Approximating the Architecture of Visual Cortex in a Convolutional Network

Bryan Tripp
bptripp@uwaterloo.ca
Department of Systems Design Engineering and Centre for Theoretical Neuroscience, University of Waterloo, Waterloo, ON N2L 3G1

Deep convolutional neural networks (CNNs) have certain structural, mechanistic, representational, and functional parallels with primate visual cortex and also many differences. However, perhaps some of the differences can be reconciled. This study develops a cortex-like CNN architecture, via (1) a loss function that quantifies the consistency of a CNN architecture with neural data from tract tracing, cell reconstruction, and electrophysiology studies; (2) a hyperparameter-optimization approach for reducing this loss, and (3) heuristics for organizing units into convolutional-layer grids. The optimized hyperparameters are consistent with neural data. The cortex-like architecture differs from typical CNN architectures. In particular, it has longer skip connections, larger kernels and strides, and qualitatively different connection sparsity. Importantly, layers of the cortex-like network have one-to-one correspondences with cortical neuron populations. This should allow unambiguous comparison of model and brain representations in the future and, consequently, more precise measurement of progress toward more biologically realistic deep networks.

1 Introduction

Computational models can help to clarify how neural cell and circuit mechanisms contribute to behavior. However, while the brain's purpose is to orchestrate sophisticated interactions with complex environments, it is hard to develop models with comparable behavior. Despite having developed somewhat independently of neuroscience, deep networks are capable of more realistic behavior than other models. For example, such systems can extract a wide variety of ethologically relevant information from natural visual scenes (Krizhevsky, Sutskever, & Hinton, 2012; Žbontar & LeCun, 2016; He, Gkioxari, Dollar, & Girshick, 2017; Kheradpisheh, Ghodrati, & Ganjtabesh, 2016). When combined with reinforcement learning, they can also interact successfully with visually complex environments (Mnih et al., 2015). So, importantly, from a functional perspective, deep networks are currently the most realistic brain models.

Neural Computation **31, 1551–1591** (2019)
doi:10.1162/neco_a_01211
© 2019 Massachusetts Institute of Technology

Deep networks, however are mechanistically quite different from the brain. Deep convolutional neural networks (CNNs) are typically little more than hierarchies of linear-nonlinear units, with fairly straightforward feedforward structures. This limits their usefulness as brain models. For example, they cannot be used to study the role of contour integration in naturalistic environments if they lack contour integration. Such mechanistic differences are also a likely source of differences in behavior (Karpathy, 2014; Nayebi & Ganguli, 2017; Lake, Ullman, Tenenbaum, & Gershman, 2016; Hassabis, Kumaran, Summerfield, & Botvinick, 2017; Rajalingham et al., 2018).

A related limitation is that existing deep networks have different architectures than cortical networks do, including different populations and connections. This prevents the use of deep networks to study the functional roles of specific brain areas. Architectural differences also impede the comparison of representations in deep networks and the brain. Deep networks account for a surprising amount of detail in neural representations. In particular, CNNs trained for object recognition are good linear predictors of neural activity at multiple points in the ventral visual stream (Yamins et al., 2014; Khaligh-Razavi & Kriegeskorte, 2014) and of functional magnetic resonance imaging data from large areas of human visual cortex (Güçlü & van Gerven, 2015; Eickenberg, Gramfort, Varoquaux, & Thirion, 2016; Seeliger et al., 2017; Shi, Wen, Zhang, Han, & Liu, 2017; Wen, Shi, Chen, & Liu, 2017). They also represent category-orthogonal information similar to primate inferotemporal cortex (Hong, Yamins, Majaj, & DiCarlo, 2016). There are also differences between deep networks and the ventral stream, including differences in representational similarity (Cadieu et al., 2014), response sparsity (Dong, Liu, & Hu, 2017), and other response statistics (Tripp, 2017), as well as dynamics (Tamura & Tanaka, 2001; Brincat & Connor, 2006; Issa, Cadieu, & DiCarlo, 2018). However, because layers of existing deep networks do not have one-to-one analogies with biological neural populations, it is not clear which layers should be compared to which groups of real neurons. This could obscure effects of network mechanisms (e.g., local response normalization) in producing more or less realistic representations. Also, since representations depend on position in the network, architectural differences may impose a ceiling on the representational similarity of artificial and biological networks.

This issue is addressed here by developing a data-driven convolutional architecture based on the primate visual cortex. Hyperparameters of convolutional neural networks (CNNs) are optimized to fit neurobiological data, including data from tract tracing, cell reconstruction, and electrophysiology studies. It is found that hyperparameters can be chosen to closely match neurophysiological data. However, the resulting architecture is qualitatively distinct from current widely used CNN architectures. In particular, it has longer skip connections; wider ranges of kernel sizes, strides, and connection densities; and qualitative differences in sparsity. The code used in this study is available at github.com/bptripp/calc.

2 Methods

CNN architectures were optimized for similarity with the visual cortex of macaque monkeys. Because standard CNNs are feedforward, only feedforward connections were included, according to the hierarchy of Felleman and Van Essen (1991). These feedforward architectures should be elaborated with lateral and feedback connections in the future, but there is no standard way to do this. Multiple approaches, (Tompson, Jain, LeCun, & Bregler, 2014; Rubin, Hooser, & Miller, 2015; Lotter, Kreiman, & Cox, 2017; Nayebi et al., 2018) could potentially be integrated and compared to the basic feedforward model developed here. The architectures included separate CNN layers corresponding to cortical layers L2/3, L4, L5, and L6. L1 was omitted because it has few excitatory cells. Feedforward interarea connections in the model originated from L2/3, L5, and L6 and terminated on neurons in L4 (Felleman & Van Essen, 1991). Within each area, interlaminar connections from L4 to L2/3 and L5, L2/3 to L5, and L5 to L6 were included (see Figure 1).

2.1 Quantification of Similarity to Cortical Architecture. A loss function was developed to clearly quantify the inconsistency between a given CNN architecture and measurable cortical properties. Optimizing the hyperparameters to minimize this loss function improves the consistency of the CNN with neural data. This is a flexible approach that allows for missing data, constraints, and priors on the hyperparameters and the potential addition of further data sets in the future.

Cortical properties were estimated from data in the neuroscience literature (see section 2.4 for details of these estimates); these are written below with a tilde overtop. Associated with the ith layer in the model (corresponding to a specific population in the brain) are $\tilde{n}^i$ (number of neurons in the layer), $\tilde{e}^i$ (number of extrinsic inputs per neuron, i.e. inputs from other brain areas), and $\tilde{w}^i_{RF}$ (the receptive field width in degrees visual angle). For interarea connections, associated with the connection from layer j to layer i is $\tilde{f}^{ij}$ (the fraction of all neurons projecting to layer i that are from layer j). A related property that is not directly estimated is $\tilde{n}^{ij}$, or the number of presynaptic neurons that contribute to the (ij)th connection. For interlaminar connections, associated with the connection from layer j to layer i is $\tilde{b}^{ij}$, the mean number of inputs from layer j that converge onto each postsynaptic neuron.

A convolutional neural network (CNN) has analogous properties, which are written using the same variable names but without the tilde. For example, the actual number of units in the network's ith layer is denoted n^i, and the goal is to make $n^i \approx \tilde{n}^i$. These parameters cannot be set directly because

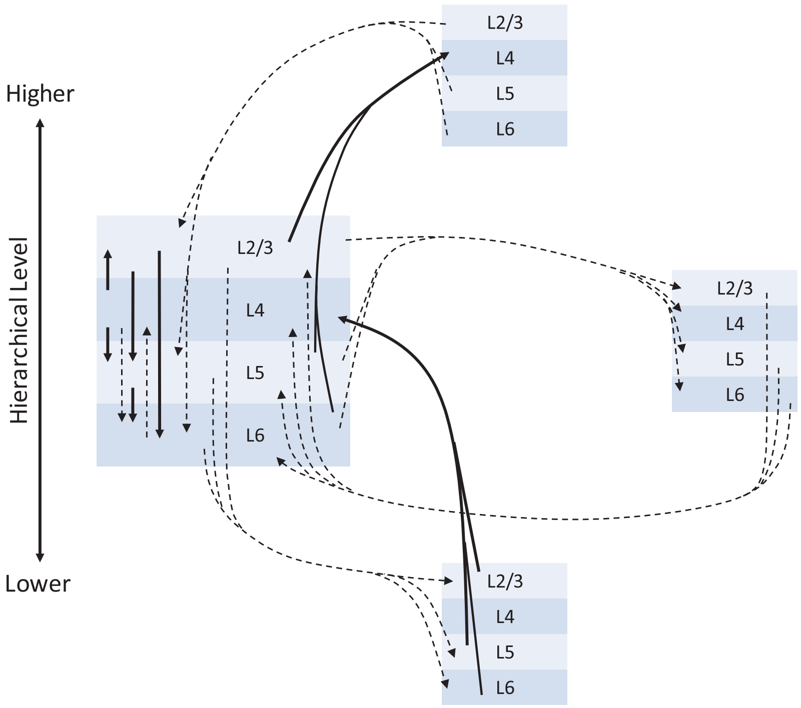

Figure 1: Sketch of a generic visual area (large box) and its relationship with other visual areas that are lower, higher, and at the same hierarchical level (smaller boxes). The model includes the feedforward connections that are indicated with solid arrows and omits feedback and lateral connections (dashed arrows). Interlaminar connections from L4 to L2/3 and L5 are counted among the feedforward ones because feedforward input arrives mainly in L4 and leaves via other layers. Ascending interlaminar connections from deep layers were omitted because they appear to be modulatory. They have broad terminations that cross functional boundaries (Callaway, 2004) and tend to target inhibitory interneurons (Thomson & Bannister, 2003). The connections sketched in this diagram are based on Felleman and Van Essen (1991) and Binzegger, Douglas, and Martin (2004).

they are functions of overlapping sets of tunable hyperparameters. Rather, the tunable hyperparameters are optimized to minimize the loss,

$$C = \sum_i \kappa_n \log^2 \frac{n^i}{\tilde{n}^i} + \sum_i \kappa_w \log^2 \frac{w^i_{RF}}{\tilde{w}^i_{RF}} + \sum_i \kappa_e \log^2 \frac{e^i}{\tilde{e}^i} + \sum_{ij \in A} \kappa_f \log^2 \frac{f^{ij}}{\tilde{f}^{ij}} + \sum_{ij \in L} \kappa_b \log^2 \frac{b^{ij}}{\tilde{b}^{ij}} + C_{constraints}, \tag{2.1}$$

where κ are importance weights, the sets A and L are interarea and interlaminar connections, respectively, and $C_{constraints}$ is a placeholder for additional cost terms such as soft constraints on the parameters. For example, sometimes a term was added to penalize the total number of parameters in the network, because numbers of parameters relate to memory requirements, run time, and overfitting. However, none of the results presented later in this letter incorporated this term. The $\log^2$ terms are symmetric in the ratio of actual versus ideal values (e.g., two times too big is as bad as two times too small). Estimates were lacking for some physiological receptive field sizes, in which case the corresponding terms were omitted from the sums (see the details in section 2.4). To obtain the results presented later, the κ terms were all set to one over the number of items in the corresponding sum.

2.1.1 Relating Cortical Properties to CNN Hyperparameters. The above measurable properties of the cortex do not correspond directly to hyperparameters of CNNs. For example, kernel size affects receptive field size but is not the same thing.

There are hyperparameters associated with each layer and connection of a CNN. Layer hyperparameters are m^i (number of feature maps, or channels, in convolutional layer i), and w^i (width of layer i in pixels; the height is assumed to be the same as the width). Standard connection hyperparameters include s^{ij} (the stride of the connection) and w_K^{ij} (kernel width). Both the i and j indices are needed for these parameters because stride and kernel width are not necessarily the same for different connections into a given layer. Two nonstandard connection parameters were used: c^{ij} (the fraction of feature maps in layer j that contribute to the ijth connection) and σ^{ij} (a pixel-wise sparsity parameter, the fraction of nonzero kernel elements in the channels that are not fully zeroed due to c^{ij}). The parameter c^{ij} is related to Scardapane, Comminiello, Hussain, & Uncini (2017); however, that work used group-sparsity regularization to encourage all of a unit's output weights to go to zero so that it could be removed from the network. In contrast, layers here typically have a number of targets, so eliminating a unit from one connection does not eliminate it from the network.

Network hyperparameters are related to physiological properties in part through

$$n^i = m^i (w^i)^2, \tag{2.2}$$

$$e^i = \sum_{j \in I_i} (w_k^{ij})^2 c^{ij} m^j \sigma^{ij}, \tag{2.3}$$

where I_i is the set of inputs to layer i. The expression for f^{ij} is more complex, so it is developed in the appendix.

Additionally, certain properties of a layer depend on properties of its incoming connections. Such relationships are straightforward in networks with a sequential feedforward structure; however, the brain has a high degree of convergence from multiple origin areas onto each target, raising the problem of consistency across these connections. Two such properties are the map width w^i (the number of units along one dimension of a feature map) and the receptive field width w^i_{RF}. In a sequential feedforward network, the map width depends on the width of the input (in image pixels) and the strides of all previous layers. It also depends on how edges are treated in the convolutions, but for simplicity, it is assumed here that edges are padded to prevent changes in resolution due to edge effects (i.e., "same" padding in Matlab terminology). In a network that lacks convergence of multiple connections onto a layer, $w^i = w^j/s^{ij}$, so the parameters w^i and s^{ij} are redundant with each other. In a network where multiple layers ($j \in I_i$) provide input to layer i, w^j and s^{ij} may be different for different inputs. However, their ratios should all be consistent with the same value of w^i. A procedure for ensuring this is described in section 2.3.

The receptive fields sizes, w^i_{RF}, could also potentially be inconsistent when calculated along different converging paths. To avoid this, w^i_{RF} are treated as hyperparameters, and w^{ij}_K are derived from them. The full receptive field grows linearly with depth. However, the edges of a unit's receptive field tend to exert a weak influence on its activity. This work considers instead the effective receptive field, which tends to be approximately gaussian in shape and to grow more slowly than the full receptive field (Luo, Li, Urtasun, & Zemel, 2016). If the stride is 1 and the kernel elements are statistically uniform, the variance $(\sigma^i)^2$ of the gaussian postsynaptic receptive field (RF) is the sum of variances of the presynaptic receptive field, $(\sigma^j)^2$, and the kernel,

$$(\sigma^i)^2 = (\sigma^j)^2 + \frac{1}{w^{ij}_K} \sum_{x=-(w^{ij}_K-1)/2}^{(w^{ij}_K-1)/2} x^2. \tag{2.4}$$

The units are layer j pixels. To facilitate optimization, the sum is approximated as an integral:

$$(\sigma^i)^2 \approx (\sigma^j)^2 + 2/w^{ij}_K \int_{x=0}^{w^{ij}_K/2} x^2 dx = (\sigma^j)^2 + (w^{ij}_K)^2/12. \tag{2.5}$$

The RF size in degrees visual angle is (accounting for strides in earlier connections)

$$w^j_{RF} = \sigma^j[w^p_{RF}(w^0/w^j)], \tag{2.6}$$

Table 1: Summary of the Cortical Population Properties Considered in This Study, with Key Sources from the Neuroscience Literature and Equations for Analogous Properties in CNNs.

Cortical Property	Neuroscience Sources	CNN Equivalent
$\tilde{n}^i$ (number of neurons)	Schmidt, Bakker, Hilgetag, Diesmann, and van Albada (2018) Garcia-Marin, Kelly, and Hawken (2017)	$n^i = m^i(w^i)^2$
$\tilde{e}^i$ (number of extrinsic inputs per neuron; inputs from other brain areas)	Schmidt et al. (2018) Garcia-Marin et al. (2017)	$e^i = \sum_{j \in I_i} (w_k^{ij})^2 c^{ij} m^j \sigma^{ij}$
$\tilde{w}_{RF}^i$ (receptive field width in degrees visual angle)	See Table 3.	$w_{RF}^j = \sigma^j[w_{RF}^p(w^0/w^j)]$ $(\sigma^i)^2 = (\sigma^j)^2 + 2/w_K^{ij} \int_{x=0}^{w_K^{ij}/2} x^2 dx$

Notes: The index i on layer properties indicates the ith layer. The indices j and i on connection properties indicate the presynaptic and postsynaptic layers, respectively. The CNN hyperparameters are: m^i (number of channels); w^i (width of layer i in pixels); s^{ij} (stride); w_K^{ij} (kernel width); c^{ij} (fraction of channels in layer j that contribute to the ijth connection); and σ^{ij} (pixel-wise kernel sparsity). w_0 is the width of the input image.

where w_{RF}^p is the size of a single image pixel in degrees visual angle (a property of the camera) and w^0 is the width of the input image (in pixels). The expression in square brackets is the size of a layer j pixel in degrees of visual angle. w_K^{ij} can therefore be found given pre- and postsynaptic receptive field widths.

Tables 1 and 2 summarize the cortical architectural properties considered here, their definitions in terms of CNN hyperparameters, and the sources in the neuroscience literature that were used to estimate them (discussed further in section 2.4).

2.1.2 Soft Constraints. Further loss terms were added to avoid underuse or overuse of individual feature maps. Specifically, $C_{constraint}$ included a term $(f_j^w - \sum_{ij \in A_j} c^{ij})^2$, where f_j^w is an estimated fraction of excitatory neurons with axons that enter the white matter for each j among the L2/3, L5, and L6 layers (sources of interarea connections), and A_j is the set of interarea connections that originate in j. For L5 and L6, $f_j^w = 1$. This encouraged the use of each feature map approximately once in an outgoing connection. Among projection neurons, different pyramidal neurons tend to project to different places (Hübener, Schwarz, & Bolz, 1990; Lur, Vinck, Tang, Cardin, & Higley, 2016) with little branching observed in anterograde tracer studies (Rockland, 2013) and little double labeling observed in retrograde tracer studies

Table 2: Summary of Cortical Connection Properties Considered Here.

Cortical Property	Neuroscience Sources	CNN Equivalent
$\tilde{f}^{ij}$ (fraction of all neurons projecting to layer i that are from layer j; for interarea connections)	Markov et al. (2014) Bakker, Wachtler, & Diesmann (2012) Schmidt et al. (2018)	$f^{ij} = \frac{n^{ij}}{\sum_{j \in I_i} n^{ij}}$ $n^{ij} = n^j c^{ij} \sigma_*^{ij} \alpha^{ij}$ $\alpha^{ij} = \begin{cases} 1, & \text{if } s^{ij} < w_K^{ij} \\ (w_K^{ij}/s^{ij})^2, & \text{otherwise} \end{cases}$ $\sigma_*^{ij} = 1 - (1 - \sigma^{ij})^{\beta^{ij} m^i}$ $\beta^{ij} = \begin{cases} (w_K^{ij}/s^{ij})^2, & \text{if } s^{ij} < w_K^{ij} \\ 1, & \text{otherwise} \end{cases}$
$\tilde{b}^{ij}$ (mean number of inputs from layer j that converge onto each postsynaptic neuron; for inter-laminar connections)	Schmidt et al. (2018) Fares and Stepanyants (2009)	$b^{ij} = (w_K^{ij})^2 m^j c^{ij} \sigma^{ij}$

Notes: The organization is the same as Table 1. The expression for f^{ij} is explained in the appendix.

(Bullier, Kennedy, & Salinger, 1984). Many pyramidal cells in these cortical layers do not contribute to feedforward interarea connections, but nonprojecting neurons in L5 and L6 were omitted from the model, as described in section 2.4. For L2/3, $f_j^w = 0.5$ (Callaway & Wiser, 1996). $C_{constraint}$ also included a term $(1 - \sum_{ij \in L_j} c^{ij})^2$ for j among the L2/3, L4, and L5 layers (sources of interlaminar connections), where L_j is the set of inter-laminar connections that originate in j.

2.2 Quantization of Receptive Field Variations. In a CNN layer, receptive field centers and kernel variations are quantized, and units are organized on a grid of three dimensions (vertical center, horizontal center, and feature map). A population of size n must be divided somehow among these dimensions. For simplicity, the height of each feature map was made equal to the width, w, so that $n = w^2 m$, where m is the number of feature maps. There is no correct choice of m because biological neurons are not organized in a three-dimensional grid of discrete features and pixels. However, some choices may be more reasonable than others.

In V1, many receptive fields are similar to various Gabor functions, and corresponding convolutional-layer channels would encode different orientations and spatial frequencies. If there were too few channels, the network would be blind to certain orientations. But if there were an excess of channels and too few pixels spanning the scene, the network would be blind to edges at periodic retinotopic positions. To avoid these extremes, the

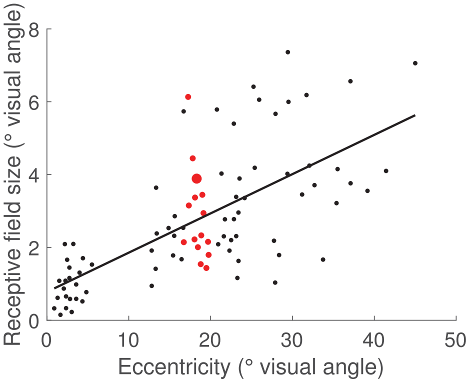

Figure 2: Scatter plot of V1 receptive field (RF) sizes versus eccentricity (data replotted from Gattass et al., 1981). The average coefficient of variation (CV) of RF sizes around a given eccentricity was estimated as 0.44. The red dots illustrate the neighborhoods used for this estimate. Around each RF (e.g., the large red dot), the CV was calculated over up to 10 other RFs (smaller red dots) with eccentricity no more than 20% larger or smaller. These CVs were averaged over the entire population.

heuristic adopted here is that receptive fields should overlap with their nearest neighbors about as much across channels as across space.

As a preliminary step in this calculation, the coefficient of variation (CV) of receptive-field sizes at a given eccentricity was estimated from data in Figure 13 of Gattass, Gross, and Sandell (1981; replotted here in Figure 2) as 0.44. From the same data set, it was estimated that half of the visual field is approximately 24 V1 receptive fields wide. (This is the integral of the inverse of the regression line, from 0 to 90 degrees.) Finally, the number of neurons in L2/3 of V1, in a single hemisphere, is estimated as 53,072,320 neurons, based on the density estimate of 47,386 neurons/mm^2 (Schmidt et al., 2018) and mean surface area of 1120 mm^2 (Felleman & Van Essen, 1991).

Correlations between neighboring receptive fields were then estimated using a recent model of receptive field variations across V1 (Goris, Simoncelli, & Movshon, 2015). For a given number of feature maps, m, the same number of random linear kernels was drawn from the model, using parameter distributions from Goris et al. (2015) and varying the receptive field size according to the estimated CV of 0.44 (above). Pairwise correlations were calculated between kernels and their nearest neighbors in visual space and feature space. To find a kernel's nearest neighbor in visual space, the kernel was shifted horizontally by a fraction of the receptive field size, $24/w$, where $w = \sqrt{n/m}$. Figure 3a shows examples of the nearest neighbors of a kernel in visual space and feature map space. Figure 3b shows how these

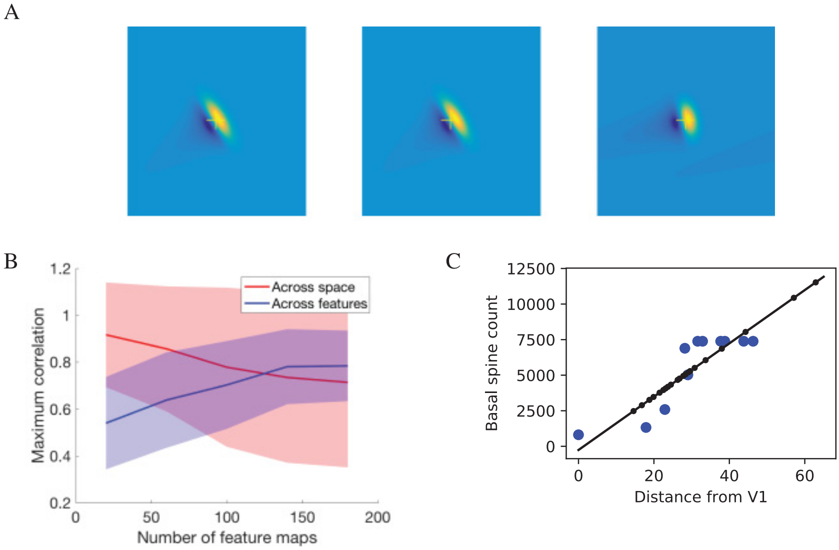

Figure 3: Heuristics for organizing neurons into grids. For V1, the number of feature maps was chosen so that differences between receptive fields would be about as large in feature space as in visual space. (a) The left panel shows an example of a model V1 receptive field kernel. The center panel shows the kernel of a nearest neighbor in visual space, that is, the same kernel shifted laterally. Its correlation with the left kernel is 0.91. The right panel shows the nearest neighbor in feature space, which has a correlation of 0.90. (b) Given a certain number of neurons, as the number of feature maps increases, the correlation between nearest neighbors in feature space increases, while the correlations in visual space decrease. The lines show means, and the colored areas show standard deviations. (c) Extrapolation to other areas was based on basal spine count. The large dots show spine counts from Elston (2007) versus distance from each area to V1 (Schmidt et al., 2018). The smaller dots are inferences for other areas, from the regression line.

correlations depend on the number of feature maps used to model V1 L2/3 in a random sample of V1-like receptive-field kernels. As the number of feature maps increases, neighbors in feature space get nearer, and neighbors in visual space get farther away because there are fewer pixels. Neighbors in feature and visual space are about equally close when there are about 125 feature maps. Over 10 groups of random kernels, this point occurred with 130 +/−6 feature maps (mean +/−standard deviation). The model of Goris et al. (2015) also includes a nonlinear component. If this is included, the result is 87 +/−5 feature maps. However, Cowley et al. (2016) found that the version of this model without the nonlinear component was more consistent with dimension reduction of recordings of V1 cells, so in this letter, L2/3 of V1 was taken to be most consistent with about 130 convolutional feature maps. However, given the large standard deviations, a wide range of choices around this value would also be fairly reasonable.

Another heuristic was needed to extrapolate this estimate outside V1. If a convolutional network were to include recurrent connections and if sparsity and kernel size were held constant, then the in-degree associated with these connections would vary linearly with the number of feature maps. The density of spines on basal dendrites in L2/3 was used as a correlate of in-degree of recurrent connections. These densities vary across visual areas by an order of magnitude (Elston, 2007). The number of feature maps in L2/3 of each area was set to be linearly proportional to these densities, interpolating unknown values according to the distance of each area from V1 (see Figure 3c). Together with the V1 model, this led to heuristics for the numbers of feature maps in each visual area.

2.3 Optimization of the Hyperparameters. Some CNN hyperparameters are integers, which makes their optimization an integer programming problem. Some of these integers, particularly the number of feature maps (m) and the kernel width (w_k), can be rounded after optimizing them as real numbers, without introducing large errors, allowing the use of gradient-based methods. However, strides cannot be handled in this way. This is partly because strides are often two or fewer, so rounding them has a relatively large effect. A more fundamental issue is that there are typically multiple paths from the input to a given layer. The product of strides should be the same along each of these paths; otherwise, different paths will produce input for inconsistent numbers of neurons. We cannot set the strides by first setting all the map widths to physiologically realistic values and then setting each stride to the ratio of presynaptic and postsynaptic map widths because this ratio will not be an integer generally. One way to resolve this inconsistency would be zero-pad lower-resolution inputs to a layer, but then each unit would receive a mixture of different features from different parts of the scene.

For this reason, only sets of hyperparameters with valid stride patterns were considered, meaning that the strides were integers, the product of strides along each path to a given layer was the same, and the product of strides along any path was no greater than the width of the input (so that each layer had at least one pixel). The integer programming problem is nonconvex because the error terms involve a difference between a target value and a product of stride parameters. This prevents the use of open source packages such as CVXPY (www.cvxpy.org). Instead, a heuristic method was used to generate a large number of valid stride patterns, and the one with the lowest cost was chosen.

Complicating this process, the ratio of invalid to valid stride patterns is exponential in the number of connections. The search was made more tractable by exploiting the fact that longer paths through the network tend

Algorithm 1: Efficient Generation of a Random Valid Stride Pattern.

```
Set all strides to NULL
while Some strides are NULL do
    Find longest path, $p_L$, that only contains NULL strides.
    Find the cumulative stride $c_1$ of the first layer in the path.
    {Determine constraint $C$ on the product of strides along $p_L$, and estimate the geometric mean stride along the path, $\mu_s$}
    if $p_L$ ends at network output then
        $C : \prod_{i \in p_L} s_i <=$image size / $c_1$
        $\mu_s =$ (image size / $c_1)^{1/l}$, where $l$ is the number of steps in the path.
    else
        $C : \prod_{i \in p_L} s_i = c_{end}/c_1$
        $\mu_s = (c_{end}/c_1)^{1/l}$, where $l$ is the number of steps in the path.
    end if
    repeat
        Set strides along path to random integers $i$ from 1 to $2\lfloor \mu_s \rfloor$, with $P(i) \propto (.1 + |1.25 - i|)^{-1}$
    until Strides satisfy $C$
end while
```

to have smaller strides at each step. Algorithm 1 gives details of the process that was used to sample random valid stride patterns. This process included a heuristic probability distribution for individual strides ($P(i)$ in algorithm 1) that sampled small values more frequently. In a long path, many of the strides should be one. For example, if the input is 256 pixels wide and the output is to be at least 1 pixel wide, no more than eight strides in a path can be more than 1.

After generating many valid stride patterns (typically 1000), a single pattern was chosen based on consistency with heuristics for the numbers of feature maps in each visual area (discussed in the previous section).

With the stride pattern determined, certain other hyperparameters could then be calculated directly. For example, the resolution of layer i, w_i, is the resolution of the input divided by the product of strides along any path to i. The number of feature maps is then $m_i = \left[\tilde{n}_i / w_i^2\right]$, rounded to the nearest integer. For a given connection, if both the presynaptic and postsynaptic receptive field sizes were known, the kernel size of the connection could also be calculated directly. However, this did not occur in the current work, because receptive field sizes were specified only for L2/3 (see section 2.4). Remaining hyperparameters were optimized in TensorFlow (Abadi et al., 2016), with the Adam algorithm (Kingma & Ba, 2014). This is a generic approach that works with or without various priors and hard or soft constraints and would be compatible with incorporation of other data sets in the future. Also, as shown below, this approach produces good fits to the data.

Kernel widths are among the integer hyperparameters, but these values were not rounded after optimization. The rationale was that real-valued kernel sizes can be approximated by rounding up and increasing sparsity at the edges. For example, a kernel size of 4.1 and sparsity σ can be approximated by a kernel of size five, with sparsity σ throughout the center and 0.55σ on the edges, producing an average of $(3 + 2(.55))\sigma = 4.1\sigma$ nonzero weights in each row.

2.4 Estimates of Physiological Properties. Schmidt et al. (2018) recently combined many data sources from the literature to estimate the network structure of macaque visual cortex, using the FV91 (Felleman & Van Essen, 1991) cortical parcellation. Many of their estimates are adopted here. Readers are referred to their paper for a thorough description of their process and the many decisions involved.

2.4.1 Numbers of Neurons. The number of neurons in each layer and area (e.g., L2/3 of V1) was estimated from the corresponding cortical surface area (in mm^2) times the neuron density of the layer (in neurons/mm^2). Both estimates were taken from Schmidt et al. (2018). Numbers of neurons in subdivisions of V1 layers (e.g., L4Cα) were estimated by dividing laminar totals from Schmidt et al. (2018) according to ratios in Garcia-Marin et al. (2017).

CNN units correspond most closely to excitatory neurons. CNN units are actually neither excitatory nor inhibitory, because the kernels can take on both positive and negative values. However, a network with such mixing of positive and negative weights can be transformed into a more physiologically realistic structure (with distinct excitatory and inhibitory units) in a way that barely affects the effective connections (Parisien, Anderson, & Eliasmith, 2008). This is done by shifting the weights in each connection until they are all positive and then adding a parallel two-synapse pathway through a new population of inhibitory neurons. (See also Tripp & Eliasmith, 2016, for additional results on the stability of recurrent networks in this scheme.) The original units become excitatory after this transform. The Parisien transform was not used in my study, as it increases computational requirements with little effect on function. However, reflecting this well-defined correspondence between excitatory neurons and mixed artificial units, targets for the numbers of units in each layer were based on excitatory neuron densities. Consistent with this interpretation, most of the convolutional units in the current model project to other cortical areas (analogous to excitatory pyramidal neurons).

The model includes only feedforward connections, so an effort was made to exclude neurons that are involved mainly in feedback. Both supragranular and infragranular layers participate in feedforward and feedback connections with other areas, although infragranular layers are a more frequent source of feedback (Felleman & Van Essen, 1991). Cortical layer L6 sends dense (Binzegger et al., 2004) modulatory (Callaway, 2004) feedback to L4, as well as the thalamus (Briggs, 2010). Different cortico-cortical cells in L5 project to different cortical areas, both higher and lower in the visual hierarchy (Kim, Juavinett, Kyubwa, Jacobs, & Callaway, 2015). To focus on neurons that contribute to feedforward cortico-cortical connections, population sizes of L5 and L6 were reduced. The L5 feedforward cortico-cortical population was estimated as 1/16 of the total L5 excitatory population. This was based on a reconstruction of 16 L5 pyramids in Callaway and Wiser (1996), of which only 3 reached the white matter, and on other work showing 3 distinct types of L5 projection neurons, only 1 of which is cortico-cortical (Lur et al., 2016). Similarly, the L6 feedforward cortico-cortical population was estimated as 0.15 of the total L6 excitatory population. This was based on Wiser and Callaway (1996), which found that 28% of L6 pyramids entered the white matter but about half of these were cortico-thalamic rather than cortico-cortical.

2.4.2 Interarea Connection Sparsity. Interarea connections in the model were based on retrograde tract-tracing data from a large, systematic study (Markov et al., 2014) that included tracer injections into 27 cortical areas. Markov et al. (2014) reported the fractions of labeled neurons found in each area extrinsic to the injection site (the FLNe). They also reported the percentage of supragranular neurons that contributed to each connection (%SLN). My model requires FLNe estimates for each feedforward interarea connection, including separate estimates for connections that originate in L2/3, L5, and L6. Supragranular and infragranular totals were obtained by multiplying the area-wise FLNe values by %SLN/100 and (1−%SLN/100), respectively. The supragranular total was assigned to L2/3. The infragranular total was divided between L5 and L6 on the basis of laminar connection strengths in CoCoMac (Bakker et al., 2012), where available, or otherwise divided evenly between L5 and L6. Laminar connections in CoCoMac can be marked as present, absent, or unknown or marked with qualitative strength labels of 1 to 3. If a laminar source was marked as present with unknown strength, it was assigned a strength of 2. The strength labels were then treated as if they were linear quantitative descriptions. For example, if L5 and L6 had strengths 2 and 3, respectively, then L5 and L6 were assigned 2/5 and 3/5 of the infragranular FLNe.

For the many visual areas that were not injected by Markov et al. (2014), Schmidt et al. (2018) included connections that are present in CoCoMac (Bakker et al., 2012). For these connections, they estimated FLNe and %SLN by regression from interarea distances and cell-density ratios, respectively, and these estimates are adopted here.

Markov et al. (2014) did not use the FV91 parcellation, so Schmidt et al. (2018) assigned FLNe to FV91 areas according to overlaps between the two parcellation schemes. For target areas that were not injected by Markov et al. (2014), they included only those connections present in CoCoMac. For areas injected by Markov et al. (2014), they included any connections that were produced by redistributing FLNe according to overlaps between areas in the two parcellation schemes. A limitation of this approach is that it seems to overestimate the number of interarea connections targeting areas injected by Markov et al. (2014). In the Markov et al. (2014) data, 66% of possible connections exist, but mapping to the FV91 parcellation leads to 97% connectivity into areas injected by Markov et al. (2014). Diverging from Schmidt et al. (2018), a more conservative estimate was obtained here by redistributing FLNe only among connections that exist in the CoCoMac database (for target areas injected by Markov et al., 2014, as well as other target areas). This approach probably misses some connections, as Markov et al. (2014) reported a number of newly found connections that are not present in CoCoMac. Furthermore, the number of connections is less important than the density; for example, a very weak connection is similar to a lack of connection. However, while the newly found connections in Markov et al. (2014) were about 100 times as weak as previously known connections on average, the connections eliminated in the current approach were only about 5 times weaker than known connections. So the current approach probably misses some of the weakest connections (about 100 times weaker than average) while avoiding somewhat stronger spurious connections (about 5 times weaker than average).

2.4.3 Convergence onto Individual Neurons. Schmidt et al. (2018) produced detailed estimates of the average numbers of synapses onto each neuron, by layer and area, including both interlaminar and interarea connections. These estimates were adopted in the present work. Specifically, Schmidt et al. (2018) estimated the number of excitatory and inhibitory inputs from other visual areas, excitatory and inhibitory interlaminar inputs from within a 1 mm^2 patch around the target neuron, and total "external" inputs, x^{ext}, including inputs from nonvisual areas and from the same area but outside the 1 mm^2 patch. For my model, estimates of total excitatory interlaminar inputs were needed, including those inside and outside a 1 mm^2 patch. Because inputs from distant cortical areas are a small minority (Markov et al., 2014), it was assumed for simplicity that all the "external" inputs were interlaminar, whereas in reality a small fraction are from nonvisual areas (these were called type IV connections in Schmidt et al., 2018). It was also assumed that the ratio of excitatory over total inputs, $r^{ex/tot}$, was the same outside as inside the patch. The estimates of interlaminar connection densities within a 1 mm^2 patch, from Schmidt et al. (2018), were therefore multiplied by a gain, $(e^{patch} + r^{ex/tot} x^{ext})/e^{patch}$, where e^{patch} is the total within-patch excitatory input. This produced estimates of the total interlaminar connection densities, including connections from both inside and outside the patch.

Connections between pairs of neurons typically involve multiple synapses, with low variance (Fares & Stepanyants, 2009; Kasthuri et al., 2015; Song, Sjostrom, Reigl, Nelson, & Chklovskii, 2005). To approximate the number of functional inputs to a neuron rather than the number of physical synapses, Schmidt et al.'s (2018) synapse estimates were divided by the mean number of synapses per connection across cases in Fares and Stepanyants (2009), which was 4.47. In the projections from LGN to V1 Garcia-Marin et al. (2017) estimated a higher redundancy of 25 to 28 synapses per functional connection, with each L4 neuron receiving input from only 7 or 8 distinct LGN neurons. This estimate was used directly for the connection from LGN to V1.

2.4.4 Receptive Field Sizes. Classical receptive field (RF) sizes have been reported for many cortical areas. Within a cortical area, mean RF sizes usually vary with eccentricity, or distance from fovea to RF center (Gattass et al., 1981), whereas RF sizes in convolutional networks are uniform across the visual field, by construction. However, this distinction was deferred for simplicity. RF sizes were taken from five degrees eccentricity (an intermediate value). RF sizes vary somewhat by cortical layer (Gilbert, 1977), with deeper layers generally having larger receptive fields. Most reports of RF sizes do not identify the layer, however, in part because the layer can be uncertain during in vivo recording. Lacking layer-wise data for most visual areas, mean RFs reported in the literature were applied to layer 2/3, and other layers were omitted from this term in the loss function. RF sizes have not been thoroughly characterized in all areas. However, without constraining kernel structure, receptive fields can only get larger along feedforward paths, so specifying a few RF sizes throughout the network constrains the remaining ones. Table 3 lists the RF sizes used and the corresponding literature sources.

2.4.5 Limitations of Physiological Estimates. In general, the accuracy of these estimates is limited. For example, regression curves were used to interpolate unknown data on FLNe and %SLN in Schmidt et al. (2018), but the measured values have large scatter around these curves. For example, FLNe of connections with lengths of between 29 mm and 31 mm vary over five orders of magnitude (data from Figure 4C of Schmidt et al., 2018).

Table 3: Mean Receptive Field (RF) Sizes, in Degrees Visual Angle, at Five Degrees Eccentricity.

Area	RF Size	Source
V1	1.3	Gattass et al. (1981)
V2	2.2	Gattass et al. (1981)
V3	2.8	Gattass, Sousa, and Gross (1988) and Felleman and Van Essen (1987)
V4	4.8	Gattass et al. (1988) and Boussaoud et al. (1991)
MT	4.2	Komatsu and Wurtz (1988) and Maunsell and Van Essen (1987)
PO	7.7	Galletti et al. (1999)
MSTd	16.0	Komatsu & Wurtz (1988)
PITd, PITv	8.9	Boussaoud et al. (1991)
CITd, CITv	38.5	Boussaoud et al. (1991)

Notes: Where multiple references are given, estimates are the average of those in each reference. Some studies did not use the FV91 parcellation. Following Table 1 of Felleman and Van Essen (1991), V6 in Galletti, Fattori, Gamberini, and Kutz (1999) is interpreted as PO, and TEO in Boussaoud, Desimone, and Ungerleider (1991) is interpreted as both PITd and PITv. Boussaoud et al. (1991) group data for TE together, but the recording sites are mostly close to TEO (see their Figure 4), so this is interpreted as spanning CITd and CITv.

Furthermore, the measurements that underlie these regressions are also somewhat uncertain. The tracer injections did not uniformly fill the injected areas, so the measurements cannot account for heterogeneity within areas. There is also a substantial spread in published estimates of cell counts and synapse densities, for example, between Garcia-Marin et al. (2017) versus O'Kusky and Colonnier (1982). There is also variation among monkeys. For example, in Van Essen and Newsome (1984), the surface area of a single hemisphere of striate cortex varied from 690 mm to 1560 mm^2, in a group of six macaque monkeys. FLNe values often vary over an order of magnitude (+/− one standard deviation) across individuals (Markov et al., 2011). In short, a fairly wide range of parameters would be within the range of individual variations, but many of the estimates used here are probably outside this range. Such uncertainties are subtle, however, compared to large qualitative differences with standard CNNs that are shown later (see section 3.2).

2.5 Single-Hemisphere Model. The methods already described could be used to develop models of various subnetworks of the brain, such as the ventral visual stream in one hemisphere or all of the visual cortex in both hemispheres. Results that follow are reported for a specific single-hemisphere model, referred to here as the macaque single-hemisphere (MSH) model. This model included all 32 vision-related areas in the FV91 parcellation (Felleman & Van Essen, 1991). Most visual areas were divided into layers L2/3 to L6, with connections as described in section 2. The lateral geniculate nucleus (LGN) was modeled with 1,27,0000 neurons divided between parvocellular (89.7%) and magnocellular (10.3%) layers (Weber, Chen, Hubbard, & Kaufman, 2000), and an additional koniocellular layer the same size as the magnocellular layer. L2/3 of V1 was divided into blobs and interblobs. L4 of V1 was divided into L4B, L4Cα, and L4Cβ (L4A was omitted as it is quite thin). V2 was divided into three groups of layers, corresponding to thick stripes, thin stripes, and pale stripes. The thick stripes received magnocellular V1 input and sent output to dorsal areas. The thin and pale stripes received parvocellular V1 input and sent output to dorsal and ventral visual areas. L2/3 blobs in V1 also received input from L4Cα, consistent with magnocellular and parvocellular convergence onto blobs, and with contributions of both systems to activity in V4 (Merigan & Maunsell, 1993). These connections are based on a classical view of connectivity in early visual cortex (Livingstone & Hubel, 1988). More recent studies have revealed many further details, including finer-grained V1 populations and greater mixing of retinal output streams in V1 (reviewed by Nassi & Callaway, 2009). However, further work is needed to develop a more realistic quantitative model of the early vision network from this newer literature.

2.6 Convolutional Network Implementation. Training of the MSH network is deferred to future work. However, to verify that the approach produced fully specified and trainable networks, smaller subnetworks were implemented in Keras (Chollet, 2015) and trained on the CIFAR-10 image classification task (Krizhevsky, 2009). Each connection included only a subset of the channels in the presynaptic layer (according to the sparsity parameter c_{ij}), which required use of Keras's "concatenate" layer to combine the desired channels. For layers with multiple inputs, a separate "Conv2D" layer was created for each input, and their activations were summed using a "sum" layer, before entering a nonlinearity.

3 Results

This section presents outcomes of the optimization process and then compares the architecture of the macaque single-hemisphere (MSH) model (section 2.5) to those of several widely used convolutional networks.

3.1 Optimization Results. The first optimization step was to generate 1000 valid stride patterns (see section 2.3) and select the one that matched heuristic targets (see section 2.2) for the number of feature maps (or channels), m, in each layer most closely. The distance metric was the root mean squared log ratio of the actual versus target products of strides. (The target products of strides were calculated from n and the m targets.) Figure 4 shows a histogram of these distances over 1000 samples. This figure also shows the loss C (see section 2.1) during optimization with five different

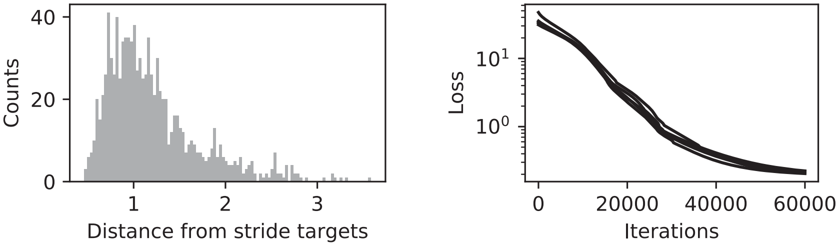

Figure 4: (Left) Histogram of distances from heuristics for numbers of feature maps in each visual area, over 1000 random stride patterns. (Right) Loss C over training iterations, with several different stride patterns. The loss is essentially independent of the stride pattern.

stride patterns. The stride pattern had little effect on C, which describes the consistency of the architecture with neurophysiological data. Rather, separately, different stride patterns varied in their agreement with heuristics for the numbers of feature maps (see section 2.2). Different stride patterns also led to different numbers of parameters in the network. This is because for given n, a larger stride means greater m everywhere downstream, and the number of parameters in a kernel is proportional to the product of presynaptic and postsynaptic m.

Figure 5 shows scatter plots of estimated physiological values, corresponding to terms in equation 2.1, versus corresponding values calculated from network hyperparameters after optimization. The scatter in FLNe is strongly correlated across independently optimized networks, even with different stride patterns ($r > .999$ between these two networks, plotted in panel E). The scatter seems to be due to the low sensitivity of FLNe to σ_{ij} values, combined with competition between the influences on c_{ij} of FLNe targets and $\sum_{ij} c_{ij}$ targets. FLNe differences from targets are moderate compared to uncertainty in FLNe due to incomplete data. Overall, most of the optimization errors are well within the range of individual variation. However, there are probably much larger errors in the target values themselves due to incomplete and uncertain data.

In the network with the best-fitting stride pattern (shown with triangle markers in Figure 5), there are approximately 40 million kernel weights (i.e., $\sum_{ij} m_i m_j w_{ij}^2 c_{ij} \sigma_{ij}$). This number is related to the capacity of the network. In a standard setting with graphical processing units, the weight-wise sparseness parameters σ_{ij} could be approximated in part using dilated convolutions (Yu & Koltun, 2015), although dilation reduces the FLNe only if the dilation step and the stride have a greatest common divisor greater than one. Somewhat more storage would be needed to approximate σ_{ij} values more closely by fixing some of the kernel entries at zero.

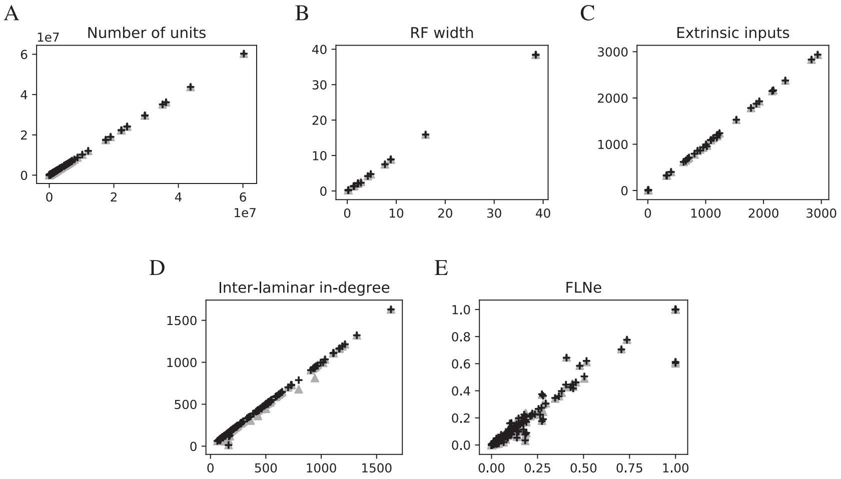

Figure 5: Optimized properties of MSH network (vertical axes) versus targets estimated from the neuroscience literature (horizontal axes). The different markers show two separate optimization results, with two different stride patterns. (A) Number of units per layer (n in section 2.1). (B) Receptive field widths (w_{rf}). (C) Number of inputs per unit from other visual areas (e). (D) In-degrees of interlaminar connections. (E) The connection sparsity measure FLNe.

3.2 Comparison with Standard Convolutional Networks. The architectures of optimized MSH models were compared with several standard and widely used convolutional networks, specifically VGG-16 (Simonyan & Zisserman, 2015), ResNet50 (He, Zhang, Ren, & Sun, 2016), InceptionV3 (Szegedy et al., 2015), and DenseNet121 (Huang, Liu, van der Maaten, & Weinberger, 2017).

Figures 6, 7, and 8 plot the connection sparsity measure FLNe (see section 2.4) for VGG-16, InceptionV3, and DenseNet121. Recall that in retrograde tracer studies, FLNe are the fractions of labeled neurons in each other cortical area (extrinsic to an injection site). This approximates the fraction of all neurons that project to the injected area, which have cell bodies in each other area. The connections in these convolutional networks involve all the units in the corresponding presynaptic layers, so the analogous values are simply the number of presynaptic units in each connection divided by the total units presynaptic to the target layer. These values are also called FLNe here. VGG-16 has a simple sequential structure. InceptionV3 and DenseNet121 have "skip connections," resulting in greater convergence onto certain layers. In fact, the key innovation of DenseNet is to include all possible skip connections. However, in deeper versions of this network, this is done in blocks, with no skip connections between blocks (Huang et al., 2017).

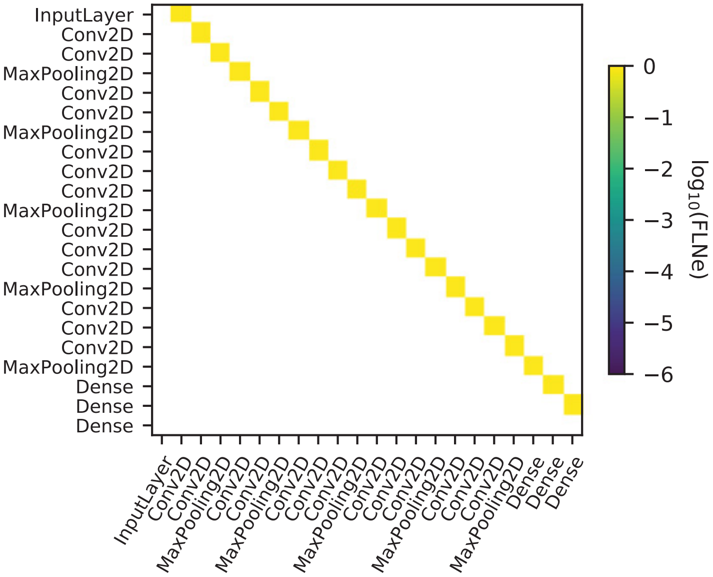

Figure 6: Connections in a widely used CNN, VGG-16 (Simonyan & Zisserman, 2015). Presynaptic layers are on the vertical axis, and postsynaptic layers are on the horizontal axis. Colors indicate the fractions of presynaptic neurons that contribute to each connection. This corresponds to a measure used in retrograde tracer studies (Markov et al., 2014), the fraction of labeled neurons extrinsic to the injection site (FLNe). In this network, all of the log-FLNe values are 0, meaning that each layer gets all of its input from one other layer, with no convergence.

Figure 9 plots the FLNe of a macaque-optimized network. To show the structure of the whole network in a single plot, the FLNe values are divided across L2/3, L5, and L6 in the source areas, according to %SLN and laminar source strengths in CoCoMac. Corresponding values are also plotted for interlaminar connections based on the ratios of $n_i c_{ij}$ in each connection to their sum. The interlaminar connections are the dense connections close to the diagonal. The vertical banded structure of the off-diagonal elements is due to the convergence of interarea connections on L4, and the horizontal banding is due to interarea connections arising from L2/3, L5, and L6 (see also Figure 1). Overall, FLNe values vary over six orders of magnitude, consistent with Markov et al. (2014). The range of numbers of connections into L4 of different areas is wide, reflecting connections recorded in the CoCoMac database (Bakker et al., 2012). For example, 7a receives input from 16 other areas (half of the network), while CITd has incoming connections only from V4. Some connections may be missing from the database,

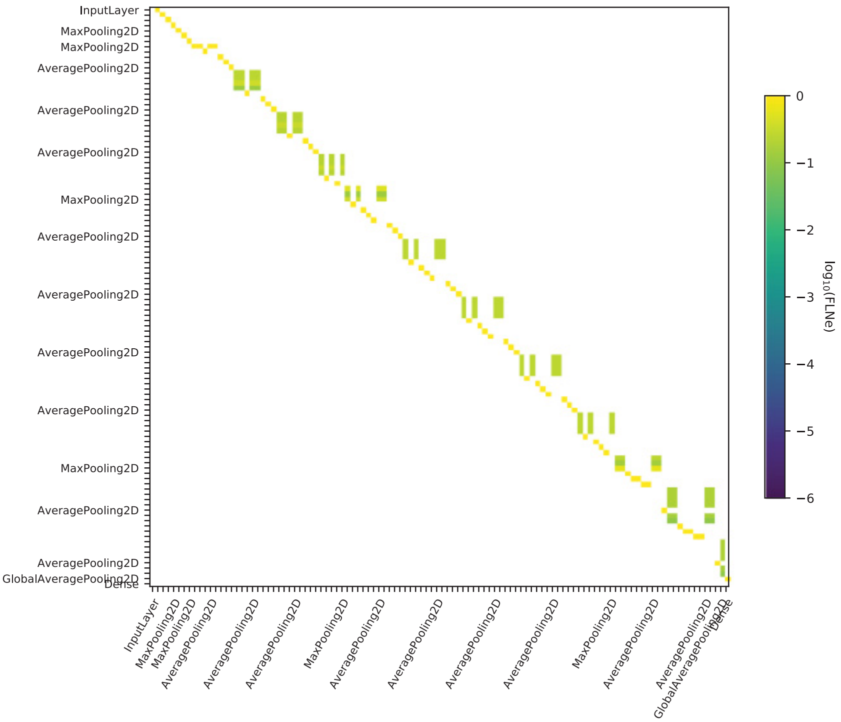

Figure 7: Connections in an Inception network (Szegedy et al., 2015). Conventions as in Figure 6. To reduce clutter, labels are not shown for the convolutional layers.

however (Markov et al., 2014). The longest path through the network is from the input to L6 of area TH, and it has 39 steps. The greatest shortcut (skip connection) is a direct connection from V2 thick stripes' L2/3 to L4 of area 46. The longest path parallel to this direct connection has 30 steps.

Comparing Figures 6 to 9, the DenseNet arguably has the most cortex-like structure, with many long skip connections and a high degree of convergence onto some layers. However, DenseNet has several qualitative differences from the MSH model. First, the DenseNet has a single input and a single output. In contrast, the visual cortex culminates in several parallel high-level areas that are involved in different functions, such as visually guided navigation and grasping, as well as visual recognition (Kravitz, Saleem, Baker, & Mishkin, 2011). Second, the DenseNet has a regular block-wise pattern of skip connections and a lack of skip connections between blocks, whereas skip connections in the MSH model span up to three-quarters of the network depth. Third, the DenseNet's FLNe values vary over one order of magnitude, whereas those in the MSH model vary over

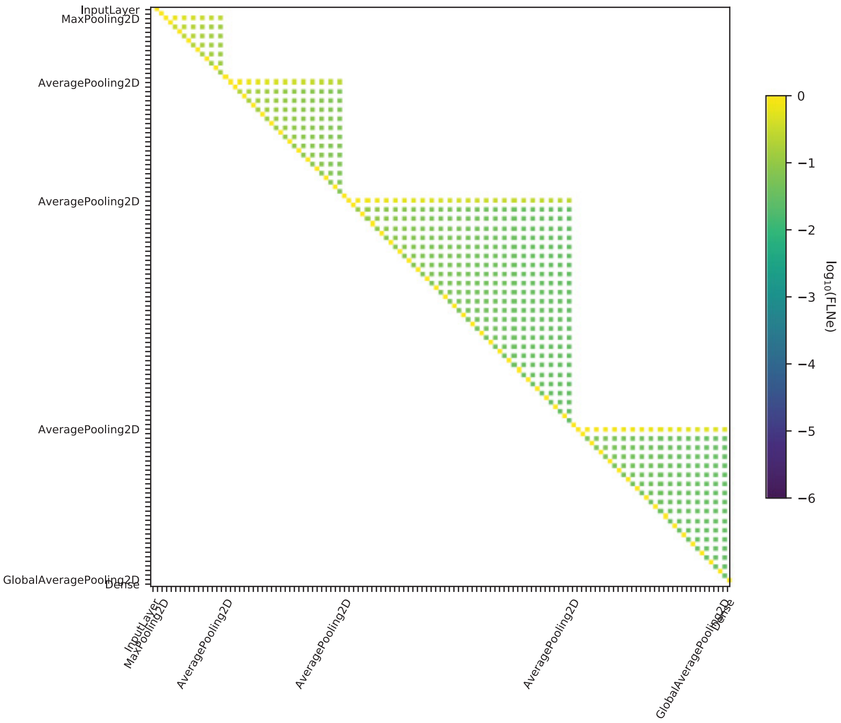

Figure 8: Connections in a DenseNet (Huang et al., 2017). Conventions as in Figure 7.

six orders of magnitude. Each DenseNet layer's FLNe values are essentially uniform, whereas in the macaque, they have a broad log-normal distribution (Markov et al., 2011). Finally, aside from statistical differences, the particular connections are different. There are no analogous layers across the two networks beyond the input layer.

Figure 10 compares the numbers of feature maps (channels) and units in each layer of the MSH model to those in the standard CNNs. In each of the standard CNNs studied here, the number of channels increases with the depth. This is also the trend in the MSH model (although there is greater scatter, due partly to differences in layer sizes within each area, and omission of nonprojecting L5 and L6 neurons). Also, in both the standard CNNs and the MSH model, the numbers of channels do not increase quickly enough to offset decreases in resolution. In each of the networks, the number of units peaks early and falls by at least an order of magnitude in later convolutional layers. In the MSH network, this reflects the tendency toward smaller visual areas higher in the macaque visual hierarchy (Felleman & Van Essen, 1991). This is also consistent with the deep-network design

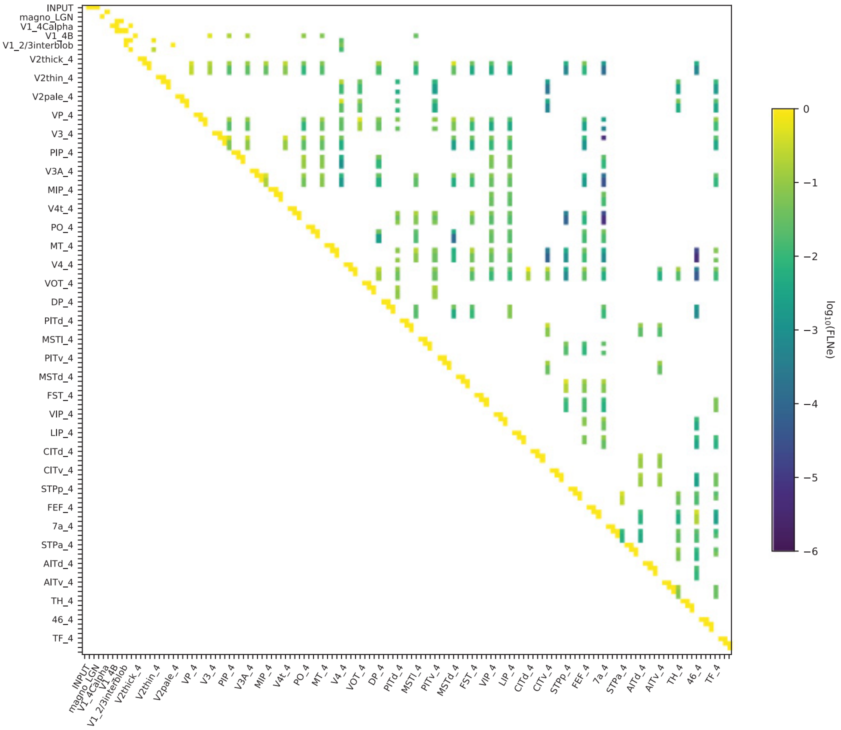

Figure 9: Connections in the cortex-like MSH network. The areas are ordered by hierarchical level (Felleman & Van Essen, 1991) and (within each level) by the number of incoming connections to L4. To show the whole network structure together, FLNe values are divided into supragranular and infragranular components and generalized to include interlaminar connections, as described in the text. To reduce clutter, only labels for L4 of each area are shown. L2/3, L5, and L6 of each area are before and after L4. Reproduced with permission from https://github.com/bptripp/calc.

heuristic of reducing the representation gradually (Szegedy, Vanhoucke, Ioffe, Shlens, & Wojna, 2016). The standard networks are all designed for the 1000-way ImageNet classification task, so the final softmax layer has exactly 1000 units in each case. Except for the softmax layer, the macaque model has a similar trend, with large V1 and V2 layers and much smaller layers at later stages.

Figure 11 plots distributions of kernel widths. In addition to convolutional layers, the standard CNNs also have pooling layers, which lack convolutional kernels. For these layers, the pool size is reported because it also defines the size of the neighborhood in the presynaptic layer that influences

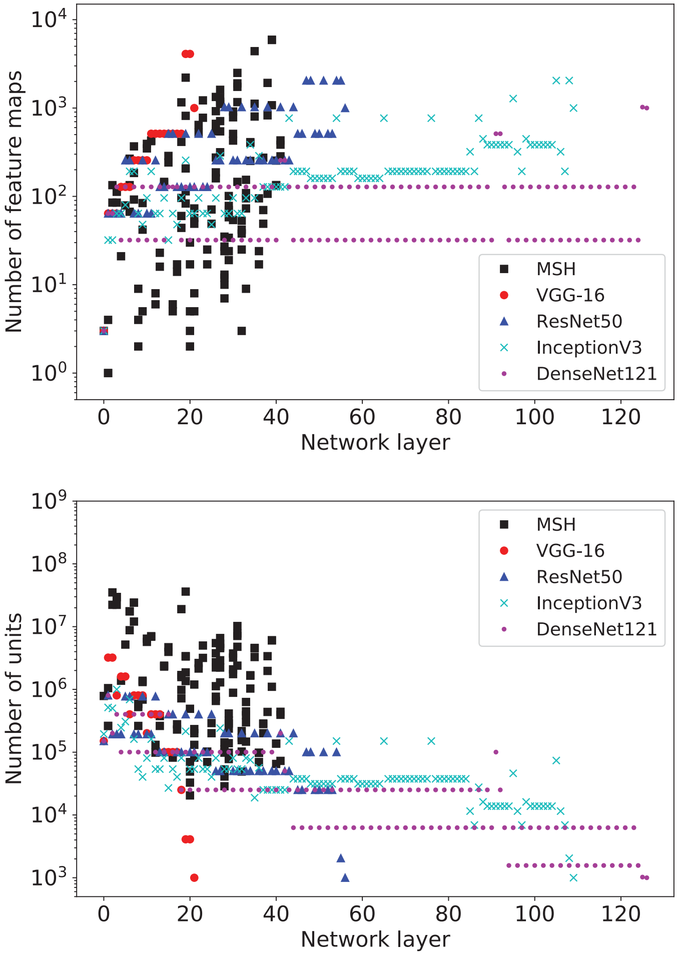

Figure 10: (Top) Number of feature maps per layer, in several standard CNNs and the macaque-like MSH model. (Bottom) Number of units per layer in these networks. The macaque-based MSH network has many parallel paths. Its layer depths are plotted based on the hierarchy in Felleman and Van Essen (1991). The input has depth 0, LGN layers have depth 1, and cortical layers have depth $1 + 4(a - 1) + l$, where a is the hierarchical level of the cortical area, and l is 1 for L4, 2 for L2/3, 3 for L5, or 4 for L6.

each postsynaptic unit. For fully connected layers with convolutional-layer input, the resolution of the convolutional layer is reported. For fully connected layers with nonspatial input, the kernel width is reported as one, as these connections are equivalent to 1×1 convolutions operating on 1×1

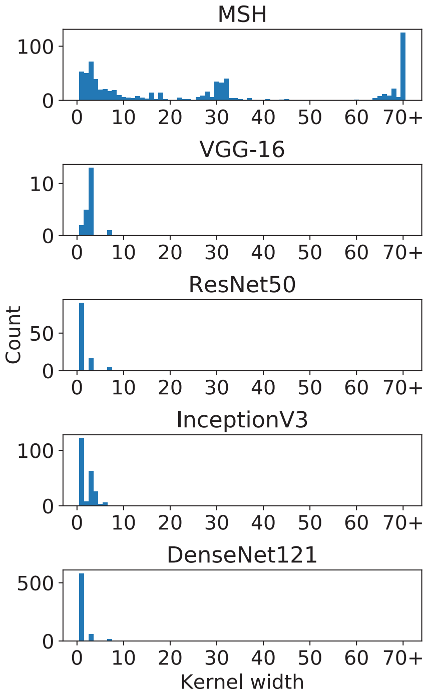

Figure 11: Histograms of kernel widths in a macaque-like MSH network (top) and several standard CNNs. The last bin includes kernels 70 or more elements wide. The largest kernels varied fairly widely in different optimizations, depending on the stride pattern. This example had particularly large kernels, with 12 kernel widths above 300.

feature maps. In modern CNNs, 3 × 3 kernels and 2 × 2 pooling neighborhoods are widely used. Inception networks have a wider variety of kernel sizes, from 1 × 1 to 5 × 5. The MSH model has a much wider range, including a substantial number of kernels larger than 30 × 30. Such kernels tend to be very sparse (see Figure 12), and to belong to long skip connections (see Figure 13). These large kernels are a consequence of direct connections between layers with very different receptive field sizes and the assumption that each input to a population affects the full receptive field. If these kernels were small, input (for example) from V2 to IT would affect only a small part of an IT neuron's receptive field.

Figure 14 shows distributions of strides. In all the networks, strides of one are most common, and strides of two are also common. The MSH model also has a small number of strides of four and eight.

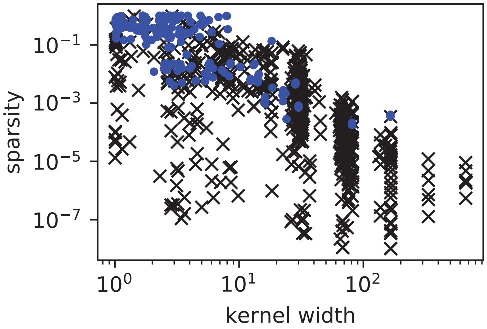

Figure 12: Scatter plot of sparsity ($c\sigma$) versus kernel width in an MSH network. Larger kernels tend to be sparser. The markers "x" and "." correspond to kernels of interarea and interlaminar connections, respectively. Interlaminar connections tend to be smaller and less sparse.

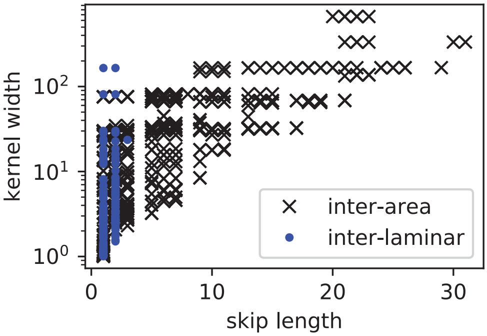

Figure 13: Scatter plot of kernel width versus length of skip connections (i.e., the longest path that is parallel to each connection). Connections with longer skip lengths have larger kernels. The markers "x" and "." correspond to kernels of interarea and interlaminar connections, respectively. Interlaminar connections have skip lengths of one or two and do not include kernels as large as the interarea connections.

3.3 Training on the CIFAR-10 Data Set. To verify that these methods produced trainable networks, a subnetwork including ventral areas up to PITd was trained to perform 10-way image classifications on the CIFAR-10 data set (Krizhevsky, Nair, & Hinton, 2014). To make training more tractable, the number of neurons in each population was reduced by a factor of 10, and connections with FLNe less than 0.15 were omitted. The accuracy of the network's predictions on the held-out validation set was 0.79, which is well above chance and well below state of the art. Future work will

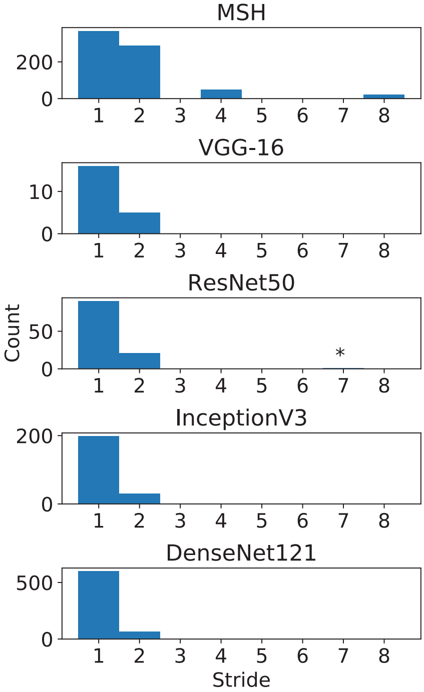

Figure 14: Histograms of strides in the macaque-like MSH network (top) and several standard CNNs. ResNet50 has one outlier, marked with an asterisk.

experiment with other regularization and learning parameters to improve performance and train the full MSH network with parallel objectives for different outputs (e.g., visual odometry, grasp planning).

4 Discussion

This study developed a method of making deep networks that have similar architectures to the visual cortex. This allowed a new and specific comparison between cortical and standard convolutional architectures. Among the standard CNNs explored here, DenseNet (Huang et al., 2017) was qualitatively the most similar to the cortex-like MSH network, in that DenseNet has more and longer skip connections than other standard networks and greater degrees of convergence onto some layers.

However, there were several qualitative differences between the MSH network and all of the standard CNNs. In the MSH network, skip connections covered a greater fraction of the network depth, kernels were larger and more varied in size, connections were sparser on average and had a wider range of sparsity, and there was greater variation in strides. The macaque-optimized network also had more units per layer than the standard CNNs.

The MSH network was similar to the standard networks in other ways. For example, the number of units per layer peaked in early layers. The longest path in the MSH model (39 steps) was within the rather wide range of depths of the standard CNNs studied here—between 22 and 127 layers. A much wider range of depths has been used in other CNNs in the literature, for example, up to 1202 layers (He et al., 2016).

In deep learning, skip connections have been used to reduce problems with training very deep networks (He et al., 2016) and to allow widespread use of features that are learned in early layers, without relearning them redundantly in later layers (Huang et al., 2017). In the brain, timing may be an additional consideration. Deeper neurons' earliest responses can lag a stimulus by more than 100 ms (Schmolesky et al., 1998). Skip connections might provide a rapid source of information to higher visual areas. For example, they might facilitate rapid recognition of objects or states that have highly salient low-level features, such as blood and brake lights.

In addition to providing a new perspective on architectural differences between visual cortex and standard CNNs, a cortex-like convolutional model is a necessary step toward clear comparisons between representations in analogous CNN and cortical areas. Such specific comparisons may be useful for understanding and reducing differences between deep networks and the brain.

Finally, biological neural networks are adapted to demanding and complex environments, so a macaque-like network may have practical advantages. For example, the MSH might complement recent work in a neural architecture search (Zoph & Le, 2016), either serving as a source of priors for hyperparameter values or suggesting a somewhat different search space, such as one with a wide range of connection sparsity.

4.1 Lack of a Unique Brainlike Set of Hyperparameters. Matching CNN hyperparameters to physiological data is inherently an underconstrained exercise because some hyperparameters (e.g., m, the number of feature maps in a layer) define a gridlike organization of units that has no analogy in the brain. Some choices of m are probably more reasonable than others, though. Two heuristics were used to guide these choices. First, the grid was chosen so that V1-like receptive fields would overlap with their neighbors about as much across channels as across space. Second, the number of channels in other areas was scaled relative to spine counts on basal dendrites, as explained in section 2.2.

A possible alternative heuristic would be to equate a V1 hypercolumn to a single pixel. A rationale for this could be that a hypercolumn spans the variety of tuning around one point in the visual field, much like all the channels of a single pixel in a convolutional layer. Each hypercolumn covers about 2 mm^2 of cortical surface, and there are (depending on the monkey) roughly 1200 mm^2 of V1 in a single hemisphere. So, instead of roughly 130 640 × 640 pixel feature maps in L2/3 of V1, as in the MSH model, this would result in about 85,000 25 × 25 pixel feature maps. Arguing against this interpretation, there is scatter in receptive field centers within a hypercolumn (Hubel & Wiesel, 1974), which could be achieved in this model only if weights were localized within subregions of oversized kernels. Furthermore, this alternative heuristic would lead to an intractable number of parameters because each feature map has its own kernels.

4.2 Connection Sparsity. In the standard CNNs studied here, every unit in a layer contributed to all of the layer's outgoing connections. Thus, the FLNe of each connection was simply the number of presynaptic units, divided by the total number of units connected to the postsynaptic layer. There was much less variation in FLNe in these networks than in the macaque brain (Markov et al., 2014), and a lack of low FLNe values. In this sense, connections are much less sparse in standard CNNs than in the macaque brain.

Interestingly, common notions of connection sparsity in CNNs are almost unrelated to FLNe measured in cortex. There has been vigorous interest in sparse connections in CNNs (LeCun, Denker, & Solla, 1989; Wen, Wu, Wang, Chen, & Li, 2016; Sun, Wang, & Tang, 2016), partly motivated by a desire for small CNNs that can run on embedded systems. Some studies achieve a reduction in parameters by kernel decomposition (Liu, Wang, Foroosh, Tappen, & Penksy, 2015), which is unrelated to FLNe. Another common approach has been to zero a fraction of the kernel entries through various means. This leads to sparsity in the same sense as the σ parameter in this study. However, because kernels are shared across units, a single nonzero entry might pass output from every presynaptic unit in a given channel (depending on the stride). So this kind of sparsity has a weak relationship with FLNe.

Dilated (or atrous) convolutions (Yu & Koltun, 2015; Chen, Papandreou, Kokkinos, Murphy, & Yuille, 2018) can strongly affect FLNe, but they typically do not. The number of presynaptic units that contribute to a connection with a dilated convolution is reduced by the square of the greatest common divisor of the dilation factor and the stride. For example, if the stride of the connection is one, then all the presynaptic units contribute, regardless of the dilation factor. In the MSH model, very sparse connections often have large strides, so setting dilation factors equal to strides would be an efficient way to approach the required sparsity.

Other deep-learning studies use a sparsity parameter that is related to the channel-wise sparsity parameter c. For example, in the early convolutional network of LeCun, Boser, et al. (1989), each feature map in the third convolutional layer received input from a distinct subset of the maps in the second layer. Recent variations of this scheme include Changpinyo, Sandler, and Zhmoginov (2017) and Zhang, Zhou, Lin, and Sun (2018). However, this scheme does not affect FLNe. Although it involves subsets of presynaptic maps, all of the maps are used somewhere in a connection from one layer to another.

Another perspective on sparsity in the deep learning literature comes from Inception networks. These networks involve parallel connections from one layer to another, each with different kernel sizes (1 × 1, 3 × 3, and 5 × 5). The authors of this scheme consider it to be another form of sparsity, because it involves many kernels and large kernels but requires fewer weights than many large kernels. However, this kind of sparsity is also unrelated to FLNe. In summary, the kind of connection sparsity that is measured in retrograde tracer studies in neuroscience is almost orthogonal to standard notions of connection sparsity in deep learning.

4.3 Limitations and Future Work. This work has omitted prominent features of cortical networks, including foveation and cortical magnification and recurrent, lateral, and feedback connections. Naturalistic foveation (with smoothly varying resolution) has not been extensively explored in deep networks (but see Wang & Cottrell, 2017; Dai et al., 2017; Rajalingham et al., 2018). There seem to be basic open questions on this topic, such as how to deal with a loss of translational equivariance in polar coordinates. So while this is an important topic, it has been omitted here for simplicity. Lateral and feedback connections (Angelucci et al., 2002) were also omitted because they are absent from standard convolutional networks. Recurrent layers (Greff, Srivastava, Koutník, Steunebrink, & Schmidhuber, 2017) and other model components with lateral (Tompson et al., 2014) and feedback (Xu et al., 2015) connections have been incorporated into convolutional networks, but there is a lack of standard approaches that have clear analogies with brain organization. However, recent work has shown that networks with such connections exhibit dynamic responses to static images that are similar to those in the ventral stream (Nayebi et al., 2018).

In addition to these missing features, the accuracy of the model is also limited by incomplete data on the macaque connectome. A particular limitation is that the estimates of interlaminar connectivity in Schmidt et al. (2018) relied largely on data from cat and mouse, including Binzegger et al. (2004). Also, many of the FLNe and %SLN values used here were interpolated from injections into about one-third of the visual areas (Schmidt et al., 2018), and estimates of cell and synapse density have not stabilized (Garcia-Marin et al., 2017). But despite these limitations, the MSH model has one-to-one analogies with macaque visual cortex, and it has hyperparameters that are probably much more consistent with macaque cortex than those of any previous convolutional network.

Future work should experiment with elaborations of the MSH model and with various training regimes and compare the resulting representations to those of monkeys. It would be interesting to see, for example, whether more realistic plasticity rules, circuit mechanisms, or experiences make the representations much more realistic than supervised learning of standard tasks.

4.4 What Insights into the Visual System Can Be Gained from the Model? A key motivation for this work is to help clarify comparisons between representations in deep networks and primate visual cortex in the future. Recently, a number of deep networks have been systematically benchmarked in terms of their ability to account for ventral stream representations (Schrimpf et al., 2018). However, these networks have nonbiological architectures. Because representations depend completely on position in a network, this may impose a ceiling on their similarity to the cortex. Furthermore, it is not really clear which layers in these networks should be compared with which neurobiological populations. This is a source of noise if one wishes to test whether specific changes to deep networks (such as adding local response normalization) result in more realistic representations. The model developed here should allow more specific comparisons in the future. This may facilitate development of more brain-like deep networks, indirectly contributing to future insights into visual cortex function.

In the meantime, since convolutional networks are the top-performing artificial systems in many vision tasks, such as object recognition, segmentation, and monocular and stereo depth estimation, they are arguably the most functionally realistic computational models of visual cortex. Understanding how convolutional networks differ from visual cortex therefore provides a valuable perspective on the visual cortex. This letter helps to clarify some architectural relationships. It shows that one can set hyperparameters to make the architecture of a convolutional network very similar to the feedforward architecture of visual cortex. Furthermore, there are a number of specific similarities and differences between the resulting MSH network and standard deep networks. For example, the number of layers in the MSH network is within the range of standard deep networks, and the MSH network has a rapid expansion in number of units per layer near the input and gradual decline in later layers, which many standard deep networks share. In contrast, standard deep networks do not have the kind of connection sparseness that exists in the cortex. A wide variety of deep network architectures are in current use, and the primate architecture is distinct from all of them but not terribly out of place.

5 Conclusion

This study developed a convolutional network architecture similar to macaque visual cortex and showed that its architecture differs qualitatively from architectures of standard convolutional networks. The cortex-like architecture has longer skip connections, qualitatively different connection sparsity, and wider ranges of sparsity and kernel size. These architectural differences may suggest directions for artificial general vision systems. Standard CNNs and cortex-like CNNs have similar increases in number of feature maps farther from the input and similar peaks in population size in early layers. Finally, layers in the architectures developed here have one-to-one relationships with neural populations in the visual cortex. This should allow more direct comparisons between representations in artificial networks and the brain.

Appendix: FLNe in CNNs

An important measure in retrograde tracer studies (Markov et al., 2014) is the fraction of labeled neurons (extrinsic to the injection site) associated with each connection (FLNe of the connection). In a convolutional network, an analogous quantity is

$$f^{ij} = \frac{n^{ij}}{\sum_{j \in I_i} n^{ij}}, \tag{A.1}$$

where I_i are the inputs to layer i, and n^{ij} is the number of presynaptic neurons that contribute to the (ij)th connection. This is

$$n^{ij} = n^j c^{ij} \sigma_*^{ij} \alpha^{ij}, \tag{A.2}$$

where n^j is the number of neurons in presynaptic layer j, c^{ij} is the fraction of channels of layer j that contribute to connection ij, and σ_*^{ij} and α^{ij} are factors related to pixel-wise sparsity and stride, respectively.

The stride affects n^{ij} only if it is greater than the kernel width—specifically,

$$\alpha^{ij} = \begin{cases} 1, & \text{if } s^{ij} \leq w_K^{ij} \\ \left(w_K^{ij}/s^{ij}\right)^2, & \text{otherwise} \end{cases}. \tag{A.3}$$

This is illustrated in Figure 15.

The factor σ_*^{ij} describes the impact of the element-wise sparseness factor σ^{ij} (the fraction of nonzero kernel entries). Specifically,

$$\sigma_*^{ij} = 1 - (1 - \sigma^{ij})^{\beta^{ij} m^i}, \tag{A.4}$$

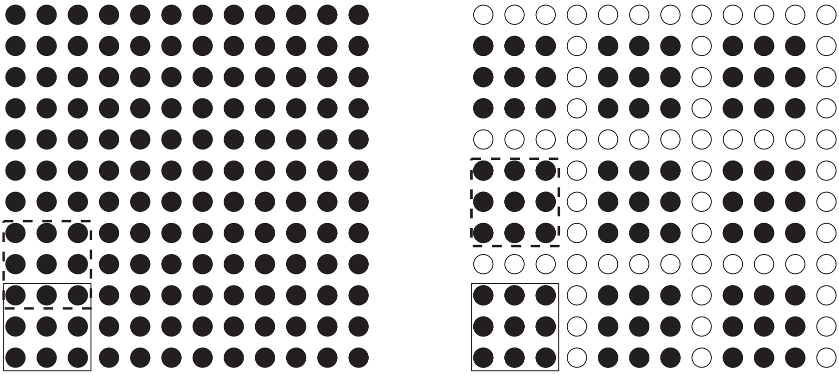

Figure 15: The stride of a connection, s^{ij}, influences FLNe only when it is greater than the kernel width (w^{ij}), as shown in the example on the right. The circles indicate units in one channel of a presynaptic layer. Units marked with filled circles contribute to the connection, and those marked with open circles do not. Solid and dashed boxes are drawn around groups of units that provide input to two different postsynaptic units. $w^{ij} = 3$ in each case. In the left example, $s^{ij} = 2$, which is less than w^{ij}, so all the presynaptic units are used. On the right, $s^{ij} = 4$, which is greater than w^{ij}, so some of the presynaptic units are not used in the connection.

where

$$\beta^{ij} = \begin{cases} \left(w_K^{ij}/s^{ij}\right)^2, & \text{if } s^{ij} < w_K^{ij} \\ 1, & \text{otherwise} \end{cases}. \tag{A.5}$$

This is due to the fact that a given presynaptic unit's output passes through different kernel entries on its way to different postsynaptic units. In the right-hand side of (A.4), $(1 - \sigma^{ij})$ is the probability that a given kernel element is zero, and $(1 - \sigma^{ij})^{\beta^{ij} m^i}$ is the probability that all kernel elements outbound from a given presynaptic unit are zero. This depends on the number of postsynaptic channels, m^i, and also on the stride-kernel width ratio via β^{ij}. Specifically, if the stride is less than the kernel width, then each presynaptic unit has multiple chances to affect different units in a given postsynaptic channel.

Acknowledgments

Thanks to Robbe Goris for sharing his code for the V1 receptive field model (Goris et al., 2015); Ricardo Fukasawa for discussing the optimization problem; and Eric Hunsberger, Brian Hu, Shamak Dutta, and two anonymous reviewers for comments on earlier versions of the letter. Eric Hunsberger pointed out a possible role of skip connections in rapid perception. This work used data from CoCoMac (Bakker et al., 2012), which draws from hundreds of tract-tracing papers. See https://github.com/bptripp/calc for additional references to these studies.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., . . . Zheng, X. (2016). TensorFlow: Large-scale machine learning on heterogeneous distributed systems. In *Proceedings of the 12th Symposium on Operating Systems Design and Implementation* (vol. 16, pp. 265–283). Berkeley, CA: USENIX.

Angelucci, A., Levitt, J. B., Walton, E. J. S., Hupe, J.-M., Bullier, J., & Lund, J. S. (2002). Circuits for local and global signal integration in primary visual cortex. *Journal of Neuroscience*, *22*(19), 8633–8646.

Bakker, R., Wachtler, T., & Diesmann, M. (2012). CoCoMac 2.0 and the future of tract-tracing databases. *Frontiers in Neuroinformatics, 6*, 30. doi:10.3389/fninf.2012.00030

Binzegger, T., Douglas, R. J., & Martin, K. A. C. (2004). A quantitative map of the circuit of cat primary visual cortex. *Journal of Neuroscience, 24*(39), 8441–8453. doi:10.1523/JNEUROSCI.1400-04.2004

Boussaoud, D., Desimone, R., & Ungerleider, L. G. (1991). Visual topography of area TEO in the macaque. *Journal of Comparative Neurology, 306*(4), 554–575. doi:10.1002/cne.903060403

Briggs, F. (2010). Organizing principles of cortical layer 6. *Frontiers in Neural Circuits, 4*(February), 1–8. doi:10.3389/neuro.04.003.2010

Brincat, S. L., & Connor, C. E. (2006). Dynamic shape synthesis in posterior inferotemporal cortex. *Neuron, 49*(1), 17–24. doi:10.1016/j.neuron.2005.11.026

Bullier, J., Kennedy, H., & Salinger, W. (1984). Branching and laminar origin of projections between visual cortical areas in the cat. *Journal of Comparative Neurology*, *228*, 329–341.

Cadieu, C. F., Hong, H., Yamins, D. L. K., Pinto, N., Ardila, D., Solomon, E. A., . . . DiCarlo, J. J. (2014). Deep neural networks rival the representation of primate IT cortex for core visual object recognition. *PLoS Computational Biology, 10*(12). doi:10.1371/journal.pcbi.1003963

Callaway, E. M. (2004). Feedforward, feedback and inhibitory connections in primate visual cortex. *Neural Networks, 17*, 625–632. doi:10.1016/j.neunet.2004.04.004

Callaway, E. M., & Wiser, A. K. (1996). Contributions of individual layer 2–5 spiny neurons to local circuits in macaque primary visual cortex. *Visual Neuroscience*, *13*, 907–922.

Changpinyo, S., Sandler, M., & Zhmoginov, A. (2017). *The power of sparsity in convolutional neural networks*. arXiv:1702, 1–13.

Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., & Yuille, A. L. (2018). DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *40*(4), 834–848.

Chollet, F. (2015). Keras. https://keras.io/.

Cowley, B. R., Smith, M. A., Kohn, A., & Yu, B. M. (2016). Stimulus-driven population activity patterns in macaque primary visual cortex. *PLoS Computational Biology, 12*(12), 1–31. doi:10.1371/journal.pcbi.1005185

Dai, J., Qi, H., Xiong, Y., Li, Y., Zhang, G., Hu, H., & Wei, Y. (2017). Deformable convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision*. Piscataway, NJ: IEEE.

Dong, Q., Liu, B., & Hu, Z. (2017). Comparison of it neural response statistics with simulations. *Frontiers in Computational Neuroscience*, *11*, 60.

Eickenberg, M., Gramfort, A., Varoquaux, G., & Thirion, B. (2016). Seeing it all: Convolutional network layers map the function of the human visual system. *NeuroImage, 152*, 184–194. doi:10.1016/j.neuroimage.2016.10.001

Elston, G. N. (2007). Specialization of the neocortical pyramidal cell during primate evolution. In J. H. Kaas (Ed.), *Evolution of nervous systems* (pp. 191–242). Orlando, FL: Academic Press.

Fares, T., & Stepanyants, A. (2009). Cooperative synapse formation in the neocortex. In *Proceedings of the National Academy of Sciences of the United States of America, 106*(38), 16463–16468. doi:10.1073/pnas.0813265106

Felleman, D. J., & Van Essen, D. C. (1987). Receptive field properties of neurons in area V3 of macaque monkey extrastriate cortex. *Journal of Neurophysiology*, *57*(4), 889–920.

Felleman, D. J., & Van Essen, D. C. (1991). Distributed hierarchical processing in the primate cerebral cortex. *Cerebral Cortex*, *1*, 1–47.

Galletti, C., Fattori, P., Gamberini, M., & Kutz, D. F. (1999). The cortical visual area V6: Brain location and visual topography. *European Journal of Neuroscience, 11*, 3922–3936. doi:10.1046/j.1460-9568.1999.00817.x

Garcia-Marin, V., Kelly, J. G., & Hawken, M. J. (2017). Major feedforward thalamic input into layer 4C of primary visual cortex in primate. *Cerebral Cortex, 29*(1), 1–16. doi:10.1093/cercor/bhx311

Gattass, R., Gross, C. G., & Sandell, J. H. (1981). Visual topography of V2 in the macaque. *Journal of Comparative Neurology, 201*(4), 519–539. doi:10.1002/cne.902010405

Gattass, R., Sousa, A. P. B., & Gross, C. G. (1988). Visuotopic organization and extent of V3 and V4 of the macaque. *Journal of Neuroscience*, *8*(6), 1831–1845.

Gilbert, C. D. (1977). Laminar differences in receptive field properties of cells in cat primary visual cortex. *J. Physiology, 268*(1977), 391–421. doi:10.1113/jphysiol.1977.sp011863

Goris, R. L., Simoncelli, E. P., & Movshon, J. A. (2015). Origin and function of tuning diversity in macaque visual cortex. *Neuron, 88*(4), 819–831. doi:10.1016/j.neuron.2015.10.009

Greff, K., Srivastava, R. K., Koutník, J., Steunebrink, B. R., & Schmidhuber, J. (2017). LSTM: A search space odyssey. *IEEE Transactions on Neural Networks and Learning Systems, 28*(10), 2222–2232. doi:10.1017/CBO9781107415324.004

Güçlü, U., & van Gerven, M. a. J. (2015). Deep neural networks reveal a gradient in the complexity of neural representations across the ventral stream. *Journal of Neuroscience, 35*(27), 10005–10014. doi:10.1523/JNEUROSCI.5023-14.2015

Hassabis, D., Kumaran, D., Summerfield, C., & Botvinick, M. (2017). Neuroscience-inspired artificial intelligence. *Neuron, 95*(2), 245–258. doi:10.1016/j.neuron.2017.06.011

He, K., Gkioxari, G., Dollar, P., & Girshick, R. (2017). Mask R-CNN. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 2980–2988). Piscataway, NJ: IEEE. doi:10.1109/ICCV.2017.322

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). Piscataway, NJ: IEEE. doi:10.3389/fpsyg.2013.00124

Hong, H., Yamins, D. L. K., Majaj, N. J., & DiCarlo, J. J. (2016). Explicit information for category-orthogonal object properties increases along the ventral stream. *Nature Neuroscience, 19*(4), 613–622. doi:10.1038/nn.4247

Huang, G., Liu, Z., van der Maaten, L., & Weinberger, K. Q. (2017). Densely connected convolutional networks. In *IEEE Conference on Computer Vision and Pattern Recognition* (pp. 4700–4708). Piscataway, NJ: IEEE. doi:10.1109/CVPR.2017.243

Hubel, D. H., & Wiesel, T. N. (1974). Uniformity of monkey striate cortex: A parallel relationship between field size, scatter, and magnification factor. *Journal of Comparative Neurology, 158*(3), 295–305. doi:10.1002/cne.901580305

Hübener, M., Schwarz, C., & Bolz, J. (1990). Morphological types of projection neurons in layer 5 of cat Vd cortex. *Journal of Comparative Neurology, 301*, 655–674. doi:10.1002/cne.903010412

Issa, E. B., Cadieu, C. F., & DiCarlo, J. J. (2018). *Neural dynamics at successive stages of the ventral visual stream are consistent with hierarchical error signals*. bioRxiv:092551.

Karpathy, A. (2014). *What I learned from competing against a ConvNet on ImageNet*. http://karpathy.github.io.

Kasthuri, N., Hayworth, K. J., Berger, D. R., Schalek, R. L., Conchello, J. A., Knowles-Barley, S., . . . Lichtman, J. W. (2015). Saturated reconstruction of a volume of neocortex. *Cell, 162*(3), 648–661. doi:10.1016/j.cell.2015.06.054

Khaligh-Razavi, S. M., & Kriegeskorte, N. (2014). Deep supervised, but not unsupervised, models may explain IT cortical representation. *PLoS Computational Biology, 10*(11). doi:10.1371/journal.pcbi.1003915

Kheradpisheh, S. R., Ghodrati, M., & Ganjtabesh, M. (2016). Deep networks can resemble human feed-forward vision in invariant object recognition. *Scientific Reports, 6*, 32672. doi:10.1038/srep32672

Kim, E. J., Juavinett, A. L., Kyubwa, E. M., Jacobs, M. W., & Callaway, E. M. (2015). Three types of cortical layer 5 neurons that differ in brain-wide connectivity and function. *Neuron, 88*(6), 1253–1267. doi:10.1016/j.neuron.2015.11.002

Kingma, D., & Ba, J. (2014). *Adam: A method for stochastic optimization*. arXiv:1412/6980 [cs], 1–15.

Komatsu, H., & Wurtz, R. H. (1988). Relation of cortical areas MT and MST to pursuit eye movements. I. Localization and visual properties of neurons. *Journal of Neurophysiology, 60*(2), 580–603. doi:10.220.33.5

Kravitz, D. J., Saleem, K. S., Baker, C. I., & Mishkin, M. (2011). A new neural framework for visuospatial processing. *Nature Reviews Neuroscience, 12*(4), 217–230. doi:10.1038/nrn3008

Krizhevsky, A. (2009). *Learning multiple layers of features from tiny images*. PhD diss., University of Toronto.

Krizhevsky, A., Nair, V., & Hinton, G. (2014). *The CIFAR-10 dataset*. http://www.cs.toronto.edu/kriz/cifar.html.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, & K. Q. Weinberger (Eds.), *Advances in neural information processing systems, 25* (pp. 1097–1105). Red Hook, NY: Curran.

Lake, B. M., Ullman, T. D., Tenenbaum, J. B., & Gershman, S. J. (2016). Building machines that learn and think like people. *Behavioral and Brain Sciences, 40*, 1–101. doi:10.1017/S0140525X16001837

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., & Jackel, L. D. (1989). Handwritten digit recognition with a back-propagation network. In D. S. Touretzky (Ed.), *Neural information processing systems* (pp. 396–404). San Mateo, CA: Morgan Kaufmann.

LeCun, Y., Denker, J. S., & Solla, S. A. (1989). Optimal brain damage. In D. S. Touretzky (Ed.), *Advances in neural information processing systems* (pp. 598–605). San Mateo, CA: Morgan Kaufmann.

Liu, B., Wang, M., Foroosh, H., Tappen, M., & Penksy, M. (2015). Sparse convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 806–814). Piscataway, NJ: IEEE.

Livingstone, M., & Hubel, D. (1988). Segregation of form, color, movement, and depth: Anatomy, physiology, and perception. *Science, 240*(4853), 740–749.

Lotter, W., Kreiman, G., & Cox, D. (2017). Deep predictive coding networks for video prediction and unsupervised learning. In *Proceedings of the International Conference on Learning Representations*. https://openreview.net/group?id-ICLR.cc/2017/conference

Luo, W., Li, Y., Urtasun, R., & Zemel, R. (2016). Understanding the effective receptive field in deep convolutional neural networks. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, & R. Garnett (Eds.), *Advances in neural information processing systems, 29* (pp. 4898–4906). Red Hook, NY: Curran.

Lur, G., Vinck, M. A., Tang, L., Cardin, J. A., & Higley, M. J. (2016). Projection-specific visual feature encoding by layer 5 cortical subnetworks. *Cell Reports, 14*(11), 2538–2545. doi:10.1016/j.celrep.2016.02.050

Markov, N. T., Ercsey-Ravasz, M. M., Ribeiro Gomes, a. R., Lamy, C., Magrou, L., Vezoli, J., . . . Kennedy, H. (2014). A weighted and directed interareal connectivity matrix for macaque cerebral cortex. *Cerebral Cortex, 24*(1), 17–36. doi:10.1093/cercor/bhs270

Markov, N. T., Misery, P., Falchier, A., Lamy, C., Vezoli, J., Quilodran, R., . . . Knoblauch, K. (2011). Weight consistency specifies regularities of macaque cortical networks. *Cerebral Cortex, 21*(6), 1254–1272. doi:10.1093/cercor/bhq201

Maunsell, J. H. R., & Van Essen, D. C. (1987). Topographic organization of the middle temporal visual area in the macaque monkey: Representational biases and the relationship to callosal connections and myeloarchitectonic boundaries. *Journal of Comparative Neurology, 266*, 535–555.

Merigan, W., & Maunsell, J. H. (1993). How parallel are the primate visual pathways? *Annual Review of Neuroscience, 16*, 369–402.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., . . . Hassabis, D. (2015). Human-level control through deep reinforcement learning. *Nature, 518*(7540), 529–533. doi:10.1038/nature14236

Nassi, J. J., & Callaway, E. M. (2009). Parallel processing strategies of the primate visual system. *Nature Reviews Neuroscience, 10*, 361–372. doi:10.1038/nrn2619

Nayebi, A., Bear, D., Kubilius, J., Kar, K., Ganguli, S., Sussillo, D., . . . Yamins, D. L. K. (2018). *Task-driven convolutional recurrent models of the visual system*. arXiv:1807.00053.

Nayebi, A., & Ganguli, S. (2017). *Biologically inspired protection of deep networks from adversarial attacks*. arXiv:1706.

O'Kusky, J., & Colonnier, M. (1982). A laminar analysis of the number of neurons, glia, and synapses in the visual-cortex (area-17) of adult macaque monkeys. *Journal of Comparative Neurology*, *210*(3), 278–290.

Parisien, C., Anderson, C. H., & Eliasmith, C. (2008). Solving the problem of negative synaptic weights in cortical models. *Neural Computation, 20*(6), 1473–1494. doi:10.1162/neco.2008.07-06-295

Rajalingham, R., Issa, E. B., Bashivan, P., Kar, K., Schmidt, K., & DiCarlo, J. J. (2018). Large-scale, high-resolution comparison of the core visual object recognition behavior of humans, monkeys, and state-of-the-art deep artificial neural networks. *Journal of Neuroscience, 38*(33), 7255–7269. doi:10.1523/JNEUROSCI.0388-18.2018

Rockland, K. S. (2013). Collateral branching of long-distance cortical projections in monkey. *Journal of Comparative Neurology, 521*(18), 4112–4123. doi:10.1002/cne.23414

Rubin, D. B., Hooser, S. D. V., & Miller, K. D. (2015). The stabilized supralinear network: A unifying circuit motif underlying multi-input integration in sensory cortex. *Neuron, 85*, 402–417. doi:10.1016/j.neuron.2014.12.026

Scardapane, S., Comminiello, D., Hussain, A., & Uncini, A. (2017). Group sparse regularization for deep neural networks. *Neurocomputing, 241*(2017), 81–89. doi:10.1016/j.neucom.2017.02.029

Schmidt, M., Bakker, R., Hilgetag, C. C., Diesmann, M., & van Albada, S. J. (2018). Multi-scale account of the network structure of macaque visual cortex. *Brain Structure and Function, 223*(3), 1409–1435. doi:10.1007/s00429-017-1554-4

Schmolesky, M. T., Wang, Y., Hanes, D. P., Thompson, K. G., Leutgeb, S., Schall, J. D., & Leventhal, A. G. (1998). Signal timing across the macaque visual system. *J. Neurophysiology*, *79*, 3272–3278.

Schrimpf, M., Kubilius, J., Hong, H., Majaj, N. J., Rajalingham, R., Issa, E. B., . . . DiCarlo, J. J. (2018). *Brain-score: Which artificial neural network for object recognition is most brain-like?* bioRxiv:407007.

Seeliger, K., Fritsche, M., Güçlü, U., Schoenmakers, S., Schoffelen, J. M., Bosch, S. E., & van Gerven, M. A. (2017). Convolutional neural network-based encoding and decoding of visual object recognition in space and time. *NeuroImage, 180*, 253–266. doi:10.1016/j.neuroimage.2017.07.018

Shi, J., Wen, H., Zhang, Y., Han, K., & Liu, Z. (2017). *Deep recurrent neural network reveals a hierarchy of process memory during dynamic natural vision*. doi:10.1101/177196

Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. In *Proceedings of the International Conference on Learning Representations*. Berkeley, CA: USENIX.

Song, S., Sjostrom, P. J., Reigl, M., Nelson, S., & Chklovskii, D. B. (2005). Highly nonrandom features of synaptic connectivity in local cortical circuits. *PLoS Biology, 3*(3), 0507–0519. doi:10.1371/journal.pbio.0030068

Sun, Y., Wang, X., & Tang, X. (2016). Sparsifying neural network connections for face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 4856–4864). Piscataway, NJ: IEEE. doi:10.1109/CVPR.2016.525

Szegedy, C., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1–9). Piscataway, NJ: IEEE. doi:10.1109/CVPR.2015.7298594

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 2818–2826). Piscataway, NJ: IEEE. doi:10.1109/CVPR.2016.308

Tamura, H., & Tanaka, K. (2001). Visual response properties of cells in the ventral and dorsal parts of the macaque inferotemporal cortex. *Cerebral Cortex*, *11*(5), 384–399.

Thomson, A. M., & Bannister, A. P. (2003). Interlaminar connections in the neocortex. *Cerebral Cortex*, *13*(1), 5–14.

Tompson, J., Jain, A., LeCun, Y., & Bregler, C. (2014). Joint training of a convolutional network and a graphical model for human pose estimation. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, & K. Q. Weinberger (Eds.), *Advances in neural information processing systems, 27* (pp. 1799–1807). Red Hook, NY: Curran.

Tripp, B. (2017). Similarities and differences between stimulus tuning in the inferotemporal visual cortex and convolutional networks. In *Proceedings of the International Joint Conference on Neural Networks* (pp. 3551–3560). Piscataway, NJ: IEEE.

Tripp, B., & Eliasmith, C. (2016). Function approximation in inhibitory networks. *Neural Networks, 77*, 95–106. doi:10.1016/j.neunet.2016.01.010

Van Essen, D. C., & Newsome, W. T. (1984). The visual field representation in the striate cortex of the macaque monkey: Asymmetries, anisptropies, and indiviual variability. *Vision Research, 24*(5), 429–448. doi:10.1016/0042-6989(84)90041-5

Wang, P., & Cottrell, G. W. (2017). Central and peripheral vision for scene recognition: A neurocomputational modeling exploration. *Journal of Vision, 17*(4), 9. doi:10.1167/17.4.9

Weber, A. J., Chen, H., Hubbard, W. C., & Kaufman, P. L. (2000). Experimental glaucoma and cell size, density, and number in the primate lateral geniculate nucleus. *Investigative Ophthalmology and Visual Science*, *41*(6), 1370–1379.

Wen, H., Shi, J., Chen, W., & Liu, Z. (2017). *Transferring and generalizing deep-learning-based neural encoding models across subjects*. doi:10.1101/171017

Wen, W., Wu, C., Wang, Y., Chen, Y., & Li, H. (2016). Learning structured sparsity in deep neural networks. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, & R. Garnett (Eds.), *Neural information processing systems, 29* (pp. 1–9). Red Hook, NY: Curran.

Wiser, A. K., & Callaway, M. (1996). Contributions of individual layer 6 pyramidal neurons to local circuitry in macaque primary visual cortex. *Journal of Neuroscience*, *16*(8), 2724–2739.

Xu, K., Ba, J. L., Kiros, R., Cho, K., Courville, A., Salakhutdinov, R., . . . Bengio, Y. (2015). Show, attend and tell: Neural image caption generation with visual attention. In *Proceedings of the International Conference on Machine Learning* (pp. 2048–2057). New York: ACM. doi:10.1109/72_279181.

Yamins, D. L. K., Hong, H., Cadieu, C. F., Solomon, E. A., Seibert, D., & Dicarlo, J. J. (2014). Performance-optimized hierarchical models predict neural responses in higher visual cortex. *PNAS, 111*, 8619–8624. doi:10.1073/pnas.1403112111

Yu, F., & Koltun, V. (2018). *Multi-scale context aggregation by dilated convolutions* arXiv:1511.07122.

Žbontar, J., & LeCun, Y. (2016). Stereo matching by training a convolutional neural network to compare image patches. *Journal of Machine Learning Research*, *17*, 1–32.

Zhang, X., Zhou, X., Lin, M., & Sun, J. (2017). ShuffleNet: An extremely efficient convolutional neural network for mobile devices. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 8619–8624). Piscataway, NJ: IEEE.

Zoph, B., & Le, Q. V. (2016). *Neural architecture search with reinforcement learning*. arXiv:1611.01578.

Received September 12, 2018; accepted April 6, 2019.