A 3D shape inference model matches human visual object similarity judgments better than deep convolutional neural networks

Goker Erdogan (gerdogan@bcs.rochester.edu)
Robert A. Jacobs (robbie@bcs.rochester.edu)
Department of Brain and Cognitive Sciences
University of Rochester
Rochester, NY USA

Abstract

In the past few years, deep convolutional neural networks (CNNs) trained on large image data sets have shown impressive visual object recognition performances. Consequently, these models have attracted the attention of the cognitive science community. Recent studies comparing CNNs with neural data from cortical area IT suggest that CNNs may—in addition to providing good engineering solutions—provide good models of biological visual systems. Here, we report evidence that CNNs are, in fact, not good models of human visual perception. We show that a 3D shape inference model explains human performance on an object shape similarity task better than CNNs. We argue that deep neural networks trained on large amounts of image data to maximize object recognition performance do not provide adequate models of human vision.

Keywords: shape perception; object recognition; neural networks; 3D shape; deep learning;

Introduction

Despite decades of research, we know little about the neural representations underlying visual perception (Peissig & Tarr, 2007; Kourtzi & Connor, 2011). This is especially true of high-level representations involved in visual object identification and recognition. Although we understand little about how our brains accomplish visual perception, we are able to build engineering solutions that approach, and in some cases match, human performance on some visual tasks. Recently, multi-layered artificial neural networks known as convolutional neural networks (CNNs) have shown impressive object recognition performances when trained on large image data sets. Importantly for the cognitive science community, there seems to be evidence suggesting that these computer vision models may also be good models of biological visual systems (Kriegeskorte, 2015). Several studies have shown that CNNs provide good accounts of neural data from both monkey and human inferotemporal (IT) cortex, explaining almost all of the variance in some cases (Baldassi et al., 2013; Cadieu et al., 2014; Khaligh-Razavi & Kriegeskorte, 2014; Yamins et al., 2014). Here, we present evidence suggesting that CNNs are, in fact, not good models of human visual perception. We show that CNNs fail to capture people's responses on an object shape similarity task. Moreover, we show that a 3D shape inference model outperforms CNNs, suggesting that 3D structure is an important feature of people's visual object representations that CNNs fail to capture.

CNNs implement a sequence of convolution and subsampling operations to extract useful visual representations when trained in a supervised manner on large image data sets. Due to their huge impact on computer vision research, these models have now started to attract attention in cognitive science and neuroscience where they are actively investigated as models of biological visual perception.

A recent study by Khaligh-Razavi and Kriegeskorte (2014) compared a large set of models from computer vision and neuroscience to human fMRI and monkey neural data from IT. Similarity matrices calculated from each model were correlated with similarity matrices from human and monkey neural data. They found that AlexNet (Krizhevsky, Sutskever, & Hinton, 2012), a deep CNN trained on 1.2 million images, had the highest correlation with IT data. An ensemble model combining the outputs of each layer of AlexNet with scores from multiple categorization models trained on the features learned by AlexNet was able to capture the entire variance in IT data. The scores from animate/inanimate, face/nonface, and body/nonbody categorization models were needed to emphasize the differences between these categories, since it seems that IT gives more weight to these categorical distinctions than AlexNet did. These authors also showed that these results are not purely driven by the category structure in IT. AlexNet on its own did, in fact, capture some of the within-category structure. It is remarkable that a model trained to maximize object recognition accuracy is able to provide a good model of biological visual systems. This raises the interesting possibility that biological visual systems might be optimized primarily for object recognition. If so, a high-performing model of visual object categorization may also be a good model of biological visual systems.

Yamins et al. (2014) recently offered evidence for this claim. They showed that models that are better at categorization explain neural responses better. Instead of using a fixed set of models, they defined a model space using parameters that control various features of CNNs such as number of layers, filter sizes, and activation thresholds. Examining a large number of models in this space, their results showed that categorization performance was highly correlated with IT response predictivity. However, they also showed that an "ideal" categorization model was not highly correlated with IT responses. This suggests that solely aiming for good categorization does not, by itself, result in models predictive of IT responses. The authors claimed that it is the combination of a hierarchical architecture and high categorization performance that accounts for why CNNs provide good models of IT responses.

In spite of these impressive results, there is reason to question whether CNNs provide good models of human visual systems. CNNs are trained only to maximize object recognition performance. However, human visual systems solve not only object recognition but a myriad of visual tasks from segmentation to extraction of 3D shape. Indeed, because 3D shape and semantic category labels are highly correlated, it is unclear whether IT representations are best thought of as shape-based or semantic (Kourtzi & Connor, 2011). Even though CNNs often account for more variance in IT responses than other models, it is possible that the driving factor behind these results is shape similarity rather than semantic features. Baldassi et al. (2013) provided evidence that shape similarity, rather than semantic information, accounts for the structure of IT representations. They demonstrated that most of the semantic category structure in IT is explained by visual shape similarities within semantic categories. This result raises the question of whether the representations learned by CNNs—which receive supervised training based solely on semantic category labels—adequately characterize representations used by human visual systems. A striking demonstration suggesting these representations are, in fact, inadequate was provided by Szegedy et al. (2013). They showed that it is possible to create pairs of images that are indistinguishable to the human eye, but nonetheless are classified by CNNs into different classes. For example, it is possible to imperceptibly perturb an image that a CNN classifies as a bus such that the CNN classifies the perturbed image as an ostrich. This finding suggests that CNNs might be solving the problem of object recognition in a way that is rather different from that of human visual systems.

Here, we report behavioral evidence from an object shape similarity task suggesting that CNNs are not good models of human vision. Moreover, we show that a 3D shape inference model provides a better account for human behavior. We argue that models trained solely to maximize object recognition performance cannot capture the nature of human visual representations. A crucial feature of these representations not captured by these models is the 3D structure of objects.

Experiment

We created a set of 10 base objects using a "shape grammar" (Figures 1 and 2) where each object consisted of multiple rectangular blocks (referred to as "parts" and denoted by P in the grammar). A base object was generated as follows. To start, a root part was assigned 0-3 neighboring parts, also referred to as child parts, using the production rules of the shape grammar. A child part connected to the root part at one of its six faces. This face was chosen at random. Similarly, the width, height, and depth of a child part were randomly chosen from the range $[0,1]$. A child part could also be assigned neighboring (or grandchild) parts using the same production rules and random selections. Note that, in this framework, an object can be characterized using a "parse tree" due to our use of a shape grammar. We constrained the parse trees for our base objects to have a depth of four, which produces objects with three levels of parts (see Figures 2a and 2b for an example base object and its parse tree).

$$P \quad \rightarrow \quad P \mid PP \mid PPP \mid \varepsilon$$

Figure 1: Production rules of the shape grammar used in generating the experimental stimuli and representing shape in our 3D shape inference model. P is the only non-terminal symbol, and ε is the Null symbol.

Each base object was then used to create 8 additional objects, called variations, by applying 1 of 4 possible manipulations, referred to as *change part size*, *add part*, *remove part*, and *change connecting face of part*. Each of these four manipulations was applied at two different levels (second and third levels) of the parse trees (see Figure 2 for examples of each manipulation). When using the *change part size* manipulation, we picked one of the parts at the desired level in the parse tree and resampled its size. When using the *add part* manipulation, a new part was added to the desired level, picking its size and connecting face (i.e., the face of its parent to which it is connected) randomly. For the *remove part* manipulation, we again randomly picked one part at the desired level and removed it and all of its children parts. Lastly, for the *change connecting face* manipulation, we randomly picked a part and chose a new connecting face for it from the empty faces of its parent. This manipulation moved the part and all of its children.

Experimental stimuli consisted of images of the 10 base objects and 80 variations (90 images in total).[1] We used Blender (`http://www.blender.org`), a 3D computer graphics and animation software package, to render each object from a random viewpoint by rotating the camera around the vertical axis keeping its distance to the origin fixed. Consequently, there was significant pose variation in our experimental stimuli.

The goal of the experiment was to collect people's object shape similarity judgments. On each trial, a subject was presented with one target and two comparison objects, and was asked to pick the comparison object that he or she thought was more similar in shape to the target object. The target object was always one of the ten base objects, and the two comparisons were two randomly picked variations of the target object. (For instance, a trial may show Figure 2a as the target object, and Figures 2c and 2d as the comparison objects.) On "catch" trials, one of the comparison objects was the same as the target object. Each subject participated in 100 trials, 16 of which were catch trials. The experiment was performed on the world wide web by 41 subjects via Amazon Mechanical Turk. Five subjects were discarded because they failed to reach 85% correct performance on catch trials.

[1] The entire set of experimental stimuli can be seen online at `http://gokererdogan.github.io/CogSci16SupplementaryMaterials/`

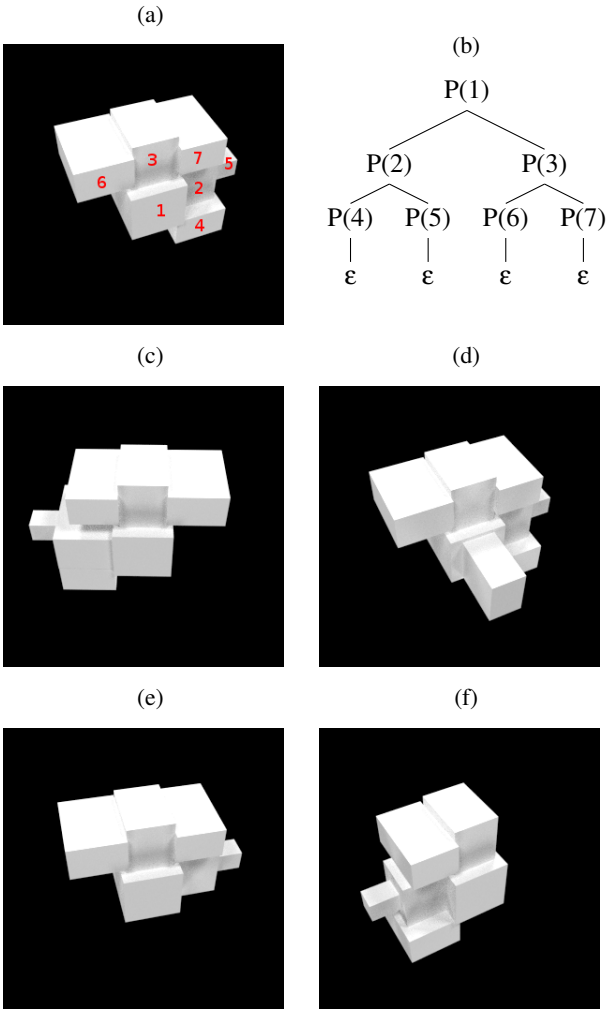

Figure 2: (a) An example base object. The numbers on parts refer to the part numbers in its parse tree. (b) Parse tree representing the object in (a). (c)-(f) Examples of *change part size* (c), *add part* (d), *change connecting face of part* (e), *remove part* (f) manipulations, respectively. The parts affected by each manipulation are Part 2 in (c), Part 4 in (e), and Part 6 in (f).

Computational Models

We compare five models on how well they account for our experimental data.

Pixel-Based model: The first one, called the Pixel-Based model, works directly on image pixel values. The dissimilarity between two objects is calculated as the Euclidean distance between their images in pixel space. The predictions of the Pixel-Based model are determined by calculating the distances between each comparison object and the target, and choosing the comparison that is closest to the target.

CNN models: Our main aim is to compare the performances of deep CNNs and a 3D shape inference model. For this purpose, we use two CNNs.[2] The first one is the eight-layer (five convolutional, three fully connected layers) CNN by Krizhevsky et al. (2012), referred to as AlexNet, trained on 1.2 million images in the ImageNet dataset. AlexNet achieved the best performance on the 2012 ImageNet Large Scale Visual Recognition Challenge. We treat each of its layers as a separate mini-model. There are, in total, 14 layers (making the three max-pooling and two normalization layers explicit). Using the standard terminology in the deep neural network literature, these layers are: *conv1, pool1, norm1, conv2, pool2, norm2, conv3, conv4, conv5, pool5, fc6, fc7, fc8*, and *prob*. The last layer, *prob*, is a 1000-dimensional vector encoding the probability of belonging to each of 1000 object categories in ImageNet. The second deep CNN that we test is by Szegedy et al. (2014), named GoogLeNet, which set the state-of-the-art performance on the 2014 ImageNet Large Scale Visual Recognition Challenge. GoogLeNet has 22 layers (with an additional five pooling layers). Our simulations used 16 layers: *pool1, conv2, inception3a-b, pool3, inception4a-e, pool5, inception5a-b, pool5, loss3* and *prob*. To make predictions from AlexNet and GoogLeNet, we input each image to a CNN and perform a bottom-up pass to calculate each layer's responses. The dissimilarity between two objects is computed as the Euclidean distance between these responses. When presented with a trial from our experiment, a mini-model chooses the comparison object that is closest in its response space to the target object.

Our 3D shape inference model: We developed a shape perception model that aims to infer 3D shape from 2D input images. Similar to our previously published 3D shape inference models (Yildirim & Jacobs, 2013; Erdogan, Yildirim, & Jacobs, 2015), this model combines a representational language characterizing 3D shape with forward models mapping from shape representations to 2D images. Using Bayesian inference, we invert this forward 3D-to-2D mapping and extract 3D shape from 2D images. Formally, a shape representation H consists of a string T from our shape grammar (Figure 1) and a spatial model S that associates a size vector ($s \in \mathbb{R}^3$) and a connecting face ($f \in \{1,2,3,4,5,6\}$) with each P node in T. The probability of H is

$$p(H) = p(S|T)p(T) \tag{1}$$

where $p(T)$ is the probability of producing parse tree T from the shape grammar. We assume production probabilities to be uniform[3] which gives the following expression for $p(T)$

$$p(T) = \frac{1}{4^{|\mathcal{P}|}}. \tag{2}$$

The probability for spatial model S consists of the probabilities of picking part sizes and connecting faces. Since we assumed part sizes to be uniform over the interval [0, 1], we only need to focus on the probabilities for connecting faces.

[2] We use the pretrained models provided by the Caffe framework (Jia et al., 2014).

[3] Production probabilities can also be integrated out, which leads to a slightly different prior distribution. Note that our results here are significantly robust to choice of prior distribution.

For a part with k available faces and c children, there are $\binom{k}{c}$ possible combinations of face assignments to its children. Since we have six empty faces for the root P node and five for the remaining P nodes (because one face is occupied by the parent), the probability of spatial model S is

$$p(S|T) = \frac{1}{\binom{6}{|O_{\text{root}}|}\prod_{n\in\{\mathcal{P}\setminus\text{root}\}}\binom{5}{(|O_n|-1)}} \tag{3}$$

where O_i refers to the set of occupied faces of node i. To map these shape representations to 2D images, we use a forward model that takes in the 3D representation and renders it as a 2D image. Because the shape representation H does not specify the viewpoint, forward model F takes in viewpoint θ along with the shape representation H and produces a 2D image I (i.e., $F:\{H,\theta\}\rightarrow I$). We used the Visualization Toolkit (VTK; `http://www.vtk.org`), a software package for 3D computer graphics, image processing, and visualization, to implement the forward model. To define the likelihood function $\mathcal{L}(H,\theta;I)$, we assume Gaussian noise on I:

$$\mathcal{L}(H,\theta;I) = p(I|H,\theta) \propto \frac{1}{\sigma^2}||I-F(H,\theta)||_F^2. \tag{4}$$

Here σ^2 denotes the variance of the noise (this is the only free parameter of the model—it was set to a value that achieves acceptance rates around 20%) on I, and $||\cdot||_F$ is the Frobenius norm. Combining the prior on shape representations and the likelihood function, we use Bayes' rule to infer likely 3D shape representations given a 2D image:

$$p(H,\theta|I) \propto p(I|H,\theta)p(H)p(\theta). \tag{5}$$

We assume $p(\theta)$ is a uniform distribution. Object similarity is computed by calculating how likely the model is to observe the image for one object given the image of the other. Denoting the images by I_1 and I_2, we calculate three similarity measures: $p(I_2|I_1)$, $p(I_1|I_2)$, and their average. We calculate $p(I_2|I_1)$ as follows (and similarly for $p(I_1|I_2)$):

$$p(I_2|I_1) = \int p(I_2|H,\theta)p(H|I_1)p(\theta)dHd\theta. \tag{6}$$

The expression inside the integral in Eqn. 6 is equivalent to inferring the 3D shape representation for I_1, picking a random viewpoint, and calculating the sum of squared error between the observed I_2 and the rendered image on the basis of inferred H and chosen viewpoint.

To sample from the posterior distribution $p(H,\theta|I)$[4] we use an MCMC procedure. We devised multiple proposal strategies to move in the hypothesis space, and used a Metropolis-Hastings (MH) algorithm to sample from the posterior.[5] These proposal strategies are: *add/remove part*, *change part size*, *change connecting face of part*, and *change viewpoint*. The *add/remove part* either adds a new P node to a random location in the tree, or removes randomly one of the P nodes with no child parts. Note that this move jumps between spaces of different dimensions; hence, we need to use a reversible-jump MCMC method. For the *change part size* move, we resample the size of a randomly picked P node. Similarly, the *change connecting face of part* move picks one P node randomly and assigns it a new random connecting face from the available faces of its parent P node. Finally, the *change viewpoint* move rotates the viewpoint around the vertical axis a random amount, which is drawn from a Gaussian distribution. Due to space limitations, we cannot go into the implementation details here.[6]

In our simulations, we ran one chain for each image used in the experiment. To speed convergence, we constrained the depth of parse trees to be at most six. Each chain was run for 200,000 iterations, and sample collection started after the first 50,000 iterations. Hence, we had 15 samples per image (see Figure 3 for two typical samples [i.e., two illustrations of the model's inferred 3D shape given an image]). To calculate the similarity between two images, we used Eqn.6, approximating the integral by a sum over samples from the posterior $p(H|I)$.

Ideal 3D observer model: As our last model, we use an "ideal" 3D observer that can perfectly extract the true 3D shape of an object from its image. Although not realistic, this model provides a useful benchmark because it defines optimal performance for our model. Two objects are compared by an alignment mechanism that rotates one object and finds the viewpoint that matches the image of the other object best (as in Eqn. 6). If we assume that shape matching is done on the basis of only the MAP sample, this ideal observer model sets the performance upper bound for our 3D shape inference model.

Results and Discussion

We calculated the predictions of each model as described in the previous section. For the experimental data, we gathered the data from all subjects and, for each trial, chose the majority response. We measured the accuracy of each model by calculating the percentage of correctly predicted trials (where a trial is correctly predicted if a model's response matches the subjects' majority response).

The results are shown in Figure 4. The Pixel-Based model has the lowest accuracy with 58%. CNNs achieve accuracies of 62% (AlexNet, using responses of the output layer) and 64% (GoogLeNet, using responses of the layer *inception5a*). Our 3D shape inference model achieves 72% accuracy using the similarities calculated by averaging $p(\text{Comparison}|\text{Target})$ and $p(\text{Target}|\text{Comparison})$. This performance is significantly better than both AlexNet's (binomial test, $p < 0.001$) and GoogLeNet's performance ($p =$

[4]To calculate Eqn. 6, we need samples from $p(H|I)$. However, $p(H|I) \approx p(H,\theta_{\text{MAP}})$ because there is only a single viewpoint from which an object H looks close to its image I. The results reported here do not change if we integrate out θ instead of using the MAP sample.

[5]The code for our shape inference model is available at `https://github.com/gokererdogan/Infer3DShape`

[6]Readers interested in these details can contact the first author.

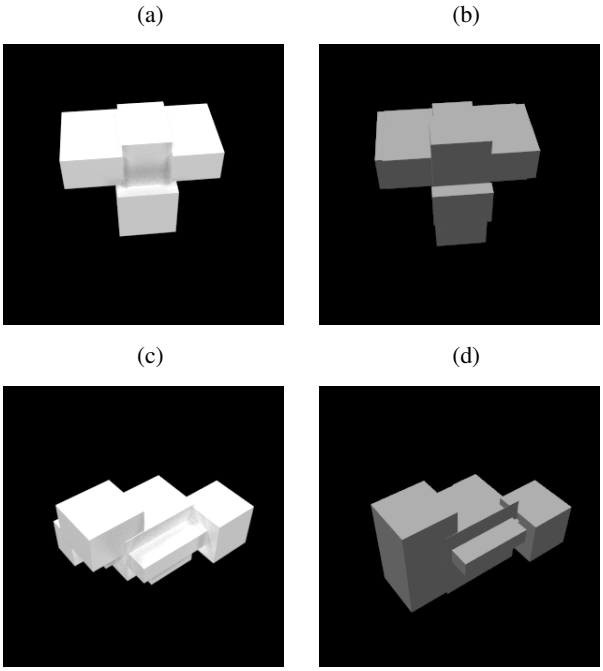

Figure 3: Sample runs of our 3D shape inference model. (a) An example input image. (b) One sample from our model for the image in (a). (c)-(d) Another example input image and a sample from our model.

0.004). The 3D ideal observer reaches an accuracy of 76%. However, the performances of the 3D ideal observer model and of our model are not significantly different ($p = 0.21$).

Because subjects did not show a strong preference for either of the comparisons in some trials, we also measured performance on only "high confidence" trials in which at least 80% of the subjects picked the same response. There were in total 120 (out of 280) high-confidence trials. Performances were as follows (due to space constraints, we omit the graph of these results). The Pixel-Based model's accuracy is 62%. Using the outputs of layer *prob*, AlexNet performs at 73%. GoogLeNet achieves an accuracy of 68% with the outputs of layer *inception5b*. Our 3D shape inference model performs the best, matching the accuracy of the 3D ideal observer model with 87% accuracy using the average similarity measure. This performance is significantly better than both GoogLeNet's performance (binomial test, $p < 0.001$) and AlexNet's performance ($p < 0.001$).

Our comparison here might seem unfair because our model knows that stimuli are built out of blocks while we used pre-trained CNNs that have never seen similar objects. However, subjects in our experiment have also never seen objects like our stimuli. In addition, previous studies presenting CNNs as good models of our visual systems used pre-trained networks. However, in order to alleviate further concerns, we have fitted the representations learned by CNNs to subjects' data using a metric-learning (Kulis, 2013) approach.[7] Accuracies have improved slightly (2%-4% increase) but not significantly, and our model still significantly outperforms both CNNs.

Taken together, these results show that a 3D shape inference model captures human performance better than deep CNNs on an object shape similarity task. This suggests that CNNs are, in fact, not good models of human vision. Although CNNs perform significantly better than chance, we believe this is due largely to the correlations between semantic object categories and shape features (Baldassi et al., 2013). Our study casts doubt on the claim that biological visual systems are optimized chiefly for object categorization, and that a system trained solely for object categorization will learn representations that are similar to ours. To the contrary, the low performances of the intermediate layers of the CNNs in our study suggests the opposite. Why does our 3D shape inference model perform better than CNNs? We believe this is due to the 3D nature of our model's shape representations. In contrast, a CNN trained to maximize object categorization performance learns to extract 3D features only to the extent that 3D information helps discriminate object categories. Therefore, it is unclear whether shape representations learned by CNNs carry 3D shape information. Since there is substantial evidence showing that human and monkey IT are selective for 3D shape (Orban, 2011), it becomes doubtful that CNNs offer good models of biological visual systems.

Lastly, we believe that our model is better suited than CNNs to understand visual perception in its totality because it is not intended simply as a model of object categorization. Biological visual systems solve a myriad of tasks from segmentation to scene perception, and our model can be readily extended to handle these diverse set of tasks. Moreover, vision is just one aspect of perception. We believe that our model—with its combination of a rich, modality-independent representational language, a forward model, and Bayesian inference—provides a promising theoretical framework for understanding not only visual, but also multisensory perception (Yildirim & Jacobs, 2013; Erdogan et al., 2015).

[7] See Supplementary Materials for further information.

Acknowledgments

This work was supported by AFOSR (FA9550-12-1-0303) and NSF (BCS-1400784) research grants.

References

Baldassi, C., Alemi-Neissi, A., Pagan, M., DiCarlo, J. J., Zecchina, R., & Zoccolan, D. (2013). Shape Similarity, Better than Semantic Membership, Accounts for the Structure of Visual Object Representations in a Population of Monkey Inferotemporal Neurons. *PLoS Comput Biol*, *9*(8), e1003167.

Cadieu, C. F., Hong, H., Yamins, D. L. K., Pinto, N., Ardila, D., Solomon, E. A., ... DiCarlo, J. J. (2014). Deep Neural Networks Rival the Representation of Primate IT Cortex for Core Visual Object Recognition. *PLoS Comput Biol*, *10*(12), e1003963.

Erdogan, G., Yildirim, I., & Jacobs, R. A. (2015). From Sensory Signals to Modality-Independent Conceptual Rep-

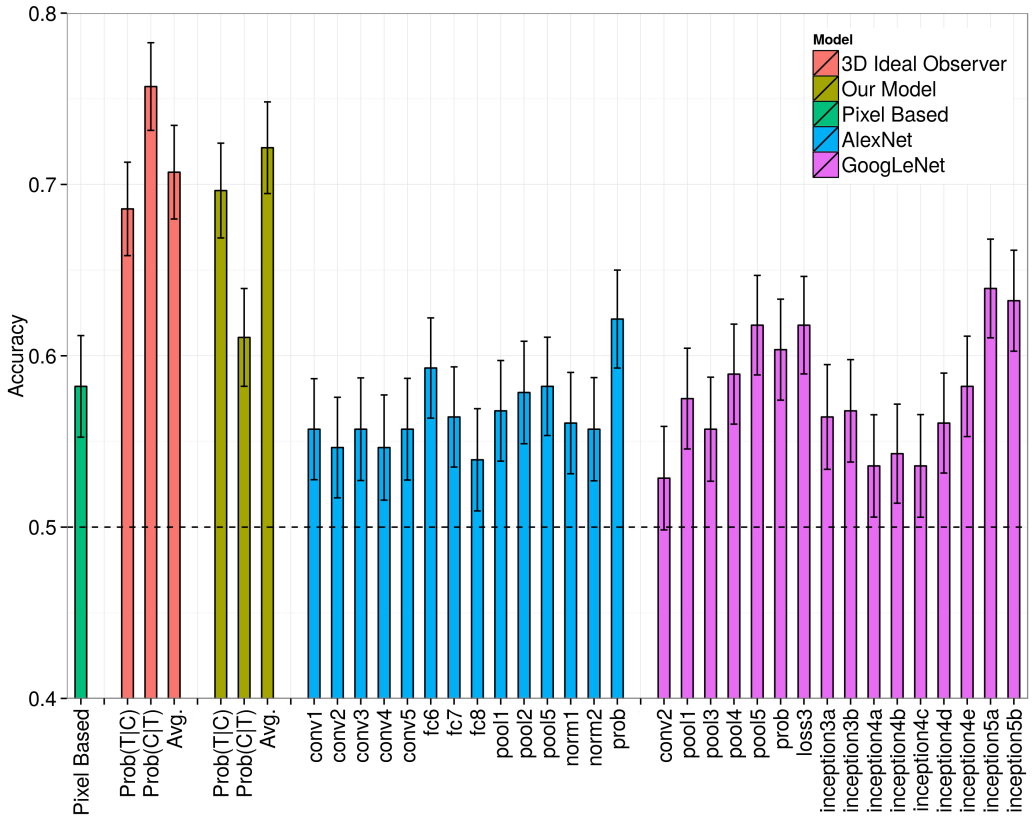

Figure 4: Prediction accuracies of each model using all experimental trials. The error bars denote SEMs and are calculated by bootstrapping with 1000 replications. P(C|T): similarity calculated from p(Comparison|Target). P(T|C): similarity calculated from p(Target|Comparison). Avg: similarity calculated from the average of these two probabilities. See the text for the layer labels for AlexNet and GoogLeNet. Note that the y-axis starts from 0.4.

resentations: A Probabilistic Language of Thought Approach. *PLoS Comput Biol*, *11*(11), e1004610.

Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., ... Darrell, T. (2014). Caffe: Convolutional Architecture for Fast Feature Embedding. *arXiv:1408.5093*.

Khaligh-Razavi, S.-M., & Kriegeskorte, N. (2014). Deep Supervised, but Not Unsupervised, Models May Explain IT Cortical Representation. *PLoS Comput Biol*, *10*(11), e1003915.

Kourtzi, Z., & Connor, C. E. (2011). Neural Representations for Object Perception: Structure, Category, and Adaptive Coding. *Annual Review of Neuroscience*, *34*(1), 45–67.

Kriegeskorte, N. (2015). Deep Neural Networks: A New Framework for Modelling Biological Vision and Brain Information Processing. *Annual Reviews of Vision Science*, *1*, 417–446.

Krizhevsky, A., Sutskever, I., & Hinton, G. (2012). Imagenet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems 25* (pp. 1097–1105).

Kulis, B. (2013). Metric Learning: A Survey. *Foundations and Trends® in Machine Learning*, *5*(4), 287–364.

Orban, G. A. (2011). The Extraction of 3d Shape in the Visual System of Human and Nonhuman Primates. *Annual Review of Neuroscience*, *34*(1), 361–388.

Peissig, J. J., & Tarr, M. J. (2007). Visual Object Recognition: Do We Know More Now Than We Did 20 Years Ago? *Annual Review of Psychology*, *58*, 75–96.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... Rabinovich, A. (2014). Going Deeper with Convolutions. *arXiv:1409.4842*.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., & Fergus, R. (2013). Intriguing Properties of Neural Networks. *arXiv: 1312.6199*.

Yamins, D. L. K., Hong, H., Cadieu, C. F., Solomon, E. A., Seibert, D., & DiCarlo, J. J. (2014). Performance-optimized Hierarchical Models Predict Neural Responses in Higher Visual Cortex. *Proceedings of the National Academy of Sciences*, *111*(23), 8619–8624.

Yildirim, I., & Jacobs, R. (2013). Transfer of Object Category Knowledge Across Visual and Haptic Modalities: Experimental and Computational Studies. *Cognition*, *126*, 135–148.