Deep Neural Networks Rival the Representation of Primate IT Cortex for Core Visual Object Recognition

Charles F. Cadieu[1]*, Ha Hong[1,2], Daniel L. K. Yamins[1], Nicolas Pinto[1], Diego Ardila[1], Ethan A. Solomon[1], Najib J. Majaj[1], James J. DiCarlo[1]

1 Department of Brain and Cognitive Sciences and McGovern Institute for Brain Research, Massachusetts Institute of Technology, Cambridge, Massachusetts, United States of America, 2 Harvard–MIT Division of Health Sciences and Technology, Institute for Medical Engineering and Science, Massachusetts Institute of Technology, Cambridge, Massachusetts, United States of America

Abstract

The primate visual system achieves remarkable visual object recognition performance even in brief presentations, and under changes to object exemplar, geometric transformations, and background variation (a.k.a. core visual object recognition). This remarkable performance is mediated by the representation formed in inferior temporal (IT) cortex. In parallel, recent advances in machine learning have led to ever higher performing models of object recognition using artificial deep neural networks (DNNs). It remains unclear, however, whether the representational performance of DNNs rivals that of the brain. To accurately produce such a comparison, a major difficulty has been a unifying metric that accounts for experimental limitations, such as the amount of noise, the number of neural recording sites, and the number of trials, and computational limitations, such as the complexity of the decoding classifier and the number of classifier training examples. In this work, we perform a direct comparison that corrects for these experimental limitations and computational considerations. As part of our methodology, we propose an extension of "kernel analysis" that measures the generalization accuracy as a function of representational complexity. Our evaluations show that, unlike previous bio-inspired models, the latest DNNs rival the representational performance of IT cortex on this visual object recognition task. Furthermore, we show that models that perform well on measures of representational performance also perform well on measures of representational similarity to IT, and on measures of predicting individual IT multi-unit responses. Whether these DNNs rely on computational mechanisms similar to the primate visual system is yet to be determined, but, unlike all previous bio-inspired models, that possibility cannot be ruled out merely on representational performance grounds.

Introduction

Primate vision achieves a remarkable proficiency in object recognition, even in brief visual presentations and under changes to object exemplar, geometric transformations, and background variation. Humans [1] and macaques [2] are known to solve this task with high accuracy at low latency for presentation times shorter than 100 ms [3,4]. This ability is likely related to the presence and rate of saccadic eye movements, which for natural viewing typically occur at a rate of one saccade every 200–250 ms [5]. Therefore, when engaged in natural viewing the primate visual system is proficient at recognizing and making rapid and accurate judgements about the objects present within a single saccadic fixation. While not encompassing all of primate visual abilities, this ability is an important subproblem that we operationally define and refer to as "core visual object recognition" [6].

A key to this primate visual object recognition ability is the representation that the cortical ventral stream creates from visual signals from the eye. The ventral stream is a series of cortical visual areas extending from primary visual area V1, through visual areas V2 and V4, and culminating in inferior temporal (IT) cortex. At the end of the ventral stream, IT cortex creates a representation of visual stimuli that is selective for object identity and tolerant to nuisance parameters such as object position, scale, pose, and background [7–10]. The responses of IT neurons are remarkable because they indicate that the ventral stream has transformed the complicated, non-linear object recognition problem at the retinae into a new neural representation that separates objects based on their category [6,11]. Results using linear classifiers have shown that the IT neural representation creates a simpler object recognition problem that can often be solved with a linear function predictive of object category [9,10]. It is thought that this transformation is achieved through the ventral stream by a series of recapitulated modules that each produce a non-linear transformation of their input that becomes selective for objects and tolerant to nuisance variables unrelated to object identity [6].

A number of bio-inspired models have sought to replicate the phenomenology observed in the primate ventral stream (see e.g.

Author Summary

Primates are remarkable at determining the category of a visually presented object even in brief presentations, and under changes to object exemplar, position, pose, scale, and background. To date, this behavior has been unmatched by artificial computational systems. However, the field of machine learning has made great strides in producing artificial deep neural network systems that perform highly on object recognition benchmarks. In this study, we measured the responses of neural populations in inferior temporal (IT) cortex across thousands of images and compared the performance of neural features to features derived from the latest deep neural networks. Remarkably, we found that the latest artificial deep neural networks achieve performance equal to the performance of IT cortex. Both deep neural networks and IT cortex create representational spaces in which images with objects of the same category are close, and images with objects of different categories are far apart, even in the presence of large variations in object exemplar, position, pose, scale, and background. Furthermore, we show that the top-level features in these models exceed previous models in predicting the IT neural responses themselves. This result indicates that the latest deep neural networks may provide insight into understanding primate visual processing.

[12–16]) and recent, related models in the machine learning community, generally referred to as "deep neural networks" share many properties with these bio-inspired models. The computational concepts utilized in these models date back to early models of the primate visual system in the work of Hubel and Wiesel [17,18], who hypothesized that within primary visual cortex more complex functional responses ("complex" cells) were constructed from more simplistic responses ("simple" cells). Models of biological vision have extended this hypothesis by suggesting that higher visual areas recapitulate this mechanism and form a hierarchy [12,13,19–22]. In the last few years, a series of visual object recognition systems have been produced that utilize deep neural networks and have achieved state-of-the-art performance on computer vision benchmarks (see e.g. [23–26]). These deep neural networks implement architectures containing successive layers of operations that resemble the simple and complex cell hierarchy first described by Hubel and Wiesel. However, unlike previous bio-inspired models, these latest deep neural networks contain many layers of computation (typically 7–9 layers, while previous models contained 3–4) and adapt the parameters of the layers using supervised learning on millions of object-labeled images (the parameters of previous models were either hand-tuned, adapted through unsupervised learning, or trained on just thousands of labeled images). Given the increased complexity of these deep neural networks and the dramatic increases in performance over previous models, it is relevant to ask, "how close are these models to achieving object recognition representational performance that is similar to that observed in IT cortex?" In this work we seek to address this question.

Our methodology directly compares the representational performance of IT cortex to deep neural networks and overcomes the shortcoming of previous comparisons. There are four areas where our approach has advantages over previous attempts. Although previous attempts have addressed one or two of these shortcomings, none has addressed all four. First, previous attempts have not corrected for a number of experimental limitations including the amount of experimental noise, the number of recorded neural sites, or the number of recorded stimulus presentations (see e.g. [9,10,27]). Our methodology makes explicit these limitations by either correcting for, or modifying model representations to arrive at a fair comparison to neural representation. We find that these corrections have a dramatic effect on our results and shed light on previous comparisons that we believe may have been misleading.

Second, previous attempts have utilized fixed complexity classifiers and have not addressed the relationship between classifier complexity and decision boundary accuracy (see e.g. [9,10,27]). In our methodology we utilize a novel extension of "kernel analysis," formulated in the works of [28–30], to measure the accuracy of a representation as a function of the complexity of the task decision boundary. This allows us to identify representations that achieve high accuracy for a given complexity and avoids a measurement confound that arises when using cross-validated accuracy: the decision boundary's complexity and/or constraints are dependent on the size and choice of the training dataset, factors that can strongly affect accuracy scores.

Third, previous attempts have not measured the variations in the neural or model spaces that are relevant to class-level object classification [31]. For example the work in [31] examined the variation present in neural populations to visual stimuli presentations and compared this variation to the variation produced in model feature spaces to the same stimuli. This methodology does not address representational performance and does not provide an accuracy-complexity analysis (however, see [32] and [33], for discussion of methodologies to account for dissimilarity matrices by class-distance matrices). Our methodology of analyzing absolute representational performance using kernel analysis provides a novel and complementary finding to the results in [27,32,34]. Because of this complementarity, in this paper we also directly measure the amount of IT neural variance captured by deep neural networks as IT encoding models and by measuring representational similarity.

Finally, our approach utilizes a dataset that is an order of magnitude larger than previous datasets, and captures a degree of stimulus complexity that is critical for assessing IT representational performance. For example, the analysis in [10] utilized 150 images and the comparison in [31] utilized 96 images, while in this work we utilize an image set of 1960 images. The larger number of images allows our dataset to span and sample a relatively high degree of stimulus variation, which includes variation due to object exemplar, geometric transformations (position, scale, and rotation/pose) and background. *Importantly this variation is critical to distinguish between models based on object classification performance*: only in the presence of high variation are models distinguishable from each other [35,36] and from IT [27].

In this work, we propose an object categorization task and establish measurements of human performance for brief visual presentations. We then present our novel extension of kernel analysis and show that the latest deep neural networks achieve higher representational performance on this visual task compared to previous generation bio-inspired models. We next compare model representational performance to the IT cortex neural representation on the same task and images by matching the number of model features to the number of IT recordings and to the amount of observed experimental noise for both multi-unit recordings and single-unit recordings. We find that the latest DNNs match IT performance whereas previous models significantly lag the IT neural representation. In addition, we replicate the findings using a linear classifier approach. Finally, we show that the latest DNNs also provide compelling models of the actual IT neural response by measuring encoding model predictions and

performing a representational similarity analysis. We conclude with a discussion of the limitations of the current approach and future directions for studying models of visual recognition and primate object recognition.

Results

To evaluate the question of representational performance we must first make a choice about the task to be analyzed. The task we examine here is visual object category recognition in a natural duration fixation. This task is a well studied subproblem in visual perception and tests a core problem of visual perception: context independent basic-level object recognition within brief visual presentation. The task is to determine the category of an object instance that is presented under the effect of image variations due to object exemplar, geometric transformations (position, scale, and rotation/pose), and background. This task is well supported by behavioral measurements: humans [1] and macaques [2] are known to solve this task with high proficiency. It is well supported by neural measurements: evidence from IT cortex indicates that the neural representation supports and performs highly on this task [37]. Furthermore, this task provides a computationally challenging problem on which previous computational models have been shown to severely underperform [35,36]. Therefore, this task is difficult computationally and is performed at high proficiency by primates, with evidence that the primate ventral visual stream produces an effective representation in IT cortex.

Methodologically, the task is defined through an image generation process. An image is constructed by first choosing one of seven categories, then one of seven 3D object exemplars from that category, then a randomly chosen background image (each background image is used only once), and finally the variation parameters are drawn from a distribution to span two full octaves of scale variation, the full width of the image for translation variation, and the full sphere for pose variation. For each object exemplar we generated 40 unique images using this process, resulting in 1960 images in total. See Fig. 1 for example images organized by object category and Methods for further description of the image generation process. The resulting image set has several advantages and disadvantages. Advantageously, this procedure eliminates dependencies between objects and backgrounds that may be found in real-world images [38], and introduces a controlled amount of variability or difficulty in the task, which we have used to produce image datasets that are known to be difficult for algorithms [35,36,39]. Though arguably not fully "natural", the resulting images are highly complex (see Discussion for further advantages and disadvantages).

In evaluating the neural representational performance we must also define the behavioral context within which the neural representation supports behavior. This definition is important because it determines specific choices in the experimental setup. The behavioral context that we seek to address is a sub-problem of general visual behavior: vision in a natural duration fixation, or visual object recognition within one fixation without contextual influence, eye movements, or shifts in attention (also called "core visual object recognition" [6]). In our neural experiments we have chosen a presentation time of 100 milliseconds (ms) so as to be relevant for this behavior (see Discussion for further justification and Supporting Information (SI) for behavioral measurements on this task).

As a first step to evaluate the neural representation, we recorded multi-unit and single-unit neural activity from awake behaving rhesus macaques during passive fixation. We recorded activity using large scale multi-electrode arrays placed in either IT cortex or visual area V4. To create a neural feature vector, which we use to assess object representational performance, we presented each image (1960 images in total) for 100 ms and measured the normalized, background subtracted firing-rate in a window from 70 ms to 170 ms post image onset, averaged over 47 repetitions (see Methods). Over two macaques we measured 168 multi-unit sites in IT cortex, and 128 multi-unit sites in V4. From these recordings we also isolated single-units from IT and V4 cortex. Using conservative criteria (see Methods), we isolated 40 single-units from IT and 40 single-units from V4 with 6 repetitions per image for each single-unit.

To evaluate the performance of neural or model representations we utilize a novel extension of kernel analysis. Kernel analysis evaluates the efficacy of the representation by measuring how the precision of the category regression problem changes as we allow the complexity of the regression function to increase [30]. Intuitively, more effective representations will achieve higher precision at the same level of complexity because they have removed irrelevant variability from the original representational space (here irrelevant variability in the original space is due to object exemplar, geometric transformation, and background). To measure precision vs. complexity of the regression function, we perform kernel ridge regression using a Gaussian kernel (see Methods for details). We define complexity as the inverse of the regularization parameter ($1/\lambda$) and precision as 1 minus the normalized mean-squared leave-one-example-out generalization error, such that a precision value of 0 is chance performance and 1 is perfect performance. The regularization parameter restricts the complexity of the resulting regression function. By choosing a Gaussian kernel we can move between regression functions that are effectively linear, to functions that interpolate between the data points (a "complex" regression function) [40]. Note that complex regression functions may not generalize if there are not enough training examples (known as "sample complexity"), which will result in saturation or reduction in accuracy as complexity increases.
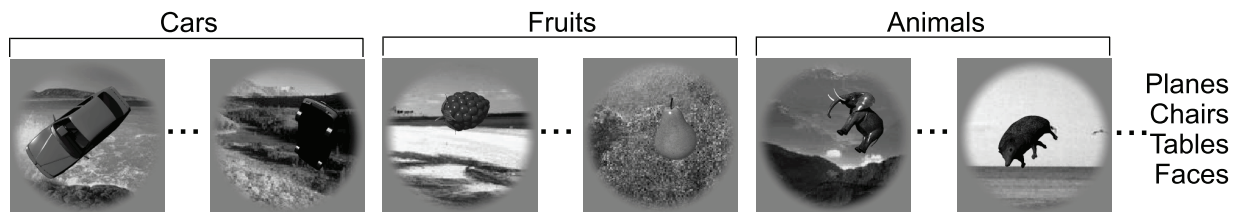
Fig. 1. Example images used to measure object category recognition performance. Two of the 1960 tested images are shown from the categories Cars, Fruits, and Animals (we also tested the categories Planes, Chairs, Tables, and Faces). Variability within each category consisted of changes to object exemplar (e.g. 7 different types of Animals), geometric transformations due to position, scale, and rotation/pose, and changes to background (each background image is unique).
doi:10.1371/journal.pcbi.1003963.g001

We compared the neural representation to three convolutional DNNs and three other biologically relevant representations. Note that the development of these representations did not utilize the 1960 images we use here for testing in any way. The three recent convolutional DNNs we examine are described in Krizhevsky et al. 2012 [24], Zeiler & Fergus 2013 [25], and Yamins et al. 2014 [27,34]. The Krizhevsky et al. 2012 and Zeiler & Fergus 2013 DNNs are of note because they have each successively surpassed the state-of-the-art performance on the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) datasets. Note that results have continued to improve on this challenge since we ran our analysis. See http://www.image-net.org/for the latest results. The DNN presented in Yamins et al. 2014 [27] is created using a supervised optimization procedure called hierarchical modular optimization (we refer to this model by the abbreviation HMO). The HMO DNN has been shown to match closely representational dissimilarity matrices of the ventral stream and to be predictive of IT and V4 neural responses [27]. We also evaluated an instantiation of the HMAX model of invariant object recognition that uses sparse localized features [41] and has previously been shown to be a relatively high performing model among artificial systems [16]. Finally, we also evaluated a V2-like model and a V1-like model that each attempt to capture a first-order account of secondary (V2) [42] and primary visual cortex (V1) [35], respectively.

Each of the three convolutional DNNs was developed, implemented, and trained by their respective researchers and for those developed outside of our group we obtained features from each DNN computed on our test images. The convolutional DNN described in Krizhevsky et al. 2012 [24] was trained by supervised learning on the ImageNet 2011 Fall release (~15 M images, 22K categories) with additional training on the LSVRC-2012 dataset (1000 categories). The authors computed the features in the penultimate layer of their model (4096 features) on the 1960 images we used to measure the neural representation. The similar 8-layer deep neural network of Zeiler & Fergus 2013 [25] was trained using supervised learning on the LSVRC-2012 dataset augmented with random crops and left-right flips. This model took advantage of hyper-parameter tuning informed by visualizations of the intermediate network layers. The 4096 dimensional feature representation was produced by taking the penultimate layer features and averaging them over 10 image crops (the 4 corners, center, and horizontal flips for each). The model of Yamins et al. 2014 [27] is an extension of the high-throughput optimization strategy described in [16] that produces a heterogeneous combination of hierarchical convolutional models optimized on a supervised object recognition task through hyperparameter optimization using boosting and error-based reweighing (see [27] for details). The total output feature space per image for the HMO model is 1250 dimensional.

Before comparing the representational performance of the neural and model representations, we first evaluate the absolute representational performance of these models on the task to verify that the task we have chosen is computationally difficult. As described in our previous work [35], we determined that a task is computationally difficult if "simple" computational models fail on the task. For the models tested here, the V1-like and V2-like models represent these computationally simple models. Using kernel analysis we evaluated both the DNNs and the bio-inspired models on the task and plot the precision vs. the complexity curves for each model representation in Fig. 2. This analysis indicates that both the V1-like and V2-like models perform near chance on this task over the entire range of complexity. Furthermore, the HMAX model performs only slightly better on this task. If we

reduce the difficulty of the task by reducing the magnitude range of the variations we introduce (not shown here, but see [35,43] for such an analysis), these models are known to perform well on this task; therefore, it is object recognition under variation that makes this problem difficult, and the magnitude range we have chosen for this task is quite difficult in that the HMAX model performs poorly. In contrast, the three DNNs perform at much higher precision levels over the complexity range. A clear ranking is observed with the Zeiler & Fergus 2013 [25] model followed by the Krizhevsky et al. 2012 [24] model and the HMO model [27]. These results indicate that these models outperform models of early visual areas, and we next ask which model, if any, can match the performance of high-level visual area IT.

In order to directly compare the representational performance of the IT neural representation to the model representations we take a number of steps to produce a fair comparison. The experimental procedure that we used to measure the neural representation is limited by the number of neural samples (sites or number of neurons) that we can measure and by noise induced by uncontrolled experimental variability and/or intrinsic neural noise. To equalize the sampling between the neural representation and the model representations we fix the number of neural samples (80 for the multi-unit analysis and 40 for the single-unit analysis) and model features (we will vary this number in later experiments). To correct for the observed experimental noise, we add noise to the model representations. To add noise to the models we estimate an experimental neural noise model. Following the observation that spike counts of neurons are approximately Poisson [44,45] and similar analyses of our own recordings, we model response variability as being proportional to the mean response. Precisely, the estimated noise model is additive to the mean response and is a zero-mean Gaussian random variable with variance being a linear function of the mean response. We estimate the parameters of the noise model from the empirical distribution of multi-unit responses and single-unit responses. Note that our empirical estimate of these quantities is influenced by both uncontrolled experimental variability (e.g. variability across recording sessions) as well as intrinsic neural noise. See Methods for further description and S1 Fig. for a verification that the noise model reduces performance more greatly than empirical noise, thus demonstrating that the noise model is conservative and over-penalizes models. To produce noise-matched model representations, we sample the model response dependent noise and measure the representational performance of the resulting representation using kernel analysis. We repeat this procedure 10 times to measure the variability produced by the additive noise model.

We compare the sample and noise corrected model representations to the multi-unit neural representations in Fig. 3A. The kernel analysis curves are plotted for neural and model representations sampled at 80 neural samples or 80 model features, respectively. The model representations have been corrected for the neural noise observed in the multi-unit IT neural measurement. Note that we do not attempt to correct the V4 sample to the noise level observed in IT because we observed similar noise between the V4 and IT neural measurements and each sample is averaged over the same number of trials (47 trials). Compared to the model representational performance in Fig. 2, model performance is reduced because of the subsampling and because of the added noise correction (without added noise and subsampling maximum precision is above 0.5 and with noise and subsampling does not pass 0.35). Consistent with previous work [10,36], we observed that the sampled IT neural representation significantly exceeds the similarly-sampled V4 neural representation. Unsurprisingly, HMAX, V2-like, and V1-like representations perform
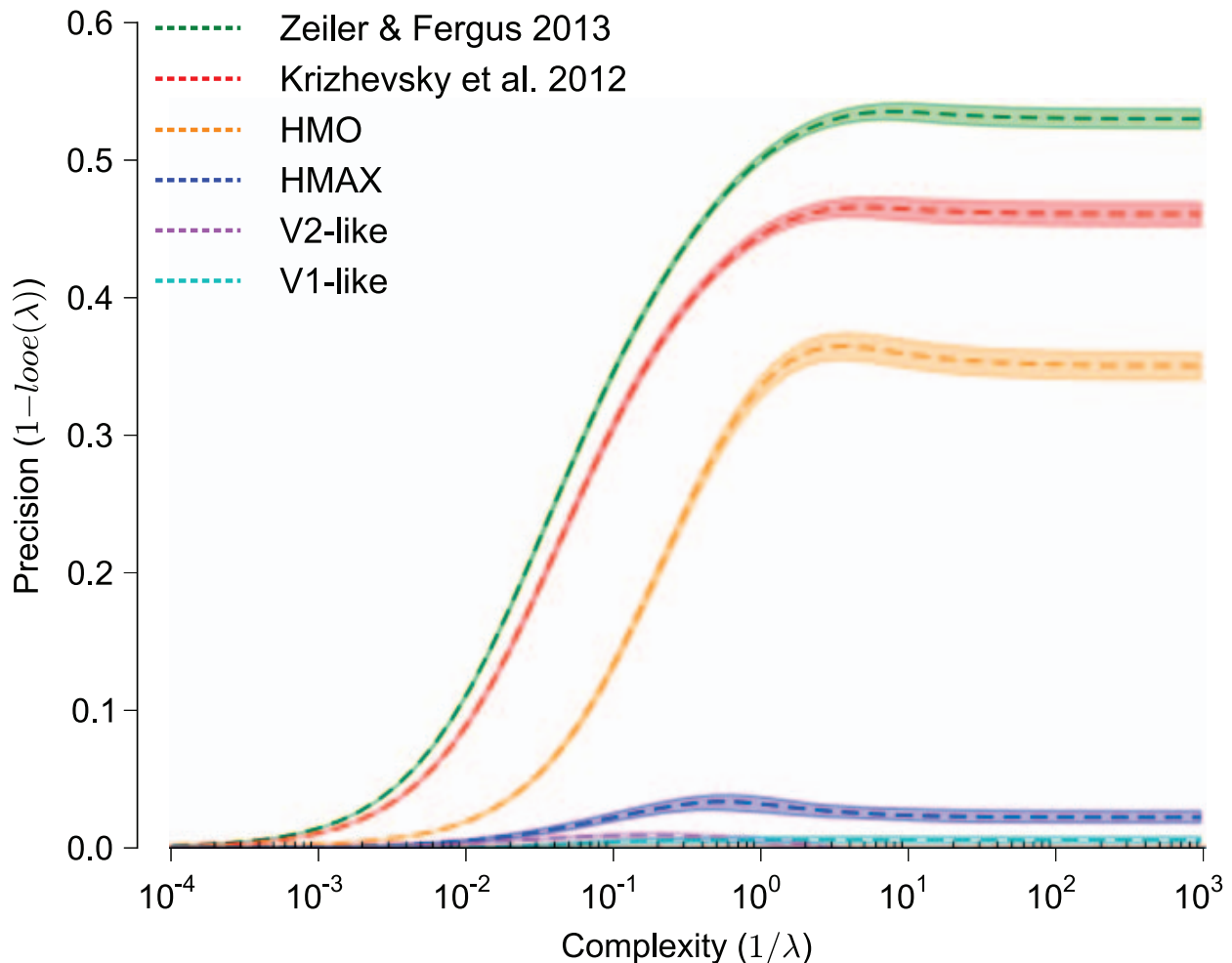
Fig. 2. Kernel analysis curves of model representations. Precision, one minus loss $(1 - looe(\lambda))$, is plotted against complexity, the inverse of the regularization parameter $(1/\lambda)$. Shaded regions indicate the standard deviation of the measurement over image set randomizations, which are often smaller than the line thickness. The Zeiler & Fergus 2013, Krizhevsky et al. 2012 and HMO models are all hierarchical deep neural networks. HMAX [41] is a model of the ventral visual stream and the V1-like [35] and V2-like [42] models attempt to replicate response properties of visual areas V1 and V2, respectively. These analyses indicate that the task we are measuring proves difficult for V1-like and V2-like models, with these models barely moving from 0.0 precision for all levels of complexity. Furthermore, the HMAX model, which has previously been shown to perform relatively well on object recognition tasks, performs only marginally better. Each of the remaining deep neural network models performs drastically better, with the Zeiler & Fergus 2013 model performing best for all levels of complexity. These results indicate that the visual object recognition task we evaluate is computationally challenging for all but the latest deep neural networks.
doi:10.1371/journal.pcbi.1003963.g002

near chance. All three recent DNNs perform better than the V4 representation. The IT representation performs quite well, especially considering the sampling and noise limitations of our recordings and would be quite competitive if directly compared to the model results in Fig. 2. After correcting for sampling and noise, the IT representation is only matched by the top performing DNN of Zeiler & Fergus 2013. Interestingly, this relationship holds for the entire complexity range.

We present the equivalent representational comparison between models and neural representations for the single-unit neural recordings in Fig. 3B. Because of the increased noise and fewer trials collected for the single-unit measurements compared to our multi-unit measurements, the single-unit noise and sample corrected model representations achieve lower precision vs. complexity curves than under the multi-unit noise and sample correction (compare to Fig. 3A). This analysis shows that the single-unit IT representation performs better than the HMO representation, slightly worse than the Krizhevsky et al. 2012

representation, and is outperformed by the Zeiler & Fergus 2013 [25] representation. Furthermore, a comparison of the relative performance of the multi-unit sample and the single-unit sample indicates that the multi-unit sample outperforms the single-unit sample. See Discussion for elaboration of this finding and S4 Fig. for trial corrected performance comparison between single- and multi-units.

In Figs. 4A and 4B we analyze the representational performance as a function of neural sites or model features for multi-unit and single-unit neural measurements. To achieve a summary number from the kernel analysis curves we compute the area-under-the-curve and we omit the HMAX, V2-like, and V1-like models because they are near zero performance in this regime. In Fig. 4A we vary the number of multi-unit recording samples and the number of features. Just as in Fig. 3A, we correct for neural noise by adding a matched neural noise level to the model representations. Fig. 4A indicates that the representational performance relationship we observed at 80 samples is robust
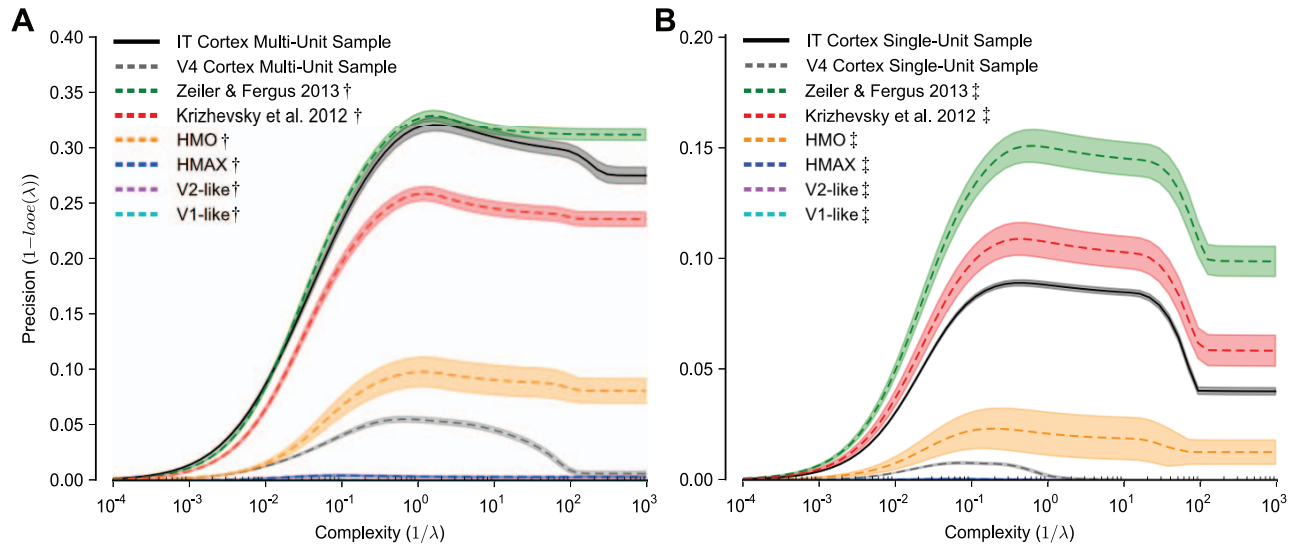
Fig. 3. Kernel analysis curves of sample and noise matched neural and model representations. Plotting conventions are the same as in Fig. 2. Multi-unit analysis is presented in panel A and single-unit analysis in B. Note that the model representations have been modified such that they are both subsampled and noisy versions of those analyzed in Fig. 2 and this modification is indicated by the † symbol for noise matched to the multi-unit IT cortex sample and by the ‡ symbol for noise matched to the single-unit IT cortex sample. To correct for sampling bias, the multi-unit analysis uses 80 samples, either 80 neural multi-units from V4 or IT cortex, or 80 features from the model representations, and the single-unit analysis uses 40 samples. To correct for experimental and intrinsic neural noise, we added noise to the subsampled model representation (no additional noise is added to the neural representations) that is commensurate to the observed noise from the IT measurements. Note that we observed similar noise between the V4 and IT Cortex samples and we do not attempt to correct the V4 cortex sample of the noise observed in the IT cortex sample. We observed substantially higher noise levels in IT single-unit recordings than multi-unit recordings due to both higher trial-to-trial variability and more trials for the multi-unit recordings. All model representations suffer decreases in accuracy after correcting for sampling and adding noise (compare absolute precision values to Fig. 2). All three deep neural networks perform significantly better than the V4 cortex sample. For the multi-unit analysis (A), IT cortex sample achieves high precision and is only matched in performance by the Zeiler & Fergus 2013 representation. For the single-unit analysis (B), both the Krizhevsky et al. 2012 and the Zeiler & Fergus 2013 representations surpass the IT representational performance.
doi:10.1371/journal.pcbi.1003963.g003
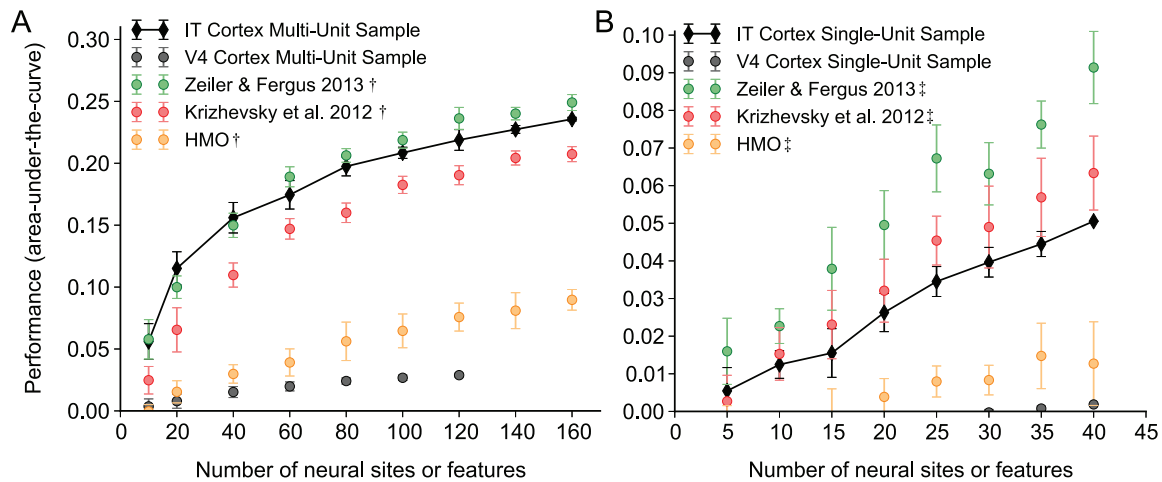
Fig. 4. Effect of sampling the neural and noise-corrected model representations. We measure the area-under-the-curve of the kernel analysis measurement as we change the number of neural sites (for neural representations), or the number of features (for model representations). Measured samples are indicated by filled symbols and measured standard deviations indicated by error bars. Multi-unit analysis is shown in panel A and single-unit analysis in B. The model representations are noise corrected by adding noise that is matched to the IT multi-unit measurements (A, as indicated by the † symbol) or single-unit measurements (B, as indicated by the ‡ symbol). For the multi-unit analysis, the Zeiler & Fergus 2013 representation rivals the IT cortex representation over our measured sample. For the single-unit analysis, the Krizhevsky et al. 2012 representation rivals the IT cortex representation for low number of features and slightly surpasses it for higher number of features. The Zeiler & Fergus 2013 representation surpasses the IT cortex representation over our measured sample.
doi:10.1371/journal.pcbi.1003963.g004

between 10 samples and 160 samples. Fig. 4B indicates that the performance of the IT single-unit representation is comparatively worse than the multi-unit, with the single-unit representation falling below the performance of the Krizhevsky et al. 2012 representation for much of the range of our analysis.

These results indicate that after correcting for noise and sampling effects, the Zeiler & Fergus 2013 DNN rivals the performance of the IT multi-unit representation and that both the Krizhevsky et al. 2012 and Zeiler & Fergus 2013 DNNs surpasses the performance of the IT single-unit representation. The performance of these two DNNs in the low-complexity regime is especially interesting because it indicates that they perform comparably to the IT representation in the low-sample regime (i.e. low number of training examples), where restricted representational complexity is essential for generalization (e.g. [46]).

To verify the results of the kernel analysis procedure we measured linear-SVM generalization performance on the same task for each neural and model representation (Fig. 5). We used a cross-validated procedure to train the linear-SVM on 80% of the images and test on 20% (regularization parameters were estimated from the training set). We repeated the procedure for 10 randomizations of the training-testing split. The linear-SVM results reveal a similar relationship to the results produced using kernel analysis (Fig. 3A). This indicates that the Zeiler & Fergus 2013 representation achieves generalization comparable to the IT multi-unit neural sample for a simple linear decision boundary. We also found near identical results to kernel analysis for the

single-unit analyses and the analysis of performance as a function of the number of neural sites or features (see S4 Fig.).

While the goal of our analysis has been to measure representational performance of neural and machine representations it is also informative to measure neural encoding metrics and measures of representational similarity. Such analyses are complementary because representational performance relates to the task goals (in this case category labels) and encoding models and representational similarity metrics are informative about a model's ability to capture image-dependent neural variability, even if this variability is unrelated to task goals. We measured the performance of the model representations as encoding models of the IT multi-unit responses by estimating linear regression models from the model representations to the IT multi-unit responses. We estimated models on 80% of the images and tested on 20%, repeating the procedure 10 times (see Methods). The median predictions averaged over the 10 splits are presented in Fig. 6A. For comparison, we also estimated regression models using the V4 multi-unit responses to predict IT multi-unit responses. The results show that the Krizhevsky et al. 2012 and the Zeiler & Fergus 2013 DNNs achieve higher prediction accuracies than the HMO model, which was previously shown to achieve high predictions on a similar test [27]. These predictions are similar in explained variance to the predictions achieved by V4 multi-units. However, no model is able to fully account for the explainable variance in the IT multi-unit responses. In Fig. 6B we show the mean explained variance of each IT multi-unit site as predicted by the
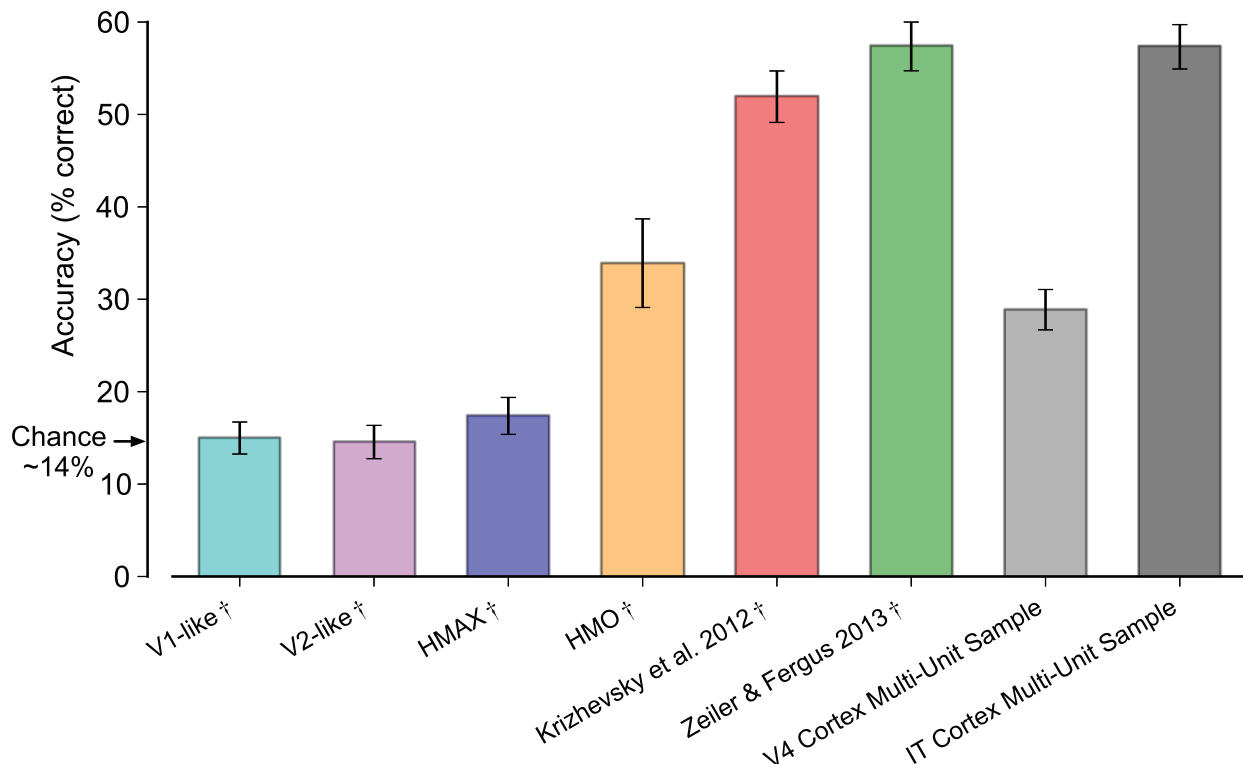
Fig. 5. Linear-SVM generalization performance of neural and model representations. Testing set classification accuracy averaged over 10 randomly-sampled test sets is plotted and error bars indicate standard deviation over the 10 random samples. Chance performance is ~14.3%. V4 and IT Cortex Multi-Unit Sample are the values measured directly from the neural samples. Following the analysis in Fig. 3A, the model representations have been modified such that they are both subsampled and have noise added that is matched to the observed IT multi-unit noise. We indicate this modification by the † symbol. Both model and neural representations are subsampled to 80 multi-unit samples or 80 features. Mirroring the results using kernel analysis, the IT cortex multi-unit sample achieves high generalization accuracy and is only matched in performance by the Zeiler & Fergus 2013 representation.
doi:10.1371/journal.pcbi.1003963.g005

Fig. 6. Neural and model representation predictions of IT multi-unit responses. A) The median predictions of IT multi-unit responses averaged over 10 train/test splits is plotted for model representations and V4 multi-units. Error bars indicate standard deviation over the 10 train/test splits. Predictions are normalized to correct for trial-to-trial variability of the IT multi-unit recording and calculated as percentage of explained, explainable va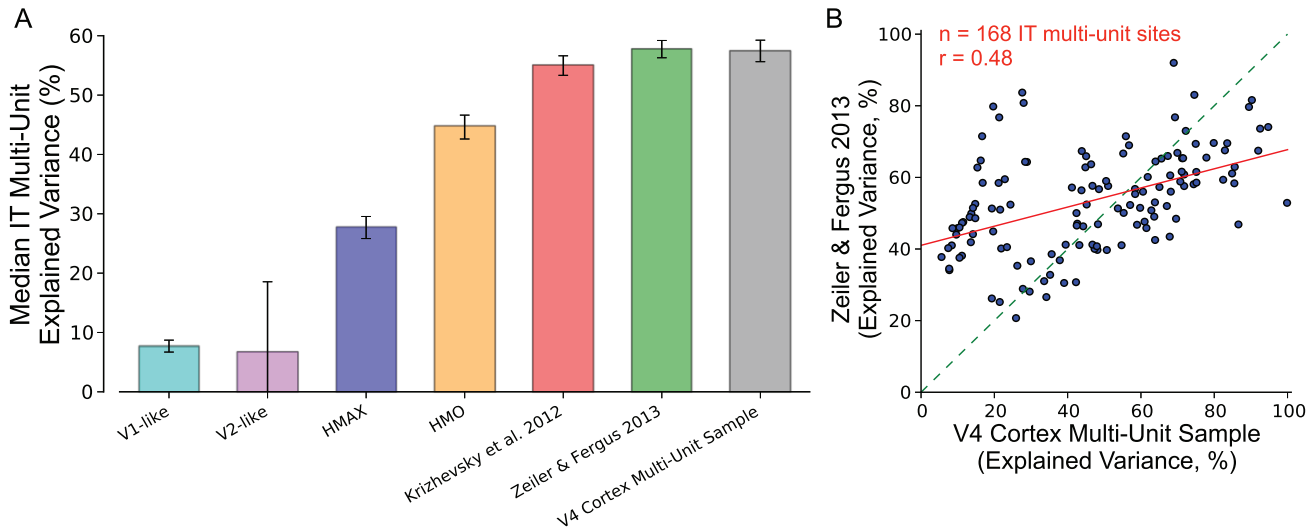riance. The HMO, Krizhevsky et al. 2012, and Zeiler & Fergus 2013 representations achieve IT multi-unit predictions that are comparable to the predictions produced by the V4 multi-unit representation. B) The mean predictions over the 10 train/test splits for the V4 cortex multi-unit sample and the Zeiler & Fergus 2013 DNN are plotted against each other for each IT multi-unit site.
doi:10.1371/journal.pcbi.1003963.g006

V4 cortex multi-unit sample and the Zeiler & Fergus 2013 DNN. There is a relatively weak relationship between the encoding performance of the neural V4 and DNN representations ($r = 0.48$ between V4 and Zeiler & Fergus 2013, compared to $r = 0.96$ and $r = 0.74$ for correlations between Krizhevsky et al. 2012 and Zeiler & Fergus 2013, and HMO and Zeiler & Fergus 2013, respectively), indicating that V4 and DNN representations may account for different sources of variability in IT (see Discussion).

Finally, we measured representational similarity using the analysis methodology proposed in [32]. This analysis methodology measures how similar two representations are and is robust to global scalings and rotations of the representational spaces. To compute the representational similarity between the IT multi-unit and model representations, we computed object-level representational dissimilarity matrices (RDMs) for model and neural representations (matrices are 49x49 dimensional as there are 49 total objects). We then measured the Spearman rank correlations between the model derived RDM and the IT multi-unit RDM (see Methods). In Fig. 7A we show the results of the representational similarity measurements for the model representations and in Fig. 7B we show depictions of the RDMs for select representations. For comparison we present the result between the V4 multi-unit representation and the IT multi-unit representation. To determine the variability due to the IT neural sample, we also present the similarity measurement between one-half of the IT multi-units and the other half (IT Cortex Split-Half). In addition, we provide results following the methodology in [27], which first predicts the IT multi-unit site responses from the model representation and then uses these predictions to form a new representation. We refer to these representations with an appended "+IT-fit". Our measurements of the HMO + IT-fit representation are in general agreement with the results in [27] but vary slightly because of differences in the image set used to produce these measurements and details of the methodology used to produce the IT predictions. Interestingly, by fitting a linear transform at the image-level to IT multi-units, the Krizhevsky et al. 2012 and Zeiler & Fergus 2013 DNNs fall within the noise limit

of the IT split-half object-level RDM measurement. However, the HMO, Krizhevsky et al. 2012, and Zeiler & Fergus 2013 representations, without the added linear mapping, have deviations from the IT representation that are unexplained by noise variation. While it is informative that a linear mapping can produce RDMs in correspondence with the IT RDM, we conclude that there remains a gap between DNN models and IT representation when measured with object-level representational similarity.

Discussion

In summary, our measurements indicate that the latest DNNs rival the representational performance of IT cortex on a rapid object category recognition task. We evaluated representational performance using a novel kernel analysis methodology, which measures precision as a function of classifier complexity. Kernel analysis allows us to measure a desirable property of a representation: a *good* representation is highly performant with a simple classification function and can thus accurately predict class labels from few examples, while a *poor* representation is only performant with complex classification functions and thus requires a large number of training examples to accurately predict (see Methods for elaboration on this point). Importantly, we made comparisons between models and neural measurements by correcting the models for experimental limitations due to sampling, noise, and trials. In this analysis we found that the Zeiler & Fergus 2013 DNN achieved comparable representational performance to the IT cortex multi-unit representation and both the Krizhevsky et al. 2012 and Zeiler & Fergus 2013 representations surpassed the performance of the IT cortex single-unit representation. These results reflect substantial progress of computational object recognition systems since our previous evaluations of model representations using a similar object recognition task [35,36]. These results extend our understanding over recent, complimentary studies, which have examined representational similarity [27], by evaluating directly absolute
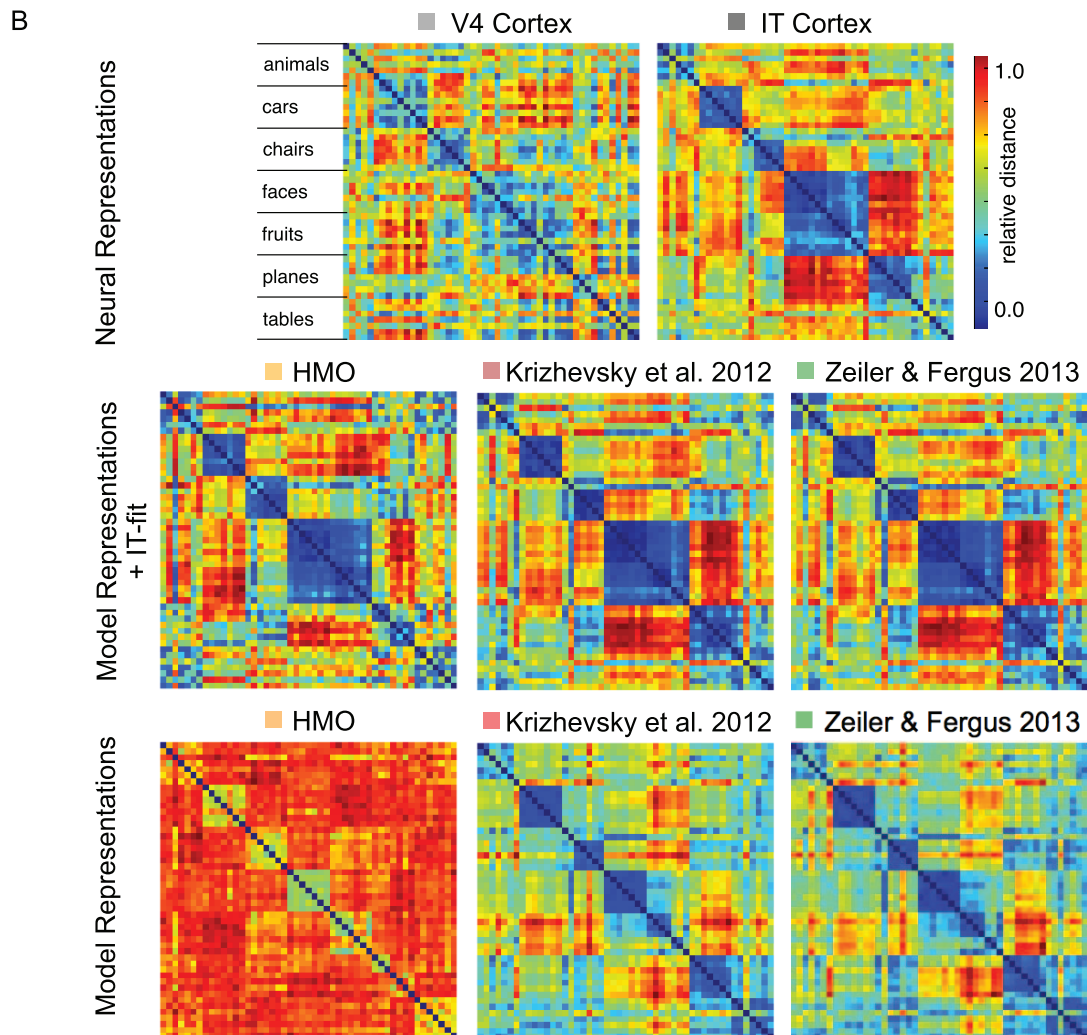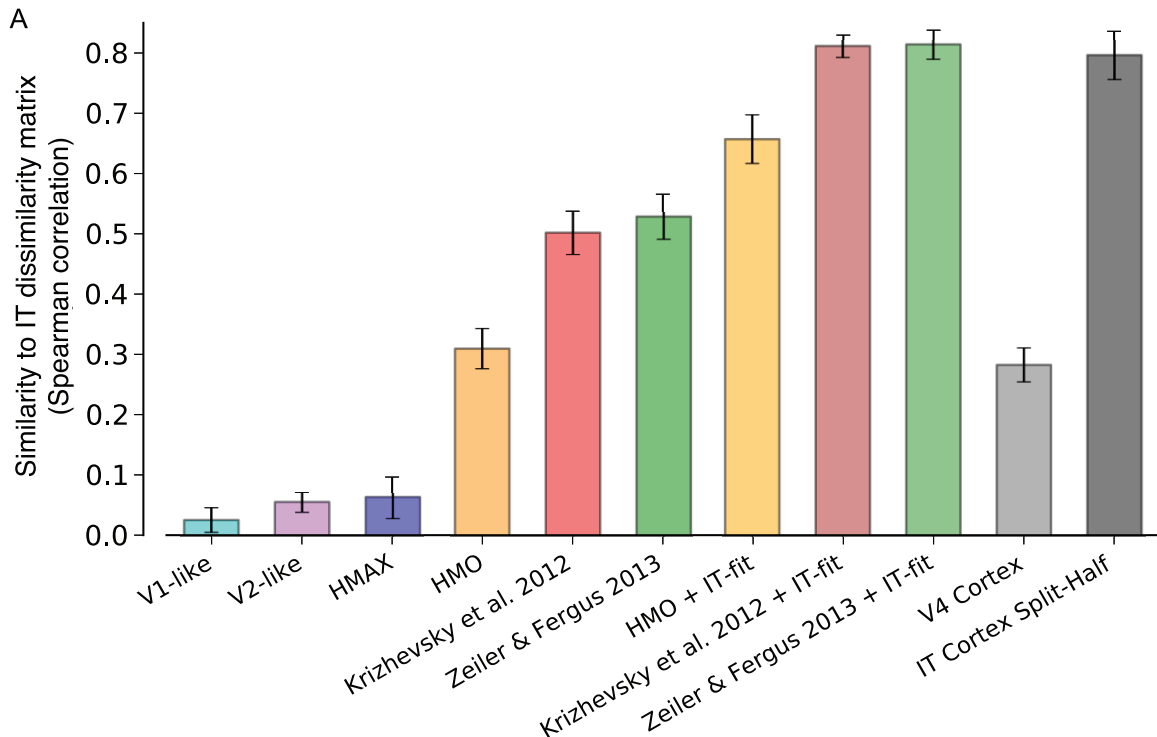
Fig. 7. Object-level representational similarity analysis comparing model and neural representations to the IT multi-unit representation. A) Following the proposed analysis in [32], the object-level dissimilarity matrix for the IT multi-unit representation is compared to the matrices computed from the model representations and from the V4 multi-unit representation. Each bar indicates the similarity between the corresponding representation and the IT multi-unit representation as measured by the Spearman correlation between dissimilarity matrices. Error bars indicate standard deviation over 10 splits. The IT Cortex Split-Half bar indicates the deviation measured by comparing half of the multi-unit sites to the other half, measured over 50 repetitions. The V1-like, V2-like, and HMAX representations are highly dissimilar to IT cortex. The HMO representation produces comparable deviations from IT as the V4 multi-unit representation while the Krizhevsky et al. 2012 and Zeiler & Fergus 2013 representations fall in-between the V4 representation and the IT cortex split-half measurement. The representations with an appended "+ IT-fit" follow the methodology in [27], which first predicts IT multi-unit responses from the model representation and then uses these predictions to form a new representation (see text). B) Depictions of the object-level RDMs for select representations. Each matrix is ordered by object category (animals, cars, chairs, etc.) and scaled independently (see color bar). For the "+ IT-fit" representations, the feature for each image was averaged across testing set predictions before computing the RDM (see Methods).
doi:10.1371/journal.pcbi.1003963.g007

representational performance for this task. In contrast to the representational performance results, all models that we have tested failed to capture the full explainable variation in IT responses (Figs. 6 and 7). Nonetheless, our results, in conjunction with the results in Yamins et al. 2014 [27], indicate that the latest DNNs provide compelling models of primate object recognition representations that predict neural responses in IT cortex [27] and rival the representational performance of IT cortex.

To address the behavioral context of core visual object recognition our neural recordings were made using 100 ms presentation times. We chose only a single presentation time (as opposed to rerunning the experiment at different presentation times) to maximize the number of images and repetitions per image given time and cost constraints in neurophysiological recordings. This choice is justified by previous results that indicate human subjects are performant on similar tasks with just 13 ms presentation times [4], that human performance on similar tasks rapidly increases from 14 ms to 56 ms and has diminishing returns between 56 ms and 111 ms [3], that decoding from IT at 111 ms presentation times achieves nearly the same performance at 222 ms presentation times [3], that for 100 ms presentation times the first spikes after stimulus onset in IT are informative and peak decoding performance is at 125 ms [9], and that maximal information rates in high-level visual cortex are achieved at a rate of 56 ms/stimulus [47]. Furthermore, we have measured human performance on our task and observed that the mean response accuracy at 100 ms presentation times is within 92% of the accuracy at 2000 ms presentation times (see S2 Fig.). While reducing presentation time below 50 ms likely would lead to reduced representational performance measurements in IT (see [3]), the presentation time of 100 ms we used for our evaluation is applicable for the core recognition behavior, has previously been shown to be performant behaviorally and physiologically, and in our own measurements on this task captures the large majority of long-presentation-time (2 second) human performance.

The images we have used to define the computational task allow us to precisely control variations to object exemplar, geometric transformations, and background; however, they have a number of disadvantages that can be improved upon in further studies. For example, this image set does not expose contextual effects that are present in the real world and may be used by both neural and machine systems, and it does not include other relevant variations, e.g. lighting, texture, natural deformations, or occlusion. We view these current disadvantages as opportunities for future datasets and neural measurements, as the approach taken here can naturally be expanded to encompass these issues.

There are a number of issues related to our measurement of macaque visual cortex, including viewing time, behavioral paradigm, and mapping the neural recording to a neural feature, that will be necessary to address in determining the ultimate representational measurement of macaque visual cortex. The

presentation time of the images shown to the animals was intentionally brief (100 ms), but is close to typical single-fixation durations during natural viewing (~200 ms), and human behavioral testing (S2 Fig.) shows that the visual system achieves high performance at this viewing time. It will be interesting to measure how the neural representational space changes with increased viewing time and multiple fixations. Another aspect to consider is that during the experimental procedure, animals were engaged in passive viewing and human subjects were necessarily performing an active task. Does actively performing a task influence the neural representation? While several studies report that such effects are present, but weak at the single-unit level [48–51], no study has yet examined the quantitative impact of these effects at the population level for the type of object recognition task we examined. Active task performance may be related to what are commonly referred to as attentional phenomena [e.g. biased competition]. In addition, the mapping from multi-unit and single-unit recordings to the neural feature vector we have used for our analysis is only one possible mapping, but it is a parsimonious first choice. Finally, visual experience or learning may impact the representations observed in IT cortex. Interestingly, the macaques involved in these studies have had little or no real-world experience with a number of the object categories used in our evaluation, though they do benefit from millions of years of evolution and years of postnatal experience. However, significant learning effects in adult IT cortex have been observed [52–54], even during passive viewing [55]. We have examined the performance of computational algorithms in terms of their absolute representational performance. It is also interesting to examine the necessary processing time and energy efficiency of these algorithms in comparison to the primate visual system. While a more in depth analysis of this issue is warranted, from a "back-of-the-envelope" calculation (see SI) we conclude that model processing times are currently competitive with primate behavioral reaction times but model energy requirements are 2 to 3 orders of magnitude higher than the primate visual system.

How do our measurements of representational performance relate to overall system performance for this task? Measuring representational performance fundamentally relies on a measure of the representation, which we have assumed is a neural measure such as single-unit response or multi-unit response. This poses difficulties for obtaining an accurate measure of human representational performance. Using only behavioral measurements the representation must be inferred, which may be possible through an investigation of the psychological space of visually presented objects. However, more direct methods may be fruitful using fMRI (see [31]), or a process that first equates macaque and human performance and uses the macaque neural representation as a proxy for the human neural representation. One approach to directly measure the overall system performance is to replicate the cross-validated procedure used to measure models in humans.

Such a procedure should control the human exposure to the training set and provide the correct labels on the training set. The procedure for measuring human performance presented in the SI does not follow this procedure. However, a comparison between human performance at 100 ms presentation times (see S2 Fig.) and overall DNN model performance on the test-set (see S3 Fig.) indicates that there is likely a gap between human performance (85% mean accuracy) and DNN performance (77% mean accuracy) on this task because allowing the human subjects exposure to 80% of the images with the correct labels is only likely to increase the human performance number. Furthermore, there is individual variability in the human performance with some individuals performing well above the mean. Therefore, while we have not attempted to make a direct comparison between human performance and DNN performance, we infer that human performance exceeds current DNN performance.

Our methodology and approach relates to the encoding and decoding approaches in systems neuroscience, which, in our view, provide complementary insights into neural visual processing. The kernel analysis methodology we use here is a neural decoding approach because it measures the relationship between the neural (or model) representation and unobserved characteristics of the stimuli (class labels). The linear-SVM methodology is also a decoding approach because it tests the generalization performance of predicting the unobserved class label from the neural (or model) representation. The approaches of predicting IT multi-unit response (Fig. 6) and measuring representational similarity to IT representation (Fig. 7) are encoding approaches because they measure the relationship between functions or measurements derived from the stimuli (pixels in the images) and the neural variation present in IT. The complementary nature of these approaches is demonstrated in our results. For example, while the Zeiler & Fergus 2013 DNN rivals the decoding performance of IT cortex, it fails to capture over 40% of the explainable variance in the IT neural sample and therefore does not produce a complete neural encoding model. Conversely, the V4 multi-unit representation severely underperforms the DNNs and IT cortex when measured with decoding approaches but produces comparable results to these representations when predicting IT multi-unit responses with an encoding approach. It is currently unclear what variation in the IT cortex multi-unit representation is not captured by DNNs or the V4 multi-unit representation. Furthermore, the IT variation that is captured by DNNs and V4 is, relative to correlations between DNN models, weakly correlated (Fig. 6B). Overall, the remaining unexplained IT variation may be exposed through a decoding approach (by, for example, exploring additional task labels), through an encoding approach (by exploring additional stimulus transformations), or through approaches that take into account intrinsic neural dynamics (e.g. [56]). The comparably high performance of the V4 multi-unit representation at predicting IT multi-unit responses may be due to its ability to capture intrinsic neural dynamics present in IT that are unrelated to stimulus derived variables.

We found that multi-units outperform single-units in our evaluation, as evidenced in the relative performance increase of DNNs over single-units in Figs. 3 and 4 and in a trial corrected comparison between multi- and single-units in S4 Fig. It may be surprising that multi-units outperform single-units on this task. A priori, we might assume that the averaging process introduced by the multi-unit recording, which aggregates the spikes from multiple spatially-localized single-units, would "wash-out" or average away neural selectivity for visual objects. However, when considering generalization performance, it is possible that averaged single-units could outperform the original single-units by

removing noise or non-class specific signal variation. In this way, multi-units may provide a form of regularization that is appropriate for this task. This regularization may be due to averaging out single-unit noise, and/or reducing variation in the representational space that is irrelevant, and therefore spurious, for the task. Alternatively, the single-unit variation may be appropriate for different tasks that we have not measured, such as fine distinctions between objects or 3-dimensional object surface representation. The spatial arrangement of functional cortical responses (topographic maps, and functional clustering) also indicates a current limitation of DNNs as models of high-level visual cortex. There is no notion of tissue or cortical space in the DNN layers that we utilize as features: the features correspond to fully connected layers which do not have convolutional topology, which may be trivially mapped to physical space, nor do they have localized operations, such as local divisive normalization. For this reason, it is non-trivial to include a biologically realistic averaging process to the DNN representations.

Could the regularizing properties of multi-unit responses be indicative of broader regularization mechanisms related to spatial organization in cortex (topographic maps, and functional clustering)? Just as our multi-unit recordings average together a number of single-units, neurons "reading-out" signals from IT cortex could average over cortical topography and thus regularize the classification decision boundary. This suggests the future goal of finding an appropriate mapping of biological phenomenology and physical mechanism to the computational concepts of kernels, regularizers, and representational spaces. The overall performance of learning algorithms strongly depends on the interconnections between the choice of kernel, regularizer, and representation. In our current work (and the predominant mode in the field), we have made specific choices on the kernel and regularizer and examined aspects of the representational space. However, a full account of biological learning and decision making must determine accurate descriptions for all three of these computational components in biological systems.

Interestingly, many of the computational concepts utilized in the high performing DNNs that we have measured extend back to early models of the primate visual system. All three DNNs we examined [24,25,27] implement architectures containing successive layers of operations that resemble the simple and complex cell hierarchy first described by Hubel and Wiesel [17,18]. In particular, max-pooling was proposed in [13] and is a prominent feature of all three DNNs. Additional computational concepts are convolution or weight sharing, which was introduced in [57] and utilized in [24,25], and backpropagation [58], which is utilized in [24,25].

The success of the Krizhevsky et al. 2012 and the Zeiler & Fergus 2013 DNNs raises a number of interesting questions. The categories we used for testing (the 7 classes used in the kernel analysis measurements) are a small fraction of the 1000 classes that these models were trained on, and it is not clear if there is a direct correspondence between the classes in the two image sets. At this point it is not clear how the non-relevant classes in the set used to train the models affects our performance estimate. As more detailed analyses are conducted it will be interesting to determine which categories are necessary to replicate ventral stream performance and similarity. For example, there may be biases in the necessary category distribution toward ecologically relevant categories, such as faces. Of biological relevance, it is not clear if natural primate development is comparable to the 15M labeled images used to train these DNNs and it seems likely that more innate knowledge of the visual world (acquired during evolution) and/or more unsupervised training (during development) are

utilized in biological systems. Finally, given their similar architectures, it is unclear why some DNNs perform well and others do not. However, our analyses provide cursory evidence that models with more layers perform better and models that effectively reduce the dimensionality of the original problem perform better. More work is necessary to determine best practices using these architectures and to determine the importance of hierarchical representations and representations that reduce dimensionality. The principled approach we have provided here allows for practical evaluations between models and neurons, and may provide a tool in assessing progress in the development of DNNs. Going forward, we would ideally like a better theoretical understanding of these architectures that would lead to more consistent implementations and would produce a detailed, mechanistic hypothesis for ventral stream processing (see [59] for an example of such an effort).

Methods

Ethics statement

This study was performed in strict accordance with the recommendations in the Guide for the Care and Use of Laboratory Animals of the National Institutes of Health. All surgical and experimental procedures were approved by the Massachusetts Institute of Technology Committee on Animal Care (Animal protocol: 0111-003-014). All human behavioral measurements were approved by the Massachusetts Institute of Technology's Committee on the Use of Humans as Experimental Subjects (number: 0812003043).

Image dataset generation

Synthetic images of objects were generated using POV-Ray, a free, high-quality ray tracing software package (http://www.povray.org). 3-d models (purchased from Dosch Design and TurboSquid) were converted to the POV-Ray format. This general approach allowed us to generate image sets with arbitrary numbers of different objects, undergoing controlled ranges of identity preserving object variation/transformation. The 2-d projection of the 3-d model was then combined with a randomly chosen background. In our image set, no two images had the same background, in some cases the background was, by chance, correlated with the object (plane on a sky background, car on a street) but more often they were uncorrelated, giving no information about the identity of the object. A circular aperture with radial fall-off was applied to create each final image.

Neural data collection

We collected neural data from V4 and IT across two adult male rhesus monkeys (*Macaca mulatta*, 7 and 9 kg) by using a multi-electrode array recording system (BlackRock Microsystems, Cerebus System). We chronically implanted three arrays per animal and recorded the 128 most visually driven neural measurement sites (determined by separate pilot images) in one animal (58 IT, 70 V4) and 168 in another (110 IT, 58 V4). During image presentation we recorded multi-unit neural responses to our images from the V4 and IT sites. Images were presented on an LCD screen (Samsung, SyncMaster 2233RZ at 120 Hz) one at a time. Each image was presented for 100 ms with a diameter of $8°$ (visual angle) at the center of the screen on top of the half-gray background and was followed by a 100 ms half-gray "blank" period. The animal's eye position was monitored by a video eye tracking system (SR Research, EyeLink II), and the animal was rewarded upon the successful completion of 6–8 image presentations while maintaining good eye fixation (within $\pm 2°$) at the

center of the screen, indicated by a small $(0.25°)$ red dot. Presentations with larger eye movements were discarded. In each experimental block, we recorded responses to all images. Within one block each image was repeated once. Over all recording sessions, this resulted in the collection of 47 image repetitions, collected over multiple days. All surgical and experimental procedures were done in accordance with the National Institute of Health guidelines and the Massachusetts Institute of Technology Committee on Animal Care.

To arrive at the multi-unit neural representation, we converted the raw multi-unit neural responses to a neural representation through the following normalization process. For each image in a block, we compute the vector of raw firing rates across measurement sites by counting the number of spikes between 70 ms and 170 ms after the onset of the image for each site. We then subtracted the background firing rate, which is the average firing rate during presentation of a gray background ("blank" image), from the evoked response. In order to minimize the effect of variable external noise, we normalize each site by the standard deviation of each site's response to a block of images. Finally, the neural representation is calculated by taking the mean across the repetitions for each image and for each site, producing a scalar valued matrix of neural sites by images. This post-processing procedure is only our current best-guess at a neural code, which has been shown to quantitatively account for human performance [60]. Therefore, it may be possible to develop a more effective neural decoding for example influenced by intrinsic cortical variability [56], or dynamics [61,62].

To arrive at the single-unit neural representation, we followed a similar normalization process as the multi-unit representation, but first conducted spike-sorting on the multi-unit recordings. We sorted single-units from the multi-unit IT and V4 data by using affinity propagation [63] together with the method described in [64]. Using a conservative criteria we isolated 160 single-units from the IT recordings and 95 single-units from the V4 recordings with 6 repetitions per image for each single-unit. Given these single-unit responses for each image we followed a processing procedure identical to the multi-unit procedure, which included counting the number of spikes between 70 ms and 170 ms after the onset of the image, subtracting the background firing rate, and normalizing by the standard deviation of the site's response to a block of images. Finally, we selected from these single-units the top 40 based on response consistency over trials on a separate image set (units with high SNR). Specifically, we measured the response to 280 images not included in our evaluations of performance but drawn from a similar stimulus distribution. These images contained 7 unique objects not contained in the original set with 1 object from each of the 7 categories. For each object there are 40 images, each with a unique background and with the object position, scale, and pose randomly sampled. For each single-unit, we separated the responses over trials into two groups, averaged across trials, and measured the correlation of these response vectors. We then sorted the single-units based on this correlation and selected the 40 with highest correlation and therefore most consistent single-units. We repeated this procedure separately for V4 and IT. The resulting consistency measurement for the IT single-units was comparable to other measurements using single-unit electrophysiology [10]. The consistency of the 40 IT single-units was higher than the 15th percentile of single-units measured in [10]. In other words, the least consistent IT single-unit in the group of 40 was more consistent than the bottom 15% of single-units analyzed in [10]. Unless otherwise noted, all analyses of single-units use these 40 high-SNR single-units.

Kernel analysis methodology

We would also like to measure accuracy as we change the complexity of the prediction function. To accomplish this, we use an extension of the work presented in [30], which is based on theory presented in [28], and [29], and we refer to as *kernel analysis*. We provide a brief description of this measure and refer the reader to those references for additional details and justification on measuring precision against complexity. This procedure is a derivative of regularized least squares [65], which arrises as a Tikhonov minimization problem [46], and can be viewed as a form of Gaussian process regression [66]. We do not elaborate on the relationships between these views, but each is a valid interpretation on the procedure.

The measurement procedure, which we refer to here as *kernel analysis*, utilizes regularized kernel ridge regression to determine how well a task in question can be solved by a regularized kernel. A highly regularized kernel will allow only a simple prediction function in the feature space, and a weakly regularized kernel will allow a complex prediction function in the feature space. A *good* representation will have high variability in relation to the task in question and can effectively perform the task even under high regularization. Therefore, if the eigenvectors of the kernel with the largest eigenvectors are effective at predicting the categories, a highly regularized kernel will still be effective for that task. In contrast, an ineffective representation will have very little variation relevant for the task in question and variation relevant for the task is only contained in the eigenvectors corresponding to the smallest eigenvalues of the kernel and only a weakly regularized kernel will be capable of performing the task efficiently. Changing the amount of regularization changes the complexity of the resulting decision function: highly regularized kernels allow for only simple decision functions in the feature space and weakly regularized kernels allow for complex decision functions. Intuitively, a good representation is one that learns a simple boundary (highly regularized) from a small number of randomly-chosen examples, while a poor representation makes a more complicated boundary (weakly regularized), requiring many examples to do so.

In our formulation, kernel analysis consists of solving a regularized least squares or equivalently a kernel regression problem over a range of regularization [65]. The regularization parameter λ controls the complexity of the function and the precision or performance is measured as the leave-one-out generalization error (*looe*). We refer to the inverse of the regularization parameter $(1/\lambda)$ as the *complexity* and $1 - looe(\lambda)$ as the *precision*. Thus, the curve $1 - looe(\lambda)$ provides us with a measurement of the precision as a function of the model complexity for the given representational space. The curves produced by different representational spaces will inform us about the simplicity of the task in that representational space, with higher curves indicating that the problem is simpler for the representation.

One of the advantages of kernel analysis is that the kernel PCA method converges favorably from a limited number of samples. [29] shows that the kernel PCA projections obtained with a finite and typically small number of samples n (images in our context) are close with multiplicative errors to those that would be obtained in the asymptotic case where $n \mapsto \infty$. This result is especially important in our setting as the number of images we can reasonably obtain from the neural measurements is comparatively low. Therefore, kernel analysis provides us with a methodology for assessing representational effectiveness that has favorable properties in the low image sample regime, here thousands of images.

We next present the specific computational procedure for computing kernel analysis. Given the learning problem $p(x,y)$ and

a set of n data points $\{(x_1,y_1),...,(x_n,y_n)\}$ drawn independently from $p(x,y)$ we evaluate a representation defined as a mapping $x \mapsto \phi(x)$. For our case, the inputs x are images, the y are normalized category labels, and the ϕ denotes a feature extraction process.

As suggested by [30], we utilize the Gaussian kernel because this kernel implies a smoothness of the task of interest in the input space [67] and does not bias against representations that may be more adapted to non-linear regression functions. We compute the kernel matrix K_σ associated to the data set as

$$K_\sigma = \begin{pmatrix} k_\sigma(\phi(x_1),\phi(x_1)) & ... & k_\sigma(\phi(x_1),\phi(x_n)) \\ \vdots & & \vdots \\ k_\sigma(\phi(x_n),\phi(x_1)) & ... & k_\sigma(\phi(x_n),\phi(x_n)) \end{pmatrix}, \qquad (1)$$

where the standard Gaussian kernel is defined as $k_\sigma(x,x') = \exp\left(-\|x-x'\|^2/2\sigma^2\right)$ with scale parameter σ.

The regularized kernel regression problem is

$$\min_{\Theta \in R^n} \frac{1}{2} \sum_{i=1}^{n} \| Y - K_\sigma \Theta \|_2^2 + \frac{\lambda}{2} \Theta^t K_\sigma \Theta, \qquad (2)$$

where Y are the normalized vector of category labels, Θ is the vector of regression parameters and λ is the regularization scalar. The solution to the regularized regression problem for a fixed σ and λ is denoted as $\Theta_\sigma(\lambda)$ and is given as

$$\Theta_\sigma^*(\lambda) = (K_\sigma + \lambda I)^{-1} Y, \qquad (3)$$

where I is the identity matrix.

The leave-one-out error can be calculated as

$$LOOE_\sigma(\lambda) = \frac{\Theta_\sigma^*(\lambda)}{diag((K_\sigma + \lambda I)^{-1})}, \qquad (4)$$

where $diag(M)$ denotes the column vector satisfying $diag(M)_i = M_{ii}$ and the division is elementwise. Note that $LOOE_\sigma(\lambda)$ is a vector of errors for each example and we compute the mean squared error over the examples as $looe_\sigma(\lambda) = \frac{1}{n}\sum_i (LOOE_\sigma(\lambda)_i)^2$, which is considered a good empirical proxy for the error on future examples (generalization error). We note that the leave-one-out error can be computed efficiently given an eigendecomposition of the kernel matrix. We refer to [65] for derivation and details on this computation.

To remove the dependence of the kernel on σ we find the value that minimizes $looe_\sigma(\lambda)$ at that value of λ: $looe(\lambda) = \min_\sigma looe_\sigma(\lambda)$. Finally, for convenience we plot *precision* $(1 - looe(\lambda))$ against *complexity* $(1/\lambda)$. Note that because we optimize the value of σ for each value of complexity, σ (the width of the Gaussian kernel) will regularize the feature space when λ does not provide sufficient regularization. Because of this effect, the precision-complexity curves in Fig. 2 plateau at high values of complexity because the optimal value of σ increases in the high complexity regime (low values of λ). We would otherwise expect that at high complexity, the regression would over fit and produce poor generalization (low $1 - looe(\lambda)$ or high error). At low values of complexity (λ) we observe that there is an optimal value for σ, which is dependent on the representation, and is robust over image splits. Therefore, it is not possible to achieve high precision at low complexity simply by

reducing σ; the variation in the representational space must be aligned with the task in order to achieve high precision at low complexity.

Note that we have chosen to use a squared error loss function for our multi-way classification problem. While it might be more appropriate to evaluate a multi-way logistic loss function, we have chosen to use the least-squares loss because it provides a stronger requirement on the representational space to reduce variance within category and to increase variance between categories, and it allows us to distinguish representations that may be identical in terms of separability for a certain complexity but still have differences in their feature mappings. The kernel analysis of deep Boltzmann machines in [68] also uses a mean squared loss function in the classification problem setting and is widely used in machine learning.

In the discussion above, $Y = (y_1, \ldots, y_n)$ represents the vector of task labels for the images $(x_1, \ldots, x_n)$. In our specific case, the y_i are normalized category identity values (normalized such that predicting 0 will result in a precision equal to 0 and perfect prediction will result in a precision equal to 1). To generalize to the case of multiway categorization, we use a version of the common one-versus-all strategy. Assuming k distinct categories, for each category j we compute the per-class leave-one-out error $looe_\sigma(\lambda)^j$ by replacing Y in equations 2 and 3 with Y_j. The overall leave-one-out error is then the average over categories of the per-category leave-one-out error. Minimization over σ then proceeds as in the single category case.

To evaluate both neural representations and machine representations using kernel analysis we measure the $1 - looe(\lambda)$ curve. The image dataset consists of 1960 images containing seven object categories with seven instances per object category. The categories are Animals, Cars, Chairs, Faces, Fruits, Planes and Tables. To measure statistical variation due to subsampling of image variation parameters we compute the $1 - looe(\lambda)$ curve ten times, each time sampling 80% of the images with replacement. The ten samples are fixed for all representations and within each subset we equalize the number of images from each category. For each representation, we maximize over the values of the Gaussian kernel σ parameter as follows. We evaluate the Gaussian kernel scale parameter σ for each representation at a range of values centered on the median distance over the distribution of distances for that representation. With σ_m denoting the value equal to the median distance, we evaluated kernels with $\sigma = \alpha \sigma_m$ for 32 values of α in the range of 0.1 to 10 with logarithmic spacing. We found that in practice the values of σ that minimized the leave-one-out error were close to σ_m and well within this range. To span a large range of complexity we evaluate $looe(\lambda)$ at 56 values of λ from 10^{-4} to 10^3 with logarithmic spacing. For each representation, this procedure produces a curve for each of the subsets, where the mean and spread (range of the values over the 10 subsets for each value of λ) are shown in Fig. 2. As a summary value, we also compute the area under the curve (KA-AUC) for each of the image set randomizations and report the mean and standard deviation. We use these mean KA-AUC values for the measurements in Figs. 4A and 4B.

Machine representations

We evaluate a number of model representations from the literature, including several recent best of breed representational learning algorithms and visual representation models. In particular we examine three recent convolutional DNNs [24,25] + Yamins. The DNNs in [24,25] are of note because they have each successively surpassed the state-of-the-art performance on the ImageNet Large Scale Visual Recognition Challenge (ILSVRC).

Note that results have continued to improve on this challenge since we ran our analysis. See http://www.image-net.org/for the latest results.

Given the similarity of these DNNs to models of biological vision and our particular interest in the primate visual system, we also tested two representations that attempt to capture ventral stream processing. The "V1-like" model attempts to replicate functional responses in the primary visual area, the "V2-like" model similarly replicates response properties of the secondary visual area, and the HMAX instantiation is a model of ventral visual processing that implements the simple and complex cell hierarchy proposed in [13].

V1-like [35]. We evaluate the V1-like representation from Pinto et al.'s V1S+ [35]. This model attempts to capture a first-order account of primary visual cortex (V1). It computes a collection of locally-normalized, thresholded Gabor wavelet functions spanning orientation and frequency. This model is a simple, baseline biologically-plausible representation, against which more sophisticated representations can be compared. This model has 86400 dimensions.

V2-like [42]. We evaluated a recent proposal for the functional role of visual area V2 [42]. This model constructs a representation from conjunctions of Gabor outputs, which is similar to the V1-like model. The Gabor outputs are combined non-linearly and averaged within receptive field windows. The representation formed by this model has been shown to correspond to visual area V2 and explains visual crowding [42]. The output representation for the V2-like model has 24316 dimensions.

HMAX instantiation [41]. We evaluate the model in [41], which is a biologically inspired hierarchical model utilizing sparse localized features. This model has been shown to perform relatively well on previous measures of invariant object recognition [16], and to explain some aspects of ventral stream responses [13,22]. This representation has 4096 dimensions. Counting the simple-complex module as a single layer, this model has two layers.

HMO, Yamins et al. 2014 [27]. The model uses hierarchical modular optimization (HMO) to develop a large, deep network that is a combination of convolutional neural networks and is described in [27]. The HMO algorithm is an adaptive boosting procedure that interleaves hyper-parameter optimization (see [27] and references therein). The specific model we examine here is identical to the one in Yamins et al. 2014 [27]. This model is developed on a screening task that contains images of objects placed on randomly selected backgrounds. This task is similar in its construction to the task we use here to evaluate representational performance, however, it contains entirely different objects in totally non overlapping semantic categories, with none of the same backgrounds and widely divergent lighting conditions and noise levels. The specific model we examine was produced by the HMO optimization procedure on this screening task and is a convolutional neural network with 1250 top-level outputs. Therefore, the total dimensionality of the HMO representation is 1250 and the model is composed of four layers. The model is identical to the one presented in Yamins et al. 2014 [27]. However, the analysis methodology in [27] has a number of important differences to the analyses presented in this paper. The following differences are noteworthy: in this paper we examine only the high variation task and a subset of the images in the high variation task (7 categories with 7 objects per category in this paper vs. 8 categories with 8 objects per category in [27]), and in this paper we compute representational similarity on the raw HMO features, while in [27] representational similarity is calculated on the HMO-based IT model population.

Krizhevsky et al. 2012 [24] (SuperVision). We evaluate the deep convolutional neural network model "SuperVision" described in [24], which is trained by supervised learning on the ImageNet 2011 Fall release (∼15M images, 22K categories) with additional training on the LSVRC-2012 dataset (1000 categories). The authors computed the features of the penultimate layer of their model (4096 features) on the testing images by cropping out the center 224 by 224 pixels (this is the input size to their model). This mimics the procedure described in [24], in which this feature is fed into logistic regression to predict category labels. This model has seven layers as tested (counting a layer for each linear-convolution or fully connected dot-product).

Zeiler and Fergus 2013 [25]. The model is a very large convolutional network with 8 layers [25] trained using supervised learning on the LSVRC-2012 dataset to predict 1000 output categories. The training data is augmented by taking random crops and flips out of the 256 by 256 pixel images as in [24] to prevent overfitting. Additionally, visualization experiments exploring what deep models learn lead to hyperparameter choices which yield better performance. The feature representation is the 4096 features input to the softmax classifier averaged over crops from the 4 corners, center, and horizontal flips (producing ten 4096 dimensional vectors that are averaged to produce the 4096 dimensional representation used here). This model also has seven layers.

Experimental noise matched model

In our evaluation of representational performance we are limited by the observed noise in the neural representation. To produce a fair comparison we alter the model representational measurements by adding a level of noise that is matched to that observed in the neural representation. Note that we are unable to fully remove noise from the neural representation, and therefore we add noise to the model representations. The sources of the observed neural noise are various and we do not make an attempt to distinguish the sources of noise. Broadly, these noise sources are experimental in nature (e.g. physical electrode movement over time) or are intrinsic to the system. Examples of noise variation that are intrinsic to the system include the arousal state of the animal, trial-to-trial variability of neural responses, and correlated neural activity that is intrinsic to the system and not related to the stimulus condition.

We estimate a rate-dependent additive noise model from either the multi-unit or single-unit neural responses. The neural responses entering this noise model estimation follow the same averaging and background subtraction step as described previously. The rate-dependent additive noise model follows the common observation that spike counts of neurons are approximately Poisson [44,45]. We first normalize the variance of the neural response to 1,

$$\frac{1}{NMT} \sum_{ijk} (r_{ijk} - \mu)^2 = 1, \tag{5}$$

where r_{ijk} indicates the neural response with indices i over N neural sites, j over M images, and k over T trials, and μ is the mean over sites, images and trials. For each neural site we estimate a linear fit of the relationship between the mean response over trials (μ_{ij}) and the variance over trials (σ_{ij}) as,

$$a_i \mu_{ij} + b_i = \sigma_{ij} \tag{6}$$

with coefficients a_i and b_i. We then average these coefficients over the sites to produce a single relationship between mean rate and variance. For the multi-unit sites we estimate $a = 0.14$ and $b = 0.92$ and for the single-unit sites we estimate $a = 0.76$ and $b = 0.71$.

The noise model assumes that the empirically observed response variation is due to an underlying and unobserved signal contribution and a noise contribution. We can estimate the total variance of the empirical response ($\sigma^2_{signal+noise}$) as,

$$\begin{aligned} \sigma^2_{signal+noise} &= \frac{1}{NM} \sum_{ij} \left[\left(\frac{1}{T} \sum_{k} r_{ijk} \right) - \mu \right]^2, \\ \sigma^2_{signal+noise} &= \sigma^2_{signal} + \sigma^2_{noise}. \end{aligned} \tag{7}$$

Note that we cannot observe σ^2_{signal} and we use the subscript "noise" to denote the standard error of the mean of the signal estimate and not ordinary trial to trial variability. Therefore, the estimate of the standard error of the signal is given as,

$$\sigma^2_{noise} = \left(\frac{1}{NM} \sum_{ij} (a\mu_{ij} + b)^2 \right) / T. \tag{8}$$

Note that we estimate the standard error of the mean using the noise model, which has the benefit of being jointly estimated from the population of neural responses. In other words, if different neural sites share a similar noise model our estimate of that noise model is improved by estimating its parameters from all of the sites jointly. However, we have verified that using the empirical estimate of σ^2_{noise} gives nearly identical results.

To add noise to the model representations we first scale the total variance of the representation such that the variance after adding noise will be approximately equal to the variance observed in the neural sample. To do this we scale the model representation such that:

$$\frac{1}{LM} \sum_{lj} (s_{lj} - v)^2 = \sigma^2_{signal+noise} - \sigma^2_{noise}, \tag{9}$$

where s_{lj} indicates the model representation value with index l over L feature dimensions and v is the mean over image and features for the model representation. Finally we add signal dependent noise to the model representation:

$$\hat{s}_{lj} = s_{lj} + \mathcal{N}(0, (as_{lj} + b)^2 / T), \tag{10}$$

where $\mathcal{N}(\mu, \sigma^2)$ indicates a sample from the normal distribution with mean μ and variance σ^2 and $\hat{s}$ indicates the noise corrupted model representation. We verified empirically that the resulting variance of the noise corrupted model representation was close to the empirical neural signal variance. We repeat this procedure for the multi-units and single-units to arrive at model representations that are corrected for the observers noise in the respective neural measurement.

See S1 Fig. for a validation of the experimental noise matched model.

Linear-SVM methodology

See Supporting Information.

Predicting IT multi-unit sites from model representations

We performed generalized linear model analysis using ridge regression to predict the IT multi-unit responses from the model representations [69]. We utilized the same cross-validation procedure as the linear-SVM analysis, estimating encoding models on 80% of the data and evaluating the performance on the remaining 20% and repeating this for 10 randomizations. For each IT multi-unit we estimated an encoding model, predicted the testing responses, and determined the explained explainable variance for that multi-unit. To measure explainable variance we measured for each multi-unit site and each training set, the Spearman-Brown corrected split-half self-consistency over image presentations (trials). To arrive at a summary statistics we took the median explained explainable variance over the multi-units for each model representation. We also used the V4 multi-unit responses to produce encoding models of the IT multi-unit response. The results of this analysis are presented in Fig. 6. Note that the predictions produced by the V2-like representation proved to be unstable and produced high variance compared to the other representations.

Representational similarity analysis

To measure representational similarity we followed the analysis methodology in [32]. We first computed a feature vector per object for each representation by averaging the representational vectors over image variations. The representational dissimilarity matrix (RDM) is defined as

$$RDM(\phi(x))_{ij} = 1 - \frac{cov(\phi(x_i),\phi(x_j))}{\sqrt{var(\phi(x_i))\cdot var(\phi(x_j))}}, \qquad (11)$$

where $\phi(x)$ indicates the representation averaged for each object, i and j index the objects, cov indicates the covariance between the vectors and var the variance of the vector. Because we have 49 unique objects in our task the resulting RDM is a 49x49 matrix. To measure the relationship between two RDMs we measured the Spearman rank correlation between the upper-triangular, non-diagonal elements of the RDMs. We computed the RDM on 20% of the images and repeated the analysis 10 times. To compute noise due to the neural sample, we computed the split-half consistency between one half of the IT multi-units and the other half. We repeated this measurement over 50 random groupings and over the 10 image splits. Following the methodology in [27], we also predicted IT multi-unit responses to form a new representation, which we measured using representational similarity analysis. To produce IT multi-unit predictions for each model representation, we followed the same methodology as described previously (Predicting IT multi-unit sites from model representations). For each image split, we estimated encoding models on 80% of the images for each of the 168 IT multi-units and produced predictions for each multi-unit on the remaining 20% of the images. We then used these predictions as a new representation and computed the object-level RDM for the 20% of held-out images. We repeated the procedure 10 times. Note that the 20% of images used for each split was identical for all RDM calculations and that the images used to estimate multi-unit encoding models did not overlap with the images used to calculate the RDM. The analysis of the representations with the additional IT multi-unit fit can be seen as a different evaluation metric to the results of predicting IT multi-units. In other words, in Fig. 6 we evaluate the IT multi-unit predictions using explained variance at the image-level, and in Fig. 7 for the "+ IT-fit" representations we evaluate the IT multi-unit predictions using an object-level representational similarity analysis.

Supporting Information

S1 Fig Effects on kernel analysis performance of empirical noise vs. induced noise model. In the top left panel we show the performance measurements, as measured by kernel analysis area-under-the-curve, of the IT cortex multi-unit sample and of the IT cortex multi-unit sample with trial dependent added noise as we vary the number of experimental trials (repetitions per image) or the trials in the noise model (T in Eq. 10). In all plots error bars indicate standard deviations of the measure over 10 repetitions of the analysis. Results are replotted and divided by the maximum performance (Relative Performance) in the lower left panel. The same analysis is performed for the IT cortex single-unit sample in the right panels. These results indicate that the noise model reduces our performance measurement over the empirically observed noise and is therefore a conservative model for inducing noise in model representations. In other words, these results indicate that the model representations with neural matched noise are likely overly penalized.
(PDF)

S2 Fig Human performance on the visual recognition task as a function of presentation time. We plot the mean block-accuracy for different stimulates presentation durations from responses measured using Amazon Mechanical Turk. The mean accuracy is plotted as diamond markers and the error bars indicate the 95% confidence interval of the standard error of the mean over block-accuracies. Chance performance is ~14% for this task. The accuracy quickly increases such that at 100 ms stimulus duration it is within 92% of the performance at 2 seconds. This indicates that on this task, human subjects are able to perform relatively highly even during brief presentations of 100 ms. We refer to this ability as "core visual object recognition" [6] and we seek to measure the neural representational performance that subserves this ability.
(PDF)

S3 Fig Linear-SVM performance of model representations without sample or noise correction. Testing set classification accuracy averaged over 10 randomly-sampled test sets is plotted and error bars indicate standard deviation over the 10 random samples. Chance performance is ~14.3%. Unlike in Fig. 5, the model representations in this figure has not been modified to correct for sampling or noise.
(PDF)

S4 Fig Effect of sampling the neural and noise-corrected model representations for the linear-SVM analysis. We measure the mean testing-set linear-SVM generalization performance as we change the number of neural sites (for neural representations), or the number of features (for model representations). Measured samples are indicated by filled symbols and measured standard deviations indicated by error bars. Multi-unit analysis is shown in panel A and single-unit analysis in B. The model representations are noise corrected by adding noise that is matched to the IT multi-unit measurements (A, as indicated by the † symbol) or single-unit measurements (B, as indicated by the ‡ symbol). This analysis reveals a similar relationship to that found using the kernel analysis methodology (compare to Fig. 5).
(PDF)

S5 Fig Comparison of IT multi-unit and single-unit representations. In the left panel we plot the kernel analysis AUC as a function of the number of single- or multi-unit sites. We

plot results for two single-unit samples. "IT Cortex Single-Unit All-SNR Sample" uses all 160 isolated single-units and "IT Cortex Single-Unit High-SNR Sample" uses the 40 most consistent (least noisy) single-units. In the right panel we show the same data, but correct the number of multi-units to reflect an estimate of the number of single-units contributing to each multi-unit recording, thus plotting against the number of estimated neurons. Unlike previous figures, these estimates have a fixed number of trials (6) for both single- and multi-unit samples. Surprisingly, multi-unit recordings surpass single-unit recordings in performance (left) and are five times better per unit in performance to the all SNR single-unit sample (right).
(PDF)

S1 Text Supporting Information for Methods and Discussion. Validation of the experimental noise matched model; Human performance on our task as a function of presentation time; Linear-SVM methodology; Comparing IT multi-unit and single-unit representations; Processing time and energy consumption of computational models.
(PDF)

Author Contributions

Conceived and designed the experiments: CFC HH NP NJM JJD. Performed the experiments: CFC HH DLKY DA EAS NJM. Analyzed the data: CFC HH DLKY DA EAS. Contributed reagents/materials/analysis tools: CFC HH DLKY NP DA EAS. Wrote the paper: CFC HH JJD.

References

1. Thorpe S, Fize D, Marlot C (1996) Speed of processing in the human visual system. Nature 381: 520–522.
2. Fabre-Thorpe M, Richard G, Thorpe SJ (1998) Rapid categorization of natural images by rhesus monkeys. Neuroreport 9: 303–308.
3. Keysers C, Xiao D, F 246 ldi 225 k P, Perrett D (2001) The Speed of Sight. Journal of Cognitive Neuroscience 13: 90–101.
4. Potter MC, Wyble B, Hagmann CE, McCourt ES (2013) Detecting meaning in RSVP at 13 ms per picture. Attention, Perception, & Psychophysics 76: 270–279.
5. Andrews TJ, Coppola DM (1999) Idiosyncratic characteristics of saccadic eye movements when viewing different visual environments. Vision Research 39: 2947–2953.
6. DiCarlo JJ, Zoccolan D, Rust NC (2012) How Does the Brain Solve Visual Object Recognition? Neuron 73: 415–434.
7. Desimone R, Albright TD, Gross CG, Bruce C (1984) Stimulus-selective properties of inferior temporal neurons in the macaque. Journal of Neuroscience 4: 2051–2062.
8. Kobatake E, Tanaka K (1994) Neuronal selectivities to complex object features in the ventral visual pathway of the macaque cerebral cortex. Journal of Neurophysiology 71: 856–867.
9. Hung CP, Kreiman G, Poggio T, DiCarlo JJ (2005) Fast readout of object identity from macaque inferior temporal cortex. Science 310: 863–866.
10. Rust NC, DiCarlo JJ (2010) Selectivity and tolerance ("invariance") both increase as visual information propagates from cortical area V4 to IT. Journal of Neuroscience 30: 12978–12995.
11. DiCarlo JJ, Cox DD (2007) Untangling invariant object recognition. Trends in Cognitive Sciences 11: 333–341.
12. Fukushima K (1980) Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. Biological Cybernetics 36: 193–202.
13. Riesenhuber M, Poggio T (1999) Hierarchical models of object recognition in cortex. Nature Neuroscience 2: 1019–1025.
14. Stringer SM, Rolls ET (2002) Invariant Object Recognition in the Visual System with Novel Views of 3D Objects. Neural Computation 14: 2585–2596.
15. Serre T, Wolf L, Bileschi S, Riesenhuber M, Poggio T (2007) Robust Object Recognition with Cortex-Like Mechanisms. IEEE Transactions on Pattern Analysis and Machine Intelligence: 411–426.
16. Pinto N, Doukhan D, DiCarlo JJ, Cox DD (2009) A High-Throughput Screening Approach to Discovering Good Forms of Biologically Inspired Visual Representation. PLoS Computational Biology 5: e1000579.
17. Hubel DH, Wiesel TN (1962) Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. The Journal of Physiology 160: 106–154.
18. Hubel DH, Wiesel TN (1968) Receptive fields and functional architecture of monkey striate cortex. The Journal of Physiology 195: 215–243.
19. Perrett DI, Oram MW (1993) Neurophysiology of shape processing. Image and Vision Computing 11: 317–333.
20. Mel BW (1997) SEEMORE: Combining Color, Shape, and Texture Histogramming in a Neurally Inspired Approach to Visual Object Recognition. Neural Computation 9: 777–804.
21. Wallis G, Rolls ET (1997) Invariant Face and Object Recognition in the Visual System. Progress in Neurobiology 51: 167–194.
22. Serre T, Kreiman G, Kouh M, Cadieu C, Knoblich U, et al. (2007) A quantitative theory of immediate visual recognition. In: Progress in Brain Research, Elsevier. pp.33–56.
23. Le QV, Monga R, Devin M, Chen K, Corrado GS, et al. (2012) Building high-level features using large scale unsupervised learning. In: ICML 2012: 29th International Conference on Machine Learning. pp.1–11.
24. Krizhevsky A, Sutskever I, Hinton G (2012) ImageNet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems 25. pp.1106–1114.
25. Zeiler MD, Fergus R (2013) Visualizing and Understanding Convolutional Networks. ArXiv.org, arXiv: 1311.2901[cs.CV]
26. Sermanet P, Eigen D, Zhang X, Mathieu M, Fergus R, et al. (2014) OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks. In: International Conference on Learning Representations. pp.1–16.
27. Yamins DLK, Hong H, Cadieu CF, Solomon EA, Seibert D, et al. (2014) Performance-optimized hierarchical models predict neural responses in higher visual cortex. Proceedings of the National Academy of Sciences 111: 8619–8624.
28. Braun ML (2006) Accurate Error Bounds for the Eigenvalues of the Kernel Matrix. The Journal of Machine Learning Research 7: 2303–2328.
29. Braun ML, Buhmann JM, Müller KR (2008) On relevant dimensions in kernel feature spaces. The Journal of Machine Learning Research 9: 1875–1908.
30. Montavon G, Braun ML, Müller KR (2011) Kernel Analysis of Deep Networks. The Journal of Machine Learning Research 12: 2563–2581.
31. Kriegeskorte N, Mur M, Ruff DA, Kiani R, Bodurka J, et al. (2008) Matching Categorical Object Representations in Inferior Temporal Cortex of Man and Monkey. Neuron 60: 1126–1141.
32. Kriegeskorte N, Mur M, Bandettini P (2008) Representational Similarity Analysis – Connecting the Branches of Systems Neuroscience. Frontiers in Systems Neuroscience 2.
33. Mur M, Ruff DA, Bodurka J, De Weerd P, Bandettini PA, et al. (2012) Categorical, yet graded–single-image activation profiles of human category-selective cortical regions. The Journal of Neuroscience 32: 8649–8662.
34. Yamins D, Hong H, Cadieu CF, DiCarlo JJ (2013) Hierarchical Modular Optimization of Convolutional Networks Achieves Representations Similar to Macaque IT and Human Ventral Stream. Advances in Neural Information Processing Systems0020: 3093–3101.
35. Pinto N, Cox DD, DiCarlo JJ (2008) Why is Real-World Visual Object Recognition Hard? PLoS Computational Biology 4: e27.
36. Pinto N, Barhomi Y, Cox DD, DiCarlo JJ (2011) Comparing state-of-the-art visual features on invariant object recognition tasks. IEEE Workshop on Applications of Computer Vision (WACV 2011): 463–470.
37. Weiskrantz L, Saunders RC (1984) Impairments of Visual Object Transforms in Monkeys. Brain 107: 1033–1072.
38. Oliva A, Torralba A (2007) The role of context in object recognition. Trends in Cognitive Sciences 11: 520–527.
39. Pinto N, Majaj N, Barhomi Y, Solomon E, DiCarlo JJ (2010) Human versus machine: comparing visual object recognition systems on a level playing field. Cosyne Abstracts 2010, Salt Lake City USA.
40. Keerthi SS, Lin CJ (2003) Asymptotic behaviors of support vector machines with Gaussian kernel. Neural Computation 15: 1667–1689.
41. Mutch J, Lowe DG (2008) Object Class Recognition and Localization Using Sparse Features with Limited Receptive Fields. International Journal of Computer Vision 80: 45–57.
42. Freeman J, Simoncelli EP (2011) Metamers of the ventral stream. Nature Neuroscience 14: 1195–1201.
43. Cadieu CF, Hong H, Yamins D, Pinto N, Majaj NJ, et al. (2013) The Neural Representation Benchmark and its Evaluation on Brain and Machine. In: International Conference on Learning Representations. pp.1–16.
44. Tolhurst DJ, Movshon JA, Dean AF (1983) The statistical reliability of signals in single neurons in cat and monkey visual cortex. Vision research.
45. Shadlen MN, Newsome WT (1998) The Variable Discharge of Cortical Neurons: Implications for Connectivity, Computation, and Information Coding. Journal of Neuroscience: 3870–3896.
46. Evgeniou T, Pontil M, Poggio T (2000) Regularization Networks and Support Vector Machines. Advances in Computational Mathematics 13: 1–50.

47. Endres D, Földiák P (2007) Bayesian binning for maximising information rate of rapid serial presentation for sensory neurons. BMC Neuroscience 8: P151.

48. Vogels R, Sáry G, Orban GA (1995) How task-related are the responses of inferior temporal neurons? Visual Neuroscience 12: 207–214.

49. Koida K, Komatsu H (2006) Effects of task demands on the responses of color-selective neurons in the inferior temporal cortex. Nature Neuroscience 10: 108–116.

50. Suzuki W, Matsumoto K, Tanaka K (2006) Neuronal Responses to Object Images in the Macaque Inferotemporal Cortex at Different Stimulus Discrimination Levels. Journal of Neuroscience 26: 10524–10535.

51. Op de Beeck HP, Baker CI (2010) Informativeness and learning: Response to Gauthier and colleagues. Trends in Cognitive Sciences 14: 236–237.

52. Kobatake E, Wang G, Tanaka K (1998) Effects of shape-discrimination training on the selectivity of inferotemporal cells in adult monkeys. Journal of Neurophysiology 80: 324–330.

53. Baker CI, Behrmann M, Olson CR (2002) Impact of learning on representation of parts and wholes in monkey inferotemporal cortex. Nature Neuroscience 5: 1210–1216.

54. Sigala N, Logothetis NK (2002) Visual categorization shapes feature selectivity in the primate temporal cortex. Nature 415: 318–320.

55. Li N, DiCarlo JJ (2010) Unsupervised Natural Visual Experience Rapidly Reshapes Size-Invariant Object Representation in Inferior Temporal Cortex. Neuron 67: 1062–1075.

56. Stevenson IH, London BM, Oby ER, Sachs NA (2012) Functional Connectivity and Tuning Curves in Populations of Simultaneously Recorded Neurons. PLoS Computational Biology 8: e1002775.

57. LeCun Y, Bottou L, Bengio Y, Haffner P (1998) Gradient-based learning applied to document recognition. Proceedings of the IEEE 86: 2278–2324.

58. Rumelhart DE, Hinton GE, Williams RJ (1986) Learning representations by back-propagating errors. Nature 323: 533–536.

59. Mallat S (2012) Group Invariant Scattering. Communications on Pure and Applied Mathematics 65: 1331–1398.

60. Majaj N, Hong H, Solomon E, DiCarlo JJ (2012) A unified neuronal population code fully explains human object recognition. Cosyne Abstracts 2012, Salt Lake City USA.

61. Churchland MM, Cunningham JP, Kaufman MT, Foster JD, Nuyujukian P, et al. (2012) Neural population dynamics during reaching. Nature 487: 51–56.

62. Canolty RT, Ganguly K, Kennerley SW, Cadieu CF, Koepsell K, et al. (2010) Oscillatory phase coupling coordinates anatomically dispersed functional cell assemblies. Proceedings of the National Academy of Sciences 107: 17356–17361.

63. Frey BJ, Dueck D (2007) Clustering by Passing Messages Between Data Points. Science 315: 972–976.

64. Quiroga RQ, Nadasdy Z, Ben-Shaul Y (2004) Unsupervised Spike Detection and Sorting with Wavelets and Superparamagnetic Clustering. Neural Computation 16: 1661–1687.

65. Rifkin RM, Lippert RA (2007) Notes on Regularized Least Squares. MIT-CSAIL Technical Report 2007-025: 1–8.

66. Rasmussen CE, Williams CKI (2006) Gaussian Processes for Machine Learning. MIT Press.

67. Smola AJ, Schölkopf B, Müller KR (1998) The connection between regularization operators and support vector kernels. Neural Networks 11: 637–649.

68. Montavon G, Müller KR (2012) Deep Boltzmann Machines and the Centering Trick. Neural Networks: Tricks of the Trade: 621–637.

69. Wu MC, David SV, Gallant JL (2006) Complete functional characterization of sensory neurons by system identification. Annual review of neuroscience 29: 477.