictions and actual neural responses is shown in fig4.4. We observed that IT neural responses are best explained by later VGG layers( 4.4A), which justifies our use of GAN models with similar low-dimensional and high-level representations. A key point to note is that, the variability in the correlation coefficients is not due to the poor predictive power of the VGG features, but is an artifact of the response quality for the neurons. This is illustrated for one of the VGG layers in 4.4D; the electrodes whose data is well predicted are also the electrodes which have a high split-half reliability (further explained in methods 6). All the GAN latents are able to predict neural activity better than the VGG features. The robust ResNet model has the highest predictive power, slightly outperforming the non-robust model. Surprisingly, StyleGAN latents are not as good at predicting neural activity (despite it being the best generative model). This is likely because StyleGAN latents also encode low-level features, which would make the representation they hold non-linearly related to the neural encoding for these images. Using the same model (trained on natural images), we also looked at how well VGG features can predict neural activity for out-of-natural-distribution images(4.4C). The predictions are worse than the natural images. This is because of two main reasons. Firstly, the out-of-natural-distribution images have different statistics than natural images, and any model trained exclusively on natural images will be overfit to the natural images. Secondly, many of the out-of-natural-distribution images are poorly represented in the CNNs, which means they don’t carry information about these images. The model generalizes quite well to categories like isolated images and black and white images that are similar to natural images. Face responses are poorly predicted by the model, which is likely because object detection models are not able to encode faces well. Other out-of-natural-distribution categories like texforms, textures, global-local patterns and symbols are also poorly predicted.

## Reconstruction of Natural Images

After looking at the similarities in representation between IT and different models, we proceeded to predict the latent vectors from neural data. The correlation coefficients between the predicted and actual latents for each dimension are shown in 4.6B using a violin plot. This violin plot illustrates the distribution of correlations in predicting the latents.

We then passed these predicted latents to the generator of the GAN model to get the reconstructed images (more details in *methods 6*). Some reconstructions are visualized in fig3 (The reconstruction for non-robust ResNet model are not shown because they are quite similar to robust ResNet). The reconstructions look quite good, and are generally able to capture the main object in the image. From visual inspection, StyleGAN-XL seems to give the best reconstructions. ResNet reconstructions are also good, but are limited by the resolution of the image, and the images don’t look as natural. BigBiGan reconstructions are often blurry/ repetitive (for example, many animals are reconstructed as cats/wolfs). More reconstructions are shown in the appendix 3.

To quantify these reconstructions we used two main metrics. The first is the VGG-SIM between the original and reconstructed images. The mean VGG-SIM for the reconstructions is shown in 4.6A. The

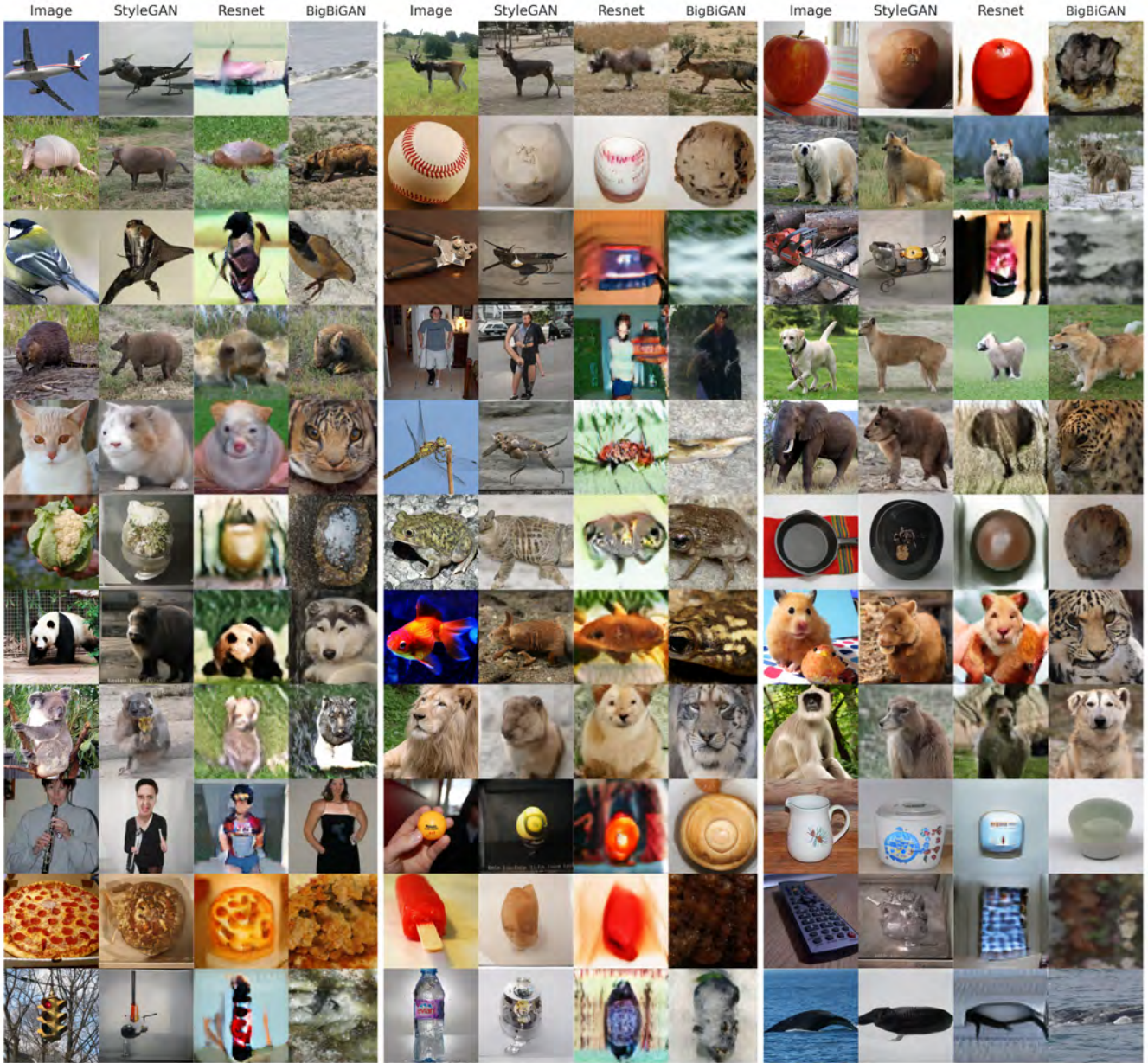


Figure 4.5: Reconstruction of natural images using different three different GAN latents for 33 sample common images. In each set of 4, the first column is the original image, followed by reconstruction using predicted latents (from neural data) from 3 different GAN models (StyleGAN-XL, ResNet GAN, BigBiGAN). The reconstruction pipeline is summarised in 6.1, and more reconstructions are in the appendix I

VGG-SIM for the reconstructions using the actual latents is also shown using dotted lines (i.e the ceiling performance). This ceiling is much higher for StyleGAN-XL compared to other models. VGG 1-4 SIM are quite similar across models, but VGG-5 Similarity establishes a clear gradient between the models (which is what is most aligned with perception). StyleGAN-XL outperforms the other models. The robust-ResNet model is second. The BigBiGAN model performs the worst. Between the two ResNet model, the robust model outperforms the non-robust model across all layers. Note that not all statistical comparisons are indicated.

The second metric is the top-5 accuracy of the reconstructions. This is found by taking the top-5 category predictions (from VGG model), and matching it against the top prediction of the original image. The orange



bars indicate the same top-5 accuracy for the latent reconstructions. The accuracy is quite low for all models ( $< 0.1$ ), and is around 10 times lower than the accuracy of the latent reconstruction. However, this isn't unexpected because the reconstructions generally fail at capturing the fine details of the image and the categorical divisions in these image classifiers are quite strict (for example there are around 40 different dog breeds). Despite this, the top-5 accuracy for StyleGAN-XL is much higher than chance. Surprisingly, the non-robust ResNet model has a higher top-5 accuracy than the robust model. BigBiGAN, which has the worst VGG-SIM, also has the worst top-5 accuracy. The values obtained are in a similar range as the IT-reconstruction model in [17], which further validates our results.

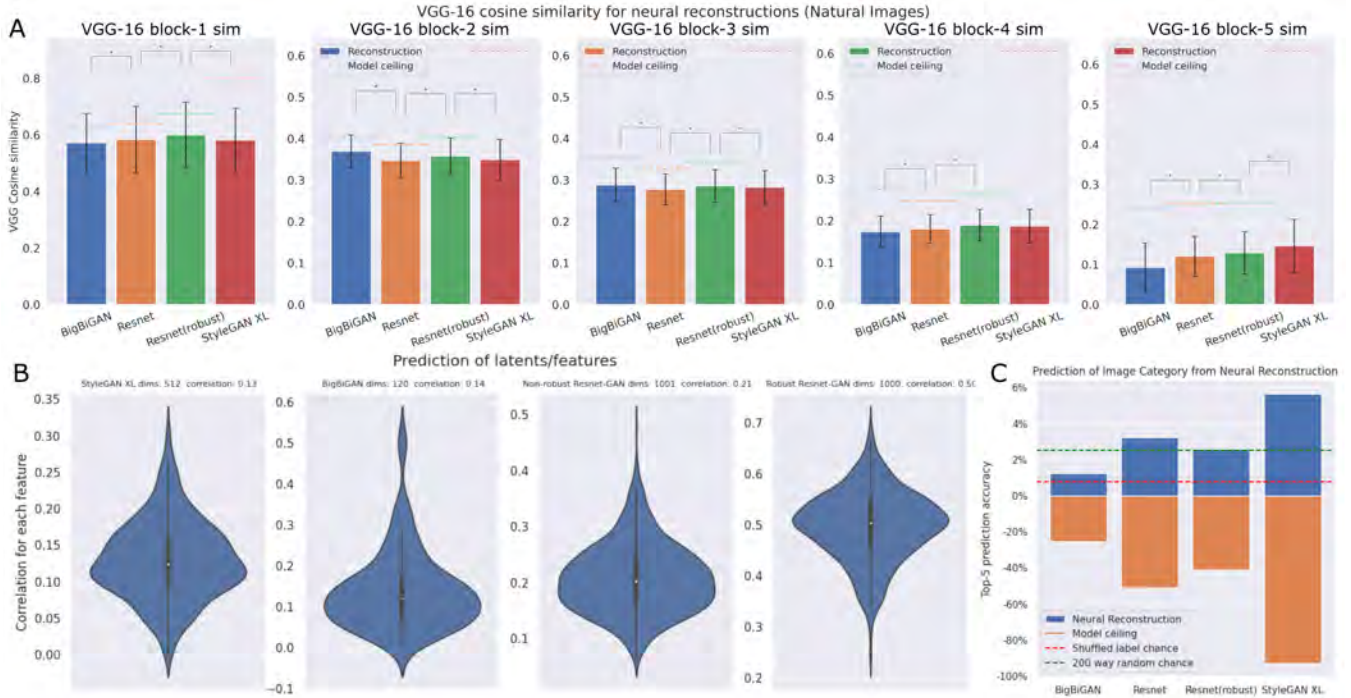


Figure 4.6: Quantification of quality of reconstruction. **A.** Mean VGG-sim between the original and reconstructed images for test images (cosine similarity between VGG layer features). The dotted lines above each bar represents the same quantification for the reconstruction of the actual latent vector =  $G(E(Img))$ . This establishes the ceiling performance using the GAN. **B.** Distribution of correlation coefficients between the predicted (from neural data) and actual latents for each latent dimension. **C.** Top-5 Accuracy of the reconstruction. This is found by taking the top-5 category predictions (from VGG model), and matching it against the top prediction of the original image. The orange bars indicate the same top-5 accuracy for the latent reconstructions. Notice that they are on different scales. The dotted lines indicate chance accuracies, Green:  $5/200$  for a 200 way classification; Red: top-5 accuracy for shuffled data and labels. (Further details are in *methods 6*)

### Category specific Reconstruction

Even though the reconstructions are the best for StyleGAN-XL, the quality was lower than what the model is capable of (indicated by the dotted lines above the bars in 4.6A). Since we have a large corpus of images, we can afford to train models for smaller subsets of images. And if the images only occupy a small region of the entire latent space, the reconstructions ought to be better (normally the decoders are swayed by large differences between categories, so they fail to capture the finer variations in each category).

We divided the 200 natural image categories into three classes: food (21/200), animals (51/200) and manmade-objects (128), and trained models independently for each of them. We have visualized some of these recon-

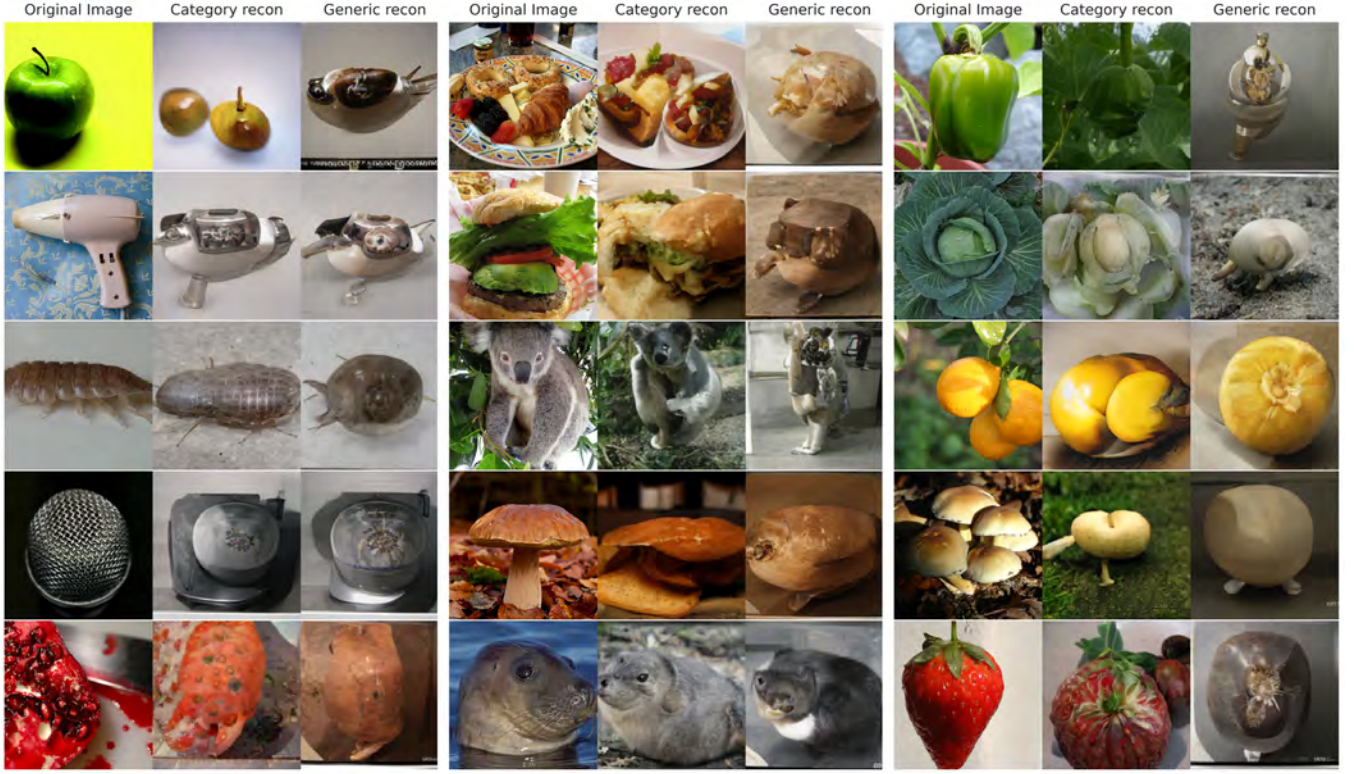


Figure 4.7: Reconstruction of category-specific StyleGAN-XL for 15 sample images. In each set of three, the columns are 1.Original image, 2.Reconstruction using a category-specific model and 3.Reconstruction using a general model.

structions in fig4.7, against the reconstruction using a general model. We see that these reconstructions capture much more detail. Infact, the improvement is the most for smallest category- Food ( $\approx 1000$  images out of 10,000). However, this is not unconditional generation (direct information about the image (the category) is used in addition to neural data for the reconstruction), so we don't quantify the quality of these reconstructions. There was no significant improvement for BigBiGAN and ResNet models for category-specific models, likely because models themselves are not that good. In the discussion we talk about extending these category-specific models to be conditioned only on the neural data.

## Reconstruction of Out-of-natural-distribution Images

Next, we looked at how well reconstruction models trained on natural images generalize to out-of-natural-distribution images. Using the same models trained on natural images, we predicted the latents for the out-of-natural-distribution images, and passed them to the generator. We quantify the VGG-5 Sim for the reconstructions in 4.8A. This is not the layer that best aligns with IT, rather it is the layer that best aligns with perception. Since several of these images are different from natural images, VGG features might not encode these images well. Thus VGG-SIM might not be the best metric to quantify the quality of these reconstructions. Therefore, we also calculated the SSIM between the original and reconstructed images in 4.8B. A very interesting observation is that natural-image like images are usually better predicted by StyleGAN-XL. But the ResNetGAN outperforms other models for classes like global-local patterns and textures (StyleGAN and BigBigan are quite bad at these).

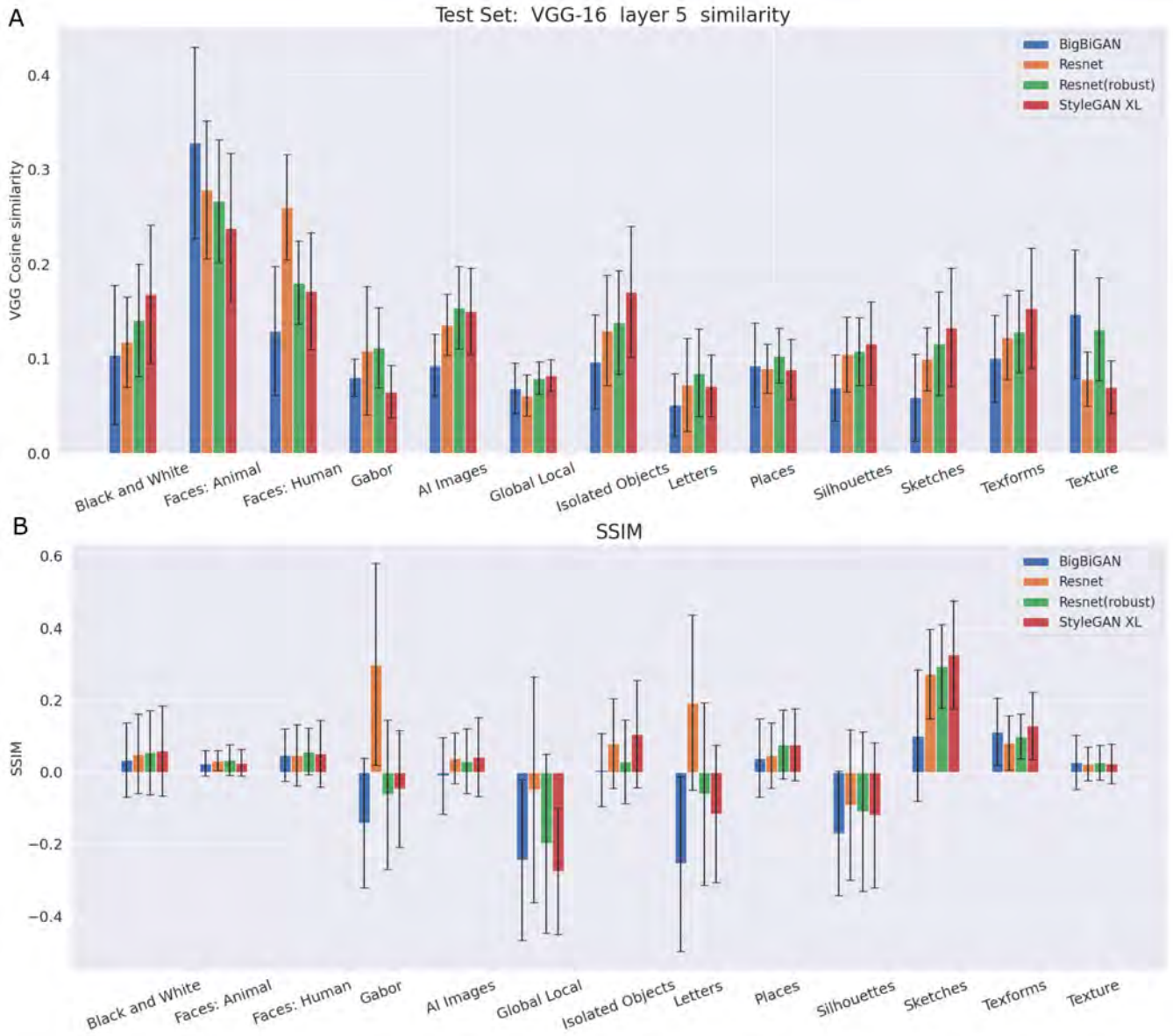


Figure 4.8: Reconstruction of out-of-natural-distribution classes using different GAN latents. For each class, we plot the mean and SEM of the similarity measure (across all images of that class) **A.** VGG-5 SIM for reconstructions (This is not the layer that best aligns with IT, rather it is the layer that best aligns with perception). **B.** SSIM for reconstructions

However, both these metrics fail to capture the quality of reconstruction that well. The reconstructions from the GANs are shown in figures 4.9-4.15. Black and white images, faces and isolated objects are faithfully reconstructed by all models. StyleGAN reconstructs both sketches and silhouettes quite well (the reconstructions are in the style of natural images, because the neural to latent decoder has only trained on natural image-latents as its range). The texforms are also reconstructed only in Stylegan-XL. The ResNet model on the other hand does quite faithful reconstructions of global local patterns, letters, textures, AI images and Gabors. This diversity in reconstructions indicate that with some fine-tuning of the models, it might be possible to get good reconstructions these out-of-natural-distribution images. The GAN models also to represent some such stimuli really well (fig I). What is more interesting is, for Gabor's and Global-local patterns, the neural reconstructions capture more information than the model reconstructions. The



Gabor orientations reconstructions (fig 4.11) are always in the same direction (which is not the case for latent reconstructions). Even in global-local patterns, the IT reconstructions are objects resembling the global/local shape (4.9), whereas the latents don't carry any such information. The appendix I has some direct latent reconstructions.

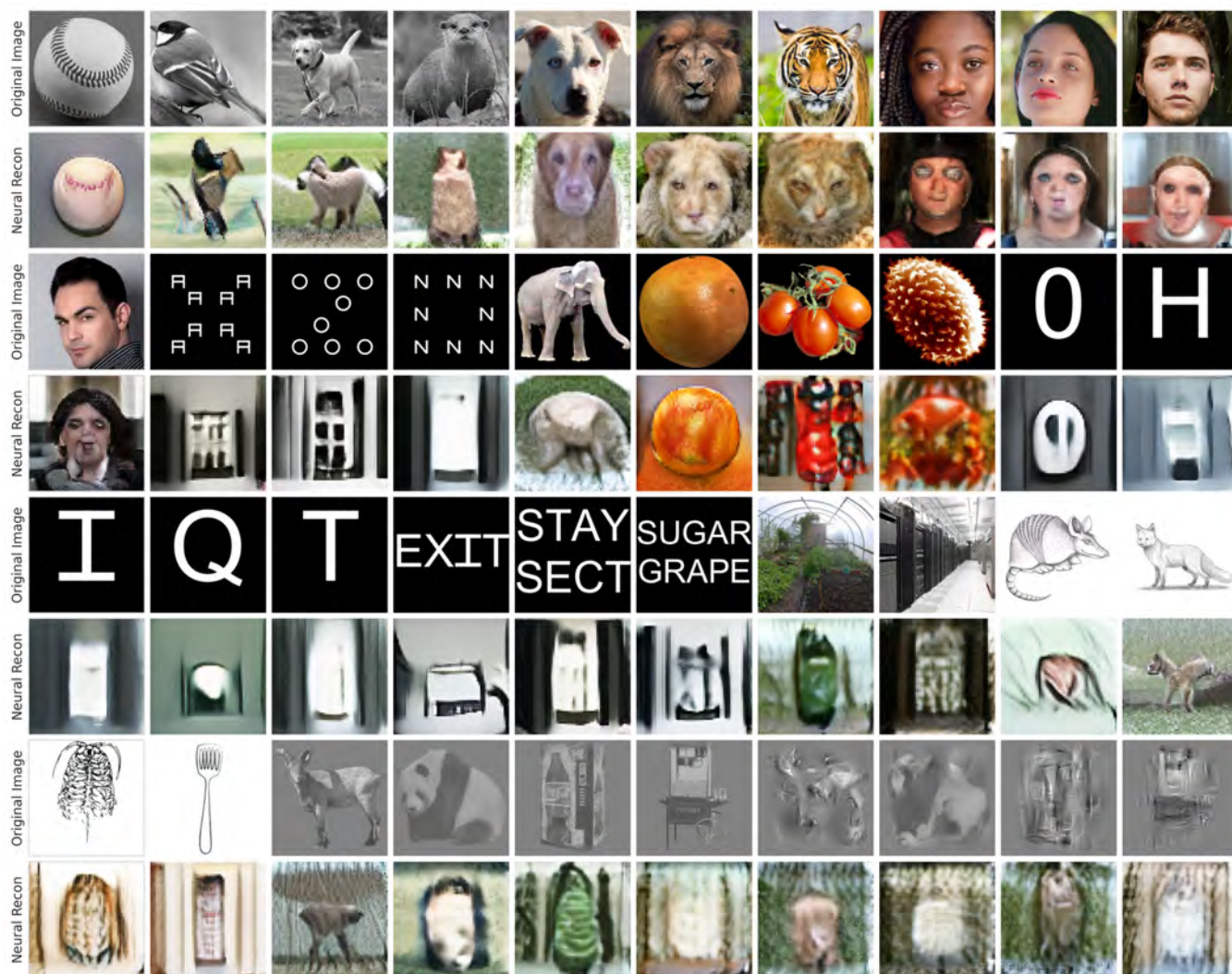


Figure 4.9: Reconstruction of out-of-natural-distribution images from neural data using Robust ResNet-GAN. Alternate rows represent the original image and the neural reconstruction using models trained on natural images. In each class, the reconstructions are predictable variations of the original images. To compare, the reconstruction from the latents themselves is in appendix I1/5

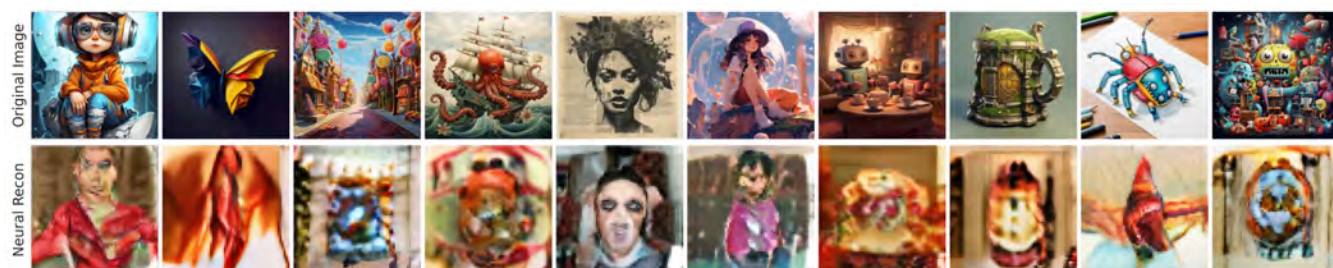


Figure 4.10: Reconstruction of AI generated images from neural data using Robust ResNet-GAN.2/5

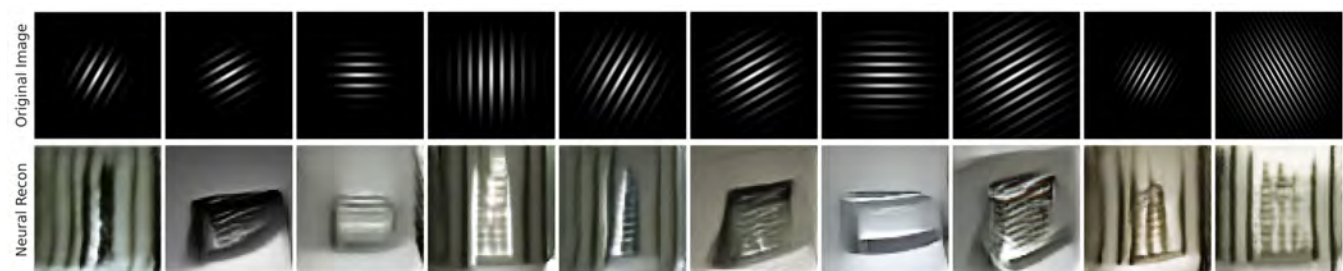


Figure 4.11: Reconstruction of Gabor images neural data using from Robust ResNet-GAN.3/5



Figure 4.12: Reconstruction of Silhouette images neural data using from Robust resnet-GAN.4/5



Figure 4.13: Reconstruction of Texture images from neural data using Robust ResNet-GAN (converted to grayscale).5/5



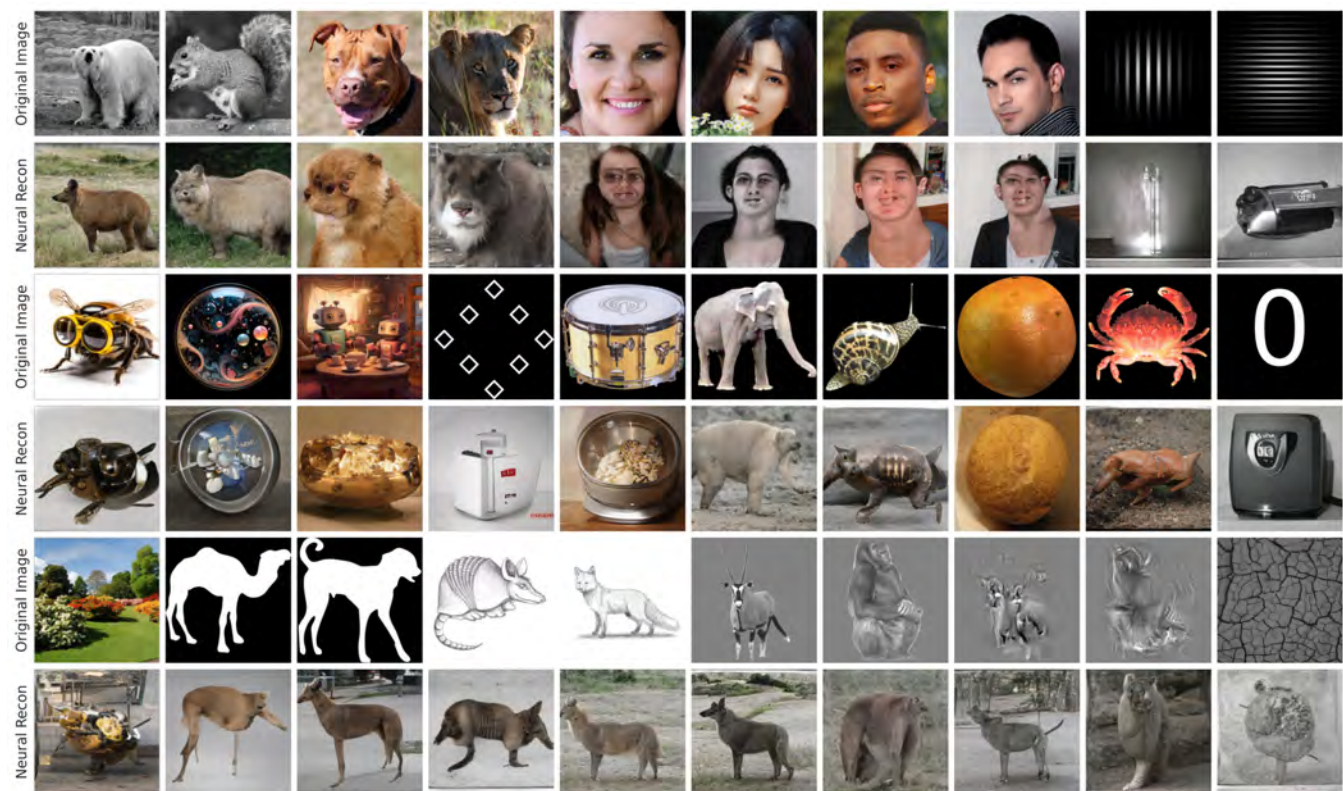


Figure 4.14: Reconstruction of out-of-natural-distribution images from neural data using StyleGAN-XL. Several reconstructions (from categories like natural images) are quite good. Texforms with high and low frequencies removed are also reconstructed quite well.

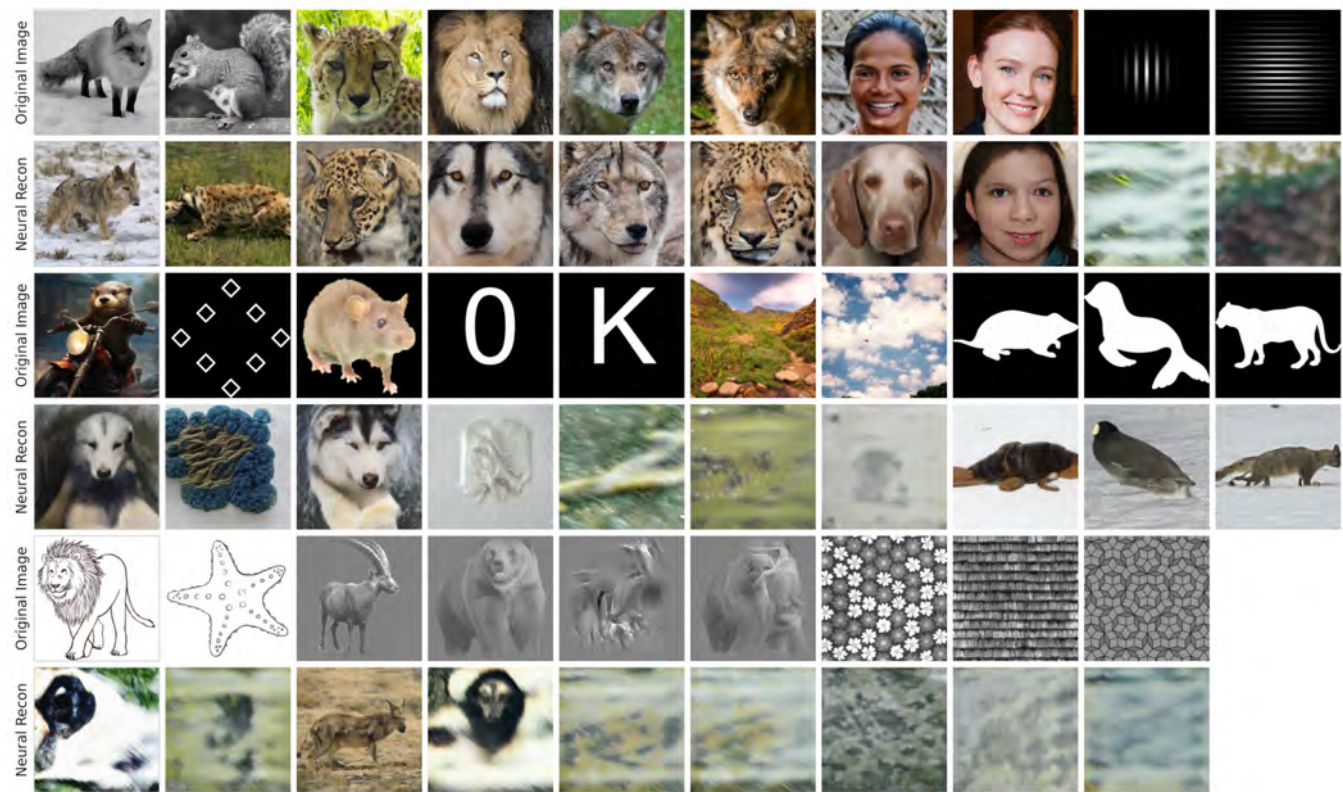


Figure 4.15: Reconstruction of out-of-natural-distribution images from BigBiGAN. Reconstruction quality is quite poor for most images.



# Discussion

In this study, we have shown that it is possible to reconstruct images from neural activity in the IT region of the brain. We show that IT neurons have a categorical responses and code for high-level visual features. We use 3 different GANs models as the backbone for this purpose and achieve fairly impressive reconstructions which are usually able to capture the object in the image.

We show that quality of reconstructions depends on various properties of the latent space. In neural response prediction, StyleGAN-XL performs the worst. Even the correlation coefficients in predicting latents (from neural data) is quite low. This implies that the *latent space of StyleGAN-XL is quite dissimilar/ non-linearly related to IT activity*. However, because the model encodes an untangled representation, these small correlations all add up to give good reconstructions, that outperform all other models. BigBiGAN, being the GAN with the lowest-dimensional latent space, is pretty bad at reconstructions. The ResNet model is the best at predicting neural responses. It's latents are also well predicted by neural data, giving quite good reconstructions. This means that the *latent space of IT is quite similar to the ResNet representation*. However since the model has low-resolution images, the reconstructions don't look the best. The ResNet model also usually captures low-level details as well (ResNet is better than StyleGAN-XL in VGG-SIM 2-3 fig4.6). The robust-model is usually better than the non-robust model (not always), which is likely because the adversarial training helps in capturing the IT representation. Nevertheless, the emergence of better alignment to IT responses as a result of adversarial training is an interesting result

We also study use these models trained on natural images to reconstruct out-of-natural-distribution images. The reconstructions are quite good for natural-image like images, especially in StyleGAN XL. We theorize that this is just a consequence of it being the best reconstruction model for natural images. However, the reconstructions are bad for image classes from natural images. The ResNet model is the best at reconstructing these images, even though CNNs themselves represent such images poorly. As we had reasoned previously, the ResNet latent space is highly linearly related to IT activity. This allows the trained linear models to still generalize to out-of-natural-distribution images and reconstruct them like natural images. The fact that this is possible, indicates that the **IT representation encodes these out-of-natural-distribution images in a similar way to natural images**. This is quite interesting, because it indicates that the *IT representation is quite robust to changes in the image statistics*. The IT can thus effectively encode these non-natural images in a manner similar to natural images (based on the fact that linear decoder trained on natural images generalizes to images with other statistics). Infact in several non-natural image classes, the IT reconstructions capture more details than the model itself.

StyleGAN-XL is an excellent reconstruction model (as indicated by the ceilings in fig6). However, it has the worst neural predictive power which we attribute to the non-linear relationship between the latent space and IT activity. Since we have a lot of training images, we grouped similar categories together and trained models only on that subset. These model better capture the local representations (within particular clus-

ters of similar images) than the model trained on all images (which mainly learn the differences between these clusters, but can't capture details) as shown in the high quality reconstructions in fig 4.7 (there was no significant improvement for BigBiGAN and ResNet models for category-specific models). This could be combined with a neural category classifier (predicting category of the image using neural data) to have full reconstructions. This would be equivalent to learning a piece-wise linear approximation of the mapping between neural space  $\mathcal{N}$  and GAN latent space  $\mathcal{L}$

## Future Work

The category specific models captured the highest detail in the food category where, all the images are visually similar (fruits or fast food). This illustrates the fact that we could train models for smaller numbers of categories and capture even more detail. However the model we have used *explicitly uses the category information*. We could possibly train a model that is conditioned only on the neural data. This can be done by training a classifier to *predict the category of the images from the neural data*. Then the overtrained category-specific models give us high-quality reconstructions. Similarly, we can also fine-tune class-specific models for different out-of-natural-distribution classes, which would allow for better such reconstructions. One could probably learn the category groupings from unsupervised clustering in the neural space. (for example all 4 legged animals are quite similar visually, and are close in neural spaces as shown from the RDMS fig 4.2 ) Such unsupervised/self-supervised approaches have been explored in some works like [25] and [26] and perform quite well. This is definitely something worth exploring, and our result has shown a proof of concept.

We have also shown that different models have different advantages. A natural question is if we can combine these models to utilize the best properties of all latent spaces. One possibility is that we could project all the reconstructions to a common space, and then combine them (eg: a median). Another interesting approach is to predict the low-level structure and high level features separately using a diffusion model (like [22]). This would allow us to capture the low-level details of the image, and then use the high-level features (like the category) as the text prompt/context to transform the image.

As mentioned before, we have also collected data for videos and natural objects (Where the monkey is free to move and interact with these objects). It would certainly be interesting to reconstruct visual activity, and capture the invariances that generalize across these different types of stimuli. In addition, analysis for M2 is also pending.

# Methods

## Data Collection

For M1, we recorded the above data over 13 sessions. On day 0 we recorded 32 repetitions to the common images. For day 1-10, we recorded 8 repetitions of the (1000 new natural + 200 common) except day 7, when 7 reps were collected. On day 11 we recorded 9 repetitions to the out-of-natural-distribution images and common images.

The neural activity was recorded from 128 electrodes in the IT region of the brain using FMA arrays. Broadband neural by the means of a wireless logger. Further details about the setup can be found in [27] and figure 2.

## Data Pre-Processing

### MUA Extraction

Multi-Unit-Activity spike-times are extracted from the raw data (recorded at 25kHz) using a custom in-lab algorithm (developed by Georgin). In brief, the raw data is first filtered between 250-3000Hz, using an anticausal zero-phase filter. The data is then thresholded at 5 times the deviation (calculated from the difference between the 75th and median value of the data).

### Firing rate calculation

The MUA is then binned into 60ms bins after aligning the spikes to the onset of the image. This data is then averaged across all repetitions of the same image to get the average response of the neuron to an image. For all relevant neural analysis other than reconstruction, we use the firing rate 120-240ms after the onset of the image. This is because the responses are stable in this time window, and the responses are not too early to be affected by the previous image. Further we z-score the responses for each neuron based on the mean and standard deviation of the responses to the common images on that day. This ensures that the responses are comparable across days, as well as across neurons.

## Neural data analysis

## Response Reliability

To quantify the reliability of the neural data, we calculate the split-half correlation of the responses to the common images. We split the responses to the common images into two halves (of repetitions), and calculate the correlation between the responses to the two halves. This is done for each electrode.

## Classification Of Good Neurons

Electrodes with reliability  $\geq 0.15$  for all 12 days. This criteria is met for 68/128 electrodes.

## Visualisation of Population Response

To visualise the population responses, we first select the responses to all good neurons, PCA (Principal Component Analysis) it down to 30 dimensions, followed by an MDS (Multi-Dimensional-Scaling) to 2 dimensions.

## Similarity between responses

To categorize the similarity between neural responses, we take the cosine similarity between the responses to any two images (cosine similarity is usually better captures selectivity patterns than Mean-squared error). We also calculate the correlation between the pairwise distance matrices for each model with the IT responses. For this we flatten the *non-diagonal* elements of the distance matrix, and calculate the correlation between the two vectors.

## Prediction of Neural activity

To predict neural activity from the latents of the models, we use L2-regularized linear regression models (to avoid overfitting). Therefore, we have 128 (number of electrodes) linear decoders, predicting the neural responses to different images. When the latent vector has a dimensionality greater than 20,000 (in the case of the initial layers of VGG which have dimensionality  $H \times W \times N_{features}$  before being flattened), we downsampled each feature image with the minimum factor such that  $H_{new} \times W_{new} \times N_{features} \leq 20,000$ . We train the models on 9000 natural images, and then test on the remaining 1000 natural images and out-of-natural-distribution images.

## Reconstruction Pipeline

An overview of the reconstruction pipeline is shown in fig 6.1. Briefly, we first use linear decoders to predict the latents of the images from the neural data. We then use these predicted latents to generate images using the generator of the GAN model.

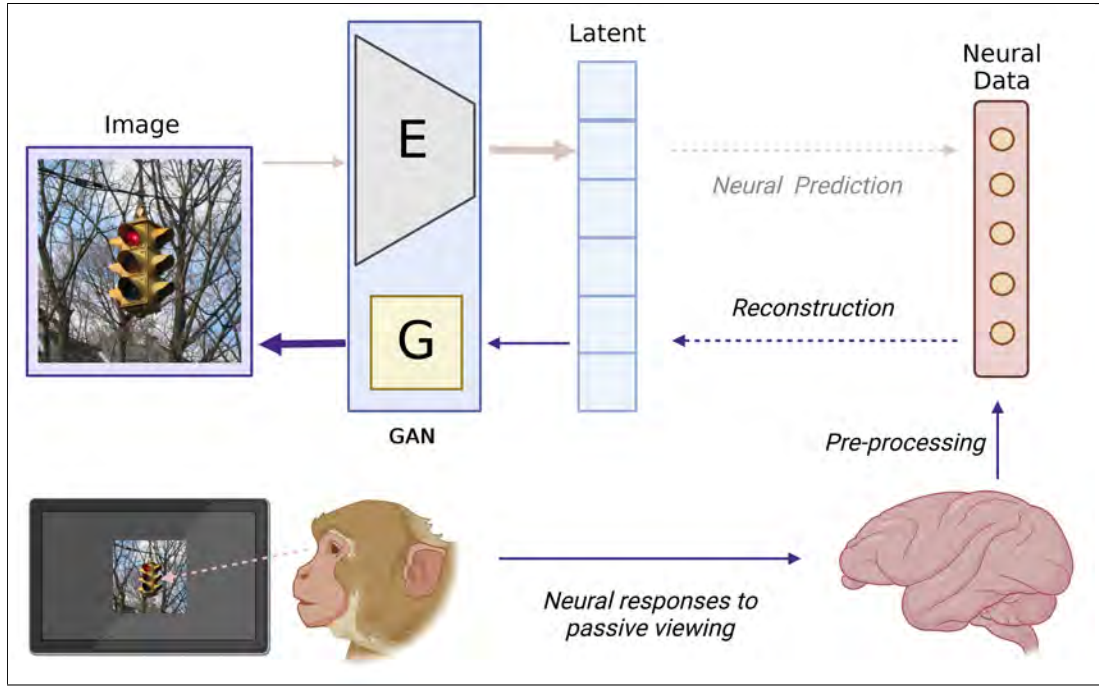


Figure 6.1: Reconstruction Pipeline

### Pre-processing of Neural data

After z-scoring the responses, as described in the previous section, we pool responses from all IT electrodes and all time bins from 60ms-360ms after the onset of the image (we pool all timebins to get as much information as possible). However, responses in different time bins, and from different electrodes are likely to be correlated, which can lead to overfitting. Therefore, we perform PCA on the responses to reduce the dimensionality of the data. We use the first 150 principal components for Reconstruction.

### Obtaining Latents for the images

In order to get the latent representation of the images for VGG16, BigBiGAN and ResNet-GAN, we perform the necessary preprocessing and then pass the image through the encoder of the model. However, StyleGAN-XL does not have an encoder to map the images.

### Latent Optimization in StyleGAN-XL

To obtain the  $w$  – latents for the images in StyleGAN-XL, we use latent optimization. The logic behind this is that we select a random latent vector and generate an image. We then look at the difference between the original and reconstructed image, and perturb the latent vector in the direction that would make the latent reconstruction closer to the image.

To be more specific, we first guess a good latent vector. This is done by first predicting the category of the image using a visual transformer model (Distilled data-efficient Image Transformer (DeiT) [28]). We then use random  $z$  – latents along with category to generate different  $w$  – latents. We initialize an

ADAM optimizer with cosine-ramped learning rates with the average of these latents. The loss function for this optimizer is the VGG-16 LPIPS [16] loss between the original image, and the reconstructed image ( $LPIPS(Image, G(w))$ ). We then perform 1000 iterations of this optimization to obtain the latent vector. Code for the same was mostly borrowed from the original paper [29].

### Linear Decoding Models

The next step in the reconstruction pipeline is mapping neural data to latents. We predict each dimension of the latent vector using an L2-regularized linear regression model, with  $\alpha = 0.01$  to avoid overfitting. Therefore, we have  $d$  linear decoders (dimensionality of latent vector) each of which predict 1 value from the 150-dimensional preprocessed neural data. We then combine the results from each decoder to get the latent vector for an image. We use 9000 natural images as the training data, and hold out the remaining 1000 images, common images and out-of-natural-distribution images for testing.

### Generating images from latents

This step is relatively straightforward. We use the predicted latent vectors and pass them through the generator of the respective GANs.

### Category specific models

To get better reconstructions, we train models on smaller subsets of the data. We divide the natural images into 3 categories: Manmade objects(128 categories), Animals(51 categories) and Food(21 categories). We then train models on these subsets of the data. The models are trained using the same pipeline as above, but only on the respective subset of the data. We then use these models to reconstruct images from the respective categories.

## Quantification of Reconstruction

### VGG-SIM

The VGG-SIM is calculated as the cosine similarity between the features of the original and reconstructed images in the different layers of the VGG-16 model. We use the features from the pooling layers of each block to calculate the VGG-SIM. The VGG-SIM is calculated for each image, and then averaged across all images to get the mean VGG-SIM.

### Top-5 Accuracy

For this part of the analysis, we first discard images for which VGG-16 does not have  $\geq 0.5$  confidence in the top prediction (this was around 25 percent of the test images). We then take the top-5 category predictions (from VGG model) for the reconstructed image, and match it against the top prediction of the original image. The top-5 accuracy is the fraction of images for which the top prediction of the original image is

in the top-5 predictions of the reconstructed image. We also calculate the top-5 accuracy for the latent reconstructions.

The chance accuracy for a 200 way classification (between 200 categories) is  $5/200$ , but since VGG-16 outputs a 1000 way classification, the chance accuracy somewhere between  $5/200$  and  $5/1000$ . Therefore, we shuffle the image and top-5 category predictions, and calculate the top-5 accuracy for the shuffled data. This gives us the chance accuracy (which is close to  $5/1000$ )

## Implementation Details

All the GAN models used were public models pretrained on Imagenet. The link to the original repositories along with the pretrained models for [StyleGAN-XL](#) [BigBiGAN](#) and [ResNet-based GAN](#) are linked. The code was implemented in a mixture of Pytorch and Tensorflow. We used a NVIDIA 2080Ti for all GPU computations.

## Part I

# Appendix



## Setup

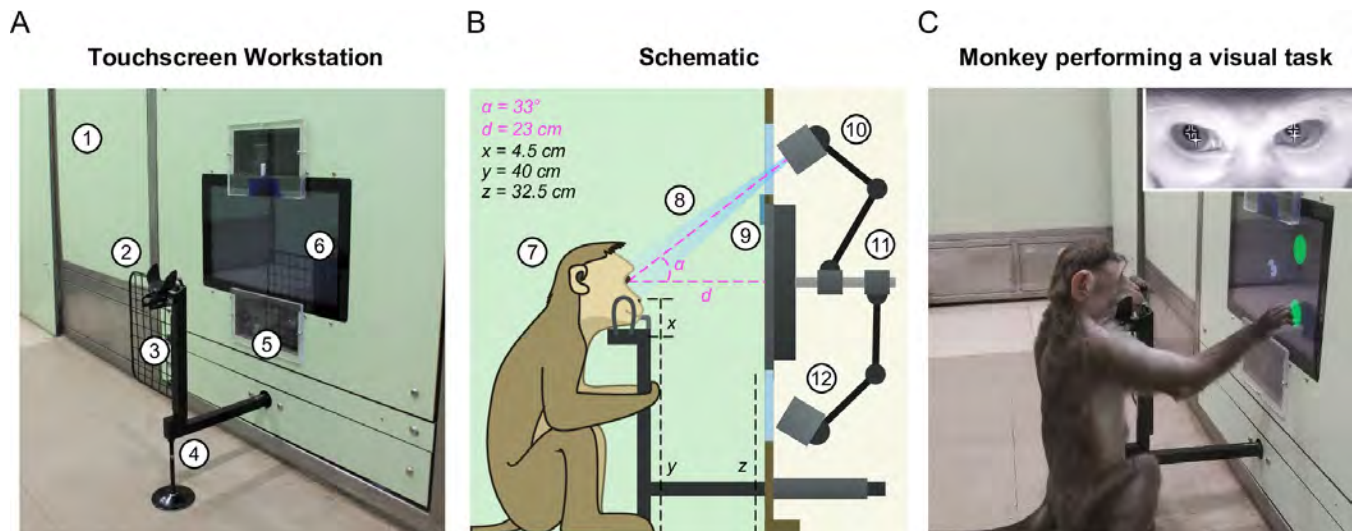


Figure 2: Setup for data collection from the paper [27]. Original Caption: Touchscreen workstation with eye tracking for unrestrained monkeys. (A) Labeled photograph of the touchscreen workstation from the monkey's side. Labels: 1: Partition panel with electromagnetic shielding; 2: Chin rest; 3: Grill to block left-hand screen access; 4: Movable reward delivery arm with concealed juice pipe; 5: Transparent viewports; 6: Touchscreen. (B) Labeled cross-section showing both monkey and experimenter sides. Labels: 7: Position of monkey at the workstation; 8: Field of view of the eye tracker; 9: Channel for mounting photodiode; 10: Eye tracker camera and additional synchronized optical video camera; 11: Adjustable arms mounted on the shaft behind touchscreen back panel; 12: Eye tracker IR illuminator. (C) Photograph of monkey M1 performing a task. Inset: Screenshot from the ISCAN IR eye tracker camera feed while monkey was doing the task, showing the detected pupil (black crosshair with white border) and corneal reflection (white crosshair with black border)

## More reconstructions

Some more selected good reconstructions for each model are shown below.

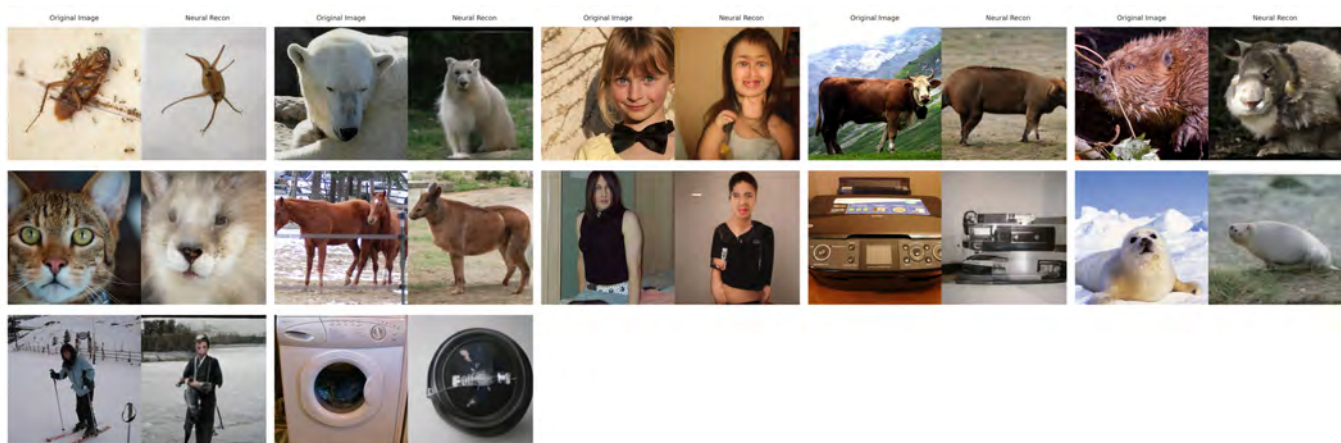


Figure 3: Reconstruction using StyleGAN-XL

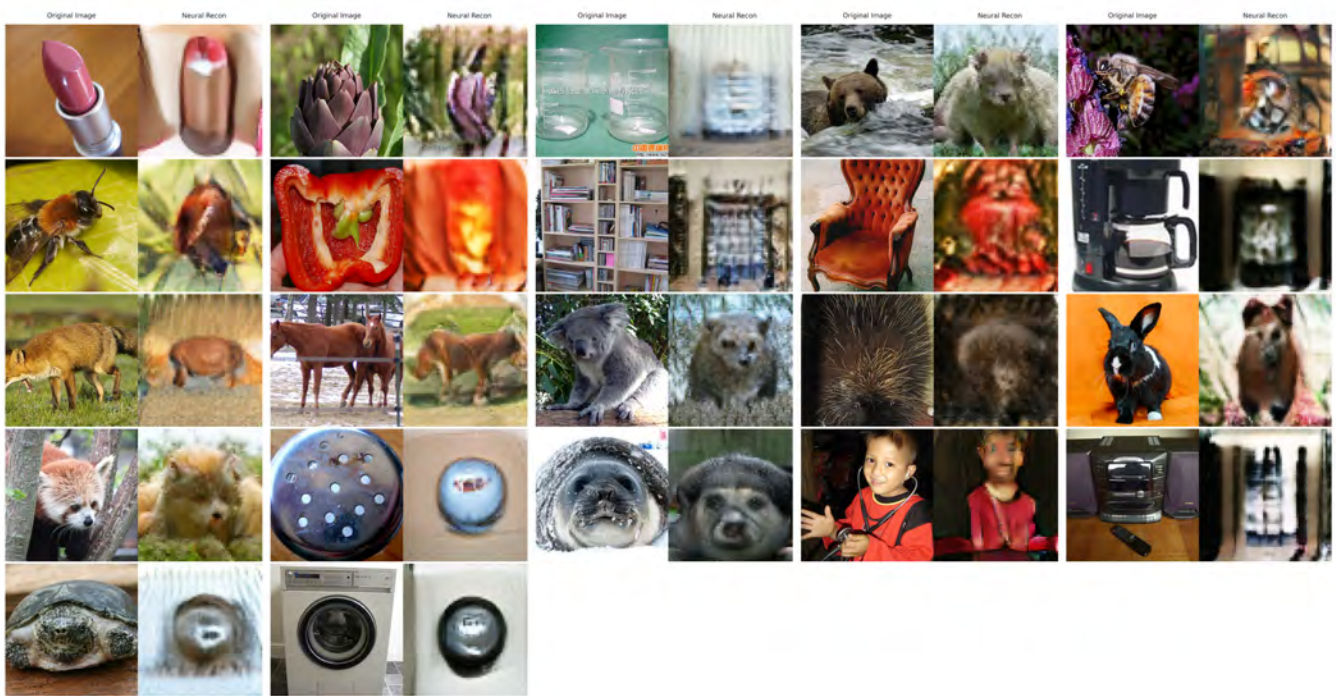


Figure 4: Reconstruction using Robust ResNet-GAN

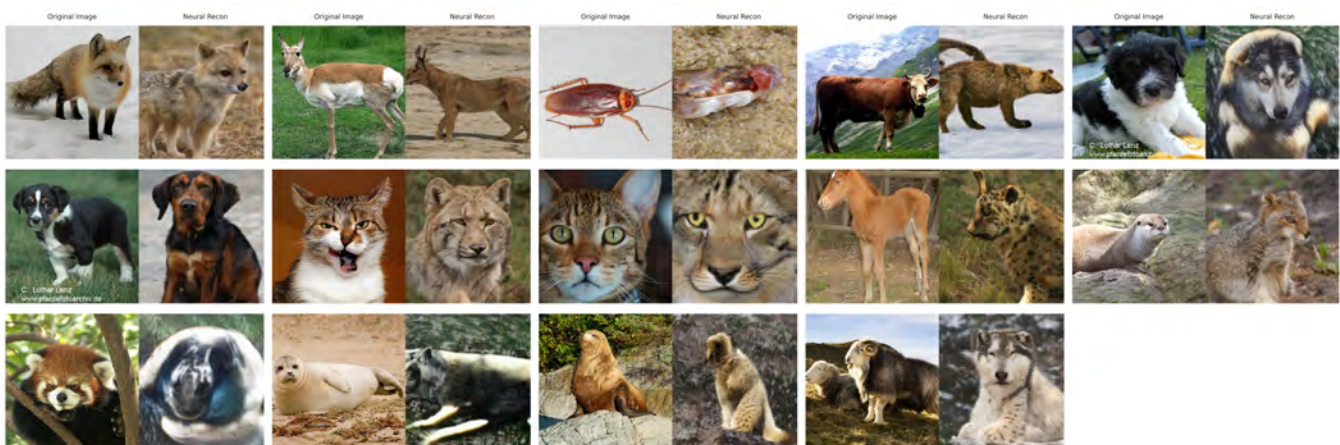


Figure 5: Reconstruction using BigBiGAN

## Latent reconstruction

This section contains the latent reconstructions for all the neural reconstruction shows. Contrasting these images with neural reconstructions indicates how much of the distortion is a result of the model itself being imperfect.

## Out-of-natural-distribution Images



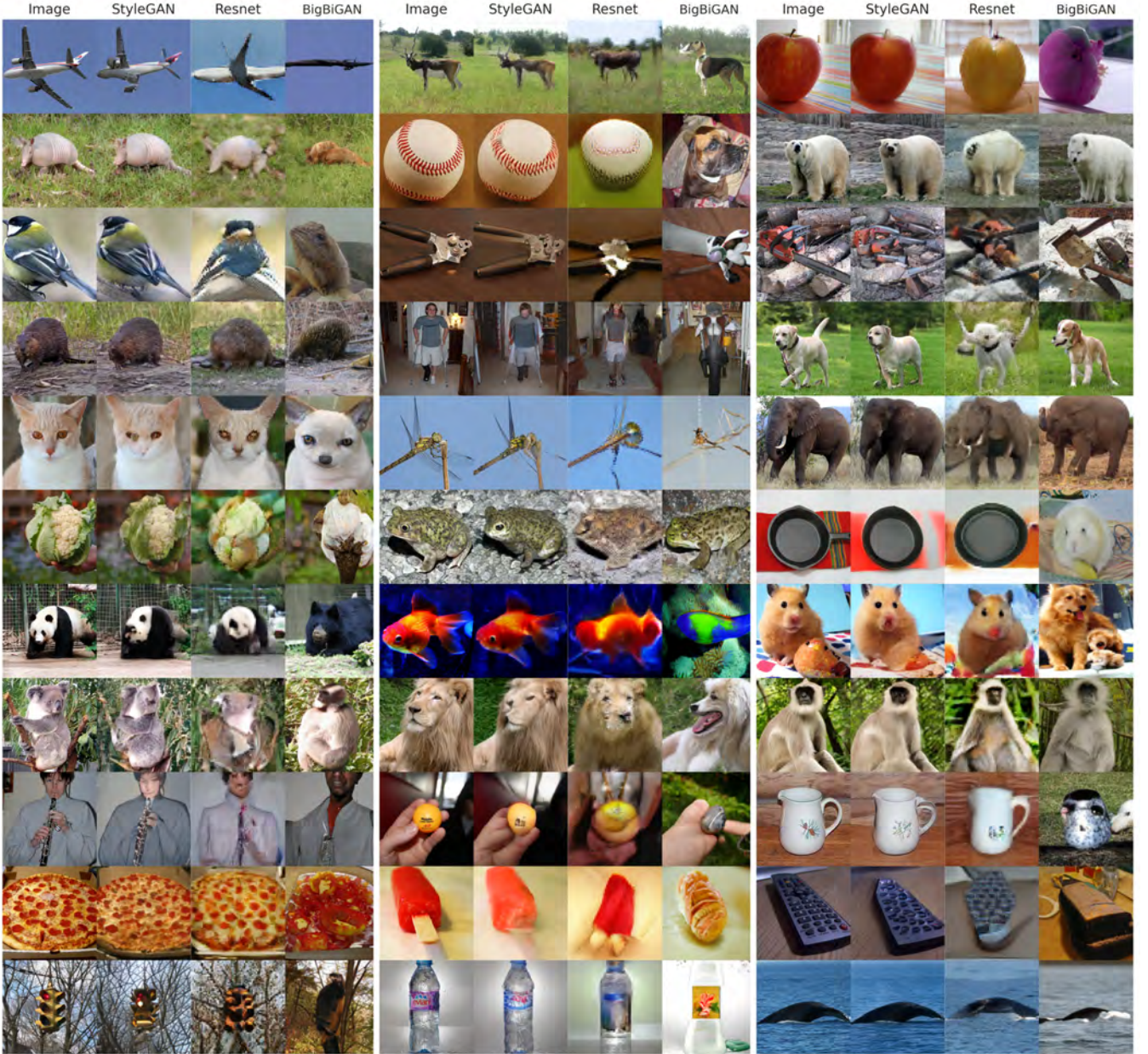


Figure 6: Reconstruction using latents of different models =  $G(E(Img))$  (without any neural data). The difference in the original and reconstructed images indicates the intrinsic loss that arises from the model itself. This establishes the ceiling performance using the GAN.



Figure 7: Latent Reconstruction of AI generated images from Robust ResNet-GAN Latents.1/5





Figure 8: Latent reconstruction of out-of-natural-distribution images from Robust ResNet-GAN Latents. Alternate rows represent the original image and the latent reconstruction using models . In each class, the reconstructions are predictable variations of the original images.1/5



Figure 9: Latent Reconstruction of Gabor images from Robust ResNet-GAN Latents.3/5





Figure 10: Latent Reconstruction of Silhouette images from Robust ResNet-GAN Latents.4/5



Figure 11: Latent Reconstruction of Texture images from Robust ResNet-GAN Latents.5/5

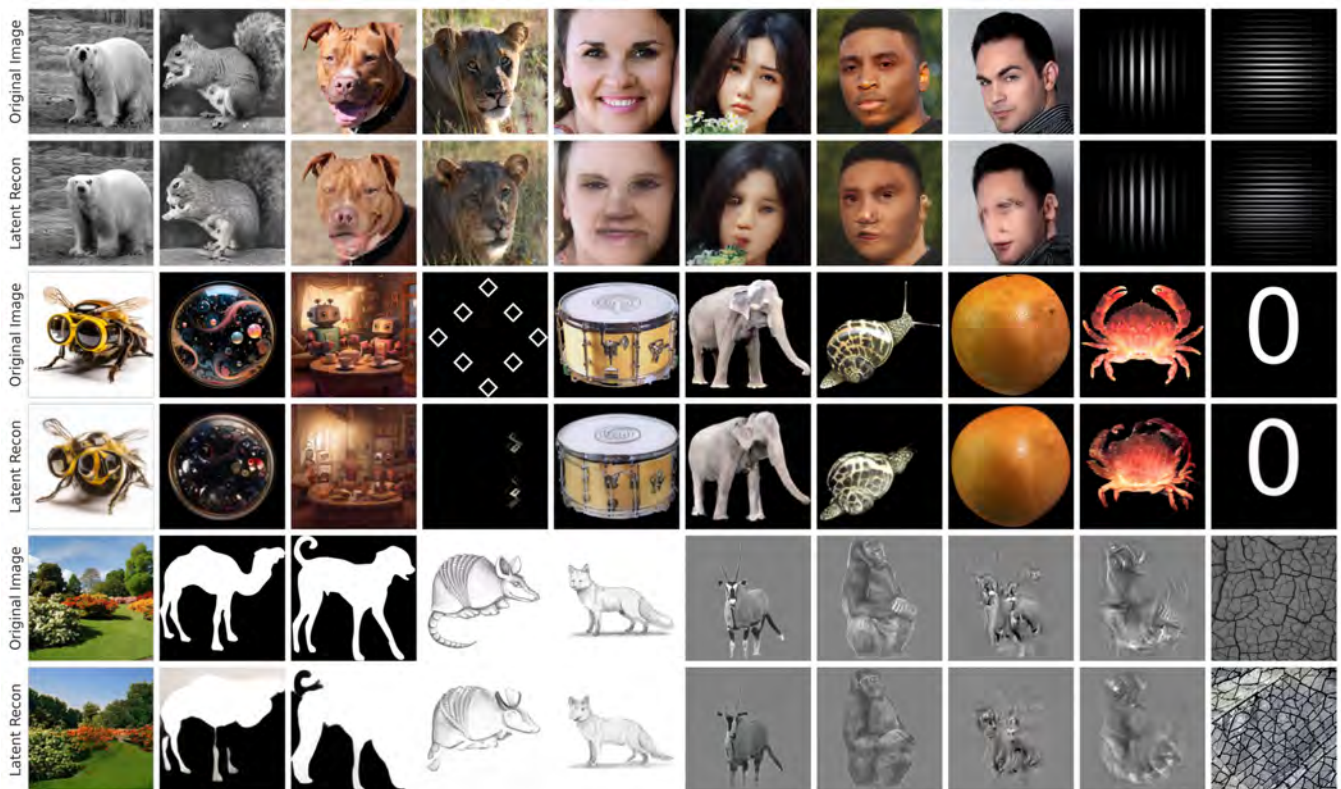


Figure 12: Latent reconstruction of out-of-natural-distribution images from StyleGAN-XL latents.

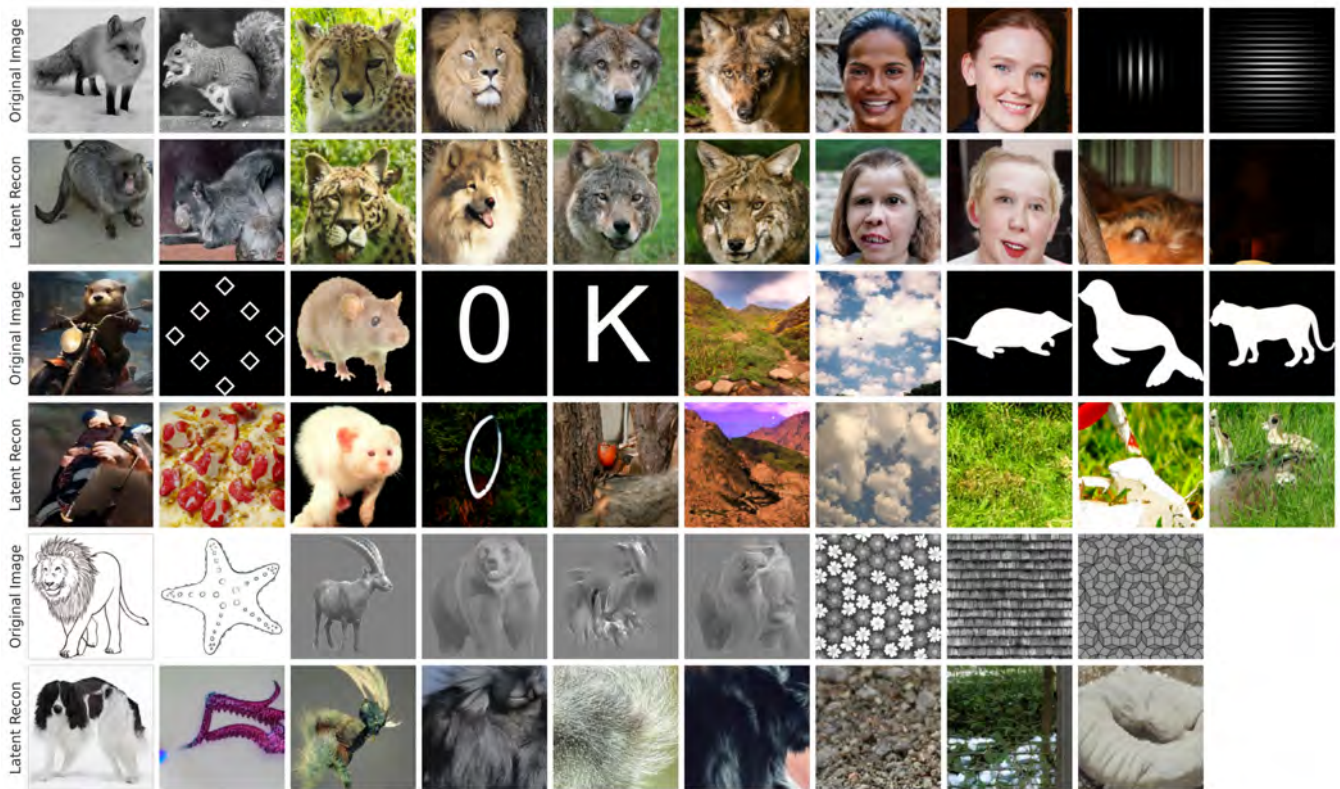


Figure 13: Latent reconstruction of out-of-natural-distribution images from BigBiGAN latents.



# Bibliography

- [1] Nikolaus Kriegeskorte. “Deep neural networks: A new framework for modeling biological vision and brain information processing”. en. In: *Annu. Rev. Vis. Sci.* 1.1 (Nov. 2015), pp. 417–446.
- [2] Jesse A Livezey and Joshua I Glaser. “Deep learning approaches for neural decoding across architectures and recording modalities”. en. In: *Brief. Bioinform.* 22.2 (Mar. 2021), pp. 1577–1591.
- [3] Shaul Hochstein and Merav Ahissar. “View from the top, hierarchies and reverse hierarchies in the visual system”. en. In: *Neuron* 36.5 (Dec. 2002), pp. 791–804.
- [4] G Wallis and E T Rolls. “Invariant face and object recognition in the visual system”. en. In: *Prog. Neurobiol.* 51.2 (Feb. 1997), pp. 167–194.
- [5] N Apurva Ratan Murty and Sripathi P Arun. “Dynamics of 3D view invariance in monkey inferotemporal cortex”. en. In: *J. Neurophysiol.* 113.7 (Apr. 2015), pp. 2180–2194.
- [6] Rufin Vogels and Irving Biederman. “Effects of illumination intensity and direction on object coding in macaque inferior temporal cortex”. en. In: *Cereb. Cortex* 12.7 (July 2002), pp. 756–766.
- [7] James J DiCarlo and David D Cox. “Untangling invariant object recognition”. en. In: *Trends Cogn. Sci.* 11.8 (Aug. 2007), pp. 333–341.
- [8] Philipp Zhu Jun-Yan, Eli Shechtman, and Alexei A. Efros. “Generative Visual Manipulation on the Natural Image Manifold”. In: *Computer Vision – ECCV 2016*. Ed. by Bastian Leibe et al. Cham: Springer International Publishing, 2016, pp. 597–613. ISBN: 978-3-319-46454-1.
- [9] Jia Deng et al. “Imagenet: A large-scale hierarchical image database”. In: *2009 IEEE conference on computer vision and pattern recognition*. Ieee. 2009, pp. 248–255.
- [10] Phil Pope et al. “The Intrinsic Dimension of Images and Its Impact on Learning”. In: *International Conference on Learning Representations*. 2021. URL: <https://openreview.net/forum?id=XJk19XzGq2J>.
- [11] Mehdi Mirza and Simon Osindero. “Conditional generative Adversarial Nets”. In: (Nov. 2014). arXiv: [1411.1784](https://arxiv.org/abs/1411.1784) [cs.LG].
- [12] Matthew D Zeiler and Rob Fergus. “Visualizing and understanding convolutional networks”. In: (Nov. 2013). arXiv: [1311.2901](https://arxiv.org/abs/1311.2901) [cs.CV].
- [13] Piotr Teterwak et al. “Understanding Invariance via Feedforward Inversion of Discriminatively Trained Classifiers”. In: *Proceedings of the 38th International Conference on Machine Learning*. Ed. by Marina Meila and Tong Zhang. Vol. 139. Proceedings of Machine Learning Research. PMLR, 2021, pp. 10225–10235. URL: <https://proceedings.mlr.press/v139/teterwak21a.html>.
- [14] Jeff Donahue and Philipp Krähenbühl. *ADVERSARIAL FEATURE LEARNING*. <http://arxiv.org/abs/1605.09782>. Accessed: 2024-4-10.
- [15] Jeff Donahue and Karen Simonyan. “Large scale adversarial representation learning”. In: (July 2019). arXiv: [1907.02544](https://arxiv.org/abs/1907.02544) [cs.CV].

- [16] Richard Zhang et al. “The Unreasonable Effectiveness of Deep Features as a Perceptual Metric”. In: *CVPR*. 2018.
- [17] Thirza Dado et al. “Brain2GAN: Feature-disentangled neural encoding and decoding of visual perception in the primate brain”. Apr. 2023.
- [18] Zarina Rakhimberdina et al. “Natural image reconstruction from fMRI using deep learning: A survey”. en. In: *Front. Neurosci.* 15 (Dec. 2021), p. 795488.
- [19] Ryusuke Hayashi and Hayaki Kawata. “Image reconstruction from neural activity recorded from monkey inferior temporal cortex using generative adversarial networks”. In: *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. Miyazaki, Japan: IEEE, Oct. 2018.
- [20] Milad Mozafari, Leila Reddy, and Rufin VanRullen. “Reconstructing Natural Scenes from fMRI Patterns using BigBiGAN”. In: *2020 International Joint Conference on Neural Networks (IJCNN)*. Glasgow, United Kingdom: IEEE, July 2020.
- [21] Yu Takagi and Shinji Nishimoto. “High-resolution image reconstruction with latent diffusion models from human brain activity”. Nov. 2022.
- [22] Yu Takagi and Shinji Nishimoto. “Improving visual image reconstruction from human brain activity using latent diffusion models via multiple decoded inputs”. In: (June 2023). arXiv: [2306.11536 \[q-bio.NC\]](#).
- [23] Bolei Zhou et al. “Places: An image database for deep scene understanding”. In: (Oct. 2016). arXiv: [1610.02055 \[cs.CV\]](#).
- [24] Bria Long, Viola S Störmer, and George A Alvarez. “Mid-level perceptual features contain early cues to animacy”. en. In: *J. Vis.* 17.6 (June 2017), p. 20.
- [25] Prajwal Singh et al. “Learning robust deep visual representations from EEG brain recordings”. In: (Oct. 2023). arXiv: [2310.16532 \[cs.CV\]](#).
- [26] Guy Gaziv et al. “Self-supervised natural image reconstruction and large-scale semantic classification from brain activity”. en. In: *Neuroimage* 254.119121 (July 2022), p. 119121.
- [27] Georgin Jacob et al. “A naturalistic environment to study visual cognition in unrestrained monkeys”. en. In: *Elife* 10 (Nov. 2021).
- [28] Hugo Touvron et al. “Training data-efficient image transformers & distillation through attention”. In: (Dec. 2020). arXiv: [2012.12877 \[cs.CV\]](#).
- [29] Axel Sauer, Katja Schwarz, and Andreas Geiger. “StyleGAN-XL: Scaling StyleGAN to Large Diverse Datasets”. In: (Feb. 2022). arXiv: [2202.00273 \[cs.LG\]](#).