Adversarial images for the primate brain

Li Yuan,1,2,6 Will Xiao,1,3,6,* Gabriel Kreiman,4 Francis E.H. Tay,5 Jiashi Feng,2 Margaret S. Livingstone1,*

1Department of Neurobiology, Harvard Medical School, Boston, MA 02115, U.S.A.
2Department of Electrical and Computer Engineering, National University of Singapore, Singapore 117583
3Department of Molecular and Cellular Biology, Harvard University, Cambridge, MA 02134, U.S.A.
4Department of Ophthalmology, Boston Children’s Hospital, Boston, MA 02115, U.S.A.
5Department of Mechanical Engineering, National University of Singapore, Singapore 117583
6These authors contributed equally: Li Yuan, Will Xiao
*To whom correspondence should be addressed;
E-mail: xiaow@fas.harvard.edu; margaret_livingstone@hms.harvard.edu

Abstract

Convolutional neural networks (CNNs) are vulnerable to adversarial attack, the phenomenon that adding minuscule noise to an image can fool CNNs into misclassifying it. Because this noise is nearly imperceptible to human viewers, biological vision is assumed to be robust to adversarial attack. Despite this apparent difference in robustness, CNNs are currently the best models of biological vision, revealing a gap in explaining how the brain responds to adversarial images. Indeed, sensitivity to adversarial attack has not been measured for biological vision under normal conditions, nor have attack methods been specifically designed to affect biological vision. We studied the effects of adversarial attack on primate vision, measuring both monkey neuronal responses and human behavior. Adversarial images were created by modifying images from one category (such as human faces) to look like a target category (such as monkey faces), while limiting pixel value change. We tested three attack directions via several attack methods, including directly using CNN adversarial images and using a CNN-based predictive model to guide monkey visual neuron responses. We considered a wide range of image change magnitudes that covered attack success rates up to > 90%. We found that adversarial images designed for CNNs were ineffective in attacking primate vision. Even when considering the best attack method, primate vision was more robust to adversarial attack than an ensemble of CNNs, requiring over 100-fold larger image change to attack successfully. The success of individual attack methods and images was correlated between monkey neurons and human behavior, but was less correlated between either and CNN categorization. Consistently, CNN-based models of neurons, when trained on natural images, did not generalize to explain neuronal responses to adversarial images. These results reveal limits of CNN-based models of primate vision through their differential response to adversarial attack, and provide clues for building better models of the brain and more robust computer vision algorithms.
Introduction

Artificial neural networks can now match, and sometimes exceed, human performance in diverse tasks ranging from visual categorization to game play. Together with this rapid development came the surprising finding that a family of ANNs loosely inspired by biological visual systems—convolutional neural networks (CNNs)—now make the best models of visual cortical neuron responses. However, an equally surprising finding is that CNNs are susceptible to adversarial attacks that do not appear to fool biological visual systems. That is, adding carefully crafted, minute noise to an image can cause CNNs to misclassify it with high confidence. Often, such adversarial noise is almost imperceptible to a human observer, suggesting that humans may be immune to adversarial images. Do adversarial images expose a fundamental difference between human vision and CNN inference? Or, more counterintuitively, could adversarial images be a clue to a previously unknown class of visual illusions? Whichever is the case, adversarial images present an opportunity for better understanding how CNNs resemble and differ from biological vision.

Here, we ask whether primate vision can be fooled by adversarial images related to those designed for CNNs. Adversarial noise tailored for CNNs is often imperceptible to humans, so adversarial images have sometimes been thought of as relative to ground truth labels assigned by humans. By this interpretation, adversarial images cannot exist for humans. However, the original definition of adversarial images was construed as a measure of sensitivity and was based solely on the amount of change to an input required to change its corresponding output. More recently, an adversarial example has been defined as ‘an input to a machine learning model that is intentionally designed by an attacker to fool the model into producing an incorrect output,’ although there is no fundamental reason why this definition should be restricted to machine learning models. When not so restricted, both definitions admit of the possibility of
adversarially attacking human vision. We follow Szegedy et al.\cite{szegedy2013intriguing} and define adversarial images relative to unambiguously classified clean images. It is important to note that, by this definition, any target class image is a trivial adversarial example of any clean image, so adversarial images always exist. However, such trivial examples usually (but not always\cite{szegedy2013intriguing}) entail large image changes. Thus, the interesting question is how little image change is required to produce successful adversarial attack.

There are indications that primate vision can be somewhat sensitive to adversarial images. Elsayed et al.\cite{elsayed2017adversarial} show that humans doing visual categorization under tight time constraints can be biased by adversarial images. Zhou et al.\cite{zhou2017dual} show that humans can decipher the attack target in adversarial images crafted for CNNs (without necessarily making mistakes if asked to categorize the images). Berardino et al.\cite{berardino2018adversarial} identify ‘eigen-distortions,’ directions of small pixel value change that most readily make images look different, although this study did not try to change the categorization of images. None of these studies specifically designed adversarial images to alter normal visual perception or measured neuronal responses to the altered images.

We studied adversarial images aimed at affecting primate vision under normal viewing conditions with relaxed time constraints. We focused on the well-characterized face-processing system in primate inferior temporal (IT) cortex\cite{pouget2019face} and created adversarial human faces to look like monkey faces, adversarial monkey faces to look like human faces, and adversarial noise images to look like human faces. To establish thresholds for successful attack, we tested a wide range of image change magnitudes, which we refer to as noise levels. Because there are practically infinite numbers of distinct images at any given (large enough) noise level, attack threshold must be a function of the method used to generate a particular attack image at a given noise level. We tested an array of attack methods that include: linear interpolation, which is guaranteed to create trivially successful images with large enough image change; variations of linear interpolation that may reduce the threshold by using affine alignment or CycleGAN\cite{zhu2017unpaired}.
translation and generating attack images for an ensemble of CNNs as in Elsayed et al. Furthermore, we tested a family of adversarial attack methods tailored to primate vision by using a model to predict visual neuron responses. We refer to these methods as *gray box attack* because they utilize partial knowledge about primate vision. Finally, we included two types of image modifications to control for the specificity of targeted attack: an array of untargeted image corruptions used in Geirhos et al., and versions of targeted attack images with noise pattern re-applied in the flipped-upside-down orientation.

We measured attack success rate at the neuronal level in monkeys by recording from face-patch neurons in macaque inferior temporal (IT) cortex using chronic electrode arrays, and at the behavior level in humans by conducting a visual categorization task on Amazon Mechanical Turk (MTurk). We found adversarial images that were miscategorized by face-selective neurons as the target category and that misled humans during visual categorization. Gray box attack and CycleGAN-based interpolation were among the most effective methods in human-to-monkey attack, which was also the most difficult attack direction by the success rates achieved. In the other two attack directions, control interpolation methods were the most effective and achieved up to > 90% success in both monkey neurons and human behavior. Comparing across visual systems, monkey neurons and human behavior were correlated in the success of attack methods and individual images. In contrast, attack success rate in models only weakly correlated with that in primates. In particular, purely CNN-based attack was usually highly effective in models, but almost completely ineffective for humans and monkey neurons. After fitting to neuronal responses, CNN-based models still did not lead to consistently effective attack images. Tellingly, CNN-based models of monkey neuron responses, when fitted on clean image data, did not generalize to predict neuron responses to attack images. The results show that there is still much room for improvement for CNNs as models of primate vision.
Results

Adversarial Human Faces Evoked Monkey Face-like Neuronal Responses

We started by trying to create adversarial human face images that would elicit monkey face-like responses (human→monkey or h2m attack) in face-selective monkey neurons. Because adversarial attack for primate vision is relatively unexplored, we tested a wide range of attack noise strength (noise level) and a variety of attack methods.

To choose an attack noise range that was likely to progress from unsuccessful to successful attack, we reasoned that adversarial attack is trivially successful if an original-class image is replaced by an image from the opposite class. Thus, we computed the distance distribution between unmodified human and monkey faces, considering 250 images in each category. Image distance was quantified by mean-squared error (MSE) using vectorized pixel values (range: 0–255). At constant image resolution, MSE is equivalent to the square of the $l_2$-norm (i.e., Euclidean distance), a common metric in the adversarial attack literature. We elected to use MSE instead of the $l_2$-norm because the latter depends on the number of pixels. The typical human-to-monkey pairwise distance in our image set was $\text{MSE} = 6500 \pm 2300$ (mean ± stdev for all numbers in text unless otherwise noted; Figure S3b). However, for successful attack, it suffices to replace a human face with the closest monkey face. With 250 images in each category, the minimum human-to-monkey distance was $2800 \pm 700$. We thus reasoned that an upper range of $\text{MSE} = 800$ was reasonable to test, as it was safely lower than the budget for simply replacing the image. We selected a lower range of $\text{MSE} = 200$ because this amount of image change was barely perceptible. We tested 10 evenly spaced noise levels covering this range of MSE.

The first attack method we considered, as a reference, was to linearly interpolate between a human face and the closest monkey face. This method is guaranteed to achieve complete success
at a high enough noise level. Second, we included a slight modification by interpolating toward the closest monkey face up to an optimized affine transform. Other transformations, such as reducing contrast, can further reduce the distance to the target image. However, we did not test further transformations because an excessively transformed image will eventually cease to represent the target category.

Third, we included a method based on CycleGAN\cite{CycleGAN}. CycleGAN can learn to translate between two image categories without any paired examples. CycleGAN does not explicitly optimize pixel-level proximity between the original and translated image. Nevertheless, the translated image is closely related to the original image, because the original (as opposed to another instance from the original class) can be approximately recovered from the translated image using a reverse translation. We trained a CycleGAN on several hundred human and monkey faces. Using the trained network, we translated each original human face into the monkey class, then linearly interpolated between the two at defined noise levels.

Fourth, we tested adversarial images created for an ensemble of CNNs. Prior work\cite{adversarial} shows that adversarial images designed for CNNs can bias human perception, but only under severe viewing time limits and backward masking where accuracy on unmodified images is reduced to around 65% (chance is 50%). We reproduced this attack method in our setting without the same viewing time limit. To create attack images between human and monkey faces, we used CNNs pre-trained on ImageNet\cite{ImageNet} and fine-tuned them on the two face categories together with the original 1000 categories. We built an ensemble of 14 fine-tuned CNNs comprising Inception, ResNet\cite{ResNet}, ResNeXt\cite{ResNeXt}, DenseNet\cite{DenseNet}, and SENet\cite{SENet}, and created adversarial images for the model ensemble using iterative gradient descent.

Lastly, we designed an attack method tailored to primate vision by building a model of macaque visual neuron responses as a substitute model for attack (Figure 1a). The model comprised a ResNet-101 from the fine-tuned CNN ensemble above, fitted with a linear map-
ping module that used features extracted from the last convolutional layer to predict neuronal responses. Such CNN-based models are the best current models of primate visual neuron responses\textsuperscript{[12]}. The linear mapping was trained on responses of 22 face patch neurons in one monkey (of two in this study) to around 1,000 pictures of objects\textsuperscript{[13]}. The model could explain around 40\% of the neural response variance on held-out images not used during fitting. We used this neuron-fitted model, which is end-to-end differentiable, to stand for the primate visual system for adversarial attack. To create attack images, we used two variants of objective functions and two variants of optimization algorithms, resulting in 4 total variants. The objective function was either 1) to maximize responses in one model neuron that produced monkey-like features in feature visualization, labelled single-neuron attack; or, 2) to match the model-predicted response pattern to the empirical neuron population response pattern to monkey faces, labelled pattern attack. The objective function was optimized using either 1) iterative gradient descent, or 2) iterative gradient descent coupled with $l_2$-projection ($l_2$-PGD), i.e., projection to a fixed noise level (MSE, equivalent to $l_2$) at each step of gradient descent. We refer to these variants collectively as gray box attack, as they use a limited amount of information about the system (primate vision) being attacked.

Each method was used to attack 40 unmodified or clean human faces, except single-neuron attack, pattern attack, and their $l_2$-PGD versions, which were each used to attack 20 images. Each method produced one attacked version of each clean image at every noise level. Thus, we tested a total of 2,400 targeted human→monkey attack images.

We presented the adversarial images, together with 250 human faces and 250 monkey faces (Figures S2), to two monkeys in a passive fixation task and recorded neuronal responses using chronically implanted multi-electrode arrays. Specifically, we recorded from face patches in inferior temporal cortex (IT), which contain neurons that are strongly selective for faces over non-face objects and that extract face features including species and identity\textsuperscript{[17,18]}. To collect
Figure 1: Overview of adversarial attack. 

a left illustrates monkey neuron-level experiments. Monkeys fixated on a red fixation point while images were presented in random order and neuronal responses were recorded. Images were presented for 100 ms; inter-stimulus interval was 150 ms for monkey 1 and 400 ms for monkey 2, whose responses extended over a longer time scale. a right illustrates behavioral experiments with human subjects. Each image was presented for 1.5 s. Humans were instructed to press a key to indicate the correct option. There was no time limit for a response. Text in the figure was not included in the experiment. b, a substitute model was fit on IT neuron responses and used to generate gray box attack images. The substitute model consisted of a pre-trained ResNet-101 (excluding the final fully-connected layer) and a linear mapping model. Adversarial images were generated by gradient-based optimization of the image to create the desired neuronal response pattern as predicted by the substitute model. 

(C) Example images for human→monkey attack (left), monkey→human attack (center), and brown noise→human face attack (right) are shown for different noise levels and different attack methods.
Figure 2: Results of adversarial attack on face-selective neurons in monkey IT. a–c corresponds to human→monkey attack. a, UMAP visualization of neuron population representation of images, showing clean human faces, monkey faces, and level 10 targeted attack images. Color code follows that in b,c. The neuron pseudopopulation included IT neurons from the face patches of two monkeys. b, Attack success rate as a function of noise level (MSE) for targeted attack methods. Points represent success rate as a fraction of images per noise level. Lines represent logistic regression fit on individual image results. c, Area under the success rate curve (AUC) per attack method. Colored points and lines indicate center values and bootstrap 95% confidence intervals (95%-CI), respectively. Triangle marker indicates method with maximum AUC. Annotations indicate statistical significance: *, p < 0.05; **, p < 0.01; ***, p < 0.001; and n.s., not significant, all p-values FDR-corrected for multiple comparisons. Annotation on each group indicates whether the coefficient in logistic fit was significantly positive by one-tailed bootstrap test; i.e., whether success increased with noise level. Annotation on horizontal lines indicates whether AUC for the best method was significantly higher than AUC of each other method by two-tailed bootstrap test. d–f are the same as a–c but for monkey→human attack. g–i are the same as a–c but for brown noise→human attack.
enough repeat presentations to reliably measure neuron responses, attack images were presented in 4 sessions of 2–3 noise levels each.

We visualized the neuron population representation of clean and adversarial images (level 10) using Uniform Manifold Approximation and Projection (UMAP)\(^{18}\) (Figure 2a). Neuronal representations of clean human and monkey faces were clearly separable. Some adversarial images were shifted away from human faces towards monkey faces. To quantify what fraction of the adversarial images would be categorized as the target category, we trained support vector machines (SVMs) to classify clean images as human or monkey based on neuronal responses. The SVMs were then used to classify held-out clean images as well as the adversarial images. The SVMs achieved high test accuracy on held-out clean human and monkey faces (98.3 ± 0.3% and 97.7 ± 0.5%, mean ± sem; Figure S3c), confirming that the recorded neurons distinctly represented human and monkey faces. Nevertheless, a fraction of human→monkey adversarial images were classified as monkey faces. We call these images successful, and calculated success rate as the fraction of successful images per method at each noise level (Figure S3a). The success rate of almost all attack methods increased with noise level (Figure 2b), as reflected in the positive coefficient in logistic regression of individual image success as a function of noise level (Figure 2c). The best attack method was CycleGAN-based interpolation, which reached a success rate of 21 ± 4% (mean ± sem) at noise level 10, followed by gray box pattern attack (19 ± 6%), pattern \(l_2\)-PGD (18 ± 5%), and affine-aligned linear interpolation (16 ± 4%; Figure S3c). To summarize and compare across methods, we calculated the area under the success rate curve (AUC) using the logistic fits (Figure S3c). We normalizing the x-value range to 0–1, so that the AUC value is roughly equivalent to the average success rate over levels. The best method, CycleGAN-based interpolation, was not significantly more successful than pattern attack (\(p = 0.068\); permutation test, FDR-corrected across 7 tests) but was more successful than all other methods (Figure 2c).
Logistic regression also allowed us to infer how much image change would be needed to achieve 50% successful attack with each method (Figure S3b). Since no method achieved 50% success rate within the tested noise levels, this midpoint estimate is necessarily an extrapolation. Nevertheless, an estimated midpoint may be useful to compare to other attack directions (below) and to the much larger MSE separating clean images. CycleGAN interpolation would require $\text{MSE} = 1030$ (bootstrap 95%-CI: 930–1150) to achieve 50% success rate. Simply replacing a human face with the closest monkey face among 250 images, while guaranteeing complete success, requires $\text{MSE} = 3210$ (95%-CI: 1670–4490; this differs from 2800 reported above because the clean images used for attack are different from the 250 images used for establishing a baseline). This closest distance increases to 4560 and 6800 for closest among 25 and 3 respectively, a roughly 1.5-fold increase for each order of magnitude; extrapolating in the other direction suggests that even if the trend continues, there may only be one in over $10^5$ monkey face images that is $\text{MSE} \approx 800$ away from an average human face image. Interpolating halfway between a human face and the closest monkey face among 250 entails $\text{MSE} = 800$ (95%-CI: 420–1120), but that likely corresponds to lower-than-50% success rate because empirically, linear interpolation at $\text{MSE} = 800$ achieved only $10 \pm 4\%$ success rate. Overall, these comparisons suggest that the attack success achieved by CycleGAN-based interpolation and pattern attack was not explained by trivially replacing the attacked image.

Were adversarial images being misclassified simply because noise degraded image quality? To control for this possibility, we tested two types of control image modifications at noise level 10. The first was image degradation, including Gaussian noise, Gaussian blur, phase scrambling, and Eidolon images at 3 coherence levels. The second was versions of targeted attack images where the image change was re-applied flipped upside-down, although flipping the adversarial noise may not be a complete control because clues indicating an upside-down monkey face may still contribute to indicating a monkey face.
Considered together, targeted attack images achieved significantly higher success rate than either clean human face images or non-targeted image modifications ($p = 1 \times 10^{-4}$; permutation test, FDR-corrected across 2 tests; Figure S3c). Considering each method individually, flipped CycleGAN images achieved the highest success rate (16 ± 3%) among control images, a success rate that was statistically no lower than success in all targeted attack methods (all pairwise $p > 0.13$; permutation test, uncorrected). This notwithstanding, the next best among 14 control methods, flipped single neuron attack, achieved only 4 ± 3% success rate. Thus, we speculate that flipped CycleGAN images still features of a monkey face to which neurons are sensitive. To anticipate, results below from human categorization support the conclusion the flipped CycleGAN images were not perceived as monkey faces.

**Adversarial Attack in Two Other Directions Also Led to Target Category-like Responses**

So far, we described human faces adversarially attacked to look like monkey faces (human → monkey or h2m attack). In adversarial attack in CNNs, one category can be attacked to target any other category. Therefore, we attempted two other attack directions: the reverse monkey→human (m2h) attack; and the brown noise→human face (b2h or noise→human) attack between two distant categories. We tested these attack directions with the same methods, except that for the gray box method, we only tested the match-response-pattern objective and not the excite-single-neuron objective.

Unexpectedly, monkey→human attack on neuronal responses was generally easier than human→monkey attack, as reflected by overall higher success rates for most methods at most noise levels (Figure 2e,f) and by the highest success rate achieved at noise level 10 (69 ± 34% for affine-aligned linear interpolation; Figure S3f). UMAP visualization (Figure 2d) shows that, unlike in human→monkey attack, many monkey→human attack images had the target
category (clean human faces) as the majority of near neighbors. However, attack success was also high for control image modifications, including untargeted image degradation (Figure S3f). This cannot be because neurons did not separate the categories of monkey and human faces; the categories were clearly separated, as we established above in human→monkey attack and further verified with the particular experimental sessions here (accuracy on human and monkey faces: 98 ± 5% and 98 ± 8%; Figure S3f). Instead, we speculate that the idiosyncrasies of the neuronal selectivity or the particular monkey images used in attack could have prevented SVM generalization. Failing to certify the SVM-based quantification, we could not meaningfully compare different attack methods or estimate the noise threshold for successful attack. We will show below that human behavior was able to separate effective targeted attack images from control image modifications.

Further counter to our prediction, brown noise→human face attack was still easier than the previous two attack directions. The most successful method judged by AUC was affine-aligned linear interpolation, which was significantly better than all other targeted methods (p ≤ 0.014; Figure 2i). It reached 91 ± 17% success rate at level 10, approaching the accuracy on human face images (98 ± 5%; Figure S3i). Linear interpolation (level 10 success rate: 87 ± 20%) and CycleGAN based-interpolation (76 ± 34%) were also highly effective, followed by gray box attack methods (pattern: 81 ± 24%; pattern $l_2$-PGD: 74 ± 25%). Model-ensemble attack was ineffective, achieving 0.6 ± 4% success rate at level 10, close to the error rate on brown noise images (0.4 ± 2%). UMAP visualization (Figure 2g) qualitatively corroborates the quantification, with most interpolation-based images intermixing with clean human faces, gray box images close to but forming a distinct cluster from human faces, and model ensemble images remaining embedded in the noise images cluster. The most successful control image modification (Gaussian blur) achieved only 16 ± 4% success at level 10, significantly lower than success rates of all targeted methods (all p = 2 × 10^{-5}; permutation test, FDR-corrected
across 6 tests) excluding model ensemble attack \((p = 0.175)\), indicating that targeted attacks were specific (Figure S3). Using fitted logistic regression, the estimated noise threshold for 50% successful attack by the best method (affine-aligned interpolation) was \(\text{MSE} = 470\) (95%-CI: 450–490), much lower than that for simply replacing the noise image with the closest human face \((\text{MSE} = 1850, 95\%-\text{CI}: 1530–2220)\), although close to that for interpolating half way between the two \((\text{MSE} = 460, 95\%-\text{CI}: 380–550)\). Post hoc, we could rationalize the relative ease of noise→human attack by suggesting that a face superimposed on noise can still look like a face. This may be attributed to primates’ heightened sensitivity to detecting faces (pareidolia) and/or the fact that noise images do not themselves belong to any category that can supply competing evidence to the evidence of a face.

To summarize, we tested changing image categorization by monkey neuronal responses through targeted adversarial attack in three directions. Within the range of image change tested, we achieved moderate success in one direction (human→monkey) and high-to-complete success in the other two (monkey→human and brown noise→human face). In human→monkey attack, although the highest success rate was lower than 50%, the success was not explained by the control method of interpolating the original image toward the target. In noise→human attack, close to complete attack success was achieved by control interpolation methods. We could not interpret the results in monkey→human attack since most types of non-targeted image degradation also achieved high success. We conclude that the ease of adversarial attack in monkey neuron was highly dependent on the involved categories and attack direction. In all attack directions, adversarial images tailored to CNNs (model ensemble attack) did not affect monkey neuronal representation, consistent with general intuition. There was no consistent evidence that model ensemble attack was more successful in the initial part of the response in a time-resolved analysis (Figure S4).
Could the same adversarial images also mislead human judgment? We recruited human subjects on Amazon Mechanical Turk (MTurk) to categorize these adversarial images in a two-way categorization task. Adversarial images were shown intermixed with an equal number of clean images, which should encourage the subjects to make their best attempt at categorizing every image and provided us a way to monitor accuracy. Subjects had ample time to examine the images (1 s, no backward masking) and make the choice (4 s). For each attack image, we collected responses from 4–5 subjects who completed the task and had high performance on clean images. Each subject was tested on only one noise level of attack images.

In human→monkey attack, the four variants of gray box attack were the most effective methods. At noise level 10, these methods achieved up to 46–65% success rate (Figure 3a). The best method was pattern attack, with 64 ± 6% success rate at noise level 10 and a success rate AUC that was higher than for all other methods, in particular linear interpolation and affine-aligned linear interpolation (both p = 3 × 10^{-5}; permutation test, FDR-corrected across 7 tests).

In monkey→human attack, affine-aligned linear interpolation was the most successful method, achieving 77 ± 29% success rate at noise level 10. In noise→human attack, the most successful method was linear interpolation (without affine-alignment) with 94 ± 9% success at noise level 10. In this direction, success rate started at noise level 1 at around 60% for the most effective methods and around 15% for the least effective method (model ensemble). This high sensitivity may be due to humans’ tendency to see pareidolia faces combined with a priming effect in the instruction to ‘determine whether each image is a face image or a non-face image.’

Unlike the corresponding results in monkey neuron attack, in human behavior in all attack directions, control image modifications always resulted in lower success rates than the best targeted attack methods (Figure S5c,f,i), indicating that the targeted methods were specific and allowing us to estimate the threshold noise level for 50% successful attack. The non-targeted
Figure 3: Results of adversarial attack on human behavior in a visual categorization task. Format of the plot follows that in Figure 2. a, b correspond to human→monkey attack. a shows attack success rate as a function of noise level (MSE) for targeted attack methods. Points represent method-level (image-averaged) attack success rate per noise level. Lines represent logistic regression fit on individual image results. b shows attack success AUC per attack method. Colored points and lines indicates center values and bootstrap 95%-CI, respectively. Triangle marker indicates method with maximum AUC. Annotations indicate statistical significance: *, p < 0.05; **, p < 0.01; ***, p < 0.001; and n.s., not significant, all p-values FDR-corrected for multiple comparisons. Annotation on each group indicate whether success increased with noise level. Annotation on horizontal lines indicate whether AUC for the best method was significantly higher than AUC of each other method by two-tailed bootstrap test. c, d are the same as a, b but for monkey→human attack. e, f are the same as a, b but for brown noise→human attack.
modification that led to the highest success rate was Eidolon 3 (i.e., Eidolon with coherence $= 0$; success rate $14 \pm 5\%$) for human→monkey attack, Gaussian blur (success rate $31 \pm 21\%$) for monkey→human attack, and flipped linear interpolation (success rate $18 \pm 4\%$) for noise→human attack. The threshold noise level for the most successful attack method in each direction was MSE = 750 (95%-CI: 710–790) for human→monkey attack, 550 (95%-CI: 510–600) for monkey→human attack, and 70 (95%-CI: 10–140; outside the tested range of noise and thus an extrapolation) for noise→human attack.

To summarize, we changed human visual categorization through targeted adversarial attack, achieving > 50% success in all three attack directions. As in monkey neuron-level results, we found a better attack method than interpolation-based control methods only in the most difficult human→monkey attack direction. Here, gray box pattern attack was the best method, while CycleGAN-based interpolation was much less effective. Also consistent with monkey neuron-level results, some attack directions were easier than others, with noise→human attack being the easiest attack direction, followed by monkey→human and lastly human→monkey attack. Finally, as in monkey neurons, CNN-tailored adversarial images were generally ineffective.

**Attack Success Was Correlated between Monkey Neuron Responses and Human Behavior**

To more directly compare the results of adversarial attack in monkey neurons and human behavior, we correlated attack success across methods (image-averaged) or across individual images (subject-averaged). To capture a more graded measure of attack success at the neuron level, we used the projection of an image’s trial-averaged response vector onto the norm of the decision hyperplane of linear SVMs. The projection value was normalized and shifted so that the mean location of source and target class images was 0 and 1 respectively; thus the projection value was usually but not always between 0 and 1. For human behavior, there was no equivalent
Figure 4: Adversarial attack success was correlated between monkey neuron responses and human behavior. The y-axis shows success in attacking human behavior; the x-axis shows the success rate for attacking monkey neuronal responses (see text for definition of neural SVM projection); dotted lines indicate identity. Dots are colored by the attack method with the same scheme as in Figure 2. a–c show method-level success (image-averaged) at noise level 10. Because many points are occluded due to the discrete y-values, linear regression of the points is indicated with solid lines. d–f show per-image success at noise level 10. a, d correspond to human→monkey attack. b, e are the same as a, d but for monkey→human attack. c, f are the same as a, d but for brown noise→human attack.
image-level measure (each subject only saw each attack image in one trial, and chose either one or the other category), so we quantified the success of each image as the fraction of subjects who chose the target category.

Success was positively correlated between at the money neuron- and human behavior-levels, with generally high correlation values ($r > 0.34$ on method-level and $r > 0.24$ on single image-level; Figure 4). Except on the method-level in human→monkey and monkey→human attack, the correlations were statistically significant ($p < 0.006$; exact test against null hypothesis that two samples are from independent Gaussian distributions, uncorrected). In fact, human-to-monkey correlation was as high as possible after taking into account the between-subject consistency (Figure 5a–d).

These analyses were limited to level 10 images only. Because success rate generally increased with noise level, including images from all noise levels introduces a confounding factor into correlation values. We repeated the analyses with all noise levels in Figure S6.

### Attack Images Reveal Mismatch between Primate Vision and CNNs

CNNs are expected to differ from primates in responding to adversarial attack, not least because CNNs should be highly sensitive to CNN-tailored adversarial images (model ensemble attack), which barely affected humans and monkeys. We measured CNN adversarial robustness in the same setting as the primate experiments, testing the same ensemble of 14 fine-tuned CNNs used to generate model ensemble attack images. Individual network outputs (logits) were averaged, then converted into a confidence vector over 1002 classes using the softmax function.

In human→monkey and monkey→human directions with model ensemble attack, success rate was already saturated at the lowest noise level 1 (Figure S7a,d), which was expected since noise level 1 is already large relative to the typical threshold reported in machine learning literature. Testing additional images at lower noise levels, we verified that CNNs could be attacked
Figure 5: Adversarial images reveal mismatch between primate vision and CNNs. a, b compare method-level (image-averaged) attack success rate at noise level 10 among 3 visual systems: monkey neuron responses (‘neural’), human behavior (‘mturk’), and CNN model output (‘model’). a compares human and model to monkey. b compares monkey and model to human. Horizontal lines and shaded area indicate inter-subject split-half self-consistency (center value and 95%-CI). Dots and vertical lines indicate correlation across visual systems (center and CI). c, d are the same as a, b, but compare image-level attack success rate. e shows the (lack of) generalization in state-of-the-art CNN-based models fitted to predict monkey neuron responses. The heat map indicates model performance quantified by fraction of response predicted by linear transformations fitted to predict neuron responses from features extracted by a ResNet-101\(^{11}\). In some configurations as described on the x-axis, a subset of image categories were included in both training and testing (individual images are always distinct between training and testing). Those cases are indicated by small grey squares in cells in the heat map. Models were tested on all categories, and prediction accuracy was evaluated separately for each category. Prediction accuracy was higher for held-out images from the same category (categories) that was (were) included in training than for images from excluded category (categories). Method details are described in text. f, results in e are summarized and combined over different attack directions. Small dots indicate individual cells as in e, color coded by the image category. Larger dots with whiskers indicate mean and bootstrap 95%-CI within each training configuration, color coded by whether the result corresponds to interpolation (lighter color) or extrapolation (darker color) performance; and whether the tested category was clean (blue) or attack (orange) images.
with a minuscule amount of image change (Figure S7b,e), with 50% success thresholds at MSE = 2.1 (95%-CI: 2.0–2.2) and 4.3 (95%-CI: 4.1–4.5) respectively for the two attack directions. Compared to primate attack threshold as the minimum over methods, CNN thresholds were still over 100 times lower. Granted, this ratio is a likely overestimation because we could much more readily optimize attack image for CNNs than for primates. Nevertheless, this ratio is a direct, quantitative comparison of model and primate adversarial robustness.

Unexpectedly, brown noise→human face attack for CNN models was difficult, requiring an attack threshold at MSE = 5310 (95%-CI: 5290–5340; Figure S7h)—even higher than the budget for simply replacing the noise image with the closest human face (MSE = 1880; 95%-CI: 1620–2210), an approach that would have led to perfect success (Figure S7h). This shows that a simple gradient descent algorithm could not find the optimal attack image at such high noise levels. The criterion for noise→human attack success might be more stringent than in other attack directions, because attack counted as successful only if confidence in the human category was higher than confidence in all other 1000 categories (excluding the monkey category). In contrast, in human→monkey attack for example, confidence in the human category only needed to exceed confidence in the monkey category. Most adversarial attack studies use the stricter one-vs-all criterion. However, to the best of our knowledge, we are not aware of studies that attempted adversarial attack starting from a noise image. Brown noise images may be particularly hard to attack (compared to Gaussian noise) in our preliminary tests.

How did CNNs respond to other attack methods? Gray box attack methods were the second-most effective group of methods in human→monkey and monkey→human attack, although not in noise→human attack (Figure S7a,d,g). In human→monkey attack, gray box methods were almost as effective as model ensemble attack and achieved 80–100% success rate starting from noise level 1. This result is unexpected, because although gray box attack used a ResNet as the feature-extraction backbone, the attack objective function was independent of weights in the
classification layer of the CNN. Indeed, gray box attack was not as effective as model ensemble attack in the monkey→human direction. In noise→human attack, linear interpolation was the most effective method, followed by CycleGAN interpolation and model ensemble attack.

Thus, the overall pattern of attack method effectiveness was different between CNNs and primates. We directly compared CNNs to primates by correlating attack success on the method- and individual image-levels as above (Figure 5a–d). At the method level, correlation between model and primate was negative in 3 out of 6 comparisons (monkey or human results, 3 attack directions). At the image level, model-primate correlation was always positive, but lower than primate-primate correlation in 5 out of 6 conditions (in the exception, monkey-human correlation was upper-bounded by low internal consistency).

This divergence in CNN and primate behavior suggests that CNN internal representation may also be less similar to neuronal representation than generally thought. To quantify this, we used the well-established approach of fitting a linear readout from CNN features to predict neuronal responses. Typically, CNN-based models for predicting neuronal responses are tested on interpolation to a held-out random subset of images that, by construction, are (in expectation) identically distributed as the training images. Here, we tested holding out entire categories of images, such as attack images, to test model extrapolation to images that likely came from a different distribution. An example scheme is shown in Figure 5e. With human→monkey attack data, we held out 1–3 categories of images (human faces, monkey faces, and/or attack images), fitted a linear model from CNN features, and validated the model on interpolation (unseen images from categories seen during training) or tested the model on extrapolation (images from unseen categories seen during training). Location on the x-axis indicates training configuration, grey squares indicate training categories, and location on the y-axis indicates validation or testing category. Models consistently performed much better on held-out images from categories included in training than on images from completely held-out categories. The same analysis,
performed and pooled over all three attack directions, is presented in Figure 5f. When testing on held-out subsets of images categories included in training, model validation performance was relatively high (first four groups in Figure 5f). However, when fitting on a subset of categories and testing on entirely held-out categories, performance was much lower (second four groups). Unexpectedly, generalization was about as poor from one clean image category to another (fifth group) as it was from attack images to clean images (sixth group) and vice versa (seventh group). Thus, although attack images revealed behavioral differences between CNNs and primates, the lack of generalization in CNN-based models of neurons could be already revealed by different categories of clean images, which likely had different statistics because they came from different image datasets.

**Discussion**

Perception does not merely mirror reality, as evinced by centuries of visual illusions and a longer history of art. Adversarial images could constitute a kind of visual illusion, namely minimally changed images that were originally unambiguous, yet come to be categorized differently. They would be distinct from phenomena such as pareidolia, which pertain to images intrinsically ambiguous or misleading. We were motivated to attempt creating adversarial images for the primate brain by the similarities between CNNs and primate vision and by the former’s high sensitivity to adversarial attack. We found adversarial images that were not expected by trivial methods in one of three targeted attack directions. These images fooled humans in visual categorization without strict time limits. Further, the altered categorization is reflected in monkey category-selective neuronal responses.

These adversarial images were created using a CNN-based model of visual neuron responses. CNNs are among the best current accounts of visual cortex function. A desirable account of the brain should be interpretable, have high predictive power, and be useful for con-
trolling the brain (a special case of prediction). Although CNNs can be said to be difficult to interpret\textsuperscript{22} (but see Richards et al.\textsuperscript{49}), they excel at predicting neuronal responses\textsuperscript{3,4,12} and, to an extent, serving as a guide for controlling those responses\textsuperscript{28,29}. Since CNNs are vulnerable to adversarial attack whereas primates are thought not to be susceptible, adversarial attack is an ideal testing ground for CNN models of the brain, by providing both images whose neuronal responses need explanation and a challenging target to which to guide neuron responses.

Using a CNN-based model to target a response pattern, we could specifically guide human categorization behavior in two attack directions (human→monkey and noise→human), achieving higher success than the best non-targeted method. This bridges the gap between previous studies that outline a link between visual neuronal activity and behavior\textsuperscript{30,31,32} and studies that show that computational models of neurons can be used to predictively control their activity\textsuperscript{28,29}.

However, this control was not consistently successful, not always better than a naive method of simply interpolating toward a target class image, and not as successful as the model predicted. Indeed, CNN-based models could not adequately predict neuron responses to adversarial images when trained only on clean images. Behaviorally, primate vision required much higher noise threshold to attack than models, even as future work may discover better methods for adversarially attacking primate vision. The pattern of which attack methods, and which individual attack images, were successful was also less similar between models and primates than within and between primate species. These results reveal a shortfall between CNN models and primate vision that is larger than previous studies suggest, and may point to directions and clues for building better models.

**Methods**

**Data and code availability** All stimuli, data underlying figures, code for data analysis, and code for generating attack images will be made available with the publication at https://
Subjects  One adult male macaca mulatta (10 kg; 13 years old) and one adult male macaca nemestrina (13 kg, 11 years old) were socially housed in standard quad cages on 12/12 hr light/dark cycles. Animals were implanted with custom-made titanium headposts before fixation training. After several weeks of fixation training, the animals underwent a second surgery for array implantation. Monkey 1 was implanted with chronic microelectrode arrays (MicroProbes, Gaithersburg, MD) targeting the medial lateral (ML) face patch. Monkey 2 was implanted with chronic brush microwire arrays targeting the anterior medial (AM) face patch. Array targets were localized using fMRI aligned to CT scans and, during surgery, using landmarks from the CT scans. Localization was confirmed after surgery by CT scans. Extracellular electrical signals were amplified and recorded using the Omniplex data acquisition system (Plexon, Dallas, TX). All surgeries were performed under general anesthesia using sterile technique. All procedures on non-human primates were approved by the Harvard Medical School Institutional Animal Care and Use Committee, and conformed to NIH guidelines provided in the Guide for the Care and Use of Laboratory Animals.

Human psychophysics experiments were conducted online on Amazon Mechanical Turk. All participants provided informed consent and received monetary compensation for participation in the experiments. All experiments were conducted according to protocols approved by the Institutional Review Board at Boston Children’s Hospital.

Clean images  We collected human and monkey faces from the web and from photographs taken in the lab to combine into 250 images per category. Monkey face images with excessively oblique head directions were excluded. Brown noise images were generated using custom code. Briefly, we generated a frequency-power spectrum that decayed as $1/f^2$, assigned random phase
per image, and generated pixel values via inverse Fourier transform. Pixel values were scaled and shifted to have a valid range.

Source images used for attack were kept separate from other clean images used for establishing baseline accuracy. Source human faces images were cropped versions from the Chicago Face Database. We used 40 different images for each of four main type of attack/manipulation method (interpolation-based, model ensemble, gray box, and non-targeted image degradation), totalling 160 source images. Source monkey faces were a set of 40 images separate from the set of 250 clean images; all attack methods used this set of 40 images. Source brown noise images were also different for each main type of attack/manipulation method and totalled 160 images separate from the 250 clean images.

**Adversarial and control images** In all, we tested 3 attack directions and 6–8 targeted methods at 10 noise levels in each direction (human→monkey attack included 8 methods; the other two directions included 6). We tested the following control methods: 6 methods for non-targeted image degradation, and versions of targeted images with the attack noise pattern flipped. Control methods were tested only at the highest noise level 10. Each method comprised 40 images at each direction and noise level, except human→monkey gray box methods, which each comprised 20 images, making a total of 8640 attack images.

**Generating adversarial images** All adversarial images were generated starting with a clean image $x$ from the source category. Linear interpolation images were generated as

$$x_{\text{adv}} = (1 - \lambda)x + \lambda x_{\text{target}},$$

where $x_{\text{target}}$ is different for each method, and $\lambda$ is a parameter from 0 to 1 tuned to give a desired noise level. For ordinary linear interpolation, $x_{\text{target}}$ is the closest image from 250 clean images of the target class. For affine-aligned linear interpolation, $x_{\text{target}}$ is the closest target
class image after an affine transformation that was optimized for each source-target image pair. Boundary mode of affine transformation (for filling uncovered pixels after transformation) was the better one of either ‘reflect’ or ‘constant,’ in the latter case with the constant fill value being an additional parameter to be optimized. Unconstrained optimization sometimes resulted in images that no longer resembled the target class, e.g., by being scaled up to become an essentially constant image. Thus, affine transformation was subject to the following constraints: scale between 3:4 and 4:3; shear and rotation between -45° and 45°; and shift between -50 and 50 pixels (out of 224). For CycleGAN-based interpolation, $x_{\text{target}} = \text{CycleGAN}_{\text{source} \rightarrow \text{target}}(x)$. We trained two CycleGAN models\(^{42}\), one for translation between human and monkey faces and the other for translation between human faces and brown noise images.

Tuning of the parameter $\lambda$ was done in Python using `scipy.optimize.minimize` with method = ‘Powell’. Finding optimal affine transformation parameters was a generally difficult problem, and we used the best parameters found by any of 7 optimization algorithms, including `scipy.optimize.minimize` options ‘L-BFGS-B,’ ‘TNC,’ ‘SLSQP,’ ‘COBYLA,’ ‘Powell,’ and ‘trust-constr’; and an CMA-ES family optimization algorithm (Algorithm 2 in \(^{54}\)) we custom-implemented.

Both model ensemble and gray box adversarial images were generated using iterative gradient descent (related to \(^{61}\) but distinct), with different objective functions. Specifically, each adversarial image $x_{\text{adv}}$ was iteratively generated as

$$
{x_{\text{adv}}^{t+1}} = x_{\text{adv}}^{t} + \epsilon \cdot \frac{g}{\text{stdev}(g) + \xi}, \quad g = \nabla_x \mathcal{L}(\theta, x_{\text{adv}}^{t}),
$$

where $x_{\text{adv}}^{t}$ is the adversarial image at iteration $t$; $g = \nabla_x \mathcal{L}(\cdot)$ is the gradient of the cost function $\mathcal{L}$; $1/\text{stdev}(g)$ keeps gradient values at roughly the same scale across loss functions; $\xi = 10^{-8}$ prevents underflow; $\epsilon$ is a learning rate that controls the amount of change in each step; and $\theta$ represents the substitute model parameters. For example, in human→monkey attack, $x_{\text{adv}}^{0} = x$. \(^{27}\)
is a clean human face. The cost function at step $t$ is

$$
\mathcal{L} = \mathcal{L}_\text{target} + \mathcal{L}_v,
\mathcal{L}_v = \sum_{i,j} (|x_{i+1,j} - x_{i,j}| + |x_{i,j+1} - x_{i,j}|),
$$

(2)

where $\mathcal{L}_\text{target}$ is a method- and target-specific loss function to be specified below, $\mathcal{L}_v$ is the total variation loss, and $(i,j)$ indexes pixels of the image. The total variation loss prevents high frequency noise from dominating the generated image features.

The target-specific loss function for gray box pattern attack was $\mathcal{L}_\text{target} = \|M_\theta(x_{adv}^t) - P_{\text{target}}\|_2^2$, where $M_\theta$ is the substitute model, $P_{\text{target}}$ is the mean population neuron response pattern averaged over target class images, and $\|\cdot\|_2^2$ is the $l_2$-norm between model-predicted and target response patterns.

The target-specific loss function for gray box single-neuron attack was $\mathcal{L}_\text{target} = -[M_\theta(x_{adv}^t)]_3$, where $[\cdot]_3$ indicates indexing the predicted activity of the third neuron (out of 22), a neuron we found to produce monkey-like features using feature visualization.34

For model ensemble attack, $\mathcal{L}_\text{target} = -[M_\theta(x_{adv}^t)]_i$, where $i$ is the index of the monkey class for human→monkey attack and the index of the human class for monkey→human and brown noise→human face attack.

For all images generated with iterative gradient descent, noise level was controlled as follows. Noise level generally increased with the number of iterations, so we kept the last image during iteration that still fell into an MSE bin for each noise level 1–10. The bins were centered on the pre-specified MSE for each noise level with width $\text{MSE} = 66.7$ (the spacing between noise levels). Final images were projected to the pre-specified central MSE value. Learning rate was 0.03 for pattern attack and 0.08 for single neuron attack and model ensemble attack. Learning rate was lower (higher) for model ensemble attack with smaller (larger) MSE range in Figure S7b,e,h. As many iterations as necessary were run to produce the specified noise levels, and generally ranged between 1,000 to 42,000 iterations.
All images generated with iterative gradient descent used jittering (40 pixels, randomized per iteration) as an additional regularizer for reducing high-frequency noise.

In $l_2$-PGD, the image during optimization was projected to the pre-specified MSE level, every iteration after that MSE level was first reached. Iteration continued until loss no longer decreased for 500 iterations. Images at the best iteration were saved. For each source image, we successively generated lower to higher noise levels, using each lower noise level attack image to seed optimization for the next higher noise level.

**Generating control images** Gaussian noise images were generated as $x_{\text{gauss}} = \text{clip}(x + \lambda g_x)$, where $g_x$ was sampled from the standard normal distribution, $\text{clip}(\cdot)$ keeps pixel values in the valid range, and $\lambda$ is tuned to yield a desired noise level.

Gaussian blur images were generated as $x_{\text{blur}} = \text{Blur}(x, \sigma)$, where $\sigma$ was the blurring filter kernel size tuned to yield a desired noise level.

Phase-scrambled images were generated as $x_{\text{phase}} = \text{PhaseShift}(x, \sigma)$, where $\sigma$ was the width of phase shift tuned to yield a desired noise level. The phase shift function was adapted from code from Geirhos et al. Briefly, it applies random uniform phase shift of the given width to the phase component of the image Fourier transform, symmetrically for the positive and negative frequency components, while leaving unchanged the amplitude component.

Eidolon images were generated as $x_{\text{phase}} = \text{Eidolon}(x, \text{reach}, \text{coherence})$, where coherence $= 1, 0.3, 0$ corresponds to what we refer to as Eidolon 1, Eidolon 2, and Eidolon 3 respectively, and reach was tuned to give a desired noise level. The Eidolon function was adapted from code from Geirhos et al., in turn based on a Python implementation of the original paper describing Eidolon images.

Upside-down flipped images were generated as $x_{\text{flip}} = \text{clip}(x + \text{flipUD}(x_{\text{adv}} - x))$. Around 0.3% to 5% of pixel values were clipped, resulting in image change values slightly under MSE.
= 800 for noise level 10.

Tuning of parameter(s) to achieve the specified noise level was done by optimization in Python using scipy\textsuperscript{55} function ‘scipy.optimize.minimize,’ as described above.

**Substitute model** To generate adversarial images, a substitute model was trained to predict IT neuronal responses. The model was fixed before generating and testing adversarial images. The substitute model comprised a pre-trained CNN (ResNet-101\textsuperscript{11} trained on ImageNet) and a linear mapping model (Figure\textsuperscript{1B}). Because the 1,000 pre-trained categories did not include our categories of interest (human face and monkey face), we collected around one thousand images for these two categories for fine-tuning the ResNet-101. Next, a linear model was trained to map extracted image features (layer conv5_3, the last convolutional layer) to neuronal responses. The linear model was factorized in the spatial and feature dimensions\textsuperscript{10}. The spatial module was a convolutional kernel $W_s$. The feature module was a mixing pointwise convolution $W_t$, i.e., a weighted sum over the feature dimension. Both $W_s$ and $W_t$ were parameterized separately for each IT neuron. Thus, the response for neuron $n = 1, \ldots, 22$ to image $x$ was modeled as

$$
\hat{y}_n = \sum (W_s^n \ast \text{ResNet}_{\text{conv5}_3}(x)) \cdot W_t^n + W_d^n,
$$

where $\ast$ denotes the convolution operation and $W_d^n$ is a bias parameter. The parameters were jointly optimized to minimize a loss function $L_e$ composed of the prediction error $L_p$, an L2-regularization loss $L_2$, and a spatial smoothness loss $L_{\text{laplace}}$:

$$
L_p = \sqrt{\sum_n (\hat{y}_n - y_n)^2}
$$

(4)

$$
L_2 = \lambda_s \sum ||W_s||_2 + \lambda_t \sum ||W_t||_2
$$

(5)
\[ \mathcal{L}_{\text{laplace}} = \lambda_l \sum W_s * \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} \]  

(6)

\[ \mathcal{L}_e = \mathcal{L}_p + \mathcal{L}_2 + \mathcal{L}_{\text{laplace}} \]  

(7)

The hyper-parameters \( \{\lambda_s, \lambda_t, \lambda_l\} = \{1, 0.1, 0.7\} \) were obtained from a grid-search to produce the highest prediction accuracy. The substitute model achieved an average correlation of 0.4 between predicted and actual IT neuron responses on held-out test images.

**Neuron-level experiment**  We recorded neuronal responses to image presentation while monkeys did a passive fixation task. Images were presented on an LCD monitor (ASUS VG248, 165 Hz) at a viewing distance of 57–61 cm. Monkeys were required to fixate within a window of 1.5–3 degrees in exchange for juice reward. Eye position was monitored using an infrared eye tracker (EyeLink, Ottawa, Canada). Images were presented at 4 degrees of visual angle in size in randomized order in the receptive fields of the neurons. Each image was presented for 100 ms, with a 150 ms interval between images for monkey 1 and 400 ms for monkey 2 due to longer response dynamics. Image onset time and spike time stamps were aligned using digital event words and analog photodiode signal.

Neuronal responses to an image were calculated as firing rates averaged over trials within a response window after stimulus onset that was automatically selected per session to maximize split-half self consistency. The response window for monkey 1 started between 60–90 ms and ended between 240–280 ms; and for monkey 2, started between 125–180 ms and ended between 340–400 ms. Visually selective neurons were selected based on split-half self consistency > 0.1. Although self consistency is a function of number of trials, the cutoff of correlation = 0.1 is generous in our experience and compared to prior work. Qualitative results were not affected when we repeated the analysis with different selection criteria. Responses per neuron were
Behavior-level experiment  Subjects on Amazon Mechanical Turk were invited to perform an image categorization task. They were instructed to determine ‘whether each image is a human face or a monkey face’ for human→monkey and monkey→human attack sessions (subjects were not informed of this), or ‘whether each image is a face image or a non-face image’ for brown noise→human face attack. Subjects were instructed to ‘Make your best guess. Sometimes, it may be hard to determine the correct answer.’ To answer, subjects pressed the left arrow key for ‘Human face’ or ‘face,’ and right arrow key for ‘non-face’ or ‘Monkey face,’ as appropriate. Each image was presented for 1.5 s. There was no time limit for a response. No feedback was provided. Adversarial images were randomly and evenly intermixed with clean images, which allowed us to monitor performance. We selected subjects with > 95% accuracy to include in further analyses. Responses were collected from 4–5 subjects for half of all attack images (20 per method per direction/noise level).

Quantification and statistics

Quantifying size of image change  Size of image change (noise level) was quantified by Mean Squared Error (MSE) as

$$
\text{MSE} = \frac{1}{N} \sum_{i,j} (x'_{i,j} - x_{i,j})^2,
$$

where $x'$ and $x$ are a pair of images, and $i, j$ index $N$ total pixels. All images were standardized to be $N = 224 \times 224$ pixels one-channel images with pixel value in the range 0–255. At this image size, MSE is related to $l_2$ (pixel value range 0–1) as

$$
l_2 = \sqrt{\sum_{i,j} \left( \frac{1}{255} \cdot (x'_{i,j} - x_{i,j}) \right)^2} = \sqrt{N/255^2} \cdot \text{MSE} \approx 0.8784 \sqrt{\text{MSE}}.
$$
Quantifying neuron- and behavior-level attack success  Neuron-level attack success was quantified using linear Support Vector Machines (SVMs) separately for each experimental session. SVM fitting was carried out using the Python package ‘scikit-learn’. SVMs were trained to categorize clean images (two-way categorization) based on the corresponding neuronal responses. For example, in monkey→human attack, the SVM was trained to classify each image as either a monkey face or a human face. We trained SVMs with balanced samples for 250 train-validation splits, leaving two images out in each split (one from each class). Moreover, we trained both linear and radial-basis function (RBF) SVMs, resulting in 500 classifiers per experimental session. The trained SVM was used to classify the held-out clean images, attack images, and control images. Attack success for each image was the fraction of classifiers that classified that image as the target class, a value that was between 0–1 but usually close to either 0 or 1. This value was further averaged across sessions and across 2 monkeys.

SVM projection was only defined for linear SVMs. It was calculated as the dot product between an image’s response pattern and SVM classification weights, then shifted and scaled so that the category-average was 0 and 1 for the source and target categories, respectively.

Behavior-level attack success was quantified, for each image, as the fraction of trials/subjects in the target category was chosen (each subject only saw each image once).

Success rates for a method at a given noise level were calculated as the average success over images. Logistic regression was fitted on individual image-level data to capture image-level success as a function of MSE for each image (MSE was tightly distributed per noise level but not identical among images). Logistic regression was carried out using the Python package ‘statsmodels’.

Time-resolved neuron responses analysis  Plotted responses were smoothed with a uniform filter of width 25 ms. Principal components (‘eigen-neurons’) were derived from standardized
neuronal responses as described above, then used to project the smoothed population response
time courses. D-prime (for selecting example neurons) was calculated as \( \frac{\mu_1 - \mu_2}{\sqrt{\frac{1}{2} (\sigma_1 + \sigma_2)}} \),
where \( \mu \) and \( \sigma \) indicate mean and standard deviations of responses to one image category. SVMs
fitting was done as described above, repeatedly for successive, non-overlapping 10 ms responses
windows.

**Model prediction of neuronal responses in Figure 5** Instead of using the method described
above to build a substitute model, for the analysis in Figure 5 we used a simpler and much faster
modeling approach following Zhuang et al\(^{52}\), which is in turn similar to prior work\(^{3,12}\). Neuronal
response models were based on features extracted by an ImageNet-trained ResNet-101 and
fitted using partial least squares (PLS) regression with 5 retained components. We searched for
the optimal ResNet layer (out of 105 including raw pixels) to use per monkey/array, using data
not included in the results. We chose layer ‘layer3.5.conv1’ for monkey 1 and ‘layer3.6.conv1’
for monkey 2. Training was done using 5-fold cross-validation, i.e., holding out each 20% of
the data in turn for testing. Cross validation was stratified; for example, when fitting on all
image categories, each category had almost the same number (difference no larger than 1) of
images in each training/validation fold. All reported model performance values were based on
held-out images. PLS regression and stratified cross-validation was implemented by the Python
package ‘scikit-learn’\(^{59}\).

Fraction of neuronal responses explained by model was quantified similar to prior work\(^{28,12,52}\).
Specifically, we calculated the fraction explained per neuron as the square of a ratio whose nom-
inator was the Pearson correlation between model prediction and neuron responses, and whose
denominator was the split-half self consistency of that neuron (Pearson’s r, Spearman-Brown
corrected). This value resembles fraction of variance explained, i.e., coefficient of determina-
tion or \( R^2 \), although they are not equivalent unless an additional optimal linear transform was
fit between the predicted and actual neuronal responses on the test set to remove any mean and scale differences. Fraction of variance explained, or $R^2$, was not suited for describing model performance on held-out data; $R^2$ was not directly used in the cited prior studies either. To produce one value for each training-testing configuration for each data set, we took the average over splits and median over neurons.

**Consistency between and across readouts in Figures 4-5** Consistency was calculated as Pearson’s correlation coefficient. Consistency within a readout was calculated by splitting the data in half by subjects, then correlating across method- (Figure 5a,b) or image-averages (Figure 5c,d), clipped at 0, then corrected using the Spearman-Brown formula.

**Center and spread estimates** Center estimates of AUC and attack threshold MSE were based on the logistic fit to the original data; confidence interval (CI) estimates were based on bootstrapping 500 times over images and fitting a logistic function to each bootstrap sample. Center and CI estimates of success rate at one noise level were based on original data and bootstrapping over images at that noise level. Center and CI estimates of Attack threshold MSE of ‘Replace,’ ‘Interp. half way,’ and ‘Erase’ was based on original data and bootstrapping over the set of source images. Center and CI estimates of consistency within and between readouts (Figure 5) were based on original data and bootstrapping over methods or images as appropriate.

**One-sample statistical tests for positive slope in logistic regression** Logistic regression was repeated with data in which noise level (MSE values) was permuted relative to attack success. Permutation was repeated 10,000 times to build a null distribution of the model ‘slope’ parameter. One-tailed p-value was calculated as the fraction of null distribution parameters that was larger than the sample value. P-values were clipped at a minimum of 1/10,000.
Two-sample statistical tests for difference between methods or method groups  For each pair of groups/conditions compared, the group label in data was permuted relative to attack success (and noise level if applicable). For tests comparing AUC between methods, logistic regression was performed with permuted data. This creates a null distribution of pairwise difference in AUC, to which the sample value was compared. For tests comparing success rate at one noise level, the sample success rate difference was compared to the permuted null distribution of success rate differences. One-tailed p-value was the fraction of null values that was larger than the sample value. P-values were clipped at a minimum of 1/10,000.

Correction for multiple comparisons  Correction was performed over each group of related tests (indicated in text) to control false discovery rate at the level of 0.05, using the two-stage Benjamini-Krieger-Yekutieli procedure as implemented in the Python library ‘statsmodels’.

Acknowledgments

This work was supported by NIH grants R01EY16187, P30EY012196, R01EY026025, and NSF STC award CCF-1231216 to the Center for Brains, Minds and Machines at MIT. Jiashi Feng was supported by grants ALSG R-263-000-D97-490 and MOE Tier-II R-263-000-D17-112. We thank Hu Zhang, David Castro, Ariana Sherdil and Peter F. Schade for discussion and assistance during the work.

Author contributions

JF, WX, and LY conceived of the study. GK, MSL, WX, and LY designed the experiments. LY developed the code for creating adversarial images. MSL and WX acquired the neuronal recording data. LY acquired the human behavior data. LY and WX analyzed the data and drafted the manuscript. All authors interpreted the data and revised the manuscript. GK, MSL,
FEHT, and JF provided funding.

**Competing interests**

The authors declare no competing interests.
References

[1] He, K., Zhang, X., Ren, S. & Sun, J. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In Proceedings of the IEEE international conference on computer vision, 1026–1034 (2015).

[2] LeCun, Y., Bengio, Y. & Hinton, G. Deep learning. nature 521, 436–444 (2015).

[3] Yamins, D. L. et al. Performance-optimized hierarchical models predict neural responses in higher visual cortex. Proceedings of the National Academy of Sciences 111, 8619–8624 (2014).

[4] Cadena, S. A. et al. Deep convolutional models improve predictions of macaque v1 responses to natural images. PLoS computational biology 15, e1006897 (2019).

[5] Szegedy, C. et al. Intriguing properties of neural networks. arXiv preprint arXiv:1312.6199 (2013).

[6] Goodfellow, I. J., Shlens, J. & Szegedy, C. Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572 (2014).

[7] Elsayed, G. et al. Adversarial examples that fool both computer vision and time-limited humans. In Advances in Neural Information Processing Systems, 3910–3920 (2018).

[8] Zhou, Z. & Firestone, C. Humans can decipher adversarial images. Nature communications 10, 1–9 (2019).

[9] Berardino, A., Laparra, V., Ballé, J. & Simoncelli, E. Eigen-distortions of hierarchical representations. In Advances in neural information processing systems, 3530–3539 (2017).

[10] Klindt, D., Ecker, A. S., Euler, T. & Bethge, M. Neural system identification for large populations separating “what” and “where”. In Advances in Neural Information Processing Systems, 3506–3516 (2017).

[11] He, K., Zhang, X., Ren, S. & Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, 770–778 (2016).

[12] Schrimpf, M. et al. Brain-score: Which artificial neural network for object recognition is most brain-like? BioRxiv 407007 (2018).

[13] Moeller, S., Freiwald, W. A. & Tsao, D. Y. Patches with links: a unified system for processing faces in the macaque temporal lobe. Science 320, 1355–1359 (2008).

[14] Deng, J. et al. Imagenet: A large-scale hierarchical image database. In 2009 IEEE conference on computer vision and pattern recognition, 248–255 (Ieee, 2009).

[15] Konkle, T., Brady, T. F., Alvarez, G. A. & Oliva, A. Conceptual distinctiveness supports detailed visual long-term memory for real-world objects. Journal of experimental Psychology: general 139, 558 (2010).

[16] Ma, D. S., Correll, J. & Wittenbrink, B. The chicago face database: A free stimulus set of faces and norming data. Behavior research methods 47, 1122–1135 (2015).
[17] Tsao, D. Y., Freiwald, W. A., Tootell, R. B. & Livingstone, M. S. A cortical region consisting entirely of face-selective cells. *Science* **311**, 670–674 (2006).

[18] McInnes, L., Healy, J. & Melville, J. UMAP: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426* (2018).

[19] Zhang, H. *et al.* Theoretically principled trade-off between robustness and accuracy. *arXiv preprint arXiv:1901.08573* (2019).

[20] Tramèr, F. *et al.* Ensemble adversarial training: Attacks and defenses. *arXiv preprint arXiv:1705.07204* (2017).

[21] Yamins, D. L. & DiCarlo, J. J. Using goal-driven deep learning models to understand sensory cortex. *Nature neuroscience* **19**, 356 (2016).

[22] Serre, T. Deep learning: the good, the bad, and the ugly. *Annual Review of Vision Science* **5**, 399–426 (2019).

[23] Ilyas, A. *et al.* Adversarial examples are not bugs, they are features. In *Advances in Neural Information Processing Systems*, 125–136 (2019).

[24] Xie, C. *et al.* Adversarial examples improve image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 819–828 (2020).

[25] Bracci, S., Ritchie, J. B., Kalfas, I. & de Beeck, H. P. O. The ventral visual pathway represents animal appearance over animacy, unlike human behavior and deep neural networks. *Journal of Neuroscience* **39**, 6513–6525 (2019).

[26] Wardle, S. G., Taubert, J., Teichmann, L. & Baker, C. I. Rapid and dynamic processing of face pareidolia in the human brain. *Nature Communications* **11**, 1–14 (2020).

[27] Bao, P., She, L., McGill, M. & Tsao, D. Y. A map of object space in primate inferotemporal cortex. *Nature* 1–6 (2020).

[28] Bashivan, P., Kar, K. & DiCarlo, J. J. Neural population control via deep image synthesis. *Science* **364**, eaav9436 (2019).

[29] Walker, E. Y. *et al.* Inception loops discover what excites neurons most using deep predictive models. *Nature neuroscience* **22**, 2060–2065 (2019).

[30] Sheinberg, D. L. & Logothetis, N. K. The role of temporal cortical areas in perceptual organization. *Proceedings of the National Academy of Sciences* **94**, 3408–3413 (1997).

[31] Afraz, S.-R., Kiani, R. & Esteky, H. Microstimulation of inferotemporal cortex influences face categorization. *Nature* **442**, 692–695 (2006).

[32] Rajalingham, R. & DiCarlo, J. J. Reversible inactivation of different millimeter-scale regions of primate it results in different patterns of core object recognition deficits. *Neuron* **102**, 493–505 (2019).

[33] Iaccarino, H. F. *et al.* Gamma frequency entrainment attenuates amyloid load and modifies microglia. *Nature* **540**, 230–235 (2016).
[34] Mahendran, A. & Vedaldi, A. Visualizing deep convolutional neural networks using natural pre-images. *International Journal of Computer Vision* **120**, 233–255 (2016).

[35] Papernot, N. *et al.* Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia conference on computer and communications security*, 506–519 (2017).

[36] Liu, Y., Chen, X., Liu, C. & Song, D. Delving into transferable adversarial examples and black-box attacks. In *Proceedings of the International Conference on Learning Representations (ICLR)* (2017).

[37] Chen, P.-Y., Zhang, H., Sharma, Y., Yi, J. & Hsieh, C.-J. Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, 15–26 (2017).

[38] Ilyas, A., Engstrom, L., Athalye, A. & Lin, J. Black-box adversarial attacks with limited queries and information. *arXiv preprint arXiv:1804.08598* (2018).

[39] Narodytska, N. & Kasiviswanathan, S. Simple black-box adversarial attacks on deep neural networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 1310–1318 (2017).

[40] Goodfellow, I. & Papernot, N. Is attacking machine learning easier than defending it. *Blog post on Feb 15*, 2017 (2017).

[41] Tramèr, F., Behrmann, J., Carlini, N., Papernot, N. & Jacobsen, J.-H. Fundamental trade-offs between invariance and sensitivity to adversarial perturbations. In *International Conference on Machine Learning*, 9561–9571 (PMLR, 2020).

[42] Zhu, J.-Y., Park, T., Isola, P. & Efros, A. A. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, 2223–2232 (2017).

[43] Geirhos, R. *et al.* Generalisation in humans and deep neural networks. *arXiv preprint arXiv:1808.08750* (2018).

[44] Koenderink, J., Valsecchi, M., van Doorn, A., Wagemans, J. & Gegenfurtner, K. Eidolons: Novel stimuli for vision research. *Journal of Vision* **17**, 7–7 (2017).

[45] Silver, D. *et al.* Mastering the game of go without human knowledge. *nature* **550**, 354–359 (2017).

[46] Xie, S., Girshick, R., Dollár, P., Tu, Z. & He, K. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1492–1500 (2017).

[47] Huang, G., Liu, Z., Van Der Maaten, L. & Weinberger, K. Q. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4700–4708 (2017).

[48] Hu, J., Shen, L. & Sun, G. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 7132–7141 (2018).
[49] Richards, B. A. et al. A deep learning framework for neuroscience. *Nature neuroscience* **22**, 1761–1770 (2019).

[50] Geirhos, R. generalisation-humans-dnns (2018). URL https://github.com/rgeirhos/generalisation-humans-DNNs/commits/master

[51] Gestaltrevision. Eidolon (2016). URL github.com/gestaltrevision/Eidolon/tree/8af01f424c2e5776479e05e42b699c4adef25f88

[52] Zhuang, C. et al. Unsupervised neural network models of the ventral visual stream. *Proceedings of the National Academy of Sciences* **118** (2021).

[53] Benjamini, Y., Krieger, A. M. & Yekutieli, D. Adaptive linear step-up procedures that control the false discovery rate. *Biometrika* **93**, 491–507 (2006).

[54] Loshchilov, I. A computationally efficient limited memory cma-es for large scale optimization. In *Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation*, 397–404 (2014).

[55] Virtanen, P. et al. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods* **17**, 261–272 (2020).

[56] Pedregosa, F. et al. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research* **12**, 2825–2830 (2011).

[57] Seabold, S. & Perktold, J. statsmodels: Econometric and statistical modeling with python. In *9th Python in Science Conference* (2010).

[58] Waskom, M. L. seaborn: statistical data visualization. *Journal of Open Source Software* **6**, 3021 (2021). URL https://doi.org/10.21105/joss.03021

[59] Pedregosa, F. et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research* **12**, 2825–2830 (2011).

[60] Kar, K., Kubilius, J., Schmidt, K., Issa, E. B. & DiCarlo, J. J. Evidence that recurrent circuits are critical to the ventral stream’s execution of core object recognition behavior. *Nature neuroscience* **22**, 974–983 (2019).

[61] Kurakin, A., Goodfellow, I., Bengio, S. et al. Adversarial examples in the physical world (2016).
Extended data figures

Figure S1: Related to Figure 1. Location of recording arrays in both monkeys is shown in relation to fMRI-defined patches selective for faces over objects. Blue indicates arrays localized by CT; red indicates face selectivity (thresholded at false discovery rate = 0.01).
Figure S2: Related to Figure 1. Shown are examples of images used in this study: human faces, and level 10 images from additional targeted or control methods not included in Figure 1.
Figure S3: Related to Figure 2. Each row indicates a different attack direction. a, d, g show mean success rate and bootstrap 95%-CI for each method and noise level without logistic regression. In b, e, h, horizontal bars show the MSE level at which success rate for each method should be 50%. Top bars are based on logistic regression on data from targeted attack methods over levels, and indicate either interpolated (inside tested MSE range of 200–800) or extrapolated (outside tested range) values. Bottom five bars indicate the MSE level associated with image manipulations that were not tested. ‘Replace’ means replacing an original image with the closest target-class example over varying numbers of target-class examples. ‘Interp. halfway’ means 50% interpolation towards the closest target-class example. Erase means converting the image to a uniform image with the same mean (which minimizes MSE over uniform images of different images). Vertical tick in each bar shows the sample estimate and extent of the bar shows 95%-CI based on repeating each analysis over bootstrap samples of images. Two underlying histograms indicate the distribution of pairwise image distances between the target class and clean images from the source class (light gray), or exact same starting images for adversarial attack from the source class (dark gray). ‘Replace’ calculations were based on the exact starting images. c, f, i shows level 10 success rate of all targeted attack methods and control manipulations, as well as accuracy on clean images (with perfect accuracy being 0 and 1 for the source and target attack category, respectively).
Figure S4: Related to Figure 2. Detailed responses over time are shown for clean and targeted attack images at level 10. Colors indicate image category as in Figure 2, dashed lines further distinguish $l_2$-PGD methods from non $l_2$-PGD methods, which have similar color. Each row indicates one experimental session. Odd roles indicate one monkey and even rows indicate the second monkey. First four columns show four separate dimensions in neuron responses: 1) first principal component of the population response; 2) second principal component; 3) the most selective unit for one category (quantified by d-prime); 4) the most selective unit for the other category. Y-axis unit is arbitrary for columns 1 and 2, and in units of spikes/s for columns 3 and 4. Column 5 shows success rate quantifying by fitting and testing SVMs on rolling, non-overlapping 10-ms windows of neuronal responses. Column 6 shows the same data as in column 5 with success rates for the clean categories normalized to 0 and 1, showing only the time window during which both clean category accuracies were over 0.75.
Figure S5: Related to Figure 3. Format of this figure is the same as Figure S3.
Figure S6: Related to Figure 4 and with similar format. This figure includes images at other noise levels, indicated by dot size.
Figure S7: Related to Figure 5. Each row indicates an attack direction. Column 1 shows success rate on an ensemble of CNNs by each attack method at each noise level; format is the same as in Figure 2b. Column 2 top shows model ensemble-attack success rate as a function of MSE for a different range covering low-to-high success rate; bottom shows the estimated threshold for 50% attack success based on logistic regression. Column 3 shows the success rate of all targeted attack methods, control image manipulations, and clean images; format is the same as in Figure S3c.