Deep convolutional models improve predictions of macaque V1 responses to natural images

Santiago A. Cadena[1,3,6,@], George H. Denfield[4,6], Edgar Y. Walker[4,6], Leon A. Gatys[1,3], Andreas S. Tolias[3,4,5,6,*], Matthias Bethge[1,2,3,6,*], and Alexander S. Ecker[1,3,6,*]

[1]Centre for Integrative Neuroscience and Institute for Theoretical Physics, University of Tübingen, Germany
[2]Max Planck Institute for Biological Cybernetics, Tübingen, Germany
[3]Bernstein Center for Computational Neuroscience, Tübingen, Germany
[4]Department of Neuroscience, Baylor College of Medicine, Houston, TX, USA
[5]Department of Electrical and Computer Engineering, Rice University, Houston, TX, USA
[6]Center for Neuroscience and Artificial Intelligence, BCM. Houston, TX, USA
[*]These authors contributed equally
[@]Corresponding author: santiago.cadena@bethgelab.org

Abstract

Despite great efforts over several decades, our best models of primary visual cortex (V1) still predict neural responses quite poorly when probed with natural stimuli, highlighting our limited understanding of the nonlinear computations in V1. At the same time, recent advances in machine learning have shown that deep neural networks can learn highly nonlinear functions for visual information processing. Two approaches based on deep learning have recently been successfully applied to neural data: transfer learning for predicting neural activity in higher areas of the primate ventral stream and data-driven models to predict retina and V1 neural activity of mice. However, so far there exists no comparison between the two approaches and neither of them has been used to model the early primate visual system. Here, we test the ability of both approaches to predict neural responses to natural images in V1 of awake monkeys. We found that both deep learning approaches outperformed classical linear-nonlinear and wavelet-based feature representations building on existing V1 encoding theories. On our dataset, transfer learning and data-driven models performed similarly, while the data-driven model employed a much simpler architecture. Thus, multi-layer CNNs set the new state of the art for predicting neural responses to natural images in primate V1. Having such good predictive *in-silico* models opens the door for quantitative studies of yet unknown nonlinear computations in V1 without being limited by the available experimental time.

1 Introduction

An essential step towards understanding visual processing in the brain is building models that accurately predict neural responses to arbitrary stimuli (Carandini et al., 2005). Primary visual cortex (V1) has been a strong focus of sensory neuroscience ever since Hubel and Wiesel's seminal studies demonstrated that neurons in V1 respond selectively to distinct image features like local orientation and contrast (Hubel and Wiesel, 1959, 1968). Our current standard model of V1 is based on linear-nonlinear models (LN) (Jones and Palmer, 1987; Heeger, 1992) and energy models (Adelson and Bergen, 1985) to explain simple and complex cells, respectively. While these simple models explain responses to simple stimuli such as gratings reasonably well, they fail to account for neural responses to natural stimuli (Olshausen and Field, 2005; Talebi and Baker, 2012).

Simple LN models fail because natural stimuli unlock nonlinear subunits that cannot be captured by a linear transformation of the stimulus. To address this issue, LN-LN cascade models have been proposed, which either learn (convolutional) subunits (Rust et al., 2005; Touryan et al., 2005; Vintch et al., 2015) or use handcrafted wavelet representations (Willmore et al., 2008). These cascade models outperform simple LN models, but they currently do not capture the full range of nonlinearities observed in V1, such as gain control mechanisms and potentially other not-yet-understood nonlinear response properties. Because experimental time is limited, LN-LN models have to be designed carefully to keep the number of parameters tractable, limiting their expressiveness to energy models for direction-selective and complex cells.

Recent advances in machine learning and computer vision using deep neural networks ('deep learning') have opened a new door to learn much more complex non-linear models of neural responses. We identify two main approaches that we refer to as goal-driven, and data-driven.

The goal-driven approach is based on transfer learning (Donahue et al., 2014), a paradigm that has been very successful in deep learning. Convolutional neural networks (CNNs) have reached human-level performance on visual tasks like object classification by training on more than one million images (Krizhevsky et al., 2012; Simonyan and Zisserman, 2014; He et al., 2016; Huang et al., 2016). These CNNs optimized for visual tasks have proven extremely useful as nonlinear feature spaces for tasks where less labeled data is available (e.g. Kümmerer et al. 2014). This transfer to a new task is achieved by (linearly) reading out the network's internal representations of the input. Yamins, DiCarlo and colleagues showed that using deep networks trained on large-scale object recognition as nonlinear feature spaces for neural system identification works remarkably well in higher areas of the ventral stream, such as V4 and IT (Yamins et al., 2014).

The deep data-driven approach is based on fitting all model parameters directly to neural data (Antolík et al., 2016; Batty et al., 2016; McIntosh et al., 2016; Klindt et al., 2017). The critical advance of deep models in neural system identification is that they can have many more parameters than classical LN cascade models discussed above, because they exploit computational similarities between different neurons. While previous approaches treated each neuron as an individual multivariate regression problem, modern CNN-based approaches learn one model for an entire population of neurons, thereby exploiting two key properties of local neural circuits: (1) they share the same presynaptic circuitry (for V1: retina and LGN) and (2) many neurons perform essentially the same computation, but at different locations (topographic organization, implemented by convolutional weight sharing).

While both, goal-driven and data-driven approaches have been shown to outperform LN models, it is currently unknown how their performance compares. Moreover, neither approach has been evaluated

in monkey V1 (see Kindel et al. 2017 for concurrent work). Here, we address this, and show that both methods perform similarly well outperforming classic LN cascade models by a substantial margin, offering an alternative to study unknown V1 nonlinear properties.

2 Materials and Methods

2.1 Electrophysiological recordings

We performed non-chronic recordings from two adult male rhesus monkeys (aged 8 and 11; weighing 10.9kg and 12.1kg) with a 32-channel linear silicon probe (NeuroNexus V1x32-Edge-10mm-60-177). The surgical methods and recording protocol were described previously (Denfield et al., 2017). Briefly, form-specific titanium recording chambers and headposts were implanted under full anesthesia and aseptic conditions. The bone was originally left intact, and only prior to recordings small trephinations (2 mm) were made over medial primary visual cortex at eccentricities ranging from 1.4 to 3.0 degrees of visual angle. Recordings were done within two weeks of each trephination. Probes were lowered using a Narishige Microdrive (MO-97) and a guide tube to penetrate the dura. Care was taken lower the probe slowly, not to penetrate the cortex with the guide tube and to minimize tissue compression (for a detailed description of the procedure, see Denfield et al. 2017). All experimental procedures complied with guidelines of the NIH and were approved by the Baylor College of Medicine Institutional Animal Care and Use Committee (permit number: AN-4367).

2.2 Data acquisition and spike sorting

Electrophysiological data were collected continuously as broadband signal (0.5–16,000 Hz) digitized at 24 bits as described previously (Ecker et al., 2010). Our spike sorting methods are based on (Ecker et al. 2014, code available at https://github.com/aecker/moksm), but with adaptations to the novel type of silicon probe as described previously (Denfield et al., 2017). Briefly, we split the linear array of 32 channels into 14 groups of 6 adjacent channels (with a stride of two), which we treated as virtual electrodes for spike detection and sorting. Spikes were detected when channel signals crossed a threshold of five times the standard deviation of the noise. After spike alignment, we extracted the first three principal components of each channel, resulting in an 18-dimensional feature space used for spike sorting. We fitted a Kalman filter mixture model (Calabrese and Paninski, 2011; Shan et al., 2017) to track waveform drift typical for non-chronic recordings. The shape of each cluster was modeled with a multivariate t-distribution ($df = 5$) with a ridge regularized covariance matrix. The number of clusters was determined based on a penalized average likelihood with a constant cost per additional cluster (Ecker et al., 2014). Subsequently, we used a custom graphical user interface to manually verify single-unit isolation by assessing the stability of the units (based on drifts and health of the cells throughout the session), identifying a refractory period, and inspecting the scatter plots of the pairs of channel principal components.

2.3 Visual stimulation and eye tracking

Visual stimuli were rendered by a dedicated graphics workstation and displayed on a CRT monitor with a 100 Hz refresh rate. The monitors were gamma corrected to have a linear luminance response profile. A camera-based, custom-built eye tracking system verified that monkeys maintained fixation within ~ 0.42 degrees around the target. Offline analysis showed that monkeys typically fixated much more accurately.

The monkeys were trained to fixate on a red target of ~ 0.15 degrees in the middle of the screen. After they maintained fixation for 300 ms, a visual stimulus appeared. If the monkeys fixated throughout the entire stimulus period, they received a drop of juice at the end of the trial.

2.4 Receptive field mapping

At the beginning of each session, we first mapped receptive fields. We used a sparse random dot stimulus for receptive field mapping. A single dot of size 0.12 degrees of visual field was presented on a uniform gray background, changing location and color (black or white) randomly every 30 ms. Each trial lasted 2 seconds. We obtained multi-unit receptive field profiles for every channel using reverse correlation. We then estimated the population receptive field location by fitting a 2D Gaussian to the spike-triggered average across channels at the time lag that maximizes the signal-to-noise-ratio. We subsequently placed our natural image stimulus at this location.

2.5 Natural image stimulus

We used an approach similar to Freeman et al. 2013. We generated stimuli with different degrees of "naturalness" by capturing different levels of higher order correlations from a local to a global scale. This was achieved by using a parametric model for texture synthesis proposed by Gatys et al. 2015 that uses the pre-trained feature maps of VGG-19 (Simonyan and Zisserman, 2014). Briefly, the algorithm consists of analysis and synthesis stages. During analysis, the summary statistics —given by the correlation matrix between feature maps (also, Gram matrix)— are computed for each layer in the net. During synthesis, by starting with a random white noise image, pixels are pushed (usually via gradient descend) in a direction that leads to Gram matrices matching those of the original image.

For visual stimuli, we randomly selected and gray-scaled 1450 images from ImageNet (Russakovsky et al., 2015). Additionally, for every image, we synthesized four new types of images using the parametric texture model (Gatys et al., 2015). For displaying and further analyses, we cropped the central 140 pixels of each image. For texture synthesis, we matched all the Gram matrices cumulatively up to conv1, conv2 and conv3 and conv4. (e.g. the conv3 model matches Gram matrices for layers conv1_1, conv2_1 and conv3_1 of VGG-19). Figure 1A shows three example images with their respective texturized versions.

The entire data set contains $1450 \times 5 = 7250$ images (original plus synthesized). During each trial, 29 images were displayed, each for 60 ms, with no blanks in between (Figure 1 B). Each image was masked by a circular mask with a diameter of 2 degrees and a soft fade-out starting at a diameter of 1 degree:

$$m(r) = \begin{cases} 1 & \text{if } r < 1 \\ 0.5\cos(\pi(r-1)) + 0.5 & \text{otherwise} \end{cases}$$

Images were randomized such that consecutive images were not of the same type or synthesized from the same image. A full pass through the dataset took 250 successful trials, after which it was traversed again in a new random order. Images were repeated between one and four times, depending on how many trials the monkeys completed in each session.

2.6 GLM with pre-trained CNN features

Our proposed model consists of two parts: feature extraction and a generalized linear model (GLM; Fig. 3). The features are the output maps of intermediate convolutional layers of VGG-19 (Simonyan and

Zisserman, 2014) to a stimulus image. We fit a separate GLM for each convolutional layer of VGG-19. We used a normalized version of VGG-19, where the weights have been rescaled such that the average activation of each feature map over a large set of natural images is equal to one (Gatys et al., 2016). The original 140 px images were first cropped to omit the 30 px border and then downsampled by a factor of two, resulting in images of size 40 px. The output of the convolutional layers is a set of K feature maps (denoted as depth in Fig. 3).

The GLM consists of linear fully connected weights w_{ijk} for each neuron that compute a dot product with the input feature maps $\psi_{ijk}(x)$, an exponential nonlinearity, and Poisson noise. Here, i and j index space, while k indexes feature maps. The weights have the same dimensionality as the feature maps. The spiking rate of a given neuron r will follow:

$$\log r(x) = \sum q_{ijk} w_{ijk} + b \tag{1}$$

Additionally, three regularization terms were applied to the weights:

1. **Sparsity**: Most weights need to be zero since we expect the spatial pooling to be localized. We use the L1 norm of the weights:

$$\mathcal{L}_{sparse} = \lambda_{sparse} \sum |w_{ijk}| \tag{2}$$

2. **Spatial Smoothness**: Together with sparseness, spatial smoothness encourages spatial locality by imposing continual regular changes in space. We computed this by an L2 penalty on the Laplacian of the weights:

$$\mathcal{L}_{laplace} = \lambda_{laplace} \sqrt{\sum (w_{:,:,k} * L)_{ij}^2}, \quad L = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} \tag{3}$$

3. **Group Sparsity**: Encourages our model to pool from a reduced set of feature maps to explain each neuron's responses:

$$\mathcal{L}_{group} = \lambda_{group} \sum_k \sqrt{w_{ijk}^2} \tag{4}$$

Considering the recorded image-response as (x, y) for one neuron, the resulting loss function is given by:

$$\mathcal{L} = -\sum y \log r(x) + r(x) + \mathcal{L}_{sparse} + \mathcal{L}_{laplace} + \mathcal{L}_{group} \tag{5}$$

We fit the model by minimizing the loss using the Adam optimizer (Kingma and Ba, 2014) on a training set consisting of 80% of the data, and reported performance on the remaining 20%. We cross-validated the parameters λ_{sparse}, λ_{lap}, λ_{group} for each neuron independently by performing a grid search over four logarithmically spaced values for each parameter. The validation was done on 20% of the training data. The same split of data for training, validation, and testing was used to fit all models in this study.

2.7 Convolutional neural network model

We followed the results of Klindt et al. 2017 and use their best-performing architecture that obtained state-of-the-art performance on a public dataset (Antolík et al., 2016). As our VGG-based model, this model also consisted of convolutional feature extraction followed by a GLM, the difference being that here the convolutional feature space was learned from neural data instead of having been trained on object recognition. The feature extraction architecture consisted of three convolutional layers with filters of receptive field size 13×13 px for the first layer and 3×3 px for the subsequent layers. Each layer had 32 feature maps (Fig. 5). As in the original publication (Klindt et al., 2017) we regularized the convolutional filters by imposing smoothness constraints on the first layer and group sparseness on the second and third. A notable difference to our VGG-based GLM is that here the readout weights are factorized in space and feature maps:

$$w_{ijk} = u_{ij} v_k$$

where u_{ij} is a spatial mask and v_k a set of feature pooling weights. We used an exponential linear unit (ELU; Clevert et al. 2015) as the output nonlinearity.

2.8 Other baseline models

The performance of the two convolutional models was compared with two alternative models: a regularized linear nonlinear Poisson model (LNP; Simoncelli et al. 2004) and the Berkeley wavelet transform (BWT) linearized model (Willmore et al., 2008). Images were down-sampled to 40 px as in our proposed model.

The LNP model was fitted using two regularization terms: smoothness and sparseness. Their corresponding parameters were cross-validated independently for each cell as above.

The BWT model (Willmore et al., 2008, 2010) uses a set of scaled, oriented, frequency- and phase-shifted wavelets to decompose the original image. We used the publicly available implementation from StrfLab (Willmore et al., 2008) and we set the temporal size and temporal velocities to one. We picked the following parameters for the Gabor wavelet bank that lead to best performance on test set in order for it to be competitive with the other methods: 16 evenly spaced orientations, 5 frequency divisions between 0.5 and 6 cycles per degree, 0.5 ratio between the Gaussian window and spatial frequency, 2.5 standard deviation of the Gaussian window of spatial separation of each wavelet. A log link and Poisson noise were used to fit the regression weights on top of the feature space.

2.9 Performance evaluation

We measured the performance of all models with the fraction of explainable variance explained FEV. That is, the ratio between the variance accounted for by the model (variance explained) and the explainable variance. The explainable variance is lower than the total variance, because observation noise prevents even a perfect model from accounting for all variance. We estimated the amount of observation noise by averaging the variance across images of responses to the same stimulus: $E_j[Var_i[y_i|x_j]]$. If our model predicted an average response of $\hat{y}$, then FEV is computed as in equation 6 for the observed spike counts y.

$$FEV = 1 - \frac{Var[\hat{y} - y] - E_i\left[Var_j[y]\right]}{Var[y] - E_i\left[Var_j[y]\right]} \tag{6}$$

3 Results

We measured the spiking activity of populations of neurons in V1 of two awake, fixating rhesus macaques using a 32-channel linear array spanning all cortical layers (Fig. 2A). We isolated 307 neurons in 23 sessions. Our stimuli consisted of synthesized images that capture different levels of high order correlations present in natural images (see Methods). Each stimulus was shown for 60ms, with no in-between blanks, and was centered on the mapped population receptive field of the neurons. The images were scaled to have the same contrast.

The entire set of stimuli has 7250 unique images and was shown one to four times for each session. We used only sessions with two or more repetitions, and applied a selection criterion to the neurons based on how much of their variability was induced by the stimulus. We estimated the observation noise by averaging the variance of responses to repeated presentations of the images. By subtracting the average trial-to-trial variance to repeated presentations from the total variance of the responses for every neuron, we obtain an estimate of the explainable variance. We discarded neurons with a ratio of explainable-to-total variance smaller than 0.15, yielding 166 isolated neurons recorded in 17 sessions. 51 neurons belonged to sessions with two repetitions and 115 to those with four.

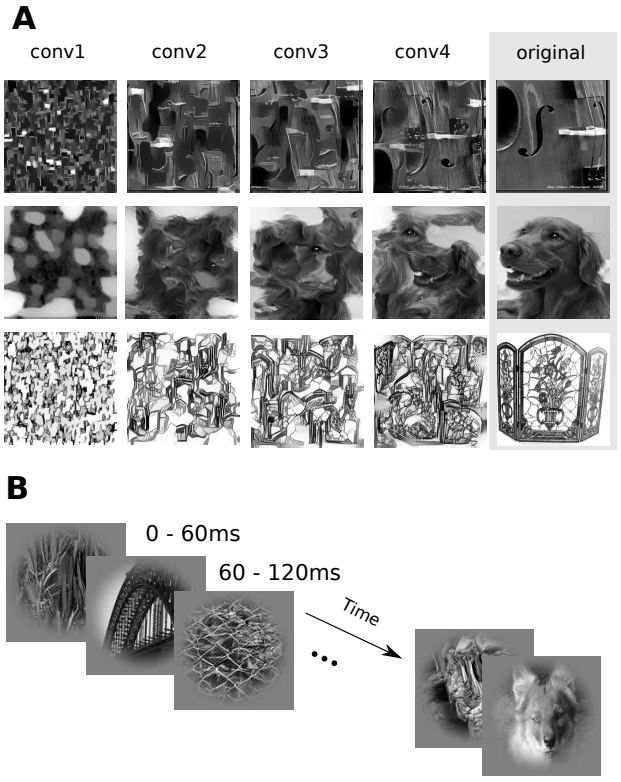

Figure 1. Stimulus paradigm A. Classes of images shown in the experiment. We used grayscale natural images (labeled 'original') from the ImageNet dataset (Russakovsky et al., 2015) along with textures synthesized from these images using the texture synthesis algorithm described by Gatys et al. 2015. Each row shows four synthesized versions of three example original images using different convolutional layers (see Materials and Methods for details). Lower convolutional layers capture more local statistics compared to higher ones. **B**. Stimulus sequence. In each trial, we showed a randomized sequence of images (each displayed for 60 ms covering 2 degrees of visual angle) centered on the receptive fields of the recorded neurons while the monkey sustained fixation on a target. The images were masked with a circular mask with cosine fadeout.

3.1 Generalized linear model with pre-trained CNN features

We used the network VGG-19 (Simonyan and Zisserman, 2014) to extract a nonlinear feature space for a generalized linear model (GLM). VGG-19 is a CNN trained on the large image classification task ImageNet (ILSVRC2012) that takes an RGB image as input and infers the class of the dominant object in the image (among 1000 possible classes). Its architecture consists of a hierarchy of linear-nonlinear transformations

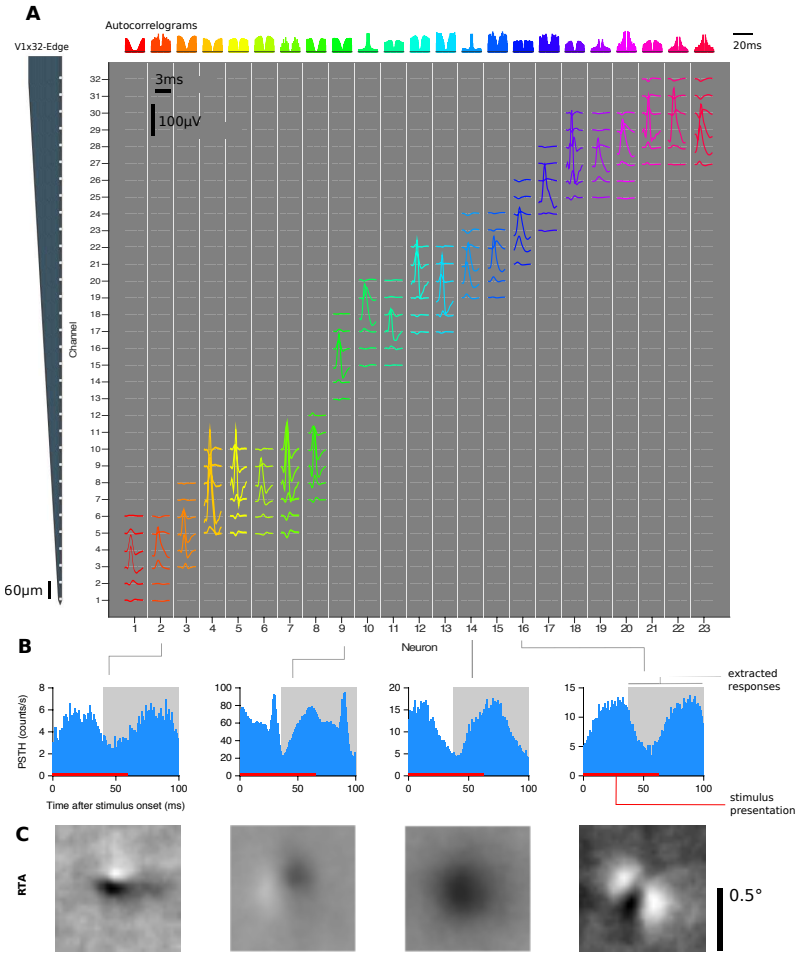

Figure 2. V1 electrophysiological responses. **A**. Isolated single-unit activity. We performed acute recordings with a 32-channel, linear array (NeuroNexus V1x32-Edge-10mm-60-177, layout shown in the left) to record in primary visual cortex of two awake, fixating macaques. The channel mean-waveform footprints of the spiking activity of 23 well-isolated neurons in one example session are shown in the central larger panel. The upper panel shows color-matched autocorrelograms. **B**. Peri-stimulus time histograms (PSTH) of four example neurons from A. Spike counts where binned with t = 1 ms, aligned to the onset of each stimulus image, and averaged over trials. The 60 ms interval where the image was displayed is shown in red. We ignored the temporal profile of the response and extracted spike counts for each image on the 40–100 ms interval after image onset (shown in light gray). **C**. The Response Triggered Average (RTA) calculated by reverse correlation of the extracted responses.

(layers) where the input is spatially convolved with a set of filters and then passed through a rectifying nonlinearity (Fig. 3). The output of such layers produces a number of feature maps that serve as input for the next layer. Additionally, the network has pooling layers where the feature maps are down-sampled by taking the local maximum values of neighboring pixels. There are 16 convolutional layers that can be grouped into five groups with 2, 2, 4, 4, 4 convolutional layers and 64, 128, 256, 512, 512 output feature maps, respectively, and a pooling layer in between each group.

For each convolutional layer of VGG-19, we fit a GLM that uses this layer's representation of the stimulus as a nonlinear feature space. To do so, we fed all images in our stimulus set through the network and extracted the feature maps of every convolutional layer (Fig. 3). We then learned a set of linear weights followed by an exponential nonlinearity to predict each neuron's response (Fig. 3). Since the convolutional feature spaces are larger than the number of pixels in the image, regularization of the readout weights is particularly important. We used three regularization terms for the weights. (1) Sparseness, because receptive fields are localized, we expect most weights to be zero; (2) smoothness, to encourage a regular spatial continuity of the receptive fields; and (3) group sparsity, which encourages the model to pool only from a small number of feature maps. We fit this model for each convolutional layer of VGG-19 to maximize the likelihood of the predicted response under a Poisson noise model and cross-validated over the three regularization terms for each cell independently.

To measure our model's performance and compare it to others, we computed the fraction of explain-

able variance explained (FEV). This metric, which ranges from 0 to 1, measures what fraction of the stimulus-driven response is explained by the model, ignoring the unexplainable trial-to-trial variability in the neurons' responses (for details see Methods).

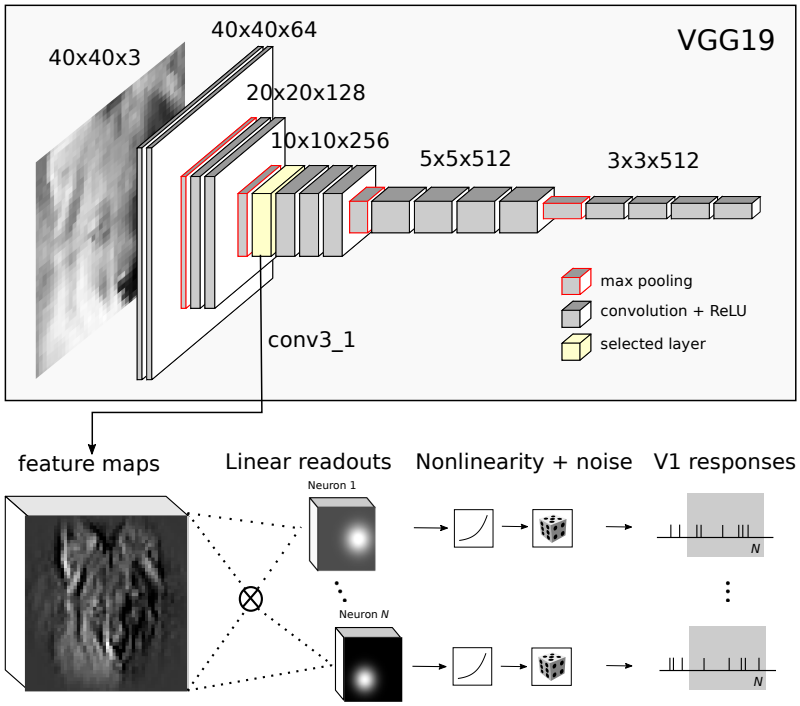

Figure 3. Our proposed model For each of the 16 convolutional layers of VGG-19 (Simonyan and Zisserman, 2014), we extract the output feature maps of the images shown to the monkey. We then train for each neuron a Generalized Linear Model with Poisson noise and log link on top of this representation to predict the observed spike counts from monkey V1. The linear readouts have the same size as the feature maps and their resulting dot product is fed to the exponential nonlinearity. The learning objective was Maximum Likelihood with three regularization terms on the weights for sparseness, spatial smoothness, and group sparseness (see Methods). This facilitated identifying a reduced set of feature maps to pool from, as well as the location of each neuron's receptive field.

3.2 Intermediate layers of VGG best predict V1 responses

The model based on the fifth (out of sixteen) layers' features (called 'conv3_1', Fig. 3) best predicted neuronal responses to novel images not seen during training (Fig. 4). This model predicted on average 50.1% of the explainable variance. In contrast, performance for the very first layer was poor (31% FEV), but increased monotonically up to conv3_1. Afterwards, the performance again decreased continually up the hierarchy (Fig. 4). These results followed our intuition that early to intermediate processing stages in a hierarchical model should match primary visual cortex, given that V1 is the third processing stage in the visual hierarchy after the retina and LGN.

One potential concern is that the performance curve may be related more to receptive field size of the units –the size of the input region which a unit depends on– than to actual nonlinear response properties. Each VGG layer convolves its inputs with a 3×3 px kernel, leading to growing receptive field sizes along the hierarchy. For example, the receptive field size of units in the first layer ('conv1_1')

is 3×3 px (which covers only 0.08 degrees of visual angle). Because nonlinear features are extracted at the scale of the units' receptive field, it may be important to match receptive field sizes between each VGG layer and V1. To address this concern, we resized the input image for each layer model such that the receptive field size of units in the layer that provided the feature space roughly matched 2 degrees of visual angle (the field of view that V1 neurons were stimulated with). In this way, the image patch each VGG unit saw was equivalent across layers. This procedure was done for the first nine convolutional layers as performance was already steadily decreasing. The resulting performance across layers agreed with the previous results (conv3_1 was still the best performing layer; Fig 4, dashed gray line).

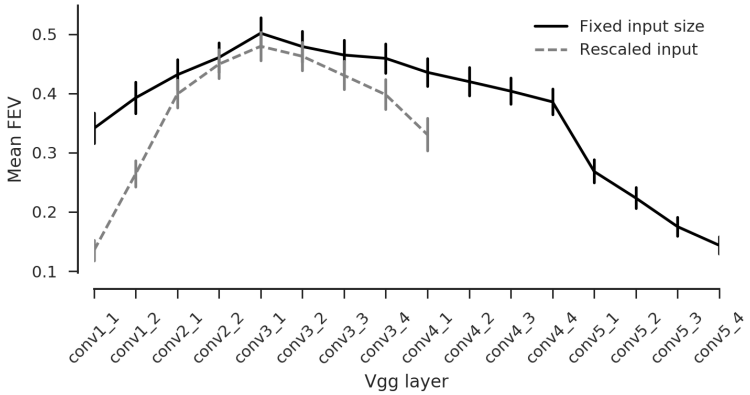

Figure 4. Model performance on test set Average fraction of explainable variance explained (FEV) on test set. Conv3_1 shows on average the highest predictive performance for both models trained with a fixed input size for all layers, and rescaled inputs to match units' receptive field sizes across layers.

3.3 Data-driven CNN model and GLM with pre-trained CNN set state of the art

We next asked how the predictive performance of our VGG-based model compared to other quantitative models of monkey V1. We therefore compared it to a classical linear-nonlinear Poisson (LNP) model, a wavelet-based model and a multi-layer CNN fit directly to the data.

We regularized the LNP model by selecting for smoothness and sparseness of the linear filters via cross-validation (see Methods). The wavelet-based model uses the Berkeley Wavelet Transform (BWT, Willmore et al. 2008, 2010), a handcrafted nonlinear feature space based on orthogonal wavelets that resemble Gabor functions. This model is the current state of the art in the neural prediction challenge for monkey V1 responses to natural images (http://neuralprediction.berkeley.edu). Because recent work has shown that multi-layer convolutional neural networks can be fit directly to neural data on natural image datasets (Antolík et al., 2016; Kindel et al., 2017; Klindt et al., 2017), we also fit a three-layer CNN identical to that of Klindt et al. 2017. This model is illustrated in Fig. 5. For more details on the models, see Methods.

We compared the models for a number of cells from a representative recording (Fig. 6A) and found a diversity of cells. For simple-like cells – cells for which the LNP had a high predictive power – all models

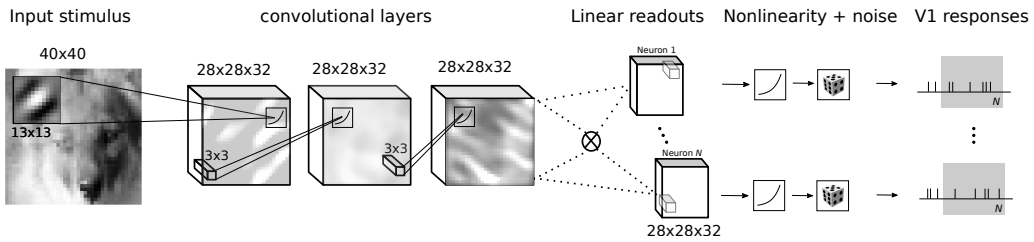

Figure 5. Convolutional neural network architecture. Following the approach of Klindt et al. 2017, we trained a three-layer convolutional neural network to produce a feature space fed to a GLM-like model. In contrast to the VGG-based model, both feature space and readout weights are trained only on the neural data.

performed approximately equally. However, the nonlinear feature spaces were able to better represent nonlinear (e.g. complex) cells, for which the BWT had a much larger performance than LNP (Fig. 6A).

The two deep learning approaches outperformed the other models of V1 (Fig. 6B, D). The LNP model achieved 17% FEV, the Berkeley Wavelet Transform model 39% FEV. The performance of the VGG-based model was comparable to that of the CNN trained directly on the data (Fig. 6C, D). On average, the VGG-based model yielded on average a slightly higher performance (50.1% FEV) than the data-driven CNN (46% FEV), but this difference was not significant ($p = 0.09$, t-test). In addition, a more extensive search of hyperparameters of the data-driven CNN architecture might lead to better performance.

4 Discussion

We fit two models based on convolutional neural networks to V1 responses to natural stimuli in awake, fixating monkeys: a goal-driven model, which uses the representations learned by a CNN trained on object recognition (VGG), and a data-driven model, which learns both the convolutional and readout parameters using stimulus-response pairs with multiple neurons simultaneously. Both approaches yielded comparable performance and outperformed the widely used LNP (Simoncelli et al., 2004) and the wavelet-decomposition model (BWT; Willmore et al. 2008), which held the previous state of the art in prediction of V1 responses to natural images. For the goal-driven model, we found that features of intermediate layers of VGG (layer conv3_1) explain V1 best.

The most successful system identification approaches to date all build on feature spaces shared by all neurons. There are three main ways in which this feature space can be chosen. First, handcrafted features have been used, which are based on existing neural encoding theories (e.g. BWT, Willmore et al. 2008; or HMAX, Riesenhuber and Poggio 1999). Second, more recent studies have learned shared feature spaces by jointly fitting the stimulus-response of all neurons in the dataset (Antolík et al., 2016; Batty et al., 2016; Kindel et al., 2017; Klindt et al., 2017; McIntosh et al., 2016). Third, inspired by the success of transfer learning in the machine learning community, researchers have borrowed representations optimized to solve a visual task like object recognition and used them to predict responses in high-level areas of the ventral stream (Yamins et al., 2014). We compared these three approaches quantitatively and showed that the last two have comparable and the highest predictive performance on our monkey V1 dataset.

This result has two important implications that we want to briefly discuss. First, the fact that deep models substantially outperformed the handcrafted feature spaces (BWT) shows that we still do not fully understand the computations performed by V1 – or at least that there still does not exist an explicit model

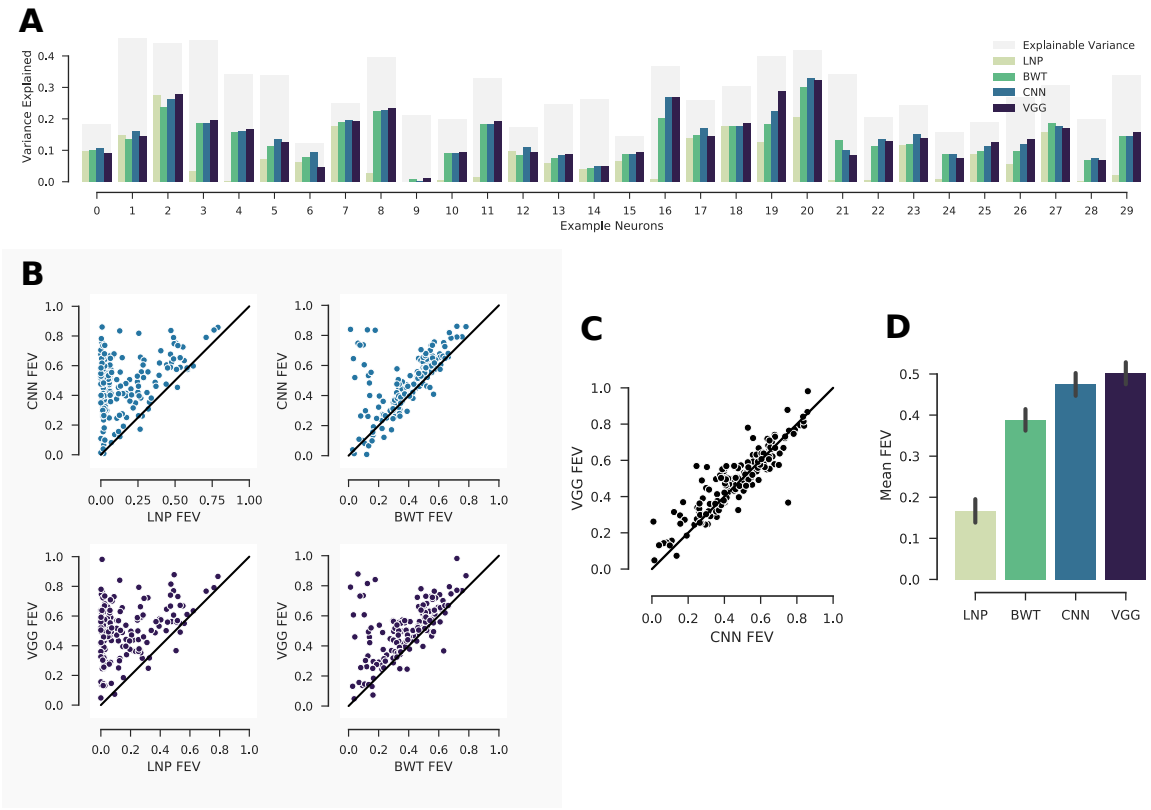

Figure 6. Deep models are the new state of the art. **A** Randomly selected cells. The normalized explainable variance (oracle) per cell is shown in gray. For each cell from left to right, the variance explained of: regularized LNP (Simoncelli et al., 2004), BWT (Willmore et al., 2008), three-layer CNN trained on neural responses, and VGG conv3_1 model (ours). **B** CNN and VGG conv3_1 models outperform for most cells LNP and BWT. Black line denotes the identity. The performance is given in FEV (see Methods). **C**. VGG conv3_1 features perform slightly better than the three-layer CNN. **D**. Average performance of the four models given in mean fraction of explainable variance explained (FEV).

applicable to natural images. Second, an architecture optimized for fitting the neural data (CNN model) did not outperform a feature space trained on a different task (object recognition). On the one hand, this result underscores the power of transfer learning and stresses the relevance of nonlinearities learned by VGG for predicting neural responses along the ventral visual stream. On the other hand, the fact that the data-driven model reached comparable performance with a shallower and less complex architecture, shows that the VGG feature space is not identical to that of V1. If it was, the VGG-based model should have outperformed the data-driven approach. Note, though, that these two approaches are each at one end of a spectrum: a hybrid approach, where a pre-trained feature space is used as initialization and subsequently fine-tuned may provide the right balance of inductive bias and flexibility, perhaps leading to even higher performance than either the data-driven or transfer-learning-based approaches. Moreover, it is possible that increasing the entropy of the stimulus set or the number of neurons (or both) could improve performance of purely data-driven models. We leave these questions for future work.

Our work contributes to a growing body of research where goal-driven deep learning models (Yamins and DiCarlo, 2016) have shown unprecedented predictive performance of higher areas of the visual stream (Cadieu et al., 2014; Yamins et al., 2014), and a hierarchical correspondence between deep networks and the ventral stream (Güçlü and van Gerven, 2015; Cichy et al., 2016). Studies based on fMRI have

established a correspondence between early layers of CNNs trained on object recognition and V1 (Güçlü and van Gerven, 2015; Kriegeskorte, 2015), somewhat in contrast to our findings that intermediate layers performed best. However, these studies are based on fMRI data, which is an average of many neurons' responses and therefore possibly more linear than individual neurons' responses. Moreover, they used a shallower and less well performing CNN (AlexNet, Krizhevsky et al. 2012), which has larger, Gabor-shaped receptive fields in its early layers.

Interestingly, the features of multiple VGG layers performed similarly well, with only a shallow peak at layer conv3_1 (Fig. 4). This result is to be expected, as it has been observed that in deep neural networks the features of subsequent convolutional layers are highly redundant. That is, one can predict the feature maps in any given layer very well by those of a previous layer. More recent state-of-the-art architectures for object recognition avoid this type of redundancy by enabling 'shortcut' connections that skip layers. These skip connections encourage each layer to extract 'new' information instead of mainly carrying along information that has already been extracted. Recent examples for such architectures are Densely Connected CNNs (Huang et al., 2016), and the residual paths of Residual Networks (He et al., 2016). Some of these novel architectures hold more similarities with known cortical circuitry. They may be exploited in the future to extract features for neural system identification in the same way and could potentially be more interpretable.

Although deep models capture nonlinearities that go beyond complex cells, they lack a minimalistic description that could be meaningfully linked to biology. However, their success over other models makes them good candidates for an *in-silico* investigation. The advantage of having a good predictive *in-silico* model is that, unlike a real brain, one can probe it with completely arbitrary stimuli and run unlimited experiments. Thus, we argue that the chances of finding simple descriptions of the nonlinear computations performed by the brain are much larger when probing a highly predictive model than when measuring activity in the brain directly without a predictive model.

Acknowledgments

Research reported in this publication was supported by the German Research Foundation (DFG) grant EC 479/1-1 to A.S.E; the Bernstein Center for Computational Neuroscience (FKZ 01GQ1002); the German Excellency Initiative through the Centre for Integrative Neuroscience Tübingen (EXC307); the National Eye Institute of the National Institutes of Health under Award Numbers R01EY026927 (A.S.T.), DP1 EY023176 (A.S.T.), and NIH-Pioneer award DP1-OD008301 (A.S.T). The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health. This research was also supported by NEI/NIH Core Grant for Vision Research (EY-002520-37), NEI training grant T32EY00700140 (G.H.D) and F30EY025510 (E.Y.W.). L.A.G was supported by German National Academic Foundation. This research was also supported by Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior/Interior Business Center (DoI/IBC) contract number D16PC00003. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoI/IBC, or the U.S. Government. The authors have no conflicts of interest to report.

References

Adelson EH, Bergen JR (1985) Spatiotemporal energy models for the perception of motion. *JOSA A* 2:284–299.

Antolík J, Hofer SB, Bednar JA, Mrsic-Flogel TD (2016) Model constrained by visual hierarchy improves prediction of neural responses to natural scenes. *PLOS Comput Biol* 12:e1004927.

Batty E, Merel J, Brackbill N, Heitman A, Sher A, Litke A, Chichilnisky E, Paninski L (2016) Multilayer recurrent network models of primate retinal ganglion cell responses .

Cadieu CF, Hong H, Yamins DL, Pinto N, Ardila D, Solomon EA, Majaj NJ, DiCarlo JJ (2014) Deep neural networks rival the representation of primate it cortex for core visual object recognition. *PLoS Comput Biol* 10:e1003963.

Calabrese A, Paninski L (2011) Kalman filter mixture model for spike sorting of non-stationary data. *Journal of neuroscience methods* 196:159–169.

Carandini M, Demb JB, Mante V, Tolhurst DJ, Dan Y, Olshausen BA, Gallant JL, Rust NC (2005) Do we know what the early visual system does? *The Journal of neuroscience* 25:10577–10597.

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

Clevert DA, Unterthiner T, Hochreiter S (2015) Fast and accurate deep network learning by exponential linear units (elus). *arXiv preprint arXiv:1511.07289* .

Denfield GH, Ecker AS, Shinn TJ, Bethge M, Tolias AS (2017) Attentional fluctuations induce shared variability in macaque primary visual cortex. *bioRxiv* p. 189282.

Donahue J, Jia Y, Vinyals O, Hoffman J, Zhang N, Tzeng E, Darrell T (2014) Decaf: A deep convolutional activation feature for generic visual recognition In *International conference on machine learning*, pp. 647–655.

Ecker AS, Berens P, Cotton RJ, Subramaniyan M, Denfield GH, Cadwell CR, Smirnakis SM, Bethge M, Tolias AS (2014) State dependence of noise correlations in macaque primary visual cortex. *Neuron* 82:235–248.

Ecker AS, Berens P, Keliris GA, Bethge M, Logothetis NK, Tolias AS (2010) Decorrelated neuronal firing in cortical microcircuits. *science* 327:584–587.

Freeman J, Ziemba CM, Heeger DJ, Simoncelli EP, Movshon JA (2013) A functional and perceptual signature of the second visual area in primates. *Nature neuroscience* 16:974–981.

Gatys L, Ecker AS, Bethge M (2015) Texture synthesis using convolutional neural networks In *Advances in Neural Information Processing Systems*, pp. 262–270.

Gatys LA, Ecker AS, Bethge M (2016) Image style transfer using convolutional neural networks In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2414–2423.

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

He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778.

Heeger DJ (1992) Half-squaring in responses of cat striate cells. *Visual neuroscience* 9:427–443.

Huang G, Liu Z, Weinberger KQ, van der Maaten L (2016) Densely connected convolutional networks. *arXiv preprint arXiv:1608.06993* .

Hubel DH, Wiesel TN (1959) Receptive fields of single neurones in the cat's striate cortex. *The Journal of physiology* 148:574–591.

Hubel DH, Wiesel TN (1968) Receptive fields and functional architecture of monkey striate cortex. *The Journal of physiology* 195:215–243.

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

Kindel WF, Christensen ED, Zylberberg J (2017) Using deep learning to reveal the neural code for images in primary visual cortex. *arXiv preprint arXiv:1706.06208* .

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

Klindt D, Ecker AS, Euler T, Bethge M (2017) Neural system identification for large populations separating "what" and "where" In *Advances in Neural Information Processing Systems (accepted)*.

Kriegeskorte N (2015) Deep neural networks: a new framework for modeling biological vision and brain information processing. *Annual Review of Vision Science* 1:417–446.

Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks In *Advances in neural information processing systems*, pp. 1097–1105.

Kümmerer M, Theis L, Bethge M (2014) Deep gaze i: Boosting saliency prediction with feature maps trained on imagenet. *arXiv preprint arXiv:1411.1045* .

McIntosh L, Maheswaranathan N, Nayebi A, Ganguli S, Baccus S (2016) Deep learning models of the retinal response to natural scenes In *Advances in Neural Information Processing Systems*, pp. 1369–1377.

Olshausen BA, Field DJ (2005) How close are we to understanding v1? *Neural computation* 17:1665–1699.

Riesenhuber M, Poggio T (1999) Hierarchical models of object recognition in cortex. *Nature neuroscience* 2:1019–1025.

Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, Huang Z, Karpathy A, Khosla A, Bernstein M et al. (2015) Imagenet large scale visual recognition challenge. *International Journal of Computer Vision* 115:211–252.

Rust NC, Schwartz O, Movshon JA, Simoncelli EP (2005) Spatiotemporal elements of macaque v1 receptive fields. *Neuron* 46:945–956.

Shan KQ, Lubenov EV, Siapas AG (2017) Model-based spike sorting with a mixture of drifting t-distributions. *bioRxiv* p. 109850.

Simoncelli EP, Paninski L, Pillow J, Schwartz O (2004) Characterization of neural responses with stochastic stimuli. *The cognitive neurosciences* 3:327–338.

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

Talebi V, Baker CL (2012) Natural versus synthetic stimuli for estimating receptive field models: a comparison of predictive robustness. *The Journal of Neuroscience* 32:1560–1576.

418 Touryan J, Felsen G, Dan Y (2005) Spatial structure of complex cell receptive fields measured with natural images.
419 *Neuron* 45:781–791.

420 Vintch B, Movshon JA, Simoncelli EP (2015) A convolutional subunit model for neuronal responses in macaque
421 v1. *The Journal of Neuroscience* 35:14829–14841.

422 Willmore B, Prenger RJ, Wu MCK, Gallant JL (2008) The berkeley wavelet transform: a biologically inspired
423 orthogonal wavelet transform. *Neural computation* 20:1537–1564.

424 Willmore BD, Prenger RJ, Gallant JL (2010) Neural representation of natural images in visual area v2. *The*
425 *Journal of neuroscience* 30:2102–2114.

426 Yamins DL, DiCarlo JJ (2016) Using goal-driven deep learning models to understand sensory cortex. *Nature*
427 *neuroscience* 19:356–365.

428 Yamins DL, Hong H, Cadieu CF, Solomon EA, Seibert D, DiCarlo JJ (2014) Performance-optimized hier-
429 archical models predict neural responses in higher visual cortex. *Proceedings of the National Academy of*
430 *Sciences* 111:8619–8624.