arXiv:1611.06448v2 [cs.CV] 22 Nov 2016

PsyPhy: A Psychophysics Driven Evaluation Framework for Visual Recognition

Brandon RichardWebster
University of Notre Dame
Notre Dame, IN 46556
brichar1@nd.edu

Samuel E. Anthony
Harvard University
Cambridge, MA 02138
santhony@wjh.harvard.edu

Walter J. Scheirer
University of Notre Dame
Notre Dame, IN 46556
walter.scheirer@nd.edu

Abstract

By providing substantial amounts of data and standardized evaluation protocols, datasets in computer vision have helped fuel advances across all areas of visual recognition. But even in light of breakthrough results on recent benchmarks, it is still fair to ask if our recognition algorithms are doing as well as we think they are. The vision sciences at large make use of a very different evaluation regime known as Visual Psychophysics to study visual perception. Psychophysics is the quantitative examination of the relationships between controlled stimuli and the behavioral responses they elicit in experimental test subjects. Instead of using summary statistics to gauge performance, psychophysics directs us to construct item-response curves made up of individual stimulus responses to find perceptual thresholds, thus allowing one to identify the exact point at which a subject can no longer reliably recognize the stimulus class. In this paper, we introduce a comprehensive evaluation framework for visual recognition models that is underpinned by this methodology. Over millions of procedurally rendered 3D scenes and 2D images, we compare the performance of well-known convolutional neural networks. Our results bring into question recent claims of human-like performance, and provide a path forward for correcting newly surfaced algorithmic deficiencies.

1. Introduction

> We often attribute "understanding" and other cognitive predicates by metaphor and analogy to cars, adding machines, and other artifacts, but nothing is proved by such attributions.
>
> *John Searle*

Imagine the following scenario: a marvelous black box algorithm has appeared that purportedly solves visual object recognition with human-like ability. As a good scientist, how might you go about falsifying this claim? By all accounts, the algorithm achieves superior performance on established benchmarks in computer vision, but its internal workings are opaque to the external observer. Such a situation is not far fetched — it should be familiar to any of us studying machine learning for visual recognition. What many of us in computer vision might not realize is that this setup happens to be the classic Chinese Room [48] problem proposed by the philosopher John Searle.

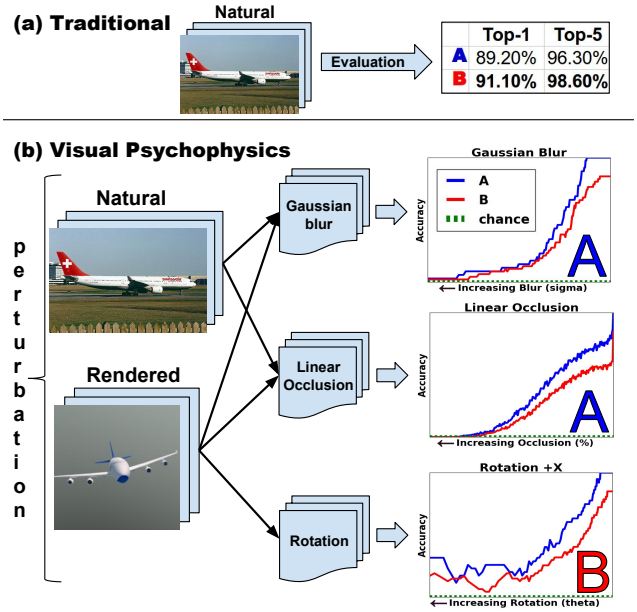

Figure 1. In this paper, the concept of applying psychophysics [28, 41] on a recognition model is introduced. (Top) In traditional dataset-based evaluation, summary statistics are generated over large sets of data, with little consideration given to specific conditions that lead to incorrect recognition instances. (Bottom) Psychophysics, a set of experimental concepts and procedures from psychology and neuroscience, helps us plot the exact relationships between perturbed test images and resulting model behavior to determine the precise conditions under which models fail. Instead of comparing summary statistics, we compare item-response curves representing performance (y-axis) versus the dimension of the image being manipulated (x-axis).

In Searle's thought experiment, a person who does not

speak Chinese is alone in a locked room and following instructions from a computer program to generate Chinese characters to respond to Chinese messages that are slipped under the door. To the message passer outside of the room, the person inside understands Chinese. However, this is not the case. The person inside the room is simply following instructions to complete the task — there is no real replication of the competency of knowing the Chinese language. Linking this back to computer vision, the summary statistics of performance from our algorithms look good on benchmark tests — enough so that we believe them to be close to human performance in some cases. But are these algorithms really solving the general problem of visual object recognition, or are they simply leveraging "instructions" provided in the form of labeled training data to solve the dataset?

Datasets in computer vision are intended to be controlled testbeds for algorithms, where the task and difficulty can be modulated to facilitate measurable progress in research. A dataset could be made up of images specifically acquired for experimentation (as is common in human biometrics [37, 36, 3]), or publicly available images crawled from the web (the predominant mode for all areas of visual recognition). Under this regime, strong advancements have been demonstrated for a number of problems [45, 19, 46, 44, 7, 29]. Deep learning is now a mainstay in computer vision thanks in part to the 2012 ImageNet Large Scale Visual Recognition Challenge [21], where AlexNet [25] reduced top-5 object classification error to 16.4% from 25.8% in the 2011 challenge [20]. When algorithms are evaluated on a common footing, it is possible to track meaningful improvements in artificial intelligence like this one. However, noticeable increases in error during cross-dataset evaluations [54] and lackluster performance in some real world applications [27] make us wonder if this is the *only* way we should be doing things.

When it comes to natural intelligence, neuroscientists and psychologists do not evaluate animals or people in the same way that computer vision scientists evaluate algorithms — and for a very good reason. With a collection of images crawled from the web, there is no straightforward way to determine the exact condition(s) that caused a subject to fail at recognizing a stimulus presented during an experiment. A natural image is the product of the physics at the instant the sensor acquired the scene; its latent parameters are largely unknown. Instead, for behavioral experiments meant to discover perceptual thresholds (*i.e.*, the average point at which subjects start to fail), the vision sciences outside of computer vision make use of the concepts and procedures from the discipline of *visual psychophysics*.

Psychophysics is the quantitative study of the relationships between controlled stimuli and the behavioral responses they elicit in a subject [28, 41]. It is a way to probe perceptual processes through the presentation of incremental and, in many cases, extremely fine-grained perturbations of visual stimuli. The properties of each stimulus are varied along one or more physical dimensions, thus controlling the difficulty of the task. The result (Fig. 1) is an *item-response curve* [13], where performance (*e.g.*, accuracy) on the y-axis is plotted against the dimension being manipulated (*e.g.*, Gaussian blur) on the x-axis. Each point on the curve reflects an individual stimulus, letting us map performance back to causal conditions in a precise manner. Psychophysics is an indispensable tool to vision science, and has been deployed to discover the minimum threshold for stimulation of a retinal photoreceptor (a single photon) [16], confirm Helmholtz's assertions on color absorption in the retina [5], and establish criteria to diagnose prosopagnosia [11] (the inability to recognize a face). As in these discoveries from biological vision, we submit that psychophysics holds much promise for discovering new aspects of the inner workings of machine learning models.

In this paper, we introduce a comprehensive evaluation framework for visual recognition that is underpinned by the principles of psychophysics. In this regime, a stimulus can be an object drawn from purely rendered data or natural scene data, and a varying physical parameter can control the amount of transformation in the subsequent set of manipulated images derived from the original stimulus. A key difference from traditional benchmarks in computer vision is that instead of looking at summary statistics (*e.g.*, average accuracy, AUC, precision, recall) to compare algorithm performance, we compare the resulting item-response curves. For complete control of the underlying parameter space, we find that procedural graphics [53, 62, 26, 59] are a useful way to generate stimuli that can be manipulated in any way we desire. Because we have the procedure that rendered each scene, we can find out where a model is failing at the parametric level. As we will see, by using this framework to explore artificial vision systems like psychologists, many interesting new findings can be surfaced about the strengths and limitations of computer vision models.

Our main contributions are as follows:

- A general evaluation framework is developed for performing visual psychophysics on computer vision models. The framework has a strong grounding in well-established work in psychology and neuroscience for behavioral experimentation.
- An investigation of procedural graphics for large-scale psychophysics experiments applied to models.
- A parallelized implementation of the psychophysics framework that is deployable as a Python package.
- A case study consisting of a battery of experiments incorporating millions of procedurally rendered images and 2D images that were perturbed, performed over

a set of well-known Convolutional Neural Network (CNN) models [25, 23, 51, 49].

2. Related Work

Methods of Evaluation from the Vision Sciences. With respect to work in computer vision directly using psychophysics, most is related to establishing human baselines for comparison to algorithmic approaches. Riesenhuber and Poggio [43] described a series of psychophysical comparisons between humans and the HMAX [42] model of visual cortex using a limited set of stimuli rendered by computer graphics. Similarly, Eberhardt et al. [12] designed an experiment to measure human accuracy and reaction time during visual categorization tasks with natural images, which were then compared to different layers of CNN models [25, 49]. With respect to low-level features, Gerhard et al. [14] introduced a new psychophysical paradigm for comparing human and model sensitivity to local image regularities.

Psychophysics can also be used for more than just performance evaluation. Scheirer et al. [47] introduced the notion of "perceptual annotation" for machine learning, whereby psychophysical measurements are used as weights in a loss function to give a training regime some *a priori* notion of sample difficulty. Using accuracy and reaction time measured via the online psychophysics testing platform TestMyBrain.org [15], perceptual annotation was shown to enhance face detection performance. Along these lines, Vondrick et al. [57] devised a method inspired by psychophysics to estimate useful biases for recognition in computer vision feature spaces.

Outside of work specifically invoking psychophysics, one can find other related methods from psychology and neuroscience for behavioral-style testing of models. 2D natural images are the most common type of data in computer vision, and thus form a good basis for algorithmic evaluation in this mode. For face recognition, Sinha et al. [50] suggested that the use of incremental manipulations to the resolution, pose, illumination, expression, lighting, and level of occlusion of face images should be considered in the design of algorithms. O'Toole et al. [33, 34, 32] and Philips et al. [39, 38] have designed controlled datasets of natural images to compare human face recognition performance against algorithms. With the focus on algorithmic consistency with human behavior, there is no explicit model vs. model comparison in the above methods.

More control in experimentation can be achieved through the use of rendered 3D scenes. Cadiue et al. [8], Yamins et al. [60, 61] and Hong et al. [18] all make use of rendered images with parametrized variation to compare the representations of models with those found in the primate brain. Pramod and Arun [40] describe a set of perceived dissimilarity measurements from humans that is used to study the systematic differences between human perception and a large number of handcrafted and learned feature representations. Because of a need for very fine-grained control of object parts and other latent parameters of scenes, procedural graphics were introduced by Tenenbaum et al. [53] for the study of one-shot learning using probabilistic generative models. The use of procedural graphics for generative models was further developed by Yildirim et al. [62], Kulkarni et al. [26], and Wu et al. [59]. These prior studies do not vary the conditions of the stimuli using the procedures of psychophysics, nor do they use large-scale renderings on the order of millions of scenes.

Other Manipulations of Stimuli in Visual Recognition Evaluations. Work coming directly out of computer vision also addresses stimulus generation for the purpose of isolating model weaknesses. Hoiem et al. [17] suggest systematically varying occlusion, size, aspect ratio, visibility of parts, viewpoint, localization error, and background to identify errors in object detectors. Wilber et al. [58] systematically apply noise, blur, occlusion, compression, textures and warping effects over 2D scenes to assess face detection performance. Finally, a whole host of approaches can be found to manipulate the inputs to CNNs in order to highlight unexpected classification errors. These include the noise patterns introduced by Szegedy et al. [52] that are imperceptible to humans, the hill climbing strategy of Tsai and Cox [55], and the fooling images produced via evolutionary algorithms that were explored by Nguyen et al. [30] and Bendale and Boult [2]. The level of control in the evaluation procedures varies between these approaches, but a common starting point based on model preference for each class is missing (*i.e.*, which object configuration produces the highest score?). We suggest in this paper that the use of computer graphics helps us address this.

3. Method: The PsyPhy Framework

Overview. Our procedure for performing psychophysics on a model largely follows from established procedures found in psychology, with a few key adaptations to accommodate artificial perception. For the purposes of this paper, our focus is on a performance-based forced-choice task that yields an interpretable item-response curve. Specifically, we choose to focus on the theoretical mapping of the *two-alternative forced choice (2AFC) match-to-sample* procedure to the problem of object classification. For descriptions of other procedures in psychophysics, see the reviews by Prins [41] and Lu and Dosher [28].

In the 2AFC match-to-sample procedure, an observer is shown a "sample" stimulus, followed by two "alternate" stimuli where one is a positive (*i.e.*, matching) stimulus and the other is a negative (*i.e.*, non-matching) stimulus. The observer is then asked to choose from the alternate stimuli the stimulus that best matches the sample — the match criterion may or may not be provided to the observer. The

observer repeats the task at different perturbed stimulus levels in either an adaptive pattern, which is like gradient descent for humans, or via the method of constants, which is a predetermined set of perturbed stimulus levels. Regardless of method, each task has two presented alternate stimuli ($N = 2$) and thus two-alternative forced-choices ($M = 2$). To complete the experiment, the same procedure is repeated for multiple observers. Analysis of the experiment would utilize the mean or median accuracy humans achieved at each stimulus level and mean or median human response time, if recorded. To evaluate visual recognition models instead of people, a mapping of the classification procedure to a more general version of the 2AFC match-to-sample procedure is required. We call this mapped classification *MAFC match-to-sample*.

In MAFC match-to-sample, the probe image in classification is equivalent to the sample stimulus and a deterministic classification model replaces the human observer. In classification, we rarely have only two classes for a model to choose from. Thus the value of M becomes the number of labeled training classes (*e.g.*, ImageNet 2012 has 1K learned classes, making $M = 1$K). Likewise, N, which is the number of presented alternate stimuli, changes to the number of images used to train the model, because this is the set of images the model is implicitly matching to (*e.g.*, for ImageNet 2012, $N = \sim$1.2M training images).

In traditional visual psychophysics, an observer's response to stimuli can be non-deterministic (due to fatigue, stress, or other factors), thus many trials with the same stimuli over different subjects are required to gain a statistical approximation of general human performance. However, when applying psychophysics to a model, we need not have a model repeat a task multiple times because of determinism. Instead, we choose to present many images (*e.g.*, instances of different classes) to the model at each perturbed stimulus level, allowing us to create an approximation of how accurate a model is at that level. We define the number of trials as the number of images presented at each stimulus level.

The last major difference is in the selection of the stimuli's default state, that is, where there is no perturbation. Blanz et al. [4] show that humans have a natural inclination towards a certain view of an object, called a canonical view. We assume in human trials that an object configuration close to a canonical view will be chosen, maximizing the probability that all observers will have no problems performing at least some part of the 2AFC match-to-sample task. However, this procedure is not as simple in image classification because we do not necessarily know if a model follows a similar canonical view. However, we can say that a model's *preferred view* is the view that produces the strongest positive response, as determined by a classification score. Choosing a preferred view is crucial to guaranteeing that when the stimulus is perturbed, the model's response will already be at its maximum for that class. Any perturbation will cause a decline (or possibly no change) in the strength of the model's response, not an increase.

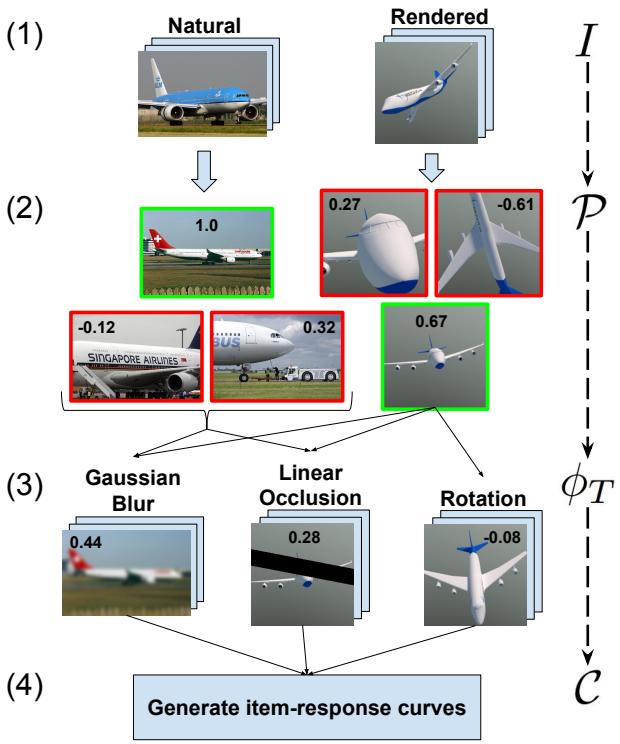

Figure 2. Major steps of the PsyPhy framework: (1) initial natural images and rendered models are used as input; (2) a model's preferred view is determined; (3) perturbations of the preferred view are generated and classified; (4) an item-response curve is plotted.

PsyPhy MAFC Framework. Inspired by software frameworks for subject testing like PsychoPy [35], we have implemented the MAFC procedure described above as a python framework for model testing called PsyPhy. Here we describe the details of each component of the framework (shown in Fig. 2). The first step is to select the initial set of stimuli for each class. For 2D natural images, this is any set of chosen images I_{2D} for a class c. For a rendered scene, a set of image specifications I_{3D} is provided to a rendering function $R(c, v)$ (implemented in this work using Mitsuba [22]) to render a single object centered in an image. The view $v \in I_{3D}$ is the parameter set $\{x, y, z, \psi\}$, where the coordinates x, y, and z are real numbers in the range $(-180.0, 180.0]$ and ψ, representing scale, is a real number in the range $(0.0, 25.0]$.

The second step is to find the preferred view for each class. For natural 2D images, the preferred view function in Eq. 1 is used. The second preferred view function, Eq. 2, uses R to create rendered images for classification. In Eq. 2, the search space is almost infinite, thus it does not find the

Algorithm 1 $\phi_T(f, V, s)$: an item-response point generation function for any image transformation function $T(s, v)$

Input: f, an input model
Input: V, a vector of preferred views
Input: s, the stimulus level
1: $\beta = \sum_{v \in V} \max(0, \lceil D(f, T(s, v), c(v)) \rceil)$
2: $a = \frac{\beta}{|V|}$
3: **return** $\{s, a\}$, an x, y coordinate pair (stimulus level, avg. accuracy over trials) for one item-response point

absolute global maximum, but rather an approximation.

$$\mathcal{P}_{2D}(I_{2D}, c) := \underset{i \in I_{2D}}{\operatorname{argmax}} D(f, i, c) \tag{1}$$

$$\mathcal{P}_{3D}(I_{3D}, c) := \underset{v \in I_{3D}}{\operatorname{argmax}} D(f, R(c, v), c) \tag{2}$$

A decision function $D(f, i, c)$ normalizes the score output of a model f to a value in the range $[-1.0, 1.0]$, which gives both a decision and a confidence associated with that decision. A value in the range $[-1.0, 0]$ is an incorrect decision and $(0, 1.0]$ is a correct decision. The parameter i is the input stimulus and c is the expected class.

A natural 2D preferred view (Eq. 1) is a single selected image $i \in I_{2D}$, where D has the strongest positive response — this is also known as the top-1 classification. A 3D preferred view (Eq. 2) is a single selected set $v = \{x_v, y_v, z_v, \psi_v\} \in I_{3D}$, where D has the strongest positive response. The major difference between Eq. 1 and Eq. 2 is the use of R in Eq. 2 to render the image prior to measuring the response from D. Invoking Eq. 1 or Eq. 2 for each class will create a vector of preferred views V.

After preferred views have been selected for all classes, whether natural or rendered, the next step is to apply perturbations to them. In this procedure, a set of preferred views is perturbed at a specific stimulus level (*i.e.*, the amount of perturbation) using a function $T(s, v)$, where T could be any image transformation function (*e.g.*, Gaussian blur, rotation). The parameter v is one preferred view — either in 2D image format or $\{x, y, z, \psi\}$ for rendered stimuli — and s is the stimulus level. The function $\phi_T(f, V, s)$ described in Alg. 1 perturbs the set of preferred views given in V and then makes a decision on each image using D. Each individual image evaluation is a trial. The value returned by ϕ_T represents one point on an item-response curve, which is an average of all trials (one trial per class).

An item-response curve is the set of x, y coordinates that represent the model behavior for a set of stimuli. Each x value is an image perturbation level, and each y value is the accuracy of the model for that image. Note that traditional psychophysics experiments with live test subjects often apply a psychometric function to interpolate between the points to generate the curve. We eschew the application

Algorithm 2 $\mathcal{C}_T(f, V, n, b_l, b_u)$: an item-response curve generation function

Input: f, an input model
Input: V, a vector of preferred views
Input: n, the number of stimulus levels
Input: b_l and b_u, the lower and upper bound values of the stimulus levels
1: **Let** S be n log-spaced stimulus levels from b_l to b_u
2: $I = \bigcup_{s \in S} \{\phi_T(f, V, s)\}$
3: **return** I, the item-response curve

of a psychometric function because we are able to perform millions of evaluations with a model in a very short window of time, compared to human experiments that typically do only a few hundred. Thus a more precise understanding of how models are performing at each stimulus level can be achieved using only the raw data.

The final stage of the framework generates the item-response curves using the function $\mathcal{C}_T(f, V, n, b_l, b_u)$. The procedure is simple, and only requires a repeated execution of ϕ_T for each stimulus level. Its steps are show in Alg. 2. The procedure will create a set of stimulus levels starting with a lower bound, b_l, and ending with an upper bound b_u. b_l is the closest stimulus level to the preferred view and b_u is the stimulus level that is farthest away. The parameter n is the number of stimulus levels to use. Typically in visual pyschophysics, log-spaced stepping is used to have a finer-grained evaluation near the canonical view; we use the same strategy for our preferred view.

4. Experiments

The first goal of our experiments was to demonstrate PsyPhy as a large-scale psychophysics evaluation framework. To do this, we processed millions of procedurally rendered 3D scenes and 2D images that were perturbed. The second goal was to demonstrate the utility of procedural graphics for large-scale psychophysics experiments. Thus we broke our experiments up into two sets: natural scene experiments and rendered scene experiments. Our final goal was to evaluate the strengths and weaknesses of well-known CNN models. All code and data for these experiments will be released following publication.

In all of our experiments, we chose to use five convolutional neural network models that were pre-trained on ImageNet 2012 [21]: AlexNet [25], CaffeNet [23], GoogleNet [51], VGG-16, and VGG-19 [49]. Each network's final layer is the Softmax function producing a confidence score from 0 to 1 — no confidence to perfect confidence — for each of ImageNet's 1K classes. Each network is wrapped in a decision function, defined by Alg. 3, for compatibility with the PsyPhy MAFC framework described

Algorithm 3 $D^*(f, i, c)$, the top-1 binary decision of the Softmax layer of one of five CNNs [21, 25, 23, 51, 49]

Input: f, one of the five network models
Input: i, an input image
Input: c, the expected class

$V = f(i)$ ▷ the Softmax vector
$d(j) := V_j$ ▷ return class response
$c^* = \text{argmax}_{j \in [0, |V|)} \, d(j)$ ▷ find class label
$s = d(c^*)$
if $c \neq c^*$ **then** ▷ incorrect class, negate response
    $s = -1 * s$
end if
return s, the decision score

in Sec. 3.

We use the following transformations for our perturbation parameters: Gaussian blur, salt & pepper noise, sharpness, contrast, brightness, rotation, scale, and linear occlusion. To perturb using Gaussian blur, we use OpenCV's `GaussianBlur` module [6], setting the blur kernel σ in the range $[0, 15]$. Salt & pepper noise utilizes the `util` module from scikit-image [56], selecting a random configuration of noise as a percentage of the total image pixels. Sharpness, contrast, and brightness are controlled using the `ImageEnhance` module of PIL [64]. The `Sharpness` is chosen within the range $[0.0, 2.0]$. For both the `Contrast` and `Brightness` functions, the controlling parameter is set between $[0.0, 15]$.

For the 3D transformations, rotation and scale, we use the standard transformation matrix from computer graphics [1]. Rotation is in the range $[-180, 180]$ and scale $(0, 25]$. The last transformation, linear occlusion, has two parameters, the percentage of occlusion and the bounding box of the object being occluded. If no bounding box is given, it defaults to the image size. In our natural image experiments, the bounding box is always assigned the default value, but in our rendered scene experiments, a specified bounding box is used. To generate an occluding bar, a point within the bounding box is randomly selected, as well as a random slope. The width of the bar is then extended until the chosen percentage of the bounding box is occluded.

4.1. 2D Natural Scene Experiments

Here we perturb images from the ImageNet 2012 [21] training dataset, which consists of ∼1.2M million images and 1K classes. We chose this set instead of the validation set or another dataset entirely because we know that each of the networks has seen all of these images during training. Ideally, the networks should have a good representation — and perfect confidence score — for each of the training images. But in practice, not all of the training images have a good representation in each network. Thus we select the top-1 image from each class independently as a preferred view. There are 1K preferred views for these experiments (one for each class) per network.

After the preferred views are selected, each image is put through a series of perturbations. The following transformations are applied: Gaussian blur, brightness, contrast, salt & pepper noise, linear occlusion, and sharpness. For each of these conditions, we created 200 perturbed images starting with the preferred view and log-spaced stepping towards increasing difficulty. The result is 201 images per class per network, or 201K images per network, or ∼1M images per condition. In total, ∼9M images were evaluated. Each image was then classified by the associated network and decision function. All classes were aggregated together on a per network and per stimuli value basis to create item-response curves for each condition.

Results. Item response curves are shown in the top row of Fig. 3 and Supplemental[1] Fig. 1. Per-class breakdowns for each condition can be found in Supplemental Spreadsheet 1. With even just small perturbations to the original training images, the models begin to fail to correctly classify the objects in all cases. Surprisingly, these are images that we expect the models to always do well on, given that their original forms were used at training time. This brings into question the level of learned invariance. From these results, we can conclude that the training regime of each model does not extrapolate far beyond the variance of the training data at hand. Moreover, notice from the images on the x-axis of each plot that the models fail at perturbation levels humans would not have trouble with.

Looking at specific model performance, the two VGG networks are very similar. The same effect can be seen with AlexNet and CaffeNet. For the VGG networks, this suggests that additional layers — beyond a certain point — do not imply better performance under degrading conditions. Likewise, switching the order of the pooling and normalization layers in CaffeNet and AlexNet [65] does not imply better performance under degrading conditions. However, there is a difference between CaffeNet/AlexNet and the VGG networks, which means some aspect of the architecture is making VGG better than CaffeNet/AlexNet. Although training data is the largest contributor to good performance overall, it is not responsible for the difference in this case, since both sets of networks were trained with the same data. In this instance, psychophysics helped us tease out the differences across networks in a way summary statistics could not have.

4.2. 3D Rendered Scene Experiments

In the experiment with rendered images, we initially selected 50 3D objects from the Blend Swap [63] library that

[1] Supplemental material is accessible at http://bjrichardwebster.com/papers/psyphy/supp.

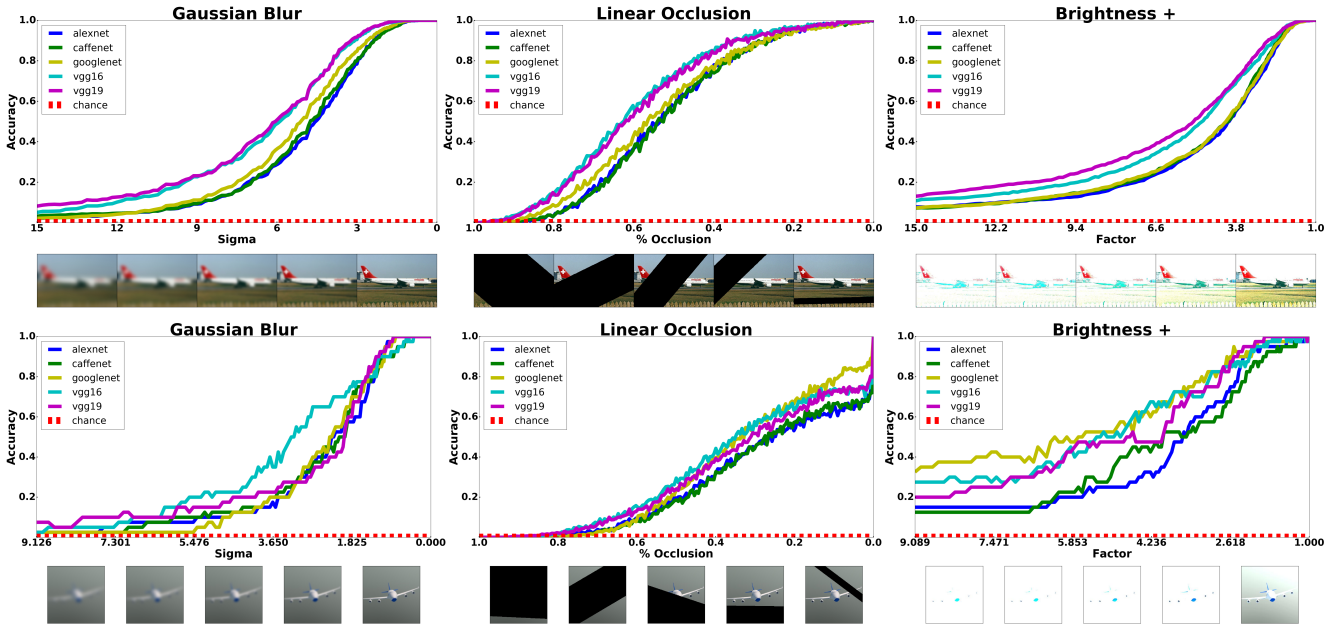

Figure 3. Three pairs of item-response curves comparing natural scenes (Top Row) to rendered scenes (Bottom Row). The natural scene experiments reflect the classification accuracy across all 1K ImageNet classes and the rendered scene experiments reflect the accuracy across 40 classes. Each experiment used five well-known CNNs [25, 23, 51, 49]. The images at the bottom of each curve show how the perturbations increase from right to left, starting with no perturbation (*i.e.*, the original image) for all conditions. A perfect curve would be a flat line at the top of the plot. Note in almost all cases humans would not start to fail under the same conditions the models are failing at. These plots (as well as the next set in Fig. 4) are best viewed in color.

corresponded to classes in ImageNet (see supp. material for a list of the classes). For each of the 3D objects, we randomly rendered 100K uniformly distributed x, y, z rotations and scales, resulting in 5M images. This rendered dataset is the basis used to find each network's set of preferred views.

Finding the preferred views using the rendered image set is nearly identical to finding it in the natural images experiment, except for the following differences. The rendered dataset is unseen to the networks, therefore there is no guarantee that there will be an image that is classified correctly and hence no preferred view. Further, we have an associated parameter set for each preferred view, which allows us to synthesize 3D transformations directly for the item-response curves. 40 3D objects yielded a preferred view across all networks, and were retained for the experiments.

After each preferred view is selected, one of the following transformations are applied by our graphics engine: rotations in the x, y, z dimensions, and scale. All are applied in the positive and negative direction. In addition, all of the transformations from the 2D natural image experiment are repeated using the rendered preferred views. For each of the 3D transformations, we rendered 200 images starting with the preferred view and log-spaced stepping towards increasing difficulty. The result is 201 images per class per network, or ∼8K images per network, or ∼40K images per transformation. The additional 2D transformations resulted in a total of ∼360K images, which brings the rendered image total to ∼683K evaluated images. All classes are aggregated together on a per network per stimuli value basis to create item-response curves for each condition.

For both finding preferred views and perturbations for the 3D scenes, the perspective projection is located 45° above the world origin and 25 standard units from the world origin. The light source is located in the Mitsuba [22] renderer's default location "sky" lighting. Finally, if a rendered object did not come with a texture, we mapped the average color of the Blendswap object icon to the surface of the object. All rendered objects used diffuse texture shading.

Results. The results are shown in the bottom row of Fig. 3, Fig. 4, and Supplemental Fig. 2. Per-class breakdowns for each condition can be found in Supplemental Spreadsheet 2. In Fig. 4 we see six different item-response curves demonstrating x, y, z rotation, scale, contrast, and salt & pepper noise. Our first observation is that in all rotations, all five networks dropped below 50% accuracy well before a human would. For this transformation, humans would probably stay near perfect on average, depending on the amount of self occlusion, thanks to the generalization properties of the visual cortex [53]. Note, however, that no network fell below chance on any rotation. In addition, networks are somewhat consistent within a rotation dimension $(+/-)$, but demonstrate different behavior across rotation

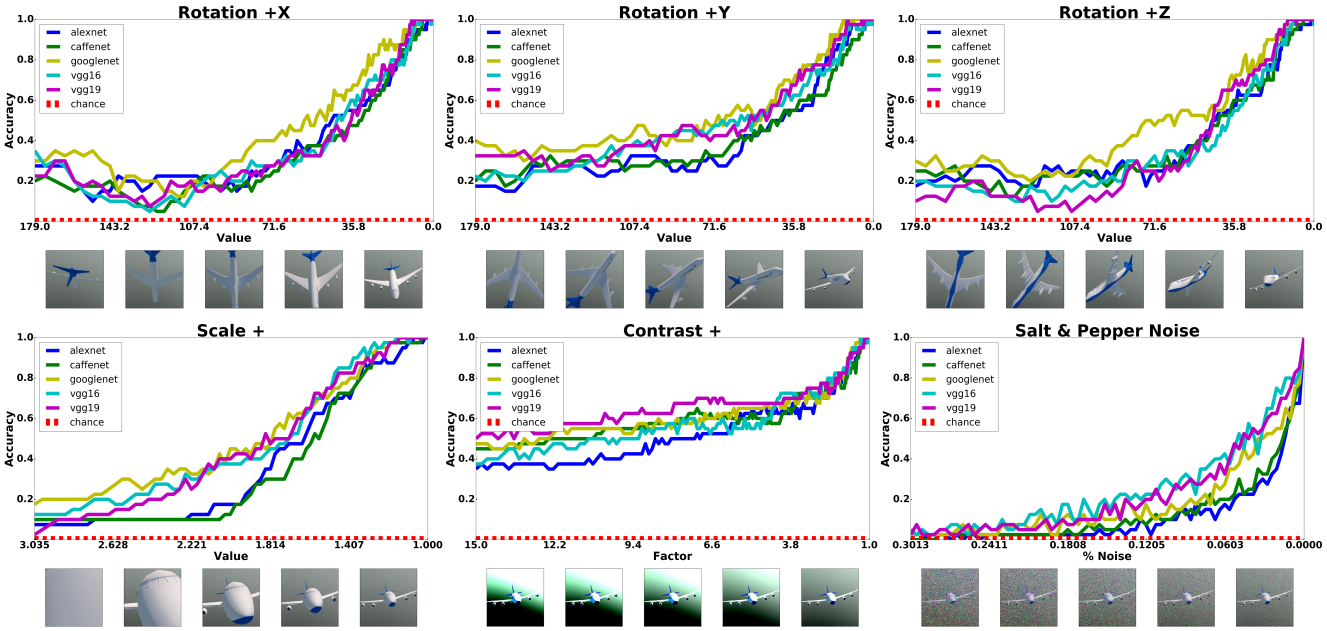

Figure 4. Six item-response curves demonstrating 2D and 3D transformations applied to rendered scenes. Each point on each curve is the classification accuracy of 40 classes. Note in all cases humans would not start to fail under the same conditions the models are failing at.

dimensions (x, y, z).

The other 3D metric, scale, has a different result than the rotations. When increasing scale, networks do well — in fact, they are arguably doing better than humans. In Fig. 4 Scale +, four of the networks are significantly above chance at all points, whereas humans probably would not do well after a certain point on the x-axis. Notice the far left image where only texture is displayed. Networks are likely learning from texture information and are able to make above chance predictions, but humans have a harder time without corresponding structural information for tasks like this.

In the 2D conditions for rendered scenes, we hypothesized that the network responses to the rendered scenes (unseen at training time) would follow a similar pattern to that of the natural scenes (seen at training time). The overall pattern of five networks did follow this trend. However, the behavior of the individual networks did not always remain the same across the experiments. Notice that unlike in the 2D natural image experiments, the VGG networks did not, in all cases, maintain similar curves, likewise of AlexNet and CaffeNet. In addition, we expected that for the rendered scenes, the network accuracies would decline much sooner than the natural scenes because they are from outside the dataset. This is confirmed to be the case; see Fig. 3.

5. Discussion

In visual psychophysics, we have a convenient and practical alternative to traditional dataset evaluation. However, the use of psychophysics testing and datasets are not mutually exclusive. How can both be combined to fuel further progress in the field? One needs datasets to form a training basis for any data-driven model. Moreover, there is major utility to having a large amount of such data — this is essential for making machine learning capture enough intraclass variance to generalize well to unseen class instances. Data augmentation [25, 9] is an obvious strategy for leveraging the rendered images that were problematic for a model during psychophysics testing to expand the scope of the training set. However, this has diminishing returns as datasets grow to sizes that exceed available memory (or even disk space) during training. Alternative formulations that use more limited training data and reinforcement learning that optimizes over item-response curves to correct for recognition errors are likely a better path forward.

Recent research has shown that CNNs are able to predict neural responses in the visual cortex of primates [60, 61]. This, coupled with excellent benchmark dataset results across multiple recognition domains, suggests that good progress is being made towards reaching human-like performance. As a strong counterpoint, our psychophysics experiments show that the most popular CNN models fail to correctly classify images that humans would never make mistakes on. What is missing from the models that is causing this behavioral discrepancy? While certainly capturing some of the operation of the ventral stream (particularly at the filtering stages of early vision), today's CNN architectures lack the intricate structure and function of biological neural networks [10]. Advances in connectomics [24], a subfield of neuroscience that is attempting to map circuits in the brain, and functional imaging [31], which allows for

the *in vivo* study of neuronal responses, are likely to inform new biologically-inspired architectures in the near future. With psychophysics as a guide to performance, candidate models can more easily be accepted or dismissed as being representative of some form of AI — making it harder for us to be fooled by the person inside of the Chinese room.

6. Acknowledgements

The authors would like to thank Lucas Parzianello for helping import Blendswap models into PsyPhy. Funding was provided under IARPA contract #D16PC00002 and NSF DGE-1313583. Hardware support was generously provided by the NVIDIA Corporation.

References

[1] E. Angel. *Interactive Computer Graphics: A Top-Down Approach with Shader-Based OpenGL (6th Edition)*. Pearson, 2011. 6

[2] A. Bendale and T. E. Boult. Towards open set deep networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 3

[3] J. R. Beveridge, P. J. Phillips, D. S. Bolme, B. A. Draper, G. H. Givens, Y. M. Lui, M. N. Teli, H. Zhang, W. T. Scruggs, K. W. Bowyer, P. Flynn, and S. Cheng. The challenge of face recognition from digital point-and-shoot cameras. In *IEEE International Conference on Biometrics: Theory, Applications and Systems (BTAS)*, 2013. 2

[4] V. Blanz, M. J. Tarr, and H. H. Bülthoff. What object attributes determine canonical views? *Perception*, 28(5):575–599, 1999. 4

[5] J. K. Bowmaker and H. J. Dartnall. Visual pigments of rods and cones in a human retina. *The Journal of Physiology*, 298(1):501–511, 1980. 2

[6] G. Bradski. Opencv. *Dr. Dobb's Journal of Software Tools*, 2000. 6

[7] Z. Bylinskii, T. Judd, A. Borji, L. Itti, F. Durand, A. Oliva, and A. Torralba. MIT saliency benchmark. 2

[8] C. F. Cadieu, H. Hong, D. Yamins, N. Pinto, N. J. Majaj, and J. J. DiCarlo. The neural representation benchmark and its evaluation on brain and machine. In *International Conference on Learning Representations (ICLR)*, 2013. 3

[9] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. Return of the devil in the details: Delving deep into convolutional nets. In *British Machine Vision Conference (BMVC*, 2014. 8

[10] D. D. Cox and T. Dean. Neural networks and neuroscience-inspired computer vision. *Current Biology*, 24(18):R921–R929, 2014. 8

[11] B. Duchaine and K. Nakayama. The Cambridge face memory test: Results for neurologically intact individuals and an investigation of its validity using inverted face stimuli and prosopagnosic participants. *Neuropsychologia*, 44(4):576–585, 2006. 2

[12] S. Eberhardt, J. Cader, and T. Serre. How deep is the feature analysis underlying rapid visual categorization? In *Advances in Neural Information Processing Systems (NIPS)*, 2016. 3

[13] S. E. Embretson and S. P. Reise. *Item response theory for psychologists*. Lawrence Erlbaum Associates, Inc., 2000. 2

[14] H. E. Gerhard, F. A. Wichmann, and M. Bethge. How sensitive is the human visual system to the local statistics of natural images? *PLoS Computational Biology*, 9(1):e1002873, 2013. 3

[15] L. Germine, K. Nakayama, B. C. Duchaine, C. F. Chabris, G. Chatterjee, and J. B. Wilmer. Is the web as good as the lab? comparable performance from web and lab in cognitive/perceptual experiments. *Psychonomic Bulletin & Review*, 19(5):847–857, 2012. 3

[16] S. Hecht, S. Shlaer, and M. H. Pirenne. Energy, quanta, and vision. *The Journal of General Physiology*, 25(6):819–840, 1942. 2

[17] D. Hoiem, Y. Chodpathumwan, and Q. Dai. Diagnosing error in object detectors. In *European Conference on Computer Vision (ECCV)*, 2012. 3

[18] H. Hong, D. Yamins, N. J. Majaj, and J. J. DiCarlo. Explicit information for category-orthogonal object properties increases along the ventral stream. *Nature Neuroscience*, 19(4):613–622, 2016. 3

[19] G. B. Huang and E. Learned-Miller. Labeled faces in the wild: Updates and new reporting procedures. Technical Report UM-CS-2014-003, University of Massachusetts, Amherst, May 2014. 2

[20] ImagetNet Large Scale Visual Recognition Challenge 2011 (ILSVRC2011). http://image-net.org/challenges/LSVRC/2011/index, Accessed: 2016-10-12. 2

[21] ImagetNet Large Scale Visual Recognition Challenge 2012 (ILSVRC2012). http://image-net.org/challenges/LSVRC/2012/index, Accessed: 2016-10-12. 2, 5, 6

[22] W. Jakob. Mitsuba renderer https://pillow.readthedocs.io/en/3.0.0/index.html, Accessed: 2016-11-05. 4, 7

[23] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014. 3, 5, 6, 7

[24] N. Kasthuri, K. J. Hayworth, D. R. Berger, R. L. Schalek, J. A. Conchello, S. Knowles-Barley, D. Lee, A. Vázquez-Reina, V. Kaynig, T. R. Jones, M. Roberts, J. L. Morgan, J. C. Tapia, H. S. Seung, W. G. Roncal, J. T. Vogelstein, R. Burns, D. L. Sussman, C. E. Priebe, H. Pfister, and J. W. Lichtman. Saturated reconstruction of a volume of neocortex. *Cell*, 162(3):648–661, 2015. 8

[25] A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2012. 2, 3, 5, 6, 7, 8

[26] T. D. Kulkarni, P. Kohli, J. B. Tenenbaum, and V. Mansinghka. Picture: A probabilistic programming language for scene perception. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 2, 3

[27] S. Lohr. A lesson of Tesla crashes? computer vision can't do it all yet. The New York Times,

September 2016. Accessed 2016-10-12 via http://www.nytimes.com/2016/09/20/science/computer-vision-tesla-driverless-cars.html. 2

[28] Z.-L. Lu and B. Dosher. *Visual Psychophysics: From Laboratory to Theory*. MIT Press, 2013. 1, 2, 3

[29] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *IEEE International Conference on Computer Vision (ICCV)*, July 2001. 2

[30] A. Nguyen, J. Yosinski, and J. Clune. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015. 3

[31] K. Ohki, S. Chung, Y. H. Ch'ng, P. Kara, and R. C. Reid. Functional imaging with cellular resolution reveals precise micro-architecture in visual cortex. *Nature*, 433(7026):597–603, 2005. 8

[32] A. J. O'Toole, X. An, J. Dunlop, V. Natu, and P. J. Phillips. Comparing face recognition algorithms to humans on challenging tasks. *ACM Transactions on Applied Perception (TAP)*, 9(4):16, 2012. 3

[33] A. J. O'Toole, P. J. Phillips, F. Jiang, J. Ayyad, N. Penard, and H. Abdi. Face recognition algorithms surpass humans matching faces over changes in illumination. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(9):1642–1646, 2007. 3

[34] A. J. O'Toole, P. J. Phillips, and A. Narvekar. Humans versus algorithms: comparisons from the face recognition vendor test 2006. In *IEEE International Conference on Automatic Face & Gesture Recognition (FG)*, 2008. 3

[35] J. W. Peirce. PsychoPy: Psychophysics software in python. *Journal of Neuroscience Methods*, 162(12):8 – 13, 2007. 4

[36] P. J. Phillips, P. J. Flynn, J. R. Beveridge, W. T. Scruggs, A. J. Otoole, D. Bolme, K. W. Bowyer, B. A. Draper, G. H. Givens, Y. M. Lui, H. Sahibzada, J. A. Scallan, III, and S. Weimer. Overview of the multiple biometrics grand challenge. In *International Conference on Biometrics (ICB)*, 2009. 2

[37] P. J. Phillips, P. J. Flynn, T. Scruggs, K. W. Bowyer, J. Chang, K. Hoffman, J. Marques, J. Min, and W. Worek. Overview of the face recognition grand challenge. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005. 2

[38] P. J. Phillips, M. Q. Hill, J. A. Swindle, and A. J. O'Toole. Human and algorithm performance on the PaSC face recognition challenge. In *IEEE International Conference on Biometrics Theory, Applications and Systems (BTAS)*, 2015. 3

[39] P. J. Phillips and A. J. O'Toole. Comparison of human and computer performance across face recognition experiments. *Image and Vision Computing*, 32(1):74–85, 2014. 3

[40] R. T. Pramod and S. P. Arun. Do computational models differ systematically from human object perception? In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 3

[41] N. Prins. *Psychophysics: a Practical Introduction*. Academic Press, 2016. 1, 2, 3

[42] M. Riesenhuber and T. Poggio. Hierarchical models of object recognition in cortex. *Nature Neuroscience*, 2(11):1019–1025, 1999. 3

[43] M. Riesenhuber and T. Poggio. The individual is nothing, the class everything: Psychophysics and modeling of recognition in object classes. Technical Report AIM-1682, Massachusetts Institute of Technology, October 2000. 3

[44] M. D. Rodriguez, J. Ahmed, and M. Shah. Action mach: A spatio-temporal maximum average correlation height filter for action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010. 2

[45] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. ImageNet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015. 2

[46] B. Sapp and B. Taskar. Modec: Multimodal decomposable models for human pose estimation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013. 2

[47] W. J. Scheirer, S. E. Anthony, K. Nakayama, and D. D. Cox. Perceptual annotation: Measuring human vision to improve computer vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI)*, 36, August 2014. 3

[48] J. R. Searle. Minds, brains, and programs. *Behavioral and Brain Sciences*, 3(03):417–424, 1980. 1

[49] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014. 3, 5, 6, 7

[50] P. Sinha, B. Balas, Y. Ostrovsky, and R. Russell. Face recognition by humans: Nineteen results all computer vision researchers should know about. *Proceedings of the IEEE*, 94(11):1948–1962, 2006. 3

[51] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 3, 5, 6, 7

[52] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. Intriguing properties of neural networks. In *International Conference on Learning Representations (ICLR)*, 2014. 3

[53] J. B. Tenenbaum, C. Kemp, T. L. Griffiths, and N. D. Goodman. How to grow a mind: Statistics, structure, and abstraction. *Science*, 331(6022):1279–1285, 2011. 2, 3, 7

[54] A. Torralba and A. A. Efros. Unbiased look at dataset bias. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011. 2

[55] C.-Y. Tsai and D. Cox. Are deep learning algorithms easily hackable? http://coxlab.github.io/ostrichinator, Accessed: 2016-10-12. 3

[56] S. van der Walt, J. L. Schönberger, J. Nunez-Iglesias, F. Boulogne, J. D. Warner, N. Yager, E. Gouillart, T. Yu, and the scikit-image contributors. scikit-image: image processing in Python. *PeerJ*, 2:e453, 6 2014. 6

[57] C. Vondrick, H. Pirsiavash, A. Oliva, and A. Torralba. Learning visual biases from human imagination. In *Advances in Neural Information Processing Systems (NIPS)*, 2015. 3

[58] M. J. Wilber, V. Shmatikov, and S. Belongie. Can we still avoid automatic face detection? In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2016. 3

[59] J. Wu, C. Zhang, T. Xue, W. T. Freeman, and J. B. Tenenbaum. Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. In *Advances in Neural Information Processing Systems (NIPS)*, December 2016. 2, 3

[60] D. Yamins, H. Hong, C. Cadieu, and J. J. DiCarlo. Hierarchical modular optimization of convolutional networks achieves representations similar to macaque it and human ventral stream. In *Advances in Neural Information Processing Systems (NIPS)*, 2013. 3, 8

[61] D. Yamins, H. Hong, C. F. Cadieu, E. A. Solomon, D. Seibert, and J. J. DiCarlo. Performance-optimized hierarchical models predict neural responses in higher visual cortex. *Proceedings of the National Academy of Sciences*, 111(23):8619–8624, 2014. 3, 8

[62] I. Yildirim, T. D. Kulkarni, W. A. Freiwald, and J. B. Tenenbaum. Efficient and robust analysis-by-synthesis in vision: A computational framework, behavioral tests, and modeling neuronal representations. In *Annual Conference of the Cognitive Science Society (CogSci)*, 2015. 2, 3

[63] Mitsuba renderer http://www.blendswap.com, 2010. 6

[64] Pillow (PIL fork) https://pillow.readthedocs.io/en/3.0.0/index.html, Accessed: 2016-11-05. 6

[65] BVLC caffe https://github.com/BVLC/caffe/tree/master/models/bvlc_reference_caffenet, Accessed: 2016-11-10. 6