ALIGNING ARTIFICIAL NEURAL NETWORKS TO THE BRAIN YIELDS SHALLOW RECURRENT ARCHITECTURES

Anonymous authors
Paper under double-blind review

ABSTRACT

Deep artificial neural networks with spatially repeated processing (a.k.a., deep convolutional ANNs) have been established as the best class of candidate models of visual processing in the primate ventral visual processing stream. Over the past five years, these ANNs have evolved from a simple feedforward eight-layer architecture in AlexNet to extremely deep and branching NASNet architectures, demonstrating increasingly better object categorization performance. Here we ask, as ANNs have continued to evolve in performance, are they also strong candidate models for the brain? To answer this question, we developed Brain-Score, a composite of neural and behavioral benchmarks for determining how brain-like a model is, together with an online platform where models can receive a Brain-Score and compare against other models. Despite high scores, typical deep models from the machine learning community are often hard to map onto the brain's anatomy due to their vast number of layers and missing biologically-important connections, such as recurrence. To further map onto anatomy and validate our approach, we built CORnet-S: an ANN guided by Brain-Score with the anatomical constraints of compactness and recurrence. Although a shallow model with four anatomically mapped areas and recurrent connectivity, CORnet-S is a top model on Brain-Score and outperforms similarly compact models on ImageNet. Analyzing CORnet-S circuitry variants revealed recurrence as the main predictive factor of both Brain-Score and ImageNet top-1 performance.

1 INTRODUCTION

In our view, the goal for computational vision systems is to be at the very least as capable as the human visual system. If we could mimic the workings of the visual system – both the outputs of the system and its internal representations – such an approach would necessarily yield powerful models: for neuroscience, these models would become mechanistic hypotheses of the processes in the brain, and for machine learning this would yield robust models that generalize across datasets and tasks. Historically, this idea of neuroscience-driven machine learning dates back to the earliest days of the field of artificial intelligence with artificial neurons being directly inspired from biological neurons (McCulloch & Pitts, 1943) and hierarchical convolutional structure of the visual system paving the road towards convolutional neural networks (Hubel & Wiesel, 1962; Fukushima, 1980). In recent years however, aligning artificial neural networks to biological ones has been less successful, partially due to the lack of robust comparison metrics and sufficiently large datasets (but see Yamins et al., 2013; 2014). Instead, progress in machine learning stemmed from strong benchmarks such as ImageNet (Deng et al., 2009) which lead to impressive models of object recognition (Russakovsky et al., 2015). In neuroscience, on the other hand, benchmarking has so far not been clearly established and quantitative model evaluations are less common.

To facilitate the cross-talk between machine learning and neuroscience, we developed *Brain-Score*, a large-scale benchmark composed of neural recordings and behavioral measurements that enables a quantified comparison between models and brain data. The primary aim of Brain-Score is to provide a benchmark that, unlike the ImageNet classification task, would allow a direct optimization of models towards a functional equivalent of the human brain. We thereby hope to encourage the

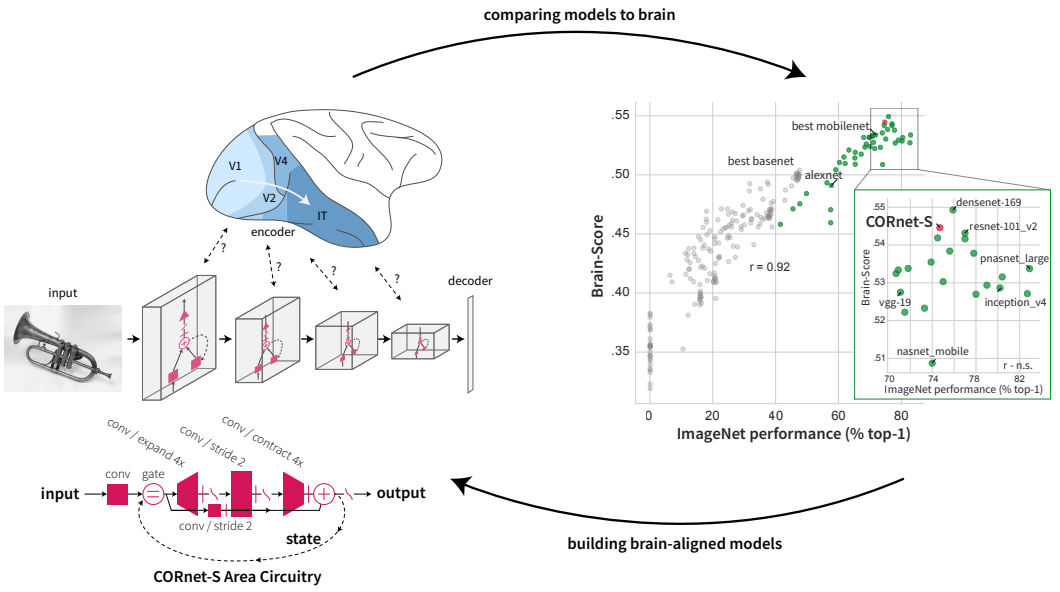

Figure 1: **Synergizing machine learning and neuroscience through Brain-Score (top).** By quantifying brain-likeness of models, we can compare models of the brain and use insights gained to inform the next generation of models. DenseNet, CORnet-S and ResNet architectures are the current winning models on Brain-Score. Correlation to ImageNet is strong at first, but non-significant for high-performance models. Green dots represent popular deep neural networks (Table 1) while gray dots correspond to various simple six-layer models at different stages of their training in order to demonstrate the relationship between ImageNet performance and Brain-Score on a wider range of performances. **CORnet-S area architecture (bottom left).** The model consists of four areas which are pre-mapped to cortical areas V1, V2, V4, and IT in the ventral stream. $V1_{COR}$ is feed-forward and acts as a pre-processor to reduce the input complexity. $V2_{COR}$, $V4_{COR}$ and IT_{COR} are recurrent (within area) to reduce the need for many layers and incorporate skip-connections, following the observation that ResNets and DenseNets are strong models on Brain-Score.

development of models that are more brain-like and thus easier to map onto the brain, to analyze, and to drive next experiments. Indeed, deep neural networks are often criticized or ignored by the neuroscience community due to the complexity in the number of layers and due to missing key elements from the ventral stream, such as recurrent and feedback connections.

To validate Brain-Score, we demonstrate that it provides robust model rankings across a diverse set of neural and behavioral datasets. We show that using this benchmark as a guiding score for building models can also yield competitive performance on ImageNet. Our model *CORnet-S* commits to a shallow recurrent anatomical structure of the ventral visual stream and is a top model on Brain-Score, yet retains a strong ImageNet top-1 performance of 73.1%. Taken together, Brain-Score benchmarking opens new possibilities for machine learning practitioners and neuroscientists to collaborate and advance both fields together.

2 Brain-Score: Comparing Models to Brain

To obtain a single scalar for brain-likeness, we built *Brain-Score*, a composite benchmark that measures how well models can predict (a) the mean neural response of each neural recording site to each and every tested naturalistic image in non-human primate visual areas V4 and IT (data from Majaj et al., 2015) and (b) mean pooled human choices when reporting a target object to each and every tested naturalistic image (data from Rajalingham et al., 2018). Individual benchmarks are briefly outlined below; see Appendix B for more details. Brain-Score is open-sourced as a platform to score neural networks on brain data through GitHub, a pip package, and a website for online submissions.

2.1 NEURAL PREDICTIVITY

A total of 2760 images containing a single object pasted randomly on a natural background were presented centrally to passively fixated monkeys for 100 ms and neural responses were obtained from 88 V4 sites and 168 IT sites. For our analyses, we used normalized time-averaged neural responses in the 70-170 ms window. A regression model was constructed for each neuron using 90% of image responses and tested on the remaining 10% in a 10-fold cross-validation strategy. The median over neurons of the Pearsons r between the predicted and actual response constituted the final neural fit score for each visual area. In our model CORnet-S, we used designated model areas and the best time point to predict corresponding neural data. In comparison models, we used the most predictive layer.

2.2 BEHAVIORAL PREDICTIVITY

A total of 2400 images containing a single object pasted randomly on a natural background were presented to 1472 humans for 100 ms and they were asked to choose from two options which object they saw. 240 of those images with around 60 responses per object-distractor pair were used in further analyses, totalling in over three hundred thousand unique responses. For all models tested, we used the outputs of the layer just prior to transformation into 1000-value ImageNet-specific category vectors to construct a linear (logistic regression) decoder from model features. We used the regressions probabilities for each class to compare model choices against actual human responses. This is a correlational measure, meaning that models that do better on behavioral predictivity are better aligned with human responses, making similar correct choices and committing similar errors.

2.3 FEEDFORWARD SIMPLICITY

Given equally predictive models, we prefer a simpler one. We considered several alternative metrics to measure model simplicity (see Appendix B.4), ultimately choosing to compute the number of convolutions and fully-connected layers along the longest path of information flow. For instance, the circuits in each of CORnet-S areas have the length of four since information is passed sequentially through four convolutional layers. Note that we counted recurrent (shared parameter) paths only once. If recurrent paths were counted the number of times information was passed through them, models with shared parameters would be no simpler than those using unique parameters at every time (i.e., feedforward models), which is counterintuitive to us.

We also wanted to emphasize that the difference between a path length of 5 and 10 is much greater than between 105 and 110, so the path length was transformed by a natural logarithm. Finally, we wanted a measure of simplicity, not complexity, so the resulting value was inverted, resulting in the following formulation of Feedforward Simplicity:

$$\text{Feedforward Simplicity} = \frac{1}{\ln\left(\text{Longest path in model}\right)}$$

Given the lack of consensus how many areas primate ventral visual pathway contains, Feedforward Simplicity is currently not included in the composite Brain-Score and is reported separately here.

3 CORNET-S: BRAIN-DRIVEN MODEL ARCHITECTURE

Using Brain-Score and Feedforward Simplicity as our guiding measures, we built CORnet-S. Specifically, our model aims to be (based on Kubilius, 2018):

(1) **Predictive**, so that it is most explanatory of neural and behavioral benchmarks. We are not only interested in having correct model outputs (behaviors) but also internals that match the brain's anatomical and functional constraints. We prefer neural network models because neurons are the units of online information transmission and models without neurons cannot be obviously mapped to neural spiking data (Yamins & DiCarlo, 2016).

(2) **Compact**, i.e. we prefer simpler models among models with similar scores as they are potentially easier to understand and more efficient to experiment with. The human and non-human primate

ventral visual pathway consists of only a handful of areas that process visual inputs: retina, LGN, V1, V2, V4, and a set of areas in the inferior temporal cortex (IT). While the exact number of areas is hard to establish, we ask that models have few areas (though each area may perform multiple operations). The model should thus obtain strong scores on Feedforward Simplicity. Also, we have no strong reason to believe that circuitry should differ across areas in the ventral visual pathway.

(3) **Recurrent**: while core object recognition was originally believed to be largely feedforward because of its fast time scale (DiCarlo et al., 2012), it has long been suspected that recurrent connections must be relevant for some aspects of object perception (Lamme & Roelfsema, 2000; Bar et al., 2006; Wagemans et al., 2012), and recent studies have shown a potential role of recurrent processes even at the fast time scale of core object recognition (Kar et al., 2018; Tang et al., 2018; Rajaei et al., 2018). Moreover, responses in the visual system have a temporal profile, so models at least should be able to produce responses *over time*.

3.1 CORNET-S MODEL SPECIFICS

CORnet-S (Fig. 1) aims to rival the best models on Brain-Score by transforming very deep feedforward architectures into a shallow recurrent model. Specifically, CORnet-S draws inspiration from ResNets that are some of the best models on our behavioral benchmark (Fig. 1; Rajalingham et al., 2018) and can be thought of as unrolled recurrent networks (Liao & Poggio, 2016). Recent studies further demonstrated that weight sharing in ResNets was indeed possible without a significant loss in CIFAR and ImageNet performance (Jastrzebski et al., 2017; Leroux et al., 2018).

Moreover, CORnet-S specifically commits to an anatomical mapping to brain areas. While for comparison models, we establish this mapping by searching for the layer in the model that best explains responses in a given brain area, ideally such mapping would already be provided by the model, leaving no free parameters to researchers. Thus, CORnet-S has four computational areas, conceptualized as analogous to the ventral visual areas in cortex: V1, V2, V4, and IT, and a linear category decoder that maps from the population of neurons in the model's last visual area to its behavioral choices. Importantly, this commitment also means that, for example, if model's area V1 is worse than area V4 at predicting neural responses in visual area V1, then this model would be falsified.

Each visual area implements a particular neural circuitry with neurons performing simple canonical computations: convolution, addition, nonlinearity, response normalization or pooling over a receptive field. In the model presented here, the circuitry is identical in each of its visual areas (except for V1), but we vary the total number of neurons in each area. The details of the layers depicted in Figure 1 are as follows. Due to high computational demands, first area $\mathrm{V1_{COR}}$ performs a 7×7 convolution with stride 2, 3×3 max pooling with stride 2, and a 3×3 convolution. Areas $\mathrm{V2_{COR}}$, $\mathrm{V4_{COR}}$ and $\mathrm{IT_{COR}}$ perform two 1×1 convolutions, a 3×3 convolution with stride 2 and a 1×1 convolution. To implement recurrence, outputs of an area are passed through it a several times to yield the final output of that area. For instance, after $\mathrm{V2_{COR}}$ processed the input once, that result is passed into $\mathrm{V2_{COR}}$ again and treated as a new input. Input changes over time are thus not implemented (denoted as "gate" in Fig. 1). $\mathrm{V2_{COR}}$ and $\mathrm{IT_{COR}}$ are repeated twice, $\mathrm{V4_{COR}}$ is repeated four times. Batch normalization (Ioffe & Szegedy, 2015) was not shared over time as suggested by Jastrzebski et al. (2017). There are no across-area bypass or across-area feedback connections in the current definition of CORnet-S and retinal and LGN processing are omitted.

The decoder part of a model implements a simple linear classifier – a set of weighted linear sums with one sum for each object category. When training on ImageNet, responses of the last model area (and last time step in the case of recurrence) are further passed through a softmax nonlinearity to perform a 1000-way classification. To reduce the amount of neural responses projecting to this classifier, we first average responses over the entire receptive field per feature map.

3.2 COMPARISON TO OTHER MODELS

Liang & Hu (2015) introduced perhaps the first deep recurrent neural network intended for object recognition by adding a variant of a simple recurrent cell to a shallow five-layer convolutional neural network backbone. Zamir et al. (2017) built a more powerful version of recurrent networks by employing LSTM cells. Moreover, they extended the idea of recurrent networks to include

biologically-realistic network unrolling over time. In their architecture, information propagates one layer at a time, so if a network has five layers, the input reaches a classifier only five time steps after it was passed in. A similar approach was used by Spoerer et al. (2017) who showed that a simple version of a recurrent net than can improve network performance on an MNIST-based task. Liao & Poggio (2016) introduced the idea that ResNets can be thought of as recurrent neural networks unrolled over time with non-shared weights, and demonstrated the first working version of a folded ResNet. Jastrzebski et al. (2017) also explored the idea of recurrent ResNets with shared weights and how they could be trained successfully.

However, all of these networks were only tested on CIFAR-100 at best. As noted by Nayebi et al. (2018), while many networks may do well on a simpler task, they may differentiate once the task becomes sufficiently difficult. Moreover, our preliminary testing indicated that non-ImageNet-trained models do not appear to score high on Brain-Score, so even for practical purposes we needed models that could be trained on ImageNet. Leroux et al. (2018) proposed probably the first recurrent architecture that performed well on ImageNet, and skips inference steps when an answer has been found. In an attempt to explore the recurrent net space in a more principled way, Nayebi et al. (2018) performed a large-scale search in the LSTM-based recurrent cell space by allowing the search to find the optimal combination of local and long-range recurrent connections. The best model demonstrated a strong ImageNet performance while being shallower than feedforward controls. In this project, we wanted to go one step further and build a maximally compact model that would nonetheless yield top Brain-Score and outperform other recurrent networks on ImageNet.

4 RESULTS

4.1 BRAIN-SCORE IS CORRELATED WITH CLASSIFICATION PERFORMANCE

We performed a large-scale model comparison using most commonly used neural network families: AlexNet (Krizhevsky et al., 2012), VGG (Simonyan & Zisserman, 2014), ResNet (He et al., 2016), Inception (Szegedy et al., 2015a;b; 2017), SqueezeNet (Iandola et al., 2016), DenseNet (Huang et al., 2017), MobileNet (Howard et al., 2017), and (P)NASNet (Zoph et al., 2017; Liu et al., 2017). These networks were taken from publicly available checkpoints: AlexNet, SqueezeNet, ResNet-{18,34} from PyTorch (Paszke et al., 2017); Inception, ResNet-{50,101,152}, (P)NASNet, MobileNet from TensorFlow-Slim (Silberman & Guadarrama, 2016); and Xception, DenseNet, VGG from Keras (Chollet et al., 2015). As such, the training procedure is different between models and our claims should be taken on those model instantiations and not on architecture families. To further map out the space of possible architectures and add a baseline, we included an in-house-developed family of models (*basenets*): lightweight AlexNet-like architectures with six convolutional layers and a single fully-connected layer, captured at various stages of training. Various hyperparameters were varied between basenets, such as the number of filter maps, nonlinearities, pooling, learning rate etc.

Figure 1 shows how models perform on Brain-Score and ImageNet. The strongest model under our current set of benchmarks is DenseNet-169 with a Brain-Score of .549, closely followed by CORnet-S with a Brain-Score of .544 and ResNet-101 with a Brain-Score of .542. The current top-performing models on ImageNet from the machine learning community all stem from the DenseNet and ResNet families of models. DenseNet-169 and ResNet-101 are also among the highest-scoring models on the IT neural predictivity and the behavioral predictivity respectively with scores of .606 on IT (DenseNet-169, layer *conv5_block31_concat*) and .389 on behavior (ResNet-101, layer *avg_pool*). VGG families win V4 with a score of .672 (VGG-19, layer *block3_pool*). Several observations for other model families are also worth noting: while ANNs from the Inception architectural family improved on ImageNet performance over subsequent versions, their Brain-Score decreased. Another natural cluster emerged with AlexNet and SqueezeNet at the bottom of the ranking: despite reasonable scores on V4 and IT neural predictivity, their behavioral scores are sub-par.

Interestingly, models that score high on brain data are also not the ones ranking the highest on ImageNet performance, suggesting a potential disconnect between ImageNet performance and fidelity to brain mechanisms. For instance, despite its superior performance of 82.90% top-1 accuracy on ImageNet, PNASNet only ranks 13$^{\text{th}}$ on the overall Brain-Score. Models with an ImageNet top-1 performance below 70% show a strong correlation with Brain-Score of .92 ($p < 10^{-14}$) but above 70% ImageNet performance there was no significant correlation ($p \gg .05$, cf. Figure 1). The same trend was apparent in our further search within the CORnet model space: although overall behav-

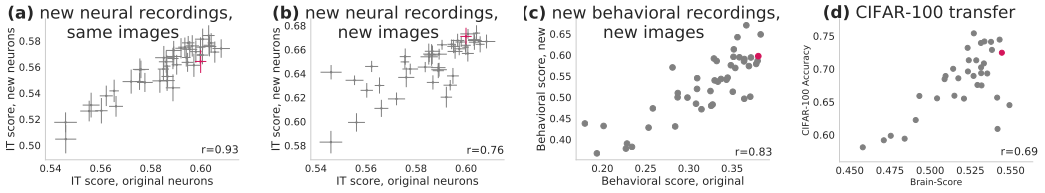

Figure 2: **Brain-Score generalization across datasets:** **(a)** to neural recordings in new subjects with the same stimulus set, **(b)** to neural recordings in new subjects with a very different stimulus set (MS COCO), **(c)** to behavioral responses in new subjects with new object categories, **(d)** to CIFAR-100 transfer performance.

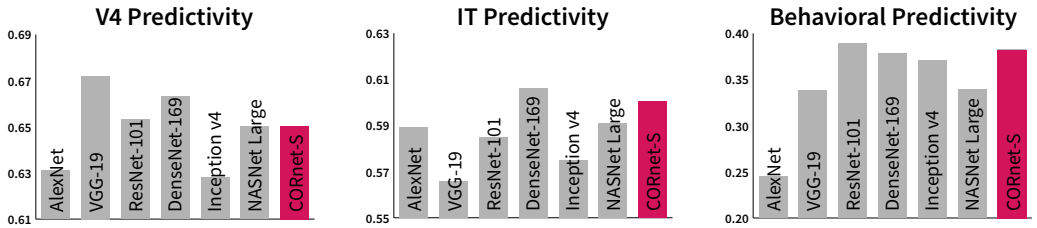

Figure 3: **Comparison of brain scores on several popular models and CORnet-S.** CORnet-S is comparable to the state-of-the-art models on neural predictivity and a top model on behavioral predictivity (see Appendix A for numerical results).

ioral scores were highly correlated with ImageNet performance, the best ImageNet models could nonetheless have only mediocre behavioral scores (Figure 7). Note however that the training procedure is different for these pre-trained models and might have an effect on Brain-Score performance.

4.2 Brain-Score is a robust measure of generalization

We further asked if Brain-Score reflects idiosyncracies of the particular datasets we included in this benchmark or instead, more desirably, provides an overall evaluation of how brain-like models are. To address this question, we performed four different tests with various generalization demands (Fig. 2). First, we compared the scores of models predicting IT neural responses to a set of new IT neural recordings (Kar et al., 2018) where new monkeys were shown the same images as before. We observed a strong correlation between the two sets (Pearson $r = .93$). When compared on predicting IT responses to a very different image set (1600 MS COCO images; Lin et al., 2014), model rankings were still strong (Pearson $r = .76$). We also found a strong correlation between model scores on our original behavioral set and a newly obtained set of behavioral responses to 20 new categories (200 images total; Pearson $r = .83$). Finally, we evaluated model feature generalization to CIFAR-100 without fine-tuning (following Kornblith et al. (2018)). Again, we observed a compelling correlation to Brain-Score values (Pearson $r = .69$). Overall, we expect that adding more benchmarks to Brain-Score will further lead scores to converge.

4.3 CORnet-S is one of the best yet much simpler brain-predicting models

CORnet-S is strong at neural as well as behavioral predictions (Fig. 3), making it one of the best models tested on Brain-Score so far. Critically, CORnet-S is substantially simpler than other top-performing models on Brain-Score (Fig. 4, left) and commits to a particular mapping between model and brain areas. To determine which elements in the circuitry are critical to CORnet-S, we attempted to alter its block structure in Fig. 5. We found that the most important factor was the presence of at least a few steps of recurrence in each block. Having a fairly wide bottleneck (at least 4x expansion) and a skip connection were other important factors. On the other hand, adding more recurrence or having five areas in the model instead of four did not improve the model or hurt its Brain-Score. Other factors affected mostly ImageNet performance, including using two convolutions instead of three within a block, having more areas in the model and using batch normalization per time step

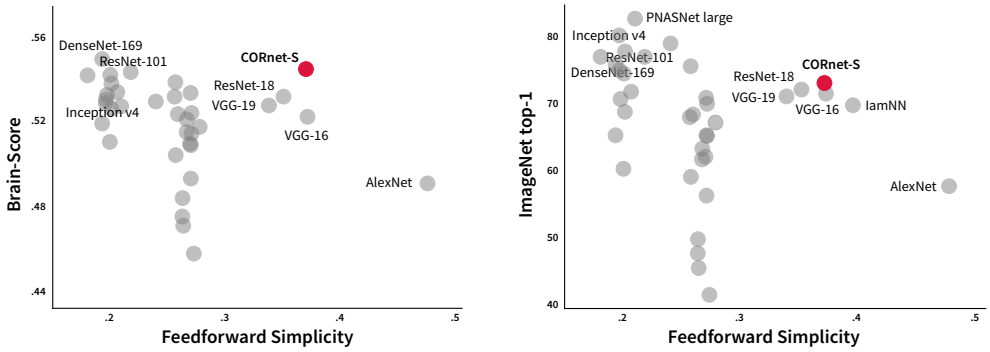

Figure 4: **Feedforward Simplicity versus Brain-Score (left) and ImageNet performance (right).** Most simple models perform poorly on Brain-Score and ImageNet, while best models for explaining brain data are complicated. CORnet-S offers the best of both worlds with the best Brain-Score and ImageNet performance and the highest degree of simplicity we could achieve to date. Note that dots were slightly jittered along the x-axis to improve visibility. Most of these jittered datapoints come from either MobileNet v1 (Feedforward Simplicity around .27) or v2 (Feedforward Simplicity around .2), so all models have the same simplicity (due to the same architecture) but varying Brain-Score (due to varying numbers of neurons).

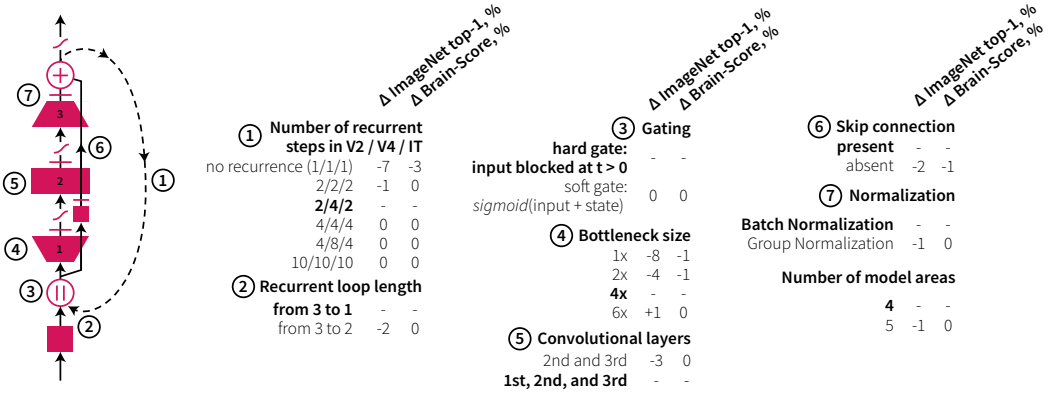

Figure 5: **CORnet-S circuitry analysis.** Each row indicates how ImageNet top-1 performance and Brain-Score change with respect to the baseline model (in bold) when a particular hyperparameter is changed.

instead of a global group normalization (Wu & He, 2018). The type of gating did not seem to matter. However, note that we kept training hyperparameters identical for all these model variants. We therefore cannot rule out that the reported differences could be minimized if more optimal hyperparameters were found.

4.4 CORNET-S IS BEST ON IMAGENET AMONG COMPACT MODELS

Due to anatomical constraints imposed by the brain, CORnet-S's architecture is much more compact than the majority of deep models in computer vision (Fig. 4, right). Compared to similar models with a path length of less than 50, CORnet-S is better in terms of Feedforward Simplicity and outperforms other models on ImageNet top-1 classification accuracy (Table 2). AlexNet and IamNN are simpler models (simplicity .48 (path length 8) and .38 (14) respectively) but suffer on classification accuracy (57.7 and 69.6 top-1 respectively) – CORnet-S provides a trade-off between the two with a Feedforward Simplicity of .37 (path length 15) and top-1 accuracy of 73.1. Several epochs later in training top-1 accuracy actually climbed to 74.4 but since we are optimizing for the brain, we chose the epoch with maximum Brain-Score. For reference, the state-of-the-art large model (Maha-

jan et al., 2018) achieves top-1 accuracy of 85.4, but at the cost of a Feedforward Simplicity of only .22 (path length 101).

5 DISCUSSION

We here presented an initial framework for quantitatively comparing any artificial neural network to the brain's neural network for visual processing. With even the relatively small number of brain benchmarks that we have included so far, the framework already reveals interesting patterns: It extends prior work showing that performance correlates with brain similarity, and our analysis of state-of-the-art networks yielded DenseNet-169, CORnet-S and ResNet-101 as the current best models of the primate visual stream. On the other hand, we also find a potential disconnect between ImageNet performance and Brain-Score, with the winning DenseNet-169 not being the best ImageNet model, and even small networks with poor ImageNet performance achieving reasonable scores.

However, it is possible that the observed lack of correlation is only specific to the way models were trained, as reported recently by Kornblith et al. (2018). For instance, they found that the presence of auxiliary classifiers or label smoothing does not affect ImageNet performance too much but significantly decreases transfer performance, in particular affecting Inception and NASNet family of models, i.e., the ones that performed worse on Brain-Score than their ImageNet performance would imply. Kornblith et al. (2018) reported that retraining these models with optimal settings markedly improved transfer accuracy. Since Brain-Score is also a transfer learning task, we cannot rule out that Brain-Score might change if we retrained the affected models classes. Thus, we reserve our claims only about the specific pre-trained models rather than the whole architecture classes.

The insights gained from Brain-Score together with anatomical constraints led us into developing a relatively shallow recurrent model CORnet-S that is among the top models on Brain-Score yet is competitive on ImageNet, combining the best of both neuroscience desiderata and machine learning engineering requirements, demonstrating that models that satisfy both communities could be developed. While we believe that CORnet-S is a closer approximation to the anatomy of the ventral visual stream than current state-of-the-art deep ANNs because we specifically limit the number of areas and we include recurrence, it is still far from complete in many ways. From a neuroscientist's point of view, on top of the lack of biologically-plausible learning mechanisms (self-supervised or unsupervised), a better model of ventral visual pathway would include more anatomical and circuitry-level details, such as retina or lateral geniculate nucleus. Similarly, adding a skip connection was not informed by cortical circuit properties but rather proposed by He et al. (2016) as a means to alleviate the degradation problem in very deep architectures (where stacking more layers results in decreased performance). But we note that not just any architectural choices work. We have tested hundreds of architectures before finding CORnet-S type of circuitries (Figure 5; Figure 7).

From a machine learning perspective, aligning models to brain data could potentially yield more robust models that, just like human visual system, work well across different datasets and tasks without fine-tuning. We tested this possibility by evaluating how stable Brain-Score scores were against several different datasets. Overall, we observed that models that score high on Brain-Score also tend to score high on other datasets, supporting the idea that Brain-Score reflects how good a model is overall, not just on the three particular neural and behavioral benchmarks that we used. The strength of this relation varies among datasets, as demonstrated by a lower correlation to the MS COCO dataset, suggesting that Brain-Score would benefit from the inclusion of more datasets, such as cartoons (Kubilius et al., 2018) or short videos (Monfort et al., 2018)).

More broadly, the utility of Brain-Score for determining how aligned a model is to primate brain as opposed to e.g. ImageNet as a proxy is that Brain-Score quantifies directly how brain-like a given model is; measuring the success directly on the system of interest, i.e. the primate brain. It is a constantly evolving benchmark and with the inclusion of more data (e.g., recordings from other brain areas, behavioral data on other image sets, inclusion of other tasks), we are never guaranteed that brain-like models will continue to follow ImageNet performance. Therefore, for those who wish to build the models of the brain, whether for better brain-machine interfaces or out of pure interest to "understand" the brain, Brain-Score is the direct measure of our progress towards that goal.

REFERENCES

Moshe Bar, Karim S Kassam, Avniel Singh Ghuman, Jasmine Boshyan, Annette M Schmid, Anders M Dale, Matti S Hämäläinen, Ksenija Marinkovic, Daniel L Schacter, Bruce R Rosen, et al. Top-down facilitation of visual recognition. *Proceedings of the national academy of sciences*, 103 (2):449–454, 2006.

François Chollet et al. Keras. `https://keras.io`, 2015.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255. IEEE, jun 2009. ISBN 978-1-4244-3992-8. doi: 10.1109/CVPR.2009.5206848.

James J DiCarlo, Davide Zoccolan, and Nicole C Rust. How does the brain solve visual object recognition? *Neuron*, 73(3):415–434, 2012.

Kunihiko Fukushima. Neocognitron: a self organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36(4):193–202, 1980.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *arXiv preprint arXiv:1704.04861*, 2017.

Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *CVPR*, volume 1, pp. 3, 2017.

David H Hubel and Torsten N Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of physiology*, 160(1):106–154, 1962.

Forrest N. Iandola, Song Han, Matthew W. Moskewicz, Khalid Ashraf, William J. Dally, and Kurt Keutzer. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and. *arXiv preprint arXiv:1602.07360*, 2016.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

Stanisaw Jastrzebski, Devansh Arpit, Nicolas Ballas, Vikas Verma, Tong Che, and Yoshua Bengio. Residual connections encourage iterative inference. *arXiv preprint arXiv:1710.04773*, 2017.

Kohitij Kar, Jonas Kubilius, Kailyn M Schmidt, Elias B Issa, and James J DiCarlo. Evidence that recurrent circuits are critical to the ventral stream's execution of core object recognition behavior. *bioRxiv*, pp. 354753, 2018.

Simon Kornblith, Jonathon Shlens, and Quoc V. Le. Do Better ImageNet Models Transfer Better? *arXiv preprint arXiv:1805.08974*, 2018.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pp. 1097–1105, 2012.

Jonas Kubilius. Predict, then simplify. *NeuroImage*, 180:110 – 111, 2018.

Jonas Kubilius, Kohitij Kar, Kailyn Schmidt, and James J DiCarlo. Can deep neural networks rival human ability to generalize in core object recognition? In *Cognitive Computational Neuroscience*, 2018. URL `https://ccneuro.org/2018/Papers/ViewPapers.asp?PaperNum=1234`.

Victor AF Lamme and Pieter R Roelfsema. The distinct modes of vision offered by feedforward and recurrent processing. *Trends in neurosciences*, 23(11):571–579, 2000.

Sam Leroux, Pavlo Molchanov, Pieter Simoens, Bart Dhoedt, Thomas Breuel, and Jan Kautz. Iamnn: Iterative and adaptive mobile neural network for efficient image classification. *arXiv preprint arXiv:1804.10123*, 2018.

Ming Liang and Xiaolin Hu. Recurrent convolutional neural network for object recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

Qianli Liao and Tomaso Poggio. Bridging the gaps between residual learning, recurrent neural networks and visual cortex. *arXiv preprint arXiv:1604.03640*, 2016.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.

Chenxi Liu, Barret Zoph, Maxim Neumann, Jonathon Shlens, Wei Hua, Li-Jia Li, Li Fei-Fei, Alan Yuille, Jonathan Huang, and Kevin Murphy. Progressive Neural Architecture Search. *arXiv preprint*, 2017. URL `https://arxiv.org/pdf/1712.00559.pdfhttp://arxiv.org/abs/1712.00559`.

Dhruv Mahajan, Ross Girshick, Vignesh Ramanathan, Kaiming He, Manohar Paluri, Yixuan Li, Ashwin Bharambe, and Laurens van der Maaten. Exploring the limits of weakly supervised pretraining. *arXiv preprint arXiv:1805.00932*, 2018.

Najib J Majaj, Ha Hong, Ethan A Solomon, and James J DiCarlo. Simple learned weighted sums of inferior temporal neuronal firing rates accurately predict human core object recognition performance. *Journal of Neuroscience*, 35(39):13402–13418, 2015.

Warren S. McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 5(4):115–133, dec 1943. ISSN 00074985. doi: 10.1007/BF02478259.

Mathew Monfort, Bolei Zhou, Sarah Adel Bargal, Tom Yan, Alex Andonian, Kandan Ramakrishnan, Lisa Brown, Quanfu Fan, Dan Gutfruend, Carl Vondrick, et al. Moments in time dataset: one million videos for event understanding. *arXiv preprint arXiv:1801.03150*, 2018.

Aran Nayebi, Daniel Bear, Jonas Kubilius, Kohitij Kar, Surya Ganguli, David Sussillo, James J DiCarlo, and Daniel LK Yamins. Task-driven convolutional recurrent models of the visual system. *arXiv preprint arXiv:1807.00053*, 2018.

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

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

Karim Rajaei, Yalda Mohsenzadeh, Reza Ebrahimpour, and Seyed-Mahdi Khaligh-Razavi. Beyond core object recognition: Recurrent processes account for object recognition under occlusion. *bioRxiv*, pp. 302034, 2018.

Rishi Rajalingham, Kailyn Schmidt, and James J DiCarlo. Comparison of object recognition behavior in human and monkey. *Journal of Neuroscience*, 35(35):12127–12136, 2015.

Rishi Rajalingham, Elias B Issa, Pouya Bashivan, Kohitij Kar, Kailyn Schmidt, and James J DiCarlo. Large-scale, high-resolution comparison of the core visual object recognition behavior of humans, monkeys, and state-of-the-art deep artificial neural networks. *Journal of Neuroscience*, pp. 0388–18, 2018.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.

N. Silberman and S. Guadarrama. Tensorflow-slim image classification model library. https://github.com/tensorflow/models/tree/master/research/slim, 2016.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Courtney J Spoerer, Patrick McClure, and Nikolaus Kriegeskorte. Recurrent convolutional neural networks: a better model of biological object recognition. *Frontiers in psychology*, 8:1551, 2017.

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 07-12-June, sep 2015a. ISBN 9781467369640. doi: 10.1109/CVPR.2015.7298594. URL http://arxiv.org/abs/1409.4842.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the Inception Architecture for Computer Vision. *arXiv preprint*, 2015b. ISSN 08866236. doi: 10.1109/CVPR.2016.308. URL https://arxiv.org/pdf/1512.00567.pdfhttp://arxiv.org/abs/1512.00567.

Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander A Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *AAAI*, volume 4, pp. 12, 2017.

Hanlin Tang, Martin Schrimpf, William Lotter, Charlotte Moerman, Ana Paredes, Josue Ortega Caro, Walter Hardesty, David Cox, and Gabriel Kreiman. Recurrent computations for visual pattern completion. *Proceedings of the National Academy of Sciences*, pp. 201719397, 2018.

Johan Wagemans, James H Elder, Michael Kubovy, Stephen E Palmer, Mary A Peterson, Manish Singh, and Rüdiger von der Heydt. A century of gestalt psychology in visual perception: I. perceptual grouping and figure–ground organization. *Psychological bulletin*, 138(6):1172, 2012.

Yuxin Wu and Kaiming He. Group normalization. *arXiv preprint arXiv:1803.08494*, 2018.

Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*, pp. 5987–5995. IEEE, 2017.

Daniel L Yamins, Ha Hong, Charles Cadieu, and James J DiCarlo. Hierarchical modular optimization of convolutional networks achieves representations similar to macaque it and human ventral stream. In *Advances in neural information processing systems*, pp. 3093–3101, 2013.

Daniel LK Yamins and James J DiCarlo. Using goal-driven deep learning models to understand sensory cortex. *Nature neuroscience*, 19(3):356, 2016.

Daniel LK Yamins, Ha Hong, Charles F Cadieu, Ethan A Solomon, Darren Seibert, and James J DiCarlo. Performance-optimized hierarchical models predict neural responses in higher visual cortex. *Proceedings of the National Academy of Sciences*, 111(23):8619–8624, 2014.

Amir R Zamir, Te-Lin Wu, Lin Sun, William B Shen, Bertram E Shi, Jitendra Malik, and Silvio Savarese. Feedback networks. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*, pp. 1808–1817. IEEE, 2017.

Barret Zoph, Vijay Vasudevan, Jonathon Shlens, and Quoc V. Le. Learning Transferable Architectures for Scalable Image Recognition. *arXiv preprint*, jul 2017. ISSN 10659471. doi: 10.1109/CVPR.2018.00907.

A NUMERICAL BRAIN-SCORES

Table 1: Brain-Scores and individual performances for state-of-the-art models

		neural predictivity		behavioral	top-1 accuracy
Brain-Score	model	V4	IT	predictivity	ImageNet
.549	densenet-169	.663	**.606**	.380	75.90
.545	CORnet-S	.653	.600	.380	74.70
.543	resnet-101_v2	.661	.585	**.383**	77.00
.542	densenet-121	.657	.597	.372	74.50
.542	densenet-201	.655	.601	.369	77.00
.538	resnet-50_v2	.663	.586	.366	75.60
.538	resnet-152_v2	.663	.586	.364	77.80
.535	inception_v2	.658	.595	.354	73.90
.534	best mobilenet	.645	.600	.356	71.80
.534	pnasnet_large	.650	.587	.364	**82.90**
.532	inception_v1	.661	.576	.358	69.80
.532	resnet-18	.648	.584	.364	69.76
.532	inception_resnet_v2	.652	.592	.351	80.40
.529	xception	.671	.565	.352	79.00
.529	inception_v4	.639	.576	.371	80.20
.528	vgg-19	**.672**	.566	.345	71.10
.527	nasnet_large	.659	.589	.334	82.70
.527	inception_v3	.660	.589	.332	78.00
.523	resnet-34	.632	.560	.378	73.30
.522	vgg-16	.669	.572	.326	71.50
.509	nasnet_mobile	.651	.594	.281	74.00
.504	best basenet	.663	.594	.256	47.64
.491	alexnet	.631	.589	.253	57.70
.470	squeezenet1_1	.654	.556	.201	57.50
.459	squeezenet1_0	.653	.546	.180	57.50

B BRAIN-SCORE BENCHMARK DETAILS

In the following section we outline the benchmarks that models are measured against. A benchmark consists of a metric applied to a specific set of experimental data, which here can be either neural recordings or behavioral measurements.

B.1 NEURAL

The purpose of neural metrics is to establish how well internal representations of a source system (e.g., a neural network model) match the internal representations in a target system (e.g., a primate). Unlike typical machine learning benchmarks, these metrics provide a principled way to prefer some models over others even if their outputs are identical. We outline here one common metric, Neural Predictivity, which is a form of a linear regression.

Neural Predictivity: Image-Level Neural Consistency Neural Predictivity is used to evaluate how well responses $\mathbf{X}$ to given images in a source system (e.g., a deep ANN) predict the responses in a target system (e.g., a single neuron's response in visual area IT). As inputs, this metric requires two assemblies of the form stimuli$\times$neuroid where neuroids can either be neural recordings or model activations. First, source neuroids are mapped to each target neuroid using a linear transformation:

$$y = \mathbf{X}w + \epsilon,$$

where w denotes linear regression weights and ϵ is the noise in the neural recordings. This mapping procedure is performed on multiple train-test splits across stimuli. In each run, the weights are fit to map from source neuroids to a target neuroid using training images, and then using these weights predicted responses y' are obtained for the held-out images. We used the neuroids from V4 and IT separately to compute these fits.

To obtain a neural predictivity score for each neuroid, we compare predicted responses y' with the measured neuroid responses y by computing the Pearson correlation coefficient r:

$$r = \frac{\sum_{i=1}^{n}(y_i - \overline{y})(y'_i - \overline{y}')}{\sqrt{\sum_{i=1}^{n}(y_i - \overline{y})^2 (y'_i - \overline{y}')^2}} \tag{1}$$

A median over all individual neuroid neural predictivity values (e.g., all measured target sites in a target brain region) is computed to obtain a predictivity score for that train-test split (median is used since responses are typically distributed non-normally). The final neural predictivity score for the target brain region is computed as the mean across all train-test splits.

We further estimate the internal consistency between neural responses by splitting neural responses in half across repeated presentations of the same image and computing Spearman-Brown-corrected Pearson correlation coefficient (Eq. 1) between the two splits across images for each neuroid.

In practice, we found that standard linear regression is comparably slow given a large dimensionality of the source system and not sufficiently robust. Thus, following Yamins et al. (2014), we use a partial least squares (PLS) regression with 25 components. We further optimized this procedure by first projecting source features into a lower-dimensional space using principal components analysis. The projection matrix is obtained for the features of a selection of ImageNet images, so that the projection is constant across train-test splits. This projection matrix is then used to transform source features. Results reported here were obtained by retaining 1000 principal components from the feature responses per layer to 1000 ImageNet validation images that captured the most variance of a source model.

Neural Recordings The neural dataset currently used in both neural benchmarks included in this version of Brain-Score is comprised of neural responses to 2,560 naturalistic stimuli in 88 V4 neurons and 168 IT neurons (cf. Figure 1), collected by Majaj et al. (2015). The image set consists of 2,560 grayscale images in eight object categories (animals, boats, cars, chairs, faces, fruits, planes, tables). Each category contains eight unique objects (for instance, the "face" category has eight unique faces). The image set was generated by pasting a 3D object model on a naturalist background. In each image, the position, pose, and size of an object was randomly selected in order to

create a challenging object recognition task both for primates and machines. A circular mask was applied to each image (see Majaj et al. (2015) for details on image generation).

Two macaque monkeys were implanted three arrays each, with one array placed in area V4 and the other two placed on the posterior-anterior axis of IT cortex. The monkeys passively observed a series of images (100 ms image duration with 100 ms of gap between each image) that each subtended approximately 8 deg visual angle. To obtain a stable estimate of the neural responses to each image, each each image was re-tested about 50 times (re-tests were randomly interleaved with other images). In the benchmarks used here, we used an average neural firing rate (normalized to a blank gray image response) in the window between 70 ms and 170 ms after image onset where the majority of object category-relevant information is contained (Majaj et al., 2015).

B.2 BEHAVIORAL

The purpose of behavioral benchmarks it to compute the similarity between source (e.g., an ANN model) and target (e.g., human or monkey) behavioral responses in any given task. For core object recognition tasks, primates (both human and monkey) exhibit behavioral patterns that differ from ground truth labels. Thus, our primary benchmark here is a behavioral response pattern metric, not an overall accuracy metric, and higher scores are obtained by ANNs that produce and predict the primate patterns of successes and failures. One consequence of this is that ANNs that achieve 100% accuracy will not achieve a perfect behavioral similarity score.

Even within the visual behavioral domain of core object recognition, there are many possible behavioral metrics. We here use the metric of the image-by-image patterns of difficulty, broken down by the object choice alternatives (termed *I2n*), because recent work (Rajalingham et al., 2018) suggests that it has the most power to distinguish among alternative ANNs (assuming that sufficient amounts of behavioral data are available).

I2n: Normalized Image-Level Behavioral Consistency Source data (model features) for a total of i images are transformed first into a $i_b \times c$ matrix of c object categories and i_b images with behavioral data available using the following procedure. First, images where behavioral responses are not available (namely, $i - i_b$ images) are used to build a c-way logistic regression from source data to a c-value probability vector for each image, where each probability is the probability that a given object is in the image. This regression is then used to estimate probabilities for the held-out i_b images. For each image, all normalized target-distractor pair probabilities are computed from the c-way probability vector. For instance, if an image contains a dog and the distractor is a bear, the target-distractor score is $\frac{p(\text{dog})}{p(\text{dog})+p(\text{bear})}$.

In order to compare source and target data, we first transform these raw accuracies in the $i_b \times c$ response matrix to a d' measure for each cell in the $i_b \times c$ matrix:

$$d' = Z(\text{Hit Rate}) - Z(\text{False Alarms Rate}),$$

where Z is the estimated z-score of responses, Hit Rate is the accuracy of a given target-distractor pair while the False Alarms Rate corresponds to how often the observers incorrectly reported seeing that target object in images where another object was presented. For instance, if a given image contains a dog and distractor is a bear, the Hit Rate for the dog-bear pair for that image comes straight from the $i_b \times c$ matrix, while in order to obtain the False Alarms Rate, all cells from that matrix that did not have dogs in the image but had a dog as a distractor are averaged, and 1 minus that value is used as a False Alarm Rate. All d' above 5 were clipped. This transformation helps to remove bias in responses and also to diminish ceiling effects (since many primate accuracies were close to 1), but empirically observed benefits of d' in this dataset are small; see Rajalingham et al. (2018) for a thorough explanation.

The resulting response matrix is further refined by subtracting the mean d' across trials of the same target-distractor pair (e.g., for dog-bear trials, their mean is subtracted from each trial). Such normalization exposes variance unique to each image and removes global trends that may be easier for models to capture. For instance, dog-bear trials on average could have been harder than dog-zebra trials. Without this normalization, a model might score very well by only capturing this tendency.

After normalization, all responses are centered around zero, and thus capturing only global trends but not each image's idiosyncrasies would be insufficient for a model to rank well.

After normalization, a Pearson correlation coefficient r_{st} between source and target data is computed using Eq. 1. We further estimate noise ceiling, that is, how well an ideal model could perform given the noise in the measured behavioral responses, by dividing target data in half across trials, computing the normalized d' $i_b \times c$ matrices for each half, and computing the Pearson correlation coefficient r_{tt} between the two halves. If source data is produced by a stochastic process, the same procedure can be carried out on the source data, resulting in the source's reliability r_{ss}.

The final behavioral predictivity score of each ANN is then computed by:

$$r = \frac{r_{st}}{\sqrt{r_{ss} r_{tt}}}$$

All models that we tested so far produced deterministic responses, thus $r_{ss} = 1$ in our scoring.

Primate behavioral data The behavioral data used in the current round of benchmarks was obtained by Rajalingham et al. (2015) and Rajalingham et al. (2018). Here we focus on only the human behavioral data, but the human and non-human primate behavioral patterns are very similar to each other (Rajalingham et al., 2015; 2018).

The image set used in this data collection was generated in a similar way as the images for V4 and IT using 24 object categories. In total, the dataset contains 2,400 images (100 per object). For this benchmark, we used 240 (10 per object) of these images for which the most trials were obtained. 1,472 human observers responded to briefly presented images on Amazon Mechanical Turk. At each trial, an image was presented for 100 ms, followed by two response choices, one corresponding to the target object present in the image and the other being one of the remaining 23 objects (i.e., a distractor object). Participants responded by choosing which object was presented in the image. Thus, over three hundred thousand responses for each target-distractor pair were obtained from multiple participants, resulting in a 240 (images) $\times 24$ (objects) response matrix when averaged across participants.

B.3 Predictors of neural scores

We compared model scores on neural (V4, IT) recordings with the scores on behavioral recordings to see if e.g. a behavioral benchmark alone would already be sufficient or if the entire set of benchmarks is necessary. We found that there was a correlation to behavior ($.65$ for V4 and $.87$ or IT) which is strong enough to connect neurons to behavior but not sufficient for behavior alone to explain the entire neural population, warranting a composite set of benchmarks.

Moreover, we tested if the number of features in model layers might predict the neural scores. Even though we PCA all features to 1,000 components, higher dimensionality might result in better scores. Following Figure 6, we found this not be the case: models with the same number of neurons have scores across the board, for all number of features.

B.4 Feedforward Simplicity

From a neuroscience point of view, simpler models can be better mapped to cortex and be better analyzed and understood with regard to the brain. Simpler models can also be better made sense of sense in terms of what components constitute a strong model by reducing models to their most essential elements. One possibility was to use the total number of parameters (weights). However, it did not seem to map well to simplicity in neuroscience terms. For instance, a single convolutional layer with many filter maps could have many parameters yet it seems much simpler than a multilayer branching structure, like the Inception block (Szegedy et al., 2017), that may have less parameters overall.

Moreover, our models are always tested on independent data sampled from different distributions than the train data. Thus, after training a model, all these parameters were fixed for the purposes of brain benchmarks, and the only free parameters are the ones introduced by the linear decoder that is trained on top of the frozen models parameters (see above for decoder details).

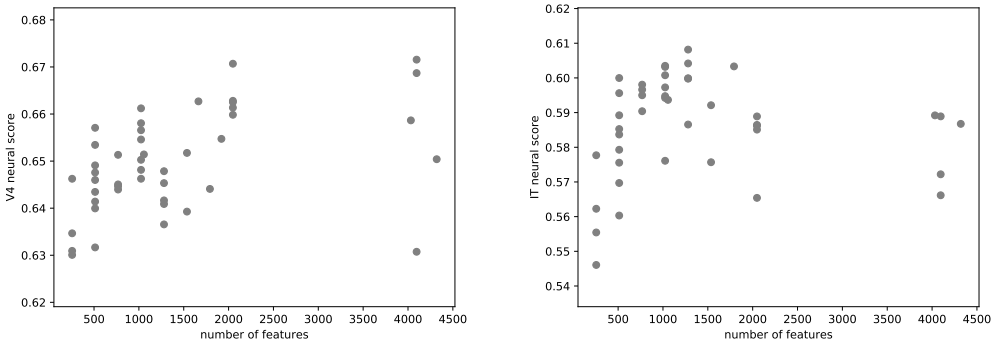

Figure 6: **Neural Scores do not depend on number of features.** We plot the number of features in models' highest-scoring layers against their neural (V4 and IT) scores. The number of neurons does not appear to be a predictor of better brain-likeness.

Table 2: **Comparison of compact models on Feedforward Simplicity and classification accuracy.** CORnet-S outperforms comparable models on simplicity and ImageNet top-1 performance.

Network	**Simplicity (path length)**	**ImageNet top-1 / 5**
AlexNet (Krizhevsky et al., 2012)	**.48 (8)**	57.7 / 79.1
VGG-16 (Simonyan & Zisserman, 2014)	.36 (16)	71.5 / 90.4
VGG-19 (Simonyan & Zisserman, 2014)	.34 (19)	72.4 / 90.9
ResNet-18 (He et al., 2016)	.35 (18)	69.8 / 89.1
SqueezeNet (Iandola et al., 2016)	.35 (18)	57.5 / 80.3
IamNN (Leroux et al., 2018)	.38 (14)	69.6 / 89.0
MobileNet-224 (Howard et al., 2017)	.27 (41)	70.6 / 89.5
CORnet-S	.37 (15)	**73.1 / 91.1**

We also considered computing the total number of convolutional and fully-connected layers, but some models, like Inception, perform some convolutions in parallel, while others, like ResNeXt (Xie et al., 2017), group multiple convolutions into a single computation. We thus decided to you the "longest path" definition as described in the main text.

B.5 BRAIN-SCORE

To evaluate how well a model is doing overall, we computed the global Brain-Score as a composite of neural V4 predictivity score, neural IT predictivity score, and behavioral I2n predictivity score (each of these scores was computed as described above). The Brain-Score presented here is the mean of the three scores. This approach of taking the does not normalize by different scales of the scores so it may be penalizing scores with low variance. However, the alternative approach of ranking models on each benchmark separately and then taking the mean rank would impose the strong assumption that for any two models with (even insignificantly) different scores, their ranks are also different. We thus chose to take the mean score to preserve the distance in values.

C CORNET-S DETAILS

C.1 IMPLEMENTATION DETAILS

We used PyTorch 0.4.1 and trained the model using ImageNet 2012 (Russakovsky et al., 2015). Images were preprocessed (1) for training, with random crops to 224 x 224 pixels, randomly flipped left and right and normalized by mean subtraction and division by standard deviation of the dataset; (2) for validation, with central crops to 224 x 224 pixels and normalized by mean subtraction and division by standard deviation of the dataset. We used a batch size of 256 images and trained on 2 GPUs (NVIDIA Titan X / GeForce 1080Ti) for 43 epochs. We use similar learning rate scheduling to ResNet with more variable learning rate updates (primarily in order to train faster): 0.1, divided by 10 every 20 epochs. For optimization, we use Stochastic Gradient Descent with momentum .9, a cross-entropy loss between image labels and model predictions (logits). We will open-source our code and weights through GitHub.

C.2 MODEL SEARCH

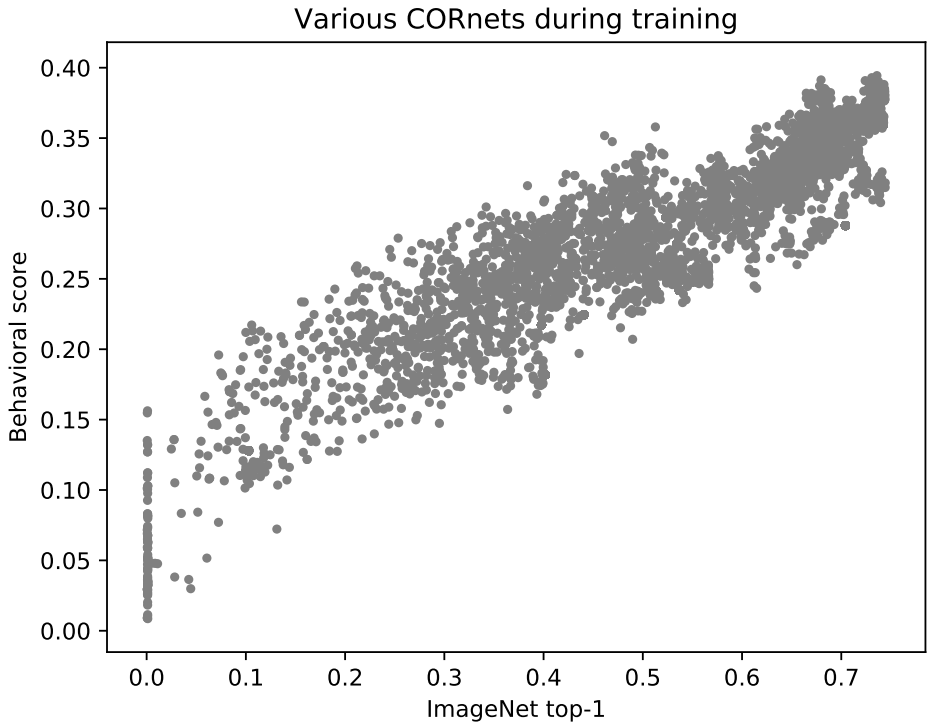

Figure 7: **ImageNet top-1 performance vs. behavioral benchmark on various CORnets.** We manually tried many different configurations of CORnet circuitry. The figure is showing how behavioral benchmark of Brain-Score is related to ImageNet top-1 performance in 106 CORnet configurations. Each dot corresponds to a particular CORnet at a particular point during training. The correlation between ImageNet top-1 performance and CORnet is robust but there is also high variance in this relationship. In particular, notice how some models achieve close to 75% ImageNet performance but show only a mediocre behavioral score. Thus, optimizing solely for ImageNet is not guaranteed at all to lead to a good alignment to brain data.

D GENERALIZATION TO OTHER DATASETS

D.1 NEURAL: NEW NEURONS, OLD IMAGES

We evaluated models on an independently collected neural dataset (288 neurons, 2 monkeys, 63 trials per image; Kar et al. (2018)) where new monkeys were presented with a subset of 640 images from the 2760 images we used for neural predictivity.

D.2 BEHAVIORAL: NEW IMAGES

We collected a new behavioral dataset, consisting of 200 images (20 objects $\times$ 10 images) from Amazon Mechanical Turk users (185,106 trials in total). We used the same experimental paradigm as in our original behavioral test but none of the objects were from the same category as before.

D.3 NEURAL: NEW NEURONS, COCO IMAGES

We obtained a neural dataset from (Kar et al., 2018) for a selection of 1600 of MS COCO images (Lin et al., 2014). These images are very dissimilar from the synthetic images we used in other tests, providing a strong means to test Brain-Score generalization. The dataset consisted of 288 neurons from 2 monkeys and 45 trials per image. Unlike our previous datasets, this one had a low internal consistency between neural responses, presumably due to the electrodes being near their end of life and producing unreasonably high amounts of noise. We therefore only used the 86 neurons with internal consistency of at least 0.9.

D.4 CIFAR-100

Following the procedure described in Kornblith et al. (2018), we tested how well these models generalize to CIFAR-100 dataset by only allowing a linear classifier to be retrained for the 100-way classification task (that is, without doing any fine-tuning). As in Kornblith et al. (2018), we used a scikit-learn implementation of a multinomial logistic regression using L-BFGS (Pedregosa et al., 2011), with the best C parameter found by searching a range from .0005 to .05 in 10 logarithmic steps (40,000 images from CIFAR-100 train set were used for training and the remaining 10,000 for testing; the search range was reduced from Kornblith et al. (2018) because in our earlier tests we found that all models had their optimal parameters in this range). Accuracies reported on the 10,000 test images.

D.5 NEURAL: EARLY AND LATE PREDICTIONS

Focusing on the temporal aspect of our neural data, we divided spike rates into an early time bin ranging from 90-110 ms and a late time bin from 190-210 ms. We found that this early-late division highlighted functional model difference more prominently than the mean temporal prediction in Nayebi et al. (2018). For instance, Figure 8 shows how IT is predicted well by strong ImageNet models at a late stage, but not at early stages. CORnet-S does well on both of these predictions.

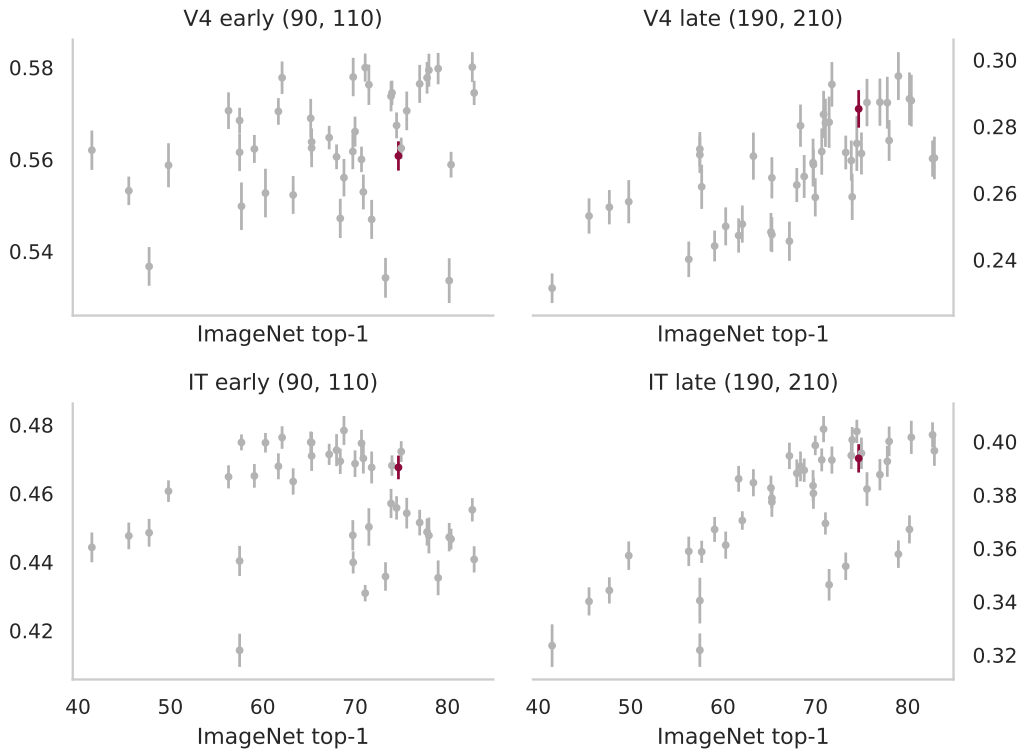

Figure 8: **Prediction correlations on early and late spike rates.** We compare ImageNet performance against pearson correlation of predicted spike rates with neural data binned into early (90-110 ms) and late (190-210 ms). Model mappings are performed separately per bin, layers are chosen based on 70-170 ms scores. Notice how better ImageNet models are better at predicting late IT responses, but not early ones.