Model metamers illuminate divergences between biological and artificial neural networks

Jenelle Feather[1,2,3], Guillaume Leclerc[4,5], Aleksander Mądry[4,5], Josh H. McDermott[1,2,3,6]

[1] Department of Brain and Cognitive Sciences, Massachusetts Institute of Technology
[2] McGovern Institute, Massachusetts Institute of Technology
[3] Center for Brains Minds and Machines, Massachusetts Institute of Technology
[4] Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology
[5] Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology
[6] Speech and Hearing Bioscience and Technology, Harvard University

Abstract
Deep neural network models of sensory systems are often proposed to learn representational transformations with invariances like those in the brain. To reveal these invariances we generated "model metamers" – stimuli whose activations within a model stage are matched to those of a natural stimulus. Metamers for state-of-the-art supervised and unsupervised neural network models of vision and audition were often completely unrecognizable to humans when generated from deep model stages, suggesting differences between model and human invariances. Targeted model changes improved human-recognizability of model metamers, but did not eliminate the overall human-model discrepancy. The human-recognizability of a model's metamers was well predicted by their recognizability by other models, suggesting that models learn idiosyncratic invariances in addition to those required by the task. Metamer recognition dissociated from both traditional brain-based benchmarks and adversarial vulnerability, revealing a distinct failure mode of existing sensory models and providing a complementary benchmark for model assessment.

Introduction

A central goal of neuroscience is to build models that reproduce brain responses and behavior. The hierarchical nature of biological sensory systems (1, 2) has motivated the use of hierarchical neural network models that transform sensory inputs into task-relevant representations (3, 4). As such models have become the top-performing machine perception systems over the last decade, they have also emerged as the leading models of both the visual and auditory systems (5–11).

One hypothesis for why artificial neural network models might replicate computations found in biological sensory systems is that they instantiate invariances that mirror those in such systems (12–14). For instance, visual object recognition must often be invariant to pose, and to the direction of illumination. Similarly, speech recognition must be invariant to speaker identity and to details of the prosodic contour. Sensory systems are hypothesized to build up invariances (15–17) that enable robust recognition. Such invariances plausibly arise in neural network models as a consequence of optimization for recognition tasks or other training objectives.

Although biological and artificial neural networks might be supposed to have similar internal invariances, there are some known human-model discrepancies that suggest the invariances of the two systems do not perfectly match. For instance, model judgments are often impaired by stimulus manipulations to which human judgments are invariant, such as additive noise (18–20) or small translations of the input (21, 22). Another such discrepancy is the vulnerability to adversarial perturbations – small changes to stimuli that alter model decisions despite being imperceptible to humans (23–26). These findings illustrate that current task-optimized models lack some of the invariances of human perception, but leave many questions unresolved. For instance, because the established discrepancies rely on only the model's output decisions, they do not reveal where in the model the discrepancies arise. It also remains unclear whether observed discrepancies are specific to supervised learning procedures that are known to deviate from biological learning. And because the known discrepancies do not point to a general method to assess model invariances in the absence of a specific hypothesis, it remains possible that current models possess many other invariances that humans lack.

In this paper we present a general test of whether the invariances present in computational models of the auditory and visual systems are also present in human perception, and apply this test to a set of contemporary and classical models. Rather than target particular known human invariances, we visualize or sonify model invariances by synthesizing stimuli that produce approximately the same activations in a model. We draw inspiration from human perceptual metamers, which have previously been characterized in the domains of color perception (27, 28), texture (29–31), cue combination (32), Bayesian decision making (33), and visual crowding (34, 35). We call the stimuli we generate "model metamers" because they are metameric for a computational model[1].

We generated model metamers from a variety of state-of-the-art deep neural network models of vision and audition by synthesizing stimuli that yield the same activations in a model stage as particular natural images or speech signals. We then evaluated human recognition of the model metamers. If the model invariances match those of humans, then humans should be able to recognize the model metamer as belonging to the same class as the natural signal to which it is matched.

---

[1] A preliminary version of some of the experiments described here was presented in a conference paper (36).

Across both visual and auditory task-optimized neural networks, metamers from late model stages were nearly always misclassified by humans, suggesting that many of their invariances are not present in human sensory systems. The same phenomenon also occurred for networks trained with unsupervised learning, demonstrating that the model failure is not specific to supervised classifiers. Model metamers could be made more recognizable to humans with selective changes to the training procedure or architecture. However, late-stage model metamers remained much less recognizable than natural stimuli in every model we tested regardless of architecture or training. Some model changes that produced more recognizable metamers did not improve conventional neural prediction metrics or evaluations of robustness, demonstrating that the metamer test provides a complementary tool to guide model improvements. The human-recognizability of a model's metamers was well predicted by other models' recognition of the same metamers, suggesting that the discrepancy with humans lies in idiosyncratic model-specific invariances. Model metamers demonstrate a qualitative gap that remains between current models of sensory systems and their biological counterparts, and provide a metric for future model evaluation.

Results

General procedure

The goal of our metamer generation procedure (Figure 1a) was to generate stimuli that produce nearly identical activations at some stage within a model, but that were otherwise unconstrained, and thus could physically differ in ways to which the model was invariant. We first measured the activations evoked by a natural image or speech signal at a particular model stage. The metamer for the natural image or speech signal was then initialized as a white noise signal (either an image or a sound waveform; white noise was chosen to sample the metamers as broadly as possible subject to the model constraints, without biasing the initialization towards a specific object class). The noise signal was then iteratively modified to minimize the difference between its activations at the model stage of interest and those for the natural signal to which it is matched. The optimization procedure performed gradient descent on the input, iteratively updating the input while holding the network parameters fixed. Model metamers can be generated in this way for any model stage constructed from differentiable operations. Because the models we considered are hierarchical, if the image or sound was matched with high fidelity at a particular stage, all subsequent stages were also matched (including the final classification layer in the case of supervised networks, yielding the same decision).

Experimental logic

The logic of our approach can be related to four sets of stimuli. For a given "reference" stimulus, there is a set of stimuli for which humans produce the same classification judgment as the reference (Figure 1b; blue shaded). A subset of these are stimuli that are indistinguishable from the reference stimulus (i.e., metameric) to human observers (blue circle). If a model performs a classification task, it will also have a set of stimuli judged to be the same category as the reference stimulus (orange shaded). However, even if the model does not itself perform classification, it could instantiate invariances that define model metamers for the reference stimulus at each model stage (orange circle).

In our experiments, we generate stimuli (sounds or images) that are metameric to a model and present these stimuli to humans performing a classification task (Figure 1c). Because we have access to the internal representations of the model, we can generate metamers for each stage of a model (i.e. sampling from the preimage of the reference stimulus at each model stage; Figure 1d). In many models there is limited invariance in the early stages (as is believed to be true of early stages of biological sensory systems (17)), with model metamers closely approximating the stimulus from which they are generated (Figure 1d, small orange sets in leftmost column). But

successive stages of a model may build up invariance, producing successively larger sets of model metamers. In a feedforward model, if two distinct inputs map onto the same representation at a given model stage, then any differences in the inputs cannot be recovered in subsequent stages, such that invariance cannot decrease from one stage to the next. If a model replicates a human sensory system, every model metamer from each stage should also be classified as the reference class by human observers (first row of Figure 1d). Such a result does not imply that all human invariances will be shared by the model, but it is a necessary condition for a model to replicate human invariances.

Discrepancies in human and model invariances could result in model metamers that are not recognizable by human observers (second row of Figure 1d). Moreover, assessing model metamers from different model stages may provide insight into how the invariances are built up over the model, and where they begin to diverge from those of humans.

Our approach differs from classical work on metamers (28) in that we do not directly assess whether model metamers are also metamers for human observers (i.e., indistinguishable). The reason for this is that a human judgment of whether two stimuli are the same or different could rely on any representations within their sensory system that distinguish the stimuli, rather than just those that are relevant to a particular behavior. By contrast, most neural network models of sensory systems are trained on a single task. As a result, we do not expect metamers of such models to be fully indistinguishable to a human observer. But if the model replicates the representations that support a particular behavioral task in humans, its metamers should nonetheless produce the same human behavioral judgment on that task, because they should be indistinguishable to the human representations that mediate the judgment. We thus use a recognition judgment as the behavioral assay of whether model metamers reflect the same invariances that are instantiated in an associated human sensory system. Moreover, if humans cannot recognize a model metamer, they would also be able to discriminate it from the reference stimulus, and the model would not pass a traditional metamerism test.

We sought to answer several questions. First, we asked whether the learned invariances of commonly used neural network models are shared by human sensory systems. Second, we asked where any discrepancies with human perception arise within models. Third, we asked whether any discrepancies between model and human invariances would also be present in models obtained without supervised learning. Fourth, we explored whether model modifications intended to improve robustness would also make model metamers more recognizable to humans. Fifth, we asked whether metamer recognition identifies model discrepancies that are not evident using other methods of model assessment, such as brain predictions or adversarial vulnerability. Sixth, we asked whether human-discrepant metamers are shared across models.

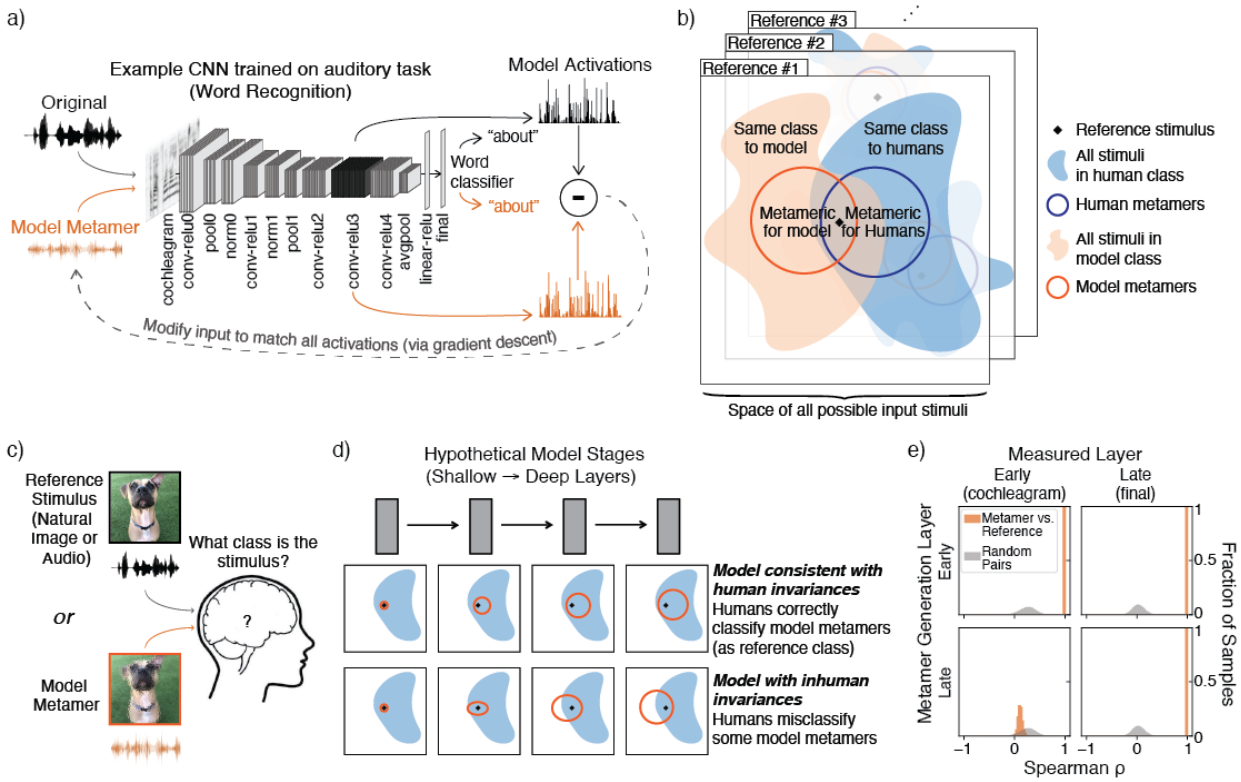


Figure 1. a) Model metamer generation. Metamers are synthesized by performing gradient descent on a noise signal to minimize the difference between its activations and those of a natural signal. The model architecture shown here is that of the CochCNN9 auditory model class used throughout the paper. b) Each reference stimulus has an associated set of stimuli in the space of all possible stimuli that are categorized as the same class by humans (blue) or by models (orange, if models have a classification decision). Metamers for humans and metamers for models are also sets of stimuli in the space of all possible stimuli (subsets of the set of same-class stimuli). c) General experimental setup. Humans make classification judgements on natural stimuli or model metamers. d) Possible scenarios for how model metamers could relate to human classification decisions. Each square panel depicts sets of stimuli in the input space. The top row depicts a model that passes our proposed behavioral test. The set of metamers for a reference stimulus grows over the course of the model, but even at the deepest stage, all model metamers are classified as the reference category by humans. The second row depicts a model whose invariances diverge from those of humans. By the late stages of the model, many model metamers are no longer recognizable by humans as the reference stimulus class. The metamer test results also constrain the model stage at which model invariances diverge from those of humans. e) Example distributions of activation similarity for pairs of metamers (a natural signal and its corresponding metamer) along with random pairs of natural signals from the training set. The latter provides a null distribution that we used to verify the success of the model metamer generation. The correlation between metamer activations in the layer used for metamer generation (top left and bottom right) falls outside the null distribution, as intended. When the metamer is generated from an early layer, the activations in a deeper layer are also well matched (top right), as expected given that the model is feedforward and deterministic. By contrast, when metamers are generated from a deep layer, activations in the deep layer are well matched, but those in the early layer are not (bottom left). This is because the model builds up invariances from layer to layer, such that metamers for the deep layer produce very different activations in an early layer.

Metamer optimization

Because the metamer generation relies on an iterative optimization procedure, it was important to measure optimization success for each metamer. We considered the metamer generation to

have succeeded only if it satisfied two conditions. First, measures of the match between the activations for the natural reference stimulus and its model metamer at the matched stage had to be much higher than would be expected by chance, quantified with a null distribution (Figure 1e, grey distributions) measured between randomly chosen pairs of examples from the training dataset. This criterion was adopted in part because it is equally applicable to networks that do not perform a task. Second, for models that performed a classification task, the metamer had to result in the same classification decision by the network as the paired natural signal. In practice we trained linear classifiers on top of all unsupervised models, such that we were able to apply this second criterion for them as well (to be conservative).

Figure 1e shows example distributions of the match fidelity (using Spearman's rho, match fidelity was also measured with two other metrics; see Methods). Activations in the matched model stage have a correlation close to 1, as intended, and are well outside the null distribution for random pairs of training examples. And as expected given the feedforward nature of the model, matching at an early layer produces matched activations in a late layer (Figure 1e, orange distributions, top row). But because the models we consider build up invariances over a series of feedforward stages, stages earlier than the matched layer need not have the same activations, and in general differ from those for the original stimulus to which the metamer was matched (Figure 1e, orange distributions, bottom row). This is particularly true for model metamers generated from late stages of the network, where many different signals can yield the same model activations.

Metamers for standard visual deep neural networks are unrecognizable to humans

We generated model metamers for multiple stages of five standard visual neural networks trained to recognize objects (using the ImageNet dataset (37)) (Figure 2a). The five network models spanned a wide range of architectural building blocks and depths: CORnet-S (38), VGG-19 (39), ResNet50 and ResNet101 (40), and AlexNet (41). These models have been posited to capture similar features as primate visual representations, and at the time the experiments were run, they respectively placed 1st, 2nd, 11th, 4th, and 59th on a neural prediction benchmark (38, 42). To evaluate human recognition of the model metamers, humans performed a 16-way categorization task on the natural stimuli and model metamers (Figure 2b) (18). In networks with residual connections, we only generated metamers at layers where all branches converge, which ensured that all subsequent model stages, and the model decision, remained matched.

Contrary to the idea that the trained networks have learned human-like invariances, we found that human recognition of the model metamers decreased across model stages, reaching near-chance performance at the deepest stages even though the model metamers remained as recognizable to the networks as the corresponding natural stimuli, as intended (Figure 2c). This reduction in human recognition of model metamers was evident as a main effect of observer (CORnet-S: $F(1,42)=424.4$, $p<0.0001$; VGG-19: $F(1,42)=1612.1$, $p<0.0001$; ResNet50: $F(1,42)=554.9$, $p<0.0001$; ResNet101: $F(1,42)=1001.7$, $p<0.0001$; AlexNet: $F(1,42)=935.4$, $p<0.0001$), and an interaction between the effect of metamer generation stage and the observer (CORnet-S: $F(5,210)=293.6$, $p<0.0001$; VGG-19: $F(9,378)=268.4$, $p<0.0001$; ResNet50: $F(7,294)=290.2$, $p<0.0001$; ResNet101: $F(7,294)=345.8$, $p<0.0001$; AlexNet: $F(8,336)=195.0$, $p<0.0001$). From visual inspection, many of the metamers from late stages resemble noise rather than natural images (Figure 2d; see Supplementary Figure 1a for examples of metamers generated from different white noise initializations). Although the specific optimization strategies we used had some effect on the subjective appearance of the model metamers, human recognition of the generated stimuli remained poor regardless of the optimization procedure (Supplementary Figure 2).

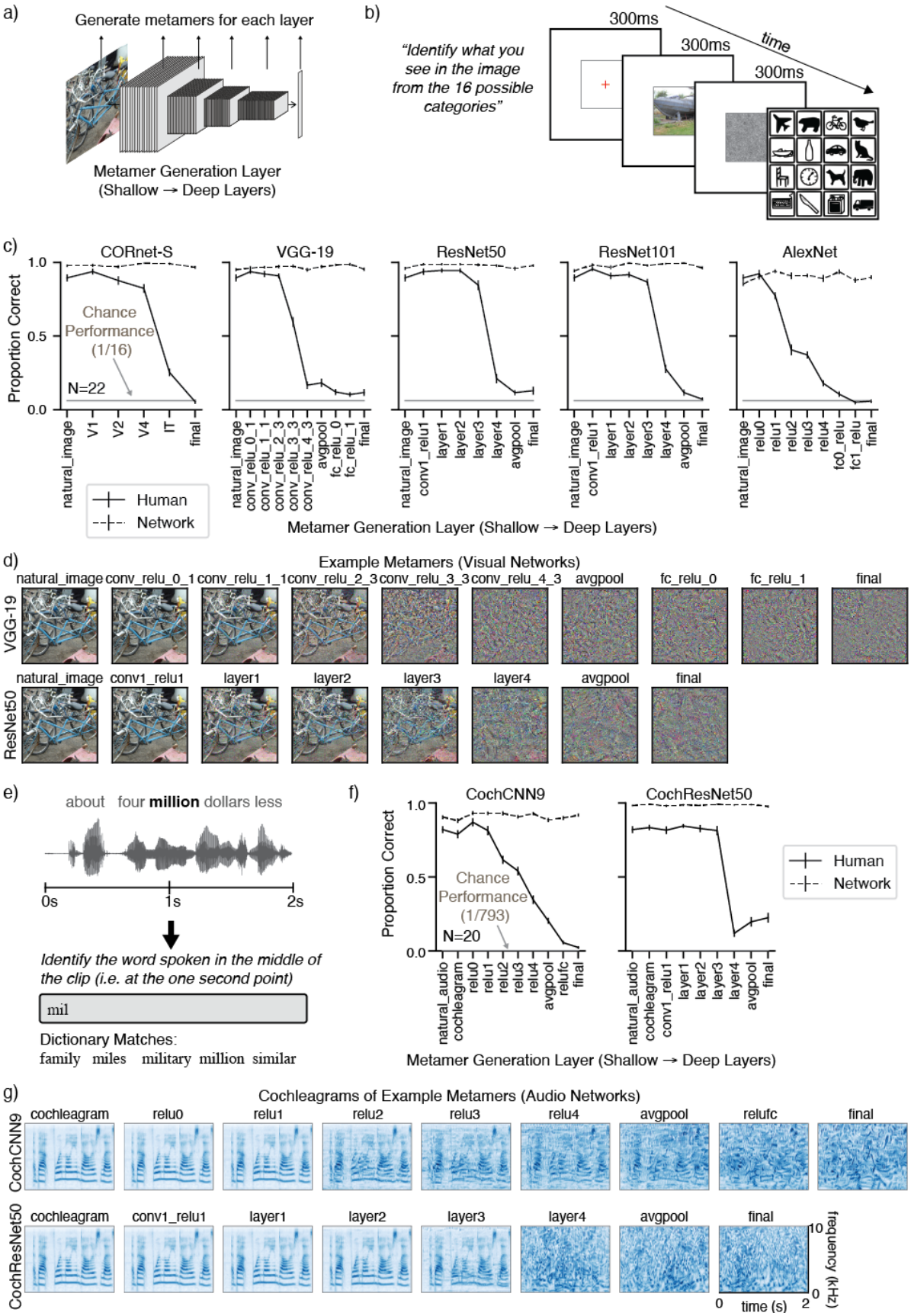

a)
Generate metamers for each layer
Metamer Generation Layer
(Shallow → Deep Layers)
b)
"Identify what you see in the image from the 16 possible categories"
300ms
300ms
300ms
time
c)
CORnet-S
VGG-19
ResNet50
ResNet101
AlexNet
Proportion Correct
1.0
0.5
0.0
Chance Performance (1/16)
N=22
natural_image
V1
V2
V4
IT
final
conv_relu_0_1
conv_relu_1_1
conv_relu_2_3
conv_relu_3_3
conv_relu_4_3
avgpool
fc_relu_0
fc_relu_1
conv1_relu1
layer1
layer2
layer3
layer4
relu0
relu1
relu2
relu3
relu4
fc0_relu
fc1_relu
Human
Network
Metamer Generation Layer (Shallow → Deep Layers)
d)
Example Metamers (Visual Networks)
VGG-19
ResNet50
e)
about four **million** dollars less
0s
1s
2s
Identify the word spoken in the middle of the clip (i.e. at the one second point)
mil
Dictionary Matches:
family miles military million similar
f)
CochCNN9
CochResNet50
Proportion Correct
Chance Performance (1/793)
N=20
natural_audio
cochleagram
relufc
Human
Network
Metamer Generation Layer (Shallow → Deep Layers)
g)
Cochleagrams of Example Metamers (Audio Networks)
CochCNN9
CochResNet50
time (s)
frequency (kHz)
10
0

Figure 2 (previous page). Metamers of standard-trained visual and audio deep neural networks are often unrecognizable to human observers. a) Model metamers are generated from different stages of the model. b) Experimental task used to assess human recognition of visual model metamers. Humans were presented with an image (either a natural image or a model metamer of a natural image) followed by a noise mask. They were then presented with 16 icons representing 16 object categories, and classified each image as belonging to one of these 16 categories by clicking on the icon. c) Human recognition of image model metamers (N=22). For all tested models, human recognition of model metamers declined for late model stages, while model metamers remained recognizable to the model (as expected). Error bars plot SEM across participants (or participant matched stimulus subsets for model curves) d) Example metamers from VGG-19 and ResNet50 visual models. e) Experimental task used to assess human recognition of auditory model metamers. Humans classified the word that was present at the midpoint of a two-second sound clip. Participants selected from 793 possible words by typing any part of the word into a response box and seeing matching dictionary entries from which to complete their response. A response could only be submitted if it matched an entry in the dictionary. f) Human recognition of audio model metamers (N=20). For both tested models, human recognition of model metamers decreased at late model stages. By contrast, model metamers remain recognizable to the audio models, as expected. When plotted, chance performance (1/793) is indistinguishable from the x-axis. Error bars plot SEM across participants (or participant matched stimulus subsets for model curves). g) Cochleagram visualizations of example audio model metamers from CochCNN9 and CochResNet50 architectures. Color intensity denotes instantaneous sound amplitude in a frequency channel (arbitrary units)

Metamers for standard auditory deep neural networks are unrecognizable to humans

To investigate whether this phenomenon generalized across modalities, we performed an analogous experiment with two audio neural networks trained to recognize speech (the word recognition task in the Word-Speaker-Noise dataset (36)). Each model consisted of a biologically-inspired "cochleagram" representation with parameters matched to estimates of the frequency tuning of the human ear (43, 44) followed by a convolutional neural network whose parameters were optimized during training. We tested two network architectures: a ResNet50 architecture (referred to here as CochResNet50) and a network with nine layers (including five convolutional layers) similar to that used in a previously published auditory neural network model (5), referred to here as CochCNN9. Model metamers were generated for clean speech examples from the validation set (Figure 1a). Humans performed a 793-way classification task (5) to identify the word in the middle of either a natural speech example or a model metamer of a speech example generated from a particular model stage (Figure 2e; humans typed responses but were only allowed to enter one of the 793 words from the models' word-recognition task).

As with the image recognition models, human recognition of the auditory model metamers decreased markedly at late model stages for both architectures (Figure 2f). As with the vision models, there was a significant main effect of human vs. model observer (CochCNN9: $F(1,38)=551.9$, $p<0.0001$; CochResNet50: $F(1,38)=467.8$, $p<0.0001$) and a significant interaction between effect of metamer generation stage and the observer (CochCNN9: $F(9,342)=189.2$, $p<0.0001$; CochResNet50: $F(8,304)=227.4$, $p<0.0001$). Subjectively, the model metamers from deeper layers sound like noise (and appear noise-like when visualized as cochleagrams; Figure 2g). This result suggests that many of the invariances present in these models are not invariances for the human auditory system.

Overall, these results demonstrate that the invariances of many common visual and auditory neural networks are substantially misaligned with those of human perception, even though these models are currently the best predictors of brain responses in each modality.

Metamers of unsupervised deep network models are also unrecognizable to humans

It is widely appreciated that the learning procedures used to train most contemporary deep neural networks deviate markedly from the learning that occurs in biological systems (45). Perhaps the

most fundamental difference lies in supervised learning, as biological systems are not provided with explicit category labels for millions of stimuli during development. Do the divergent invariances evident in neural network models result in some way from supervised training? Metamers are well-suited to address this question given that their generation is not dependent on a classifier, and thus can be generated for any sensory model.

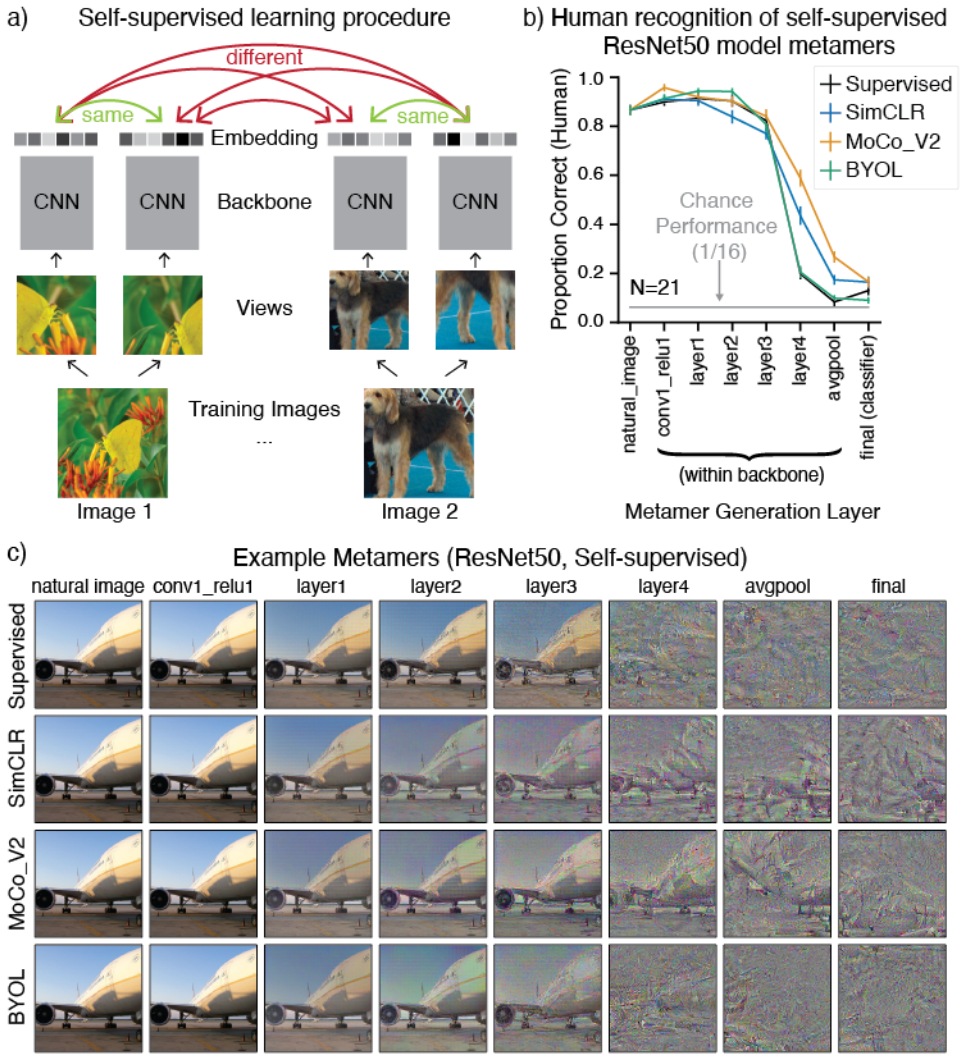


Figure 3. Metamers of self-supervised neural network models are unrecognizable to humans. a) Overview of self-supervised learning, adapted from (46). Models were trained to map multiple views of the same image to nearby points in the embedding space. The SimCLR and MoCo_V2 models also had an additional training objective that explicitly pushed apart embeddings from different images. b) Human recognition of metamers from supervised and self-supervised models (N=21). Self-supervised models (SimCLR, MoCo_V2, and BYOL) used a ResNet50 backbone; the comparison supervised model was also a ResNet50. For self-supervised models, the "final" stage was a linear classifier trained on the ImageNet task on the avgpool representation (intended to validate the representations learned via self-supervision, but included here for completeness). Model recognition curves were close to ceiling as in Figure 2, and are omitted here and in later figures for brevity. Error bars plot SEM across participants. c) Example metamers from select stages of ResNet50 supervised and self-supervised models. In all models, late-stage metamers are mostly unrecognizable.

Recent advances in certain types of unsupervised learning have produced neural networks whose representations support classification tasks without requiring millions of labeled examples during training (46). The leading such models are self-supervised, being trained with a loss function favoring representations in which variants of a single training example (different crops of an image, for instance) are similar while those from different training examples are not (Figure 3a). To assess whether this unsupervised training produces more human-like invariances compared to traditional supervised training, we generated model metamers for three such models, all with a ResNet50 architecture: SimCLR (46), MoCo_V2 (47) and BYOL (48). After training, we verified that the learned representations were sufficient to support object classification performance (on ImageNet) by training a linear classifier on the final average pooling layer (without changing any of the model weights). We measured human recognition of metamers from these models as well as those of the same ResNet50 architecture fully trained with supervision (on the ImageNet task).

As shown in Figure 3b&c, the unsupervised networks (SimCLR, MoCo_V2 and BYOL) produced similar results to those for supervised networks: human recognition of model metamers declined at deeper model stages, approaching chance levels for the final stages. Two of the self-supervised networks had somewhat more recognizable metamers at intermediate stages, with MoCo_V2 producing the largest boost (significant interaction between model type and model stage, $F(21, 420)=16.0$, $p<0.0001$). However, recognition was low in absolute terms, with the metamers bearing little resemblance to the original image they were matched to. Overall, the results suggest that the failure of standard neural network models to pass our metamer test is not specific to the supervised training procedure. This result also demonstrates the generality of the metamers method, as it can be applied to models that do not have a behavioral read-out.

Model metamers reveal invariances of classical hierarchical models of sensory systems

As a further demonstration of the generality of the model metamer method, we generated metamers for classical visual and auditory models that were designed by hand based on neuroscience and engineering principles. Although these models do not perform classification tasks as well as contemporary neural network models, their comparative simplicity might be hypothesized to yield more human-like invariances.

The HMAX vision model is a biologically-motivated architecture with cascaded filtering and pooling operations inspired by simple and complex cells in the primate visual system and was intended to capture aspects of biological object recognition (4, 16). We generated model metamers by matching all units at the S1, C1, S2, or C2 stage of the model (Figure 4a). Although HMAX is significantly shallower than the neural network models investigated in previous sections, it is evident that by the C2 model stage its model metamers are comparably unrecognizable to humans (Figure 4b&c; significant main effect of model stage, $F(4,76)=351.9$, $p<0.0001$). This classical model thus also has invariances that differ from those of the human object recognition system.

We performed an analogous experiment on a classical model of auditory cortex, consisting of a set of spectrotemporal filters applied to a cochleagram representation (Spectemp model) (49). We used a version of the model in which the convolutional responses are summarized with the mean power in each filter (5, 50, 51) (Figure 4d). Metamers from the first two stages were fully recognizable, and very similar to the original audio, indicating that these stages instantiate few invariances, as expected for overcomplete filter bank decompositions (Figure 4e&f). By contrast, metamers from the mean power representation were unrecognizable (significant main effect of model stage $F(4,76)=515.3$, $p<0.0001$), indicating that this model stage produces invariances that humans do not share (plausibly because the human speech recognition system retains information that is lost in the averaging stage).

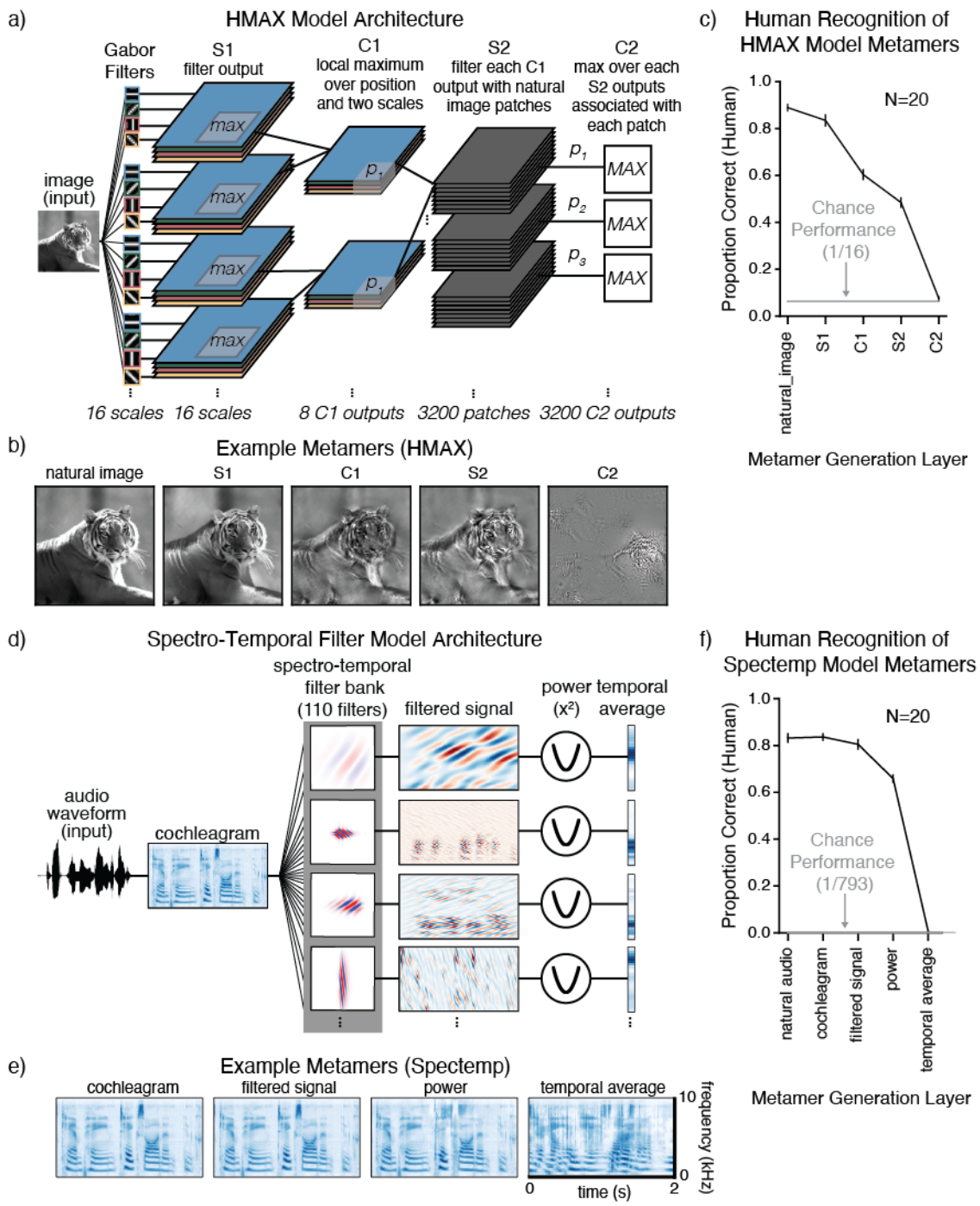


Figure 4. Metamers from classical models of sensory systems. a) Schematic of HMAX vision model, adapted from (4). b) Example HMAX model metamers. c) Human recognition of HMAX model metamers (N=20). Metamers generated from the HMAX model are recognizable at early model stages but become unrecognizable to humans by the C2 model stage. Error bars plot SEM across participants. d) Schematic of spectro-temporal auditory filterbank model (Spectemp), adapted from (49). e) Cochleagrams of example Spectemp model metamers. f) Human recognition of Spectemp model metamers (N=20). Metamers generated from the Spectemp model are recognizable at early model stages but become unrecognizable by the final (temporal average of filter power) stage. Error bars plot SEM across participants

Overall, these results show how metamers can reveal the invariances present in classical models as well as state-of-the-art deep neural networks, and demonstrate that both types of models fail to fully capture the invariances of biological sensory systems.

Adversarial training of visual models increases human recognition of model metamers

A known peculiarity of contemporary artificial neural networks is their vulnerability to small adversarial perturbations designed to change the class label assigned by a model (23–26, 52). Such perturbations are typically imperceptible to humans due to their small magnitude, but can drastically alter model decisions, and have been the subject of intense interest in part due to the security risk they pose for machine systems. One way to reduce this vulnerability is to generate perturbation-based adversarial examples during training and add them to the training set, forcing the network to learn to recognize the perturbed images as the "correct" human-interpretable class (Figure 5a) (53). This procedure has been found to yield networks that are less susceptible to adversarial examples for reasons that remain debated (54).

We asked whether reducing adversarial vulnerability in this way would improve human recognition of model metamers. A priori it was not clear what to expect. Perturbation-based adversarial examples can be viewed as the converse of a model metamer, in that they are generated by perturbations to which humans are invariant but which cause changes in model decisions. Making models robust to adversarial perturbations causes them to exhibit more of the invariances of humans (the shaded orange covers more of the blue outline in Figure 1b), but it is not obvious that this will reduce the model invariances that are not shared by humans (i.e., to decrease the orange outlined regions that don't overlap with blue shaded region in Figure 1b). Previous work visualizing latent representations of visual neural networks suggested that robust training might make model representations more human-like (55), but human recognition of model metamers had not been behaviorally evaluated.

We first generated model metamers for vision models trained to be adversarially robust (ImageNet ResNet50 and AlexNet architectures) (55). As a control, we also trained models with equal magnitude perturbations in random, rather than adversarial, directions. Such random perturbations are typically ineffective at preventing adversarial attacks (52). As intended, networks with adversarial perturbations during training were more robust to adversarial examples than the standard network or networks with random perturbations (Supplementary Figure 3a&b).

For both architectures, metamers for the robust models were significantly more recognizable than those from the standard model (Figure 5b-d; Supplementary Figure 1), evident as a significant main effect of training type (repeated measures ANOVAs comparing human recognition of standard and each adversarial model, significant main effect in each of the five models, $F(1,19)>104.61$, $p<0.0001$). Training with random rather than adversarial perturbations did not yield the same benefit (significant main effect of random vs. adversarial for each perturbation of the same ε type and size, $F(1,19)>121.38$, $p<0.0001$). Model metamers were more recognizable to humans for some adversarial training variants than others, but all variants that we tried produced a human recognition benefit. It was nonetheless the case that metamers from all variants remained less than fully recognizable to humans when generated from the deep layers. We note that performance is inflated by the use of a 16-way alternative-force-choice (AFC) task, for which above-chance performance is possible even with severely distorted images.

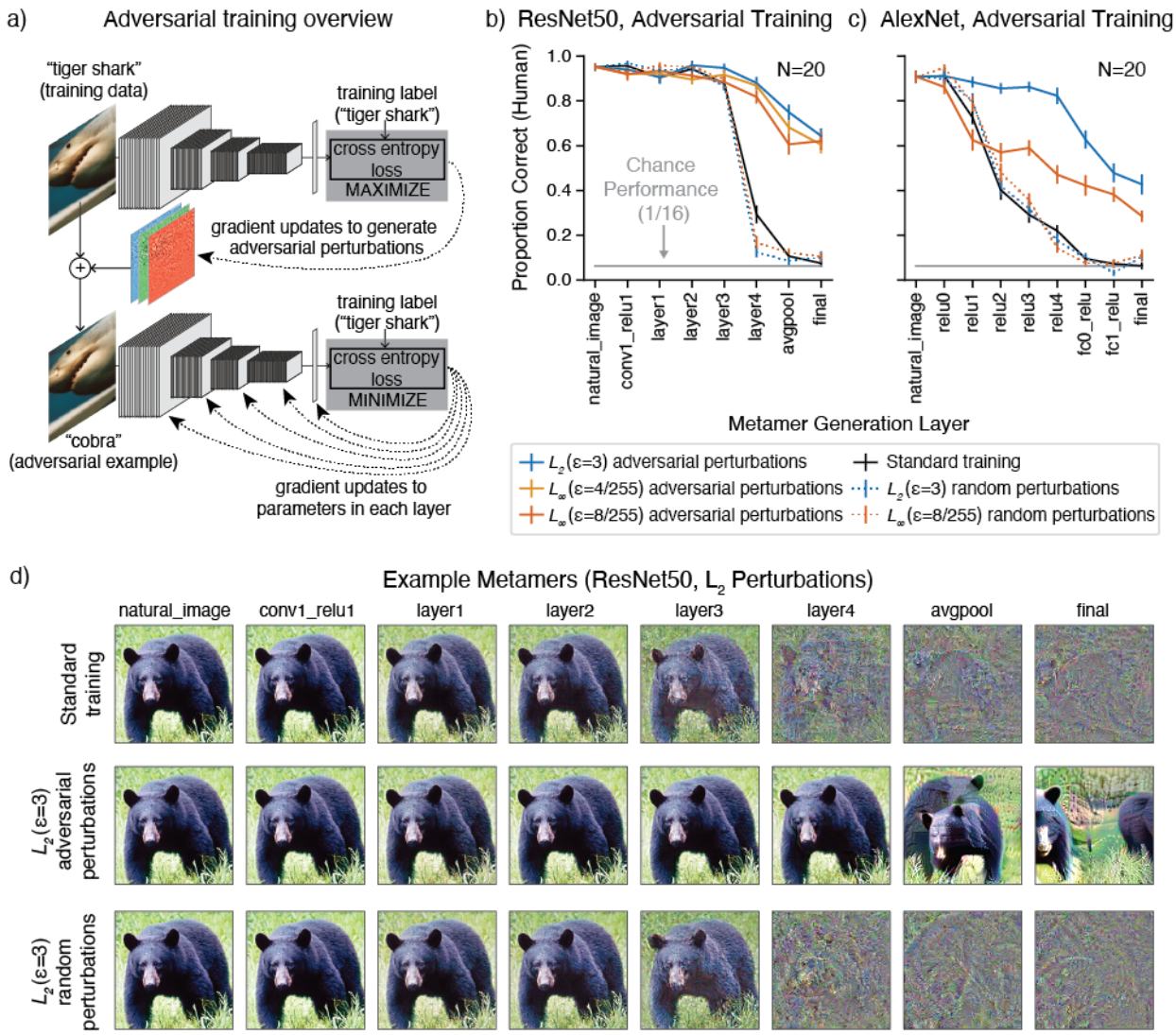


Figure 5. Adversarial training increases human recognition of visual model metamers. a) Explanation of adversarial training. Adversarial examples are derived at each training step by finding an additive perturbation to the input that moves the classification label away from the training label class (top). These derived adversarial examples are then provided to the model as training examples and used to update the model parameters (bottom). The resulting model learns to classify the adversarial examples as the training data label, and is subsequently more robust to adversarial perturbations than if standard training were used. As a control experiment, we also trained models with random perturbations to the input rather than adversarial perturbations. b) Human recognition of visual model metamers on a 16-way classification task with metamers generated from ResNet-50 models (N=20) trained with and without adversarial or random perturbations. Error bars plot SEM across participants. c) Same as c, but for AlexNet models (N=20). In both ResNet50 and AlexNet models, adversarial training leads to more recognizable metamers at the deep layers of the networks, though in both cases the metamers remain less than fully recognizable. d) Example visual model metamers for ResNet50 networks trained with and without adversarial or random L_2 perturbations.

Adversarial training of auditory models increases human recognition of model metamers

To investigate whether adversarial training also leads to more human-like auditory representations, we trained CochResNet50 and CochCNN9 architectures on the word recognition task described above, using both standard and adversarial training. Because the auditory models contain a fixed cochlear stage at their front end, there are two natural places to generate

adversarial examples (they can be added to the waveform or the cochleagram), and we explored both for completeness.

We first investigated the effects of adversarial perturbations to the waveform (Figure 6a, Supplementary Figure 3c&d). As a control experiment, we again trained networks with random, rather than adversarial, perturbations. As with the visual models, human recognition was generally better for model metamers from adversarially trained networks (Figure 6b&c; ANOVAs comparing standard and adversarial models, significant main effect in 4/5 cases: $F(1,19)>9.26$, $p<0.0075$; no significant main effect for CochResNet50 with L_2 (ε=1) perturbations: $F(1,19)=0.29$, $p=0.59$). This benefit again did not occur for models trained with random perturbations (ANOVAs comparing each random and adversarial perturbation model with the same ε type and size; significant main effect in each case: $F(1,19)>4.76$, $p<0.0444$). The model metamers from the robust models are visibly less noise-like when viewed in the cochleagram representation (Figure 6d).

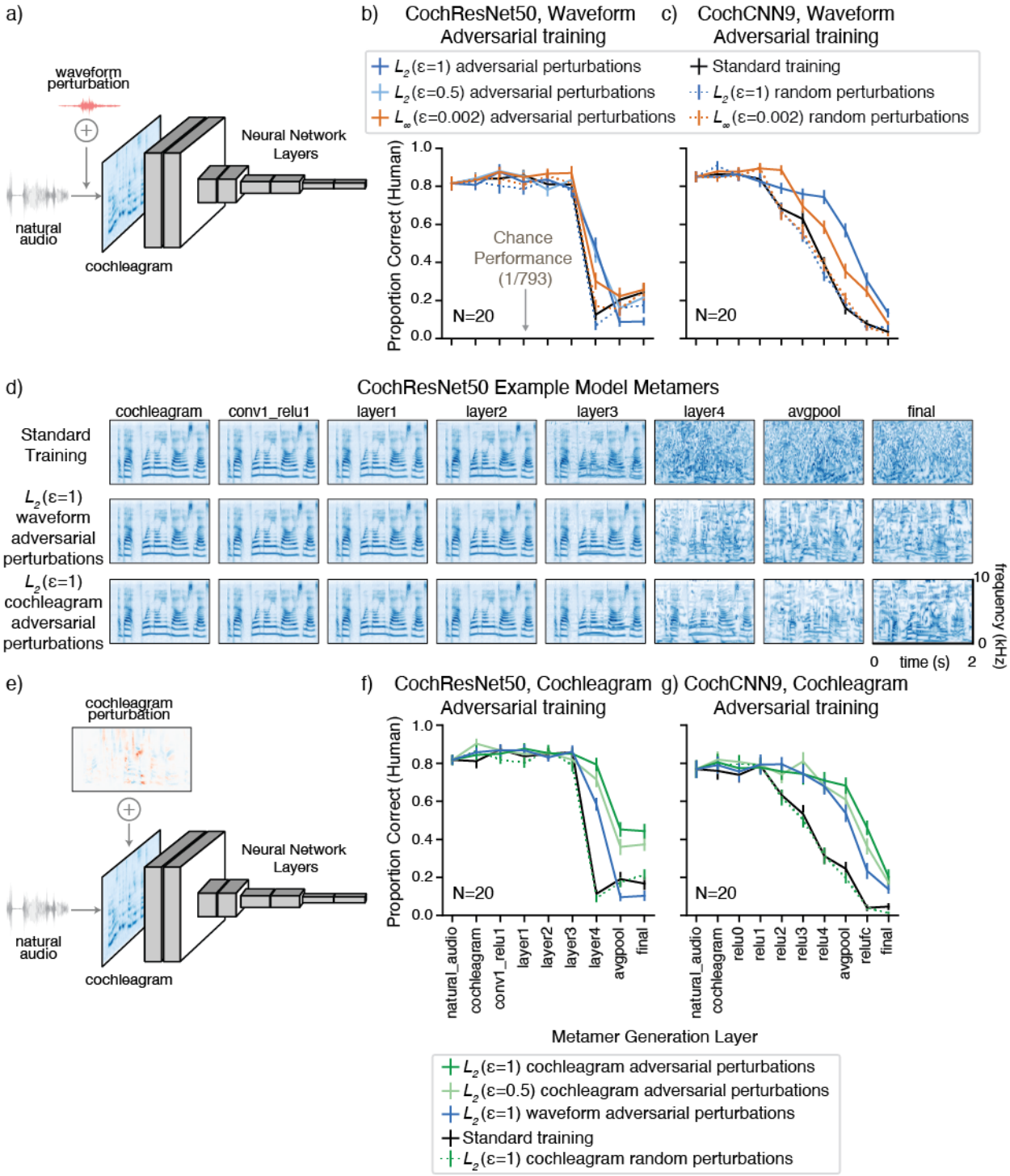
We also trained models with adversarial perturbations to the cochleagram representation, whose fixed components enabled norm-based constraints on the perturbation size analogous to those used for input-based adversarial examples (Figure 6e). Models trained on cochleagram adversarial examples had significantly more recognizable metamers than both the standard models and the models adversarially trained on waveform perturbations (Figure 6f&g; ANOVAs comparing each model trained with cochleagram perturbations vs. the same architecture trained with waveform perturbations; significant main effect in each case: $F(1,19)>4.6$, $p<0.04$; ANOVAs comparing each model trained with cochleagram perturbations to the standard model; significant main effect in each case: $F(1,19)>102.25$, $p<0.0001$). The effect on metamer recognition was again specific to adversarial perturbations (ANOVAs comparing effect of training with adversarial vs. random perturbations with the same ε type and size: $F(1,19)>145.07$, $p<0.0001$).

Although the perturbation sizes in the two model stages are not directly comparable, in each case we chose the size to be large enough that the model showed robustness to adversarial perturbations while not being so large that the model could not perform the task (as is standard for adversarial training). Further, training networks with perturbations generated at the cochleagram stage resulted in substantial robustness to adversarial examples generated at the waveform (Supplementary Figure 3e&f). These results suggest that the improvements from intermediate-stage perturbations may in some cases be more substantial than those from perturbations to the input representation. They also highlight the utility of model metamers for evaluating model modifications, in this case adversarial training with perturbations to intermediate model representations.

Overall, these results suggest that training alterations intended to mitigate undesirable characteristics of artificial neural networks (adversarial vulnerability) can cause their invariances to become more like those of humans in both the visual and auditory domain. However, substantial discrepancies remain even with these alterations. In both modalities, many model metamers from the deep layers remain unrecognizable even after adversarial training.

Figure 6 (next page). Adversarial training increases human recognition of audio model metamers. a) Schematic of auditory convolutional neural networks with adversarial perturbations applied to the waveform input. b) Human recognition of auditory model metamers from CochResNet50 (N=20) and c) CochCNN9 (N=20) models with adversarial perturbations generated in the waveform space (models trained with random perturbations are also included for comparison). When plotted here and in f&g, chance performance (1/793) is indistinguishable from the x-axis and error bars plot SEM across participants. L_2 (ε=0.5) waveform adversaries were only included in the CochResNet50 experiment. d) Cochleagrams of example model metamers from CochResNet50 models trained with waveform and cochleagram adversarial perturbations.

e) Schematic of auditory convolutional neural networks with adversarial perturbations applied to the cochleagram stage. f) Human recognition of auditory model metamers from networks trained with cochleagram adversarial perturbations are more recognizable for CochResNet50 and g) CochCNN9 models compared to those from models trained with waveform perturbations.



Human recognition of metamers dissociates from adversarial vulnerability

Although we found that adversarial training increased the human recognizability of model metamers, we also found examples where a model's susceptibility to adversarial examples was not predictive of the recognizability of its metamers. Here we present results for two models that had similar adversarial robustness, one of which had much more recognizable metamers than the other.

The first model was a CNN that was modified to reduce aliasing (LowpassAlexNet). Because many traditional neural networks contain downsampling operations (e.g. pooling) without a preceding lowpass filter, they violate the sampling theorem (36, 56) (Figure 7a). It is nonetheless possible to modify the architecture to reduce aliasing, and such modifications have been suggested to improve network robustness to small image translations (21, 22). The second model was a CNN that contained an initial processing block inspired by primary visual cortex in primates (57, 58) featuring hard-coded Gabor filters and a stochastic response component present during training (VOneAlexNet, Figure 7b). The inclusion of this block has been previously demonstrated to increase adversarial robustness. It was a priori unclear whether either model modification would improve human recognizability of the model metamers.

Both architectures were comparably robust to adversarial perturbations (Figure 7c; no significant main effect of architecture for perturbations, $F(1,8)<4.5$, $p>0.10$ for all perturbation types), and both were more robust than the standard AlexNet (main effect of architecture for perturbations $F(1,8)>137.4$, $p<0.031$ for all adversarial perturbation types for both VoneAlexNet and LowPassAlexNet). However, metamers generated from LowpassAlexNet were substantially more recognizable than metamers generated from VoneAlexNet (Figure 7d,e; main effect of architecture: $F(1,19)=71.7$, $p<0.0001$; interaction of architecture and model stage: $F(8,152)=21.8$, $p<0.0001$). This result empirically demonstrates that human recognizability of model metamers and susceptibility to adversarial examples are dissociable. The model metamer test thus provides a complementary comparison of model invariances that can differentiate models even when adversarial robustness does not.

These results may be understood in terms of the four types of stimulus sets originally shown in Figure 1b, four configurations of which are depicted in Figure 7f. Adversarial examples are stimuli that are metameric to a reference stimulus for humans but that are classified differently from the reference stimulus by a model. Adversarial robustness thus corresponds to a situation where the human metamers for a reference stimulus fall completely within the set of stimuli that are given the reference class label by a model (blue outline contained within orange shaded region in 7f, right column). This condition does not imply that all model metamers will be recognizable to humans (orange outline contained within the blue shaded region, top row). These theoretical observations motivate the use of model metamers as a complementary model test, and are confirmed by the empirical observations of this section. The effect of adversarial training on model metamers (Figures 5&6) thus appears to be somewhat specific to the training method – a model's adversarial robustness is not always predictive of the human recognizability of its metamers.

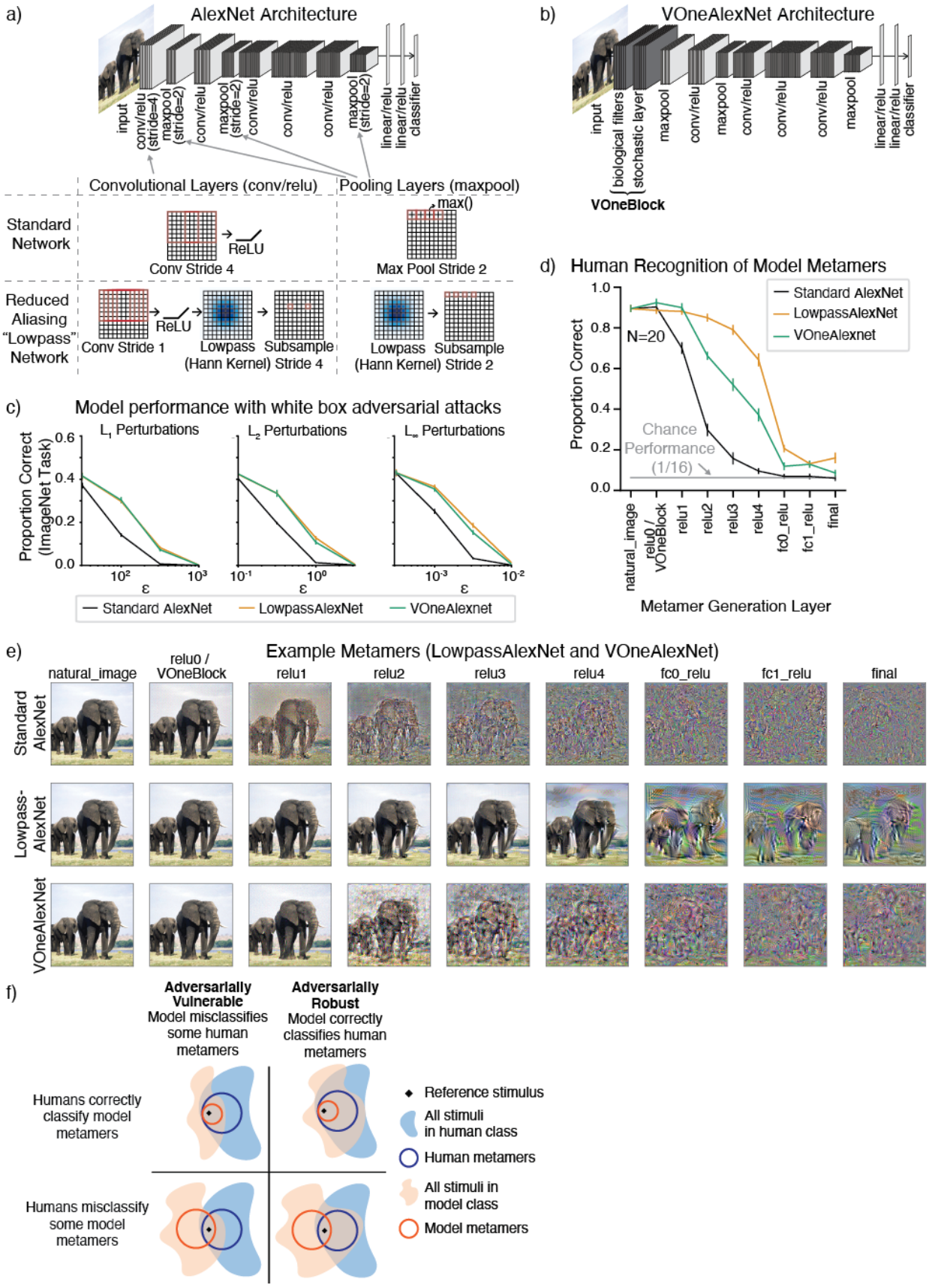

a)
AlexNet Architecture
input
conv/relu (stride=4)
maxpool (stride=2)
conv/relu
maxpool (stride=2)
conv/relu
conv/relu
conv/relu
maxpool (stride=2)
linear/relu
linear/relu
classifier
Convolutional Layers (conv/relu)
Pooling Layers (maxpool)
Standard Network
ReLU
Conv Stride 4
max()
Max Pool Stride 2
Reduced Aliasing "Lowpass" Network
Conv Stride 1
Lowpass (Hann Kernel)
Subsample Stride 4
Lowpass (Hann Kernel)
Subsample Stride 2
b)
VOneAlexNet Architecture
input
biological filters
stochastic layer
VOneBlock
maxpool
conv/relu
maxpool
conv/relu
conv/relu
conv/relu
maxpool
linear/relu
linear/relu
classifier
c)
Model performance with white box adversarial attacks
Proportion Correct (ImageNet Task)
L1 Perturbations
L2 Perturbations
L∞ Perturbations
ε
Standard AlexNet
LowpassAlexNet
VOneAlexnet
d)
Human Recognition of Model Metamers
Proportion Correct
N=20
Chance Performance (1/16)
Standard AlexNet
LowpassAlexNet
VOneAlexnet
natural_image
relu0 / VOneBlock
relu1
relu2
relu3
relu4
fc0_relu
fc1_relu
final
Metamer Generation Layer
e)
Example Metamers (LowpassAlexNet and VOneAlexNet)
natural_image
relu0 / VOneBlock
relu1
relu2
relu3
relu4
fc0_relu
fc1_relu
final
Standard AlexNet
Lowpass-AlexNet
VOneAlexNet
f)
Adversarially Vulnerable
Model misclassifies some human metamers
Adversarially Robust
Model correctly classifies human metamers
Humans correctly classify model metamers
Humans misclassify some model metamers
Reference stimulus
All stimuli in human class
Human metamers
All stimuli in model class
Model metamers

Figure 7 (previous page). Human recognition of model metamers dissociates from adversarial vulnerability. a) Operations included in AlexNet architecture to reduce aliasing. Strided convolutions were replaced with a sequence of four operations: convolution with a learnable kernel using a stride of 1, ReLU, convolution with a fixed lowpass filter, and subsampling with the original convolution stride. Max Pooling was replaced with a sequence of two operations: convolution with a fixed lowpass filter, and subsampling with the original pooling stride. The network with these modified operations more closely obeys the sampling theorem and is referred to as "LowpassAlexNet". b) Schematic of VOneBlock input to AlexNet backbone architecture. The VOneBlock includes a layer of fixed Gabor filters, and a stochastic layer with Gaussian noise. The stochastic layer was turned on for training but turned off for evaluations of adversarial robustness and for metamer generation. c) Adversarial vulnerability, assessed via recognition accuracy on 1000-way ImageNet classification task with adversarial perturbations of different sizes added to the images. LowpassAlexNet and VOneAlexNet both produce a modest increase in adversarial robustness (greater accuracy for large adversarial perturbations) compared to the standard-trained AlexNet model. Error bars plot SEM across five subsets of training images. D) Human recognition of model metamers generated from LowpassAlexNet, VoneAlexNet and standard AlexNet models on the 16-way classification task (N=20). LowpassAlexNet has substantially more recognizable model metamers than the VOneNet, demonstrating a dissociation between human recognition of model metamers and a model's susceptibility to adversarial examples. Error bars plot SEM across participants. e) Example model metamers from experiment in (d). f) Human-recognizability of model metamer is dissociable from adversarial robustness. Adversarial examples are cases where a model misclassifies stimuli that are metameric to humans (left row). This schematic depicts a reference stimulus for which the model gives the correct class label, such that the adversarial perturbation induces a classification error. Model metamers may be unrecognizable to humans even if the model is robust to adversarial examples (bottom right). Tests involving model metamers are thus complementary to those involving adversarial examples.

Standard evaluation metrics do not capture differences between auditory models

Are the differences between models shown by the metamer test similarly evident when using standard neural evaluation benchmarks (5)? To address this question, we used such benchmarks to evaluate the auditory models from the adversarial-training experiments described above. We chose to make this comparison with auditory models because we had access to a large data set of human auditory cortical responses (59) that had previously been used to evaluate neural network models of the auditory system (fMRI responses to a large set of natural sounds). We assessed predictions of the responses of individual voxels using cross-validated regularized linear regression applied to the network activations to the same set of natural sounds (Figure 8a). We used the models trained with cochleagram adversarial perturbations as these produced the largest effect on model metamer recognizability (Figure 8b).

The brain predictions of the two types of neural network models were overall similar (Figure 8c&d), with both explaining significantly more variance than a single-layer baseline model (the Spectemp model from Figure 4d; permutation test, $p<0.004$ for all three models). There was a small but statistically significant difference between the brain predictions of adversarially trained and standard models, but the difference was in the opposite direction as the metamer recognizability difference (permutation test for standard model larger than adversarial robust model; $p<0.014$ for all comparisons). Results were consistent across different regions of interest defined by selectivity for speech, music, pitch, or audio frequency (neural network models were better than Spectemp model, $p<0.004$ in all cases, and robust models were never better than standard models, $p>0.062$ in all cases). These results contrast with those of the metamer test, which showed a relatively large benefit from adversarially-robust training (compare Figure 8b&c), and indicate that the metamer test is complementary to traditional metrics of model-brain fit.

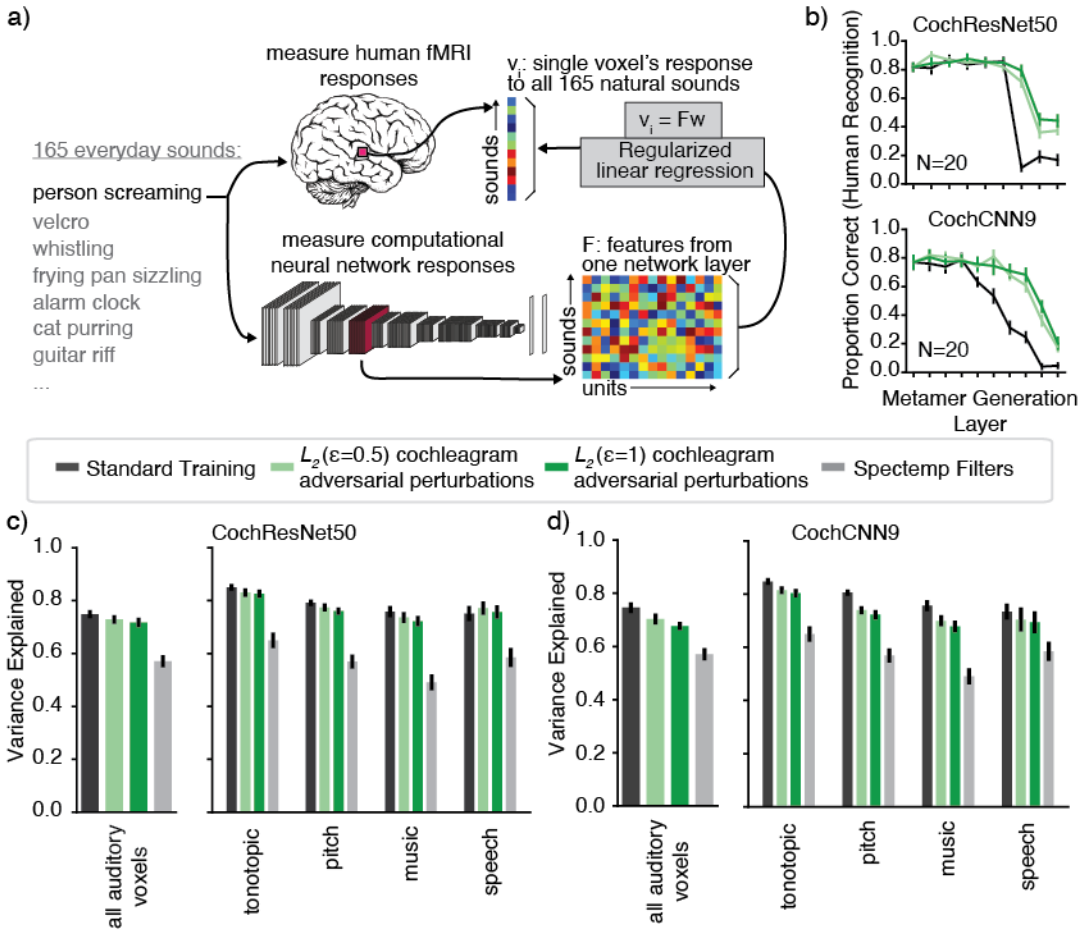


Figure 8. Human recognition of model metamers dissociates from model predictions of fMRI responses. a) 165 natural sounds presented to humans in an fMRI experiment (59) were also presented to neural network models. The model's time-averaged unit responses in each layer were used to predict each auditory cortical voxel's response by a linear mapping fit to the responses to a subset of the sounds with ridge regression. For each neural network model, the best-predicting layer was selected for each participant using independent data. Model predictions were evaluated on a set of held-out sounds. Schematic of procedure is reproduced from (5). b) Human metamer recognition results on the 793-way word classification task for neural network models included in fMRI analysis, reproduced from Figure 6e&f for ease of comparison with c and d. Recognition is better for metamers from models trained with adversarial perturbations. c) Average voxel response variance explained by CochResNet50 models across all voxels in auditory cortex (left) and a selection of auditory functional ROIs (right). The Variance Explained (R^2) was measured for the best-predicting layer of the neural network models, chosen individually for each participant and each voxel subset. For each participant, the other participants' data was used to choose the layer that yielded the best predictions, and then the R^2 from this layer for the held-out participant was included in the average. Error bars plot SEM across participants. d) Same as c, but for CochCNN9 models. Models trained with adversarial perturbations explain about the same amount of fMRI response variance despite exhibiting more human-recognizable model metamers (compare to b).

Human recognition of a model's metamers is predicted by their recognizability to other models

Are one model's metamers recognizable by other models? We addressed this issue by taking all the models we trained for one modality, holding one model out as the "generation" model, and then presenting its metamers to each of the other models ("recognition" models), measuring the accuracy of their class predictions (Figure 9a). We repeated this procedure with each model as the generation model. Accuracy for all combinations of recognition and generation models is

shown in Supplementary Figures 4 and 5. As a summary measure for each generation model, we averaged the accuracy across the recognition models (Figure 9a, right). Because we had trained the ResNet50 and CochResNet50 architectures with several variants of self-supervised and adversarial training in addition to standard supervised training, we performed further analysis of these architectures (Figure 9b&c).

In the vision models, metamers from deep stages of the standard supervised trained ResNet50 were generally not recognized by other models (Figure 9b, blue), suggesting that many of the invariances in this model are not present in the other models. A similar trend held for the models trained with self-supervision (Figure 9b, orange). By contrast, metamers from the adversarially trained models were more recognizable to other models (Figure 9b, green). We saw an analogous metamer transfer boost from the model with reduced aliasing (LowPassAlexNet), for which metamers for intermediate stages were more recognizable to other models (Supplementary Figure 6). Similar results held for audio models (Figure 9c), though metamers from the standard supervised CochResNet50 transferred better to other models than did those for the supervised vision model, perhaps due to the shared cochlear representation present in all auditory models, which could increase the extent of shared invariances.

These results suggest that models tend to contain idiosyncratic invariances, in that their metamers vary in ways that render them unrecognizable to other models. The results also clarify the effect of adversarial training. Specifically, they suggest that adversarial training is removing some of the idiosyncratic invariances of standard-trained deep neural networks, rather than learning new invariances that are not shared with other models (in which case their metamers would not have been better recognized by other models). The architectural change that reduced aliasing had a similar effect, albeit limited to the intermediate model stages.

The average model recognition of metamers generated from a given stage of another model is strikingly similar to human recognition of the metamers from that stage (compare Figures 9b and 9c to Figures 3, 5, and 6, and Supplementary Figure 6 to Figure 7d). To quantify this similarity, we plotted the average model recognition for metamers from each stage of each generating model against human recognition of the same stimuli, revealing a strong correlation for both visual (Figure 9d) and auditory (9e) models. This result suggests that the human-model discrepancy revealed by model metamers reflects invariances that are often idiosyncratic properties of a specific neural network, leading to impaired recognition by both other models and human observers. One consequence of this result is that a collection of other models may be a reasonable proxy for a human observer -- rather than running a new psychophysical experiment any time there is a new model to evaluate, one could test the model's metamers on a set of other models, facilitating the automation of searches for better models of perception.

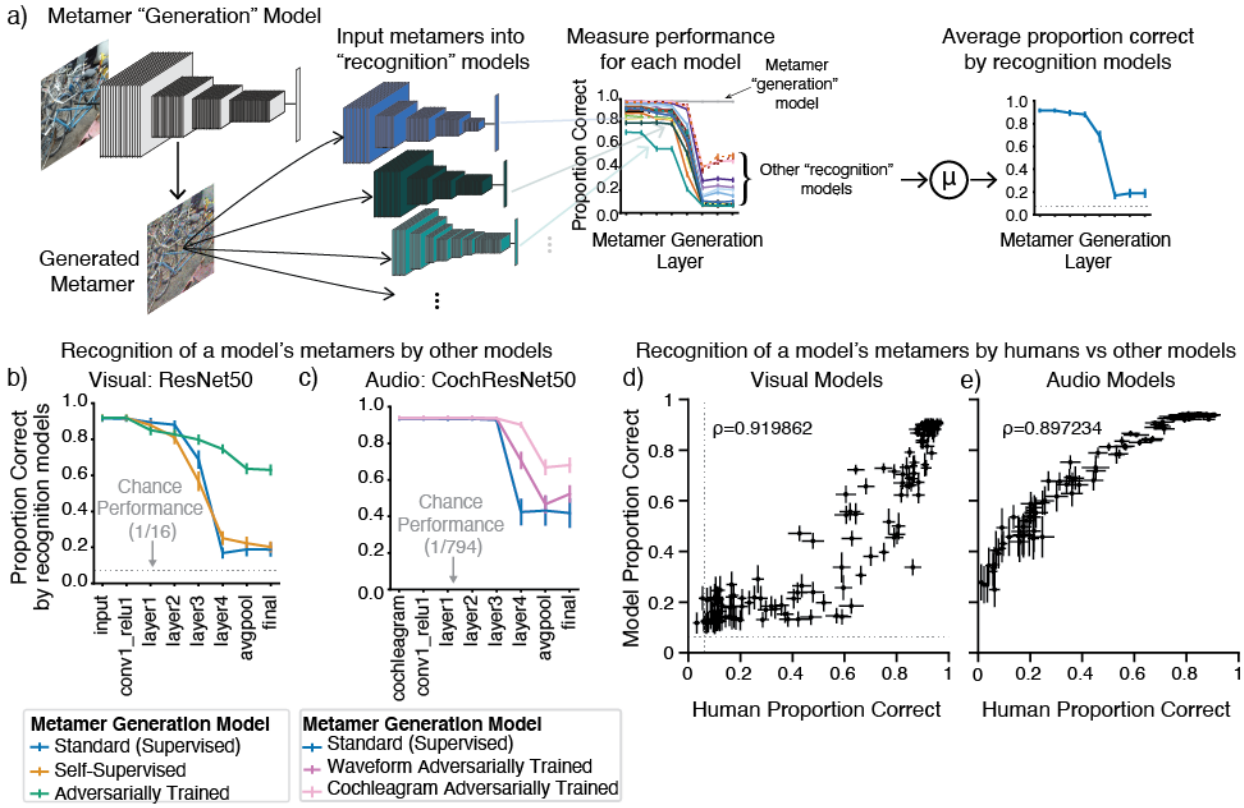


Figure 9. Human recognition of a model's metamers is correlated with their recognition by other models. a) Model metamers were generated for each layer of a "generation" model (one of the models from Figures 2c, 3, and 5 for visual model analysis, and from Figures 2f and 6 for audio model analysis). The associated metamers are input into "recognition" models (all other models from Figures 2c, 3, and 5 for visual model analysis, and from Figures 2f and 6 for audio model analysis). We measured recognition of the generating model's metamers by each recognition model, averaging the accuracy over all recognition models (excluding the generation model). This analysis is shown here for a standard-trained ResNet50 image model. Error bars are SEM over recognition models. b) Average model recognition of metamers generated from the standard ResNet50, the three self-supervised ResNet50 models, and the three ResNet50 models trained with adversarial perturbations. In model groups with multiple generating models, we averaged each recognition model's accuracy curve across all of the generating models, then averaged these curves across the recognition models. Error bars are the SEM over recognition models. Model metamers from deep stages tend to be unrecognizable to other models, but models trained with adversarial perturbations have metamers that are more recognizable by other models. c) Same as b, but for audio models, with metamers generated from the standard CochResNet50, three CochResNet50 models with waveform adversarial perturbations, and two CochResNet50 models with cochleagram adversarial perturbations. Chance performance is 1/794 for the models because (unlike the human experimental participants) they had a "null" (no speech) class label as a possible prediction in addition to the 793 word labels. d, e) Correlation between human and model recognition of another model's metamers for visual (d) and audio (e) models. The abscissa of each data point plots the average human recognition accuracy of metamers generated from one stage of a model. The ordinate of each data point plots the average recognition by other models of those metamers. In both modalities, human recognition of a model's metamers is highly correlated with the accuracy of other models at recognizing those same model metamers (quantified by spearman rank-ordered correlation coefficient).

Discussion

We used model metamers to reveal invariances of deep artificial neural networks, and compared these invariances to those of humans by measuring human recognition of visual and auditory model metamers. We found that metamers of standard deep neural networks are dominated by

invariances that are absent from human perceptual systems, in that metamers from deep model stages are typically completely unrecognizable to humans. This was true across modalities (both visual and auditory) and across training methods (supervised vs. self-supervised training). We identified ways to make model metamers more human-recognizable in both the auditory and visual domains, including a new type of adversarial training for audio models using perturbations at an intermediate model stage. Although there was a substantial effect on metamer recognition from one common training method to reduce adversarial vulnerability, we found that metamers could reveal model differences that were not evident by measuring adversarial vulnerability alone. Moreover, some model improvements revealed by model metamers were not obvious from standard brain prediction metrics. These results show that metamers provide a model comparison tool that complements the standard benchmarks that are in widespread use.

Even though some models produced significantly more recognizable metamers than others, metamers generated from late model stages remained less recognizable than natural images or sounds in all cases we tested, suggesting that substantial further improvements are needed to align model representations with those of biological sensory systems. The fact that one model generally cannot recognize another model's metamers indicates that the discrepancy is at least partially caused by invariances that are idiosyncratic to a model (being absent in both humans and other models). Might humans analogously have their own invariances that are specific to an individual? This possibility is difficult to explicitly test given that we cannot currently sample human metamers (model metamer generation relies on having access to the model's parameters and responses, which are currently beyond reach for biological systems). If idiosyncratic invariances were present in humans as well, the phenomenon we have described here might not represent a human-model discrepancy, and could instead be a general property of recognition systems. The main argument against this interpretation is that several model modifications motivated by engineering considerations – different forms of adversarial training, and architectural modifications to reduce aliasing – substantially reduced the idiosyncratic invariances present in standard deep neural network models. These results suggest that idiosyncratic invariances are not unavoidable in a recognition system. Moreover, the set of modifications explored here was far from exhaustive, and we see no reason why the idiosyncratic model invariances could not be further alleviated with alternative training or architecture changes in the future.

Relation to previous work

The methods we use to synthesize model metamers are not new. Previous neural network visualizations have also used gradient descent on the input to visualize representations (60), in some cases matching the activations at individual layers as we do here (61). However, the significance of these visualizations for evaluating neural network models of biological sensory systems has received relatively little attention. One contributing factor may be that model visualizations have often been constrained by added natural image priors or other forms of regularization (62) that help make model visualizations look more natural, but mask the extent to which they otherwise diverge from a perceptually meaningful stimulus. For this reason, we intentionally avoided priors or other regularization when generating model metamers, as they defeat the purpose of the metamer test.

Another reason the discrepancies we report here have not been widely discussed within neuroscience is that most studies of neural network visualizations have relied on small numbers of examples rather than systematically measuring recognizability to human observers (in part because these visualizations are primarily reported within computer science, where such experiments are not the norm). By contrast, our work used human perceptual experiments to quantify the divergence between human and model invariances, revealing a widespread mismatch between humans and current state-of-the-art computational models. We found

controlled experiments to be essential. Prior to running full-fledged experiments we always conducted the informal exercise of generating examples and evaluating them subjectively. Although the largest effects were evident informally, the variability of natural images and sounds made it difficult to predict with certainty how an experiment would turn out. It is thus critical to substantiate informal observation with controlled experiments in humans.

Metamers are also related to a type of adversarial example generated by adding small perturbations to an image from one class such that the activations of a classifier match those of a reference image from a different class. These "invariance-based adversarial examples" yield the same network activations at the classification layer (sometimes referred to as "feature collisions" (63)). but are seen as different classes by humans (albeit again tested informally with a small number of examples rather than a systematic experiment) (64, 65). Our method differs in probing the network invariances without any explicit bias to cause the metamers to appear different to humans, but our results may reflect some of the same properties evident in these alternative adversarial stimuli. We innovate on this prior work by quantifying human recognition with experiments, in probing invariances across model stages, by exploring a diverse set of models (e.g. those trained with self-supervision), by showing that the relevant phenomena generalize across auditory and visual domains, and by identifying many non-human invariances as model-specific. The finding that models tend to learn idiosyncratic invariances may be related to findings that the representational dissimilarity matrices for natural images can vary between individual neural network models (66).

Metamers have previously been used to validate models of visual and auditory texture (30, 67), and visual crowding (34, 35, 68, 69). A related type of model-matched stimulus has been used to test whether models can account for brain responses (51). In these previous lines of work the models instantiated hypotheses about specific types of invariances that may be present in human sensory systems, and the experiments assess human perception or brain responses to test the hypotheses. Our work differs in that the models are learned, and could in principle instantiate many different types of invariances (which are a priori unknown).

Our work also differs from prior work on metamers in that our test of model metamers is a recognition test rather than a same/different metamerism test that has classically been used for evaluating models of early sensory stages (measuring the ability to discriminate two stimuli). We view the recognition test as most appropriate for current deep neural network models that are typically trained on one or two tasks at most and that model components of high-level sensory perception, such as object or speech recognition. A same/different discrimination judgment might be performed using any representations in the observer's perceptual system, rather than just those that are relevant for a particular task. There is thus no reason to expect that metamers for representations that support a particular task would be completely indistinguishable in every respect to a human (in contrast to metamers of color vision, for instance, for which the entire visual system is constrained by the cone inputs). The classical metamer test is thus likely to be too sensitive for our purposes – models may fail the test even if they succeeded in capturing the invariances of a particular task. We note that because an unrecognizable metamer would obviously be distinguishable from the natural image or sound to which its activations are matched, a failure in our recognition test also implies a failure in the classical test of metamerism.

Effects of unsupervised training

One common criticism of task-optimized models is that supervised training on classification tasks is inconsistent with biological learning (45). Recent advances in unsupervised learning have enabled useful representations to be learned from large quantities of natural data without explicit labels, potentially providing a more biologically plausible computational theory of learning (70–

72). A priori it seemed plausible that the invariances of deep neural network models could be strongly dependent on supervised training for classification tasks, in which case models trained without supervision might be substantially more human-like according to the metamers test. However, we found that the invariances learned from self-supervised learning also diverged from the invariances of human perceptual systems.

The metamer-related model discrepancies we saw for self-supervised models are particularly striking because these models are trained with the goal of invariance, being explicitly optimized to become invariant to the augmentations performed on the input. But the metamers of these models reveal that they nonetheless encode a large number of invariances that diverge from those of human perception. This finding is consistent with evidence that the classification errors of self-supervised models are no more human-like than those of supervised models (73). We also found that the divergence with human recognition had a similar dependence on model stage irrespective of whether models are trained with or without supervision. These findings raise the possibility that factors common to supervised and unsupervised neural networks underlie the divergence with humans.

Differences in metamers across layers

The metamer test differs from some other model metrics (e.g. behavioral judgments of natural images or sounds, or measures of adversarial vulnerability) in that metamers can be generated from every stage of a model, with the resulting discrepancies associated with particular model stages. Because invariances reflect information that is discarded, and because information that is discarded at one stage of a feedforward system cannot be recovered, the set of metamers for a reference stimulus cannot shrink from one model stage to the next. But apart from this constraint it is otherwise not obvious how discrepancies might develop across stages, and we observed variation from model to model.

One application of this layer-specificity is to characterize intermediate model stages. In some cases metamers revealed that intermediate stages were more human-like in some models than others. For example, the effects of reducing aliasing produced large improvements in the human-recognizability of metamers from intermediate stages (Figure 7d). By contrast, metamers from the final stages showed little improvement. This result indicates that this model change produces intermediate representations with more human-like invariances despite not resolving the discrepancy introduced at the final model stages.

For most models, the early layers produced model metamers that were fully recognizable, but that also resemble the original image or sound they were matched to. By contrast, metamers from late stages physically deviated from the original image or sound but for some models nonetheless remained recognizable. This difference highlights two ways that a model's metamers can pass the recognition test used here – either by being perceptually indistinguishable to humans, or by being recognizable to humans as the same class despite being perceptually distinct. This distinction could be quantified in future work by combining a traditional metamer test with our recognition test.

Across all the models we considered, the final model stages tended to produce metamers that were less recognizable than natural images to humans as well as to other models. This was true irrespective of how the models were trained. This result highlights these deep stages as targets for model improvements.

Limitations

Although a model that fails our metamer test is ruled out as a description of human perception, passing the test, on its own, reveals little. For instance, a model that instantiates the identity mapping would pass our test despite not being able to account for human perceptual abilities. Traditional metrics thus remain critical, but on their own are also insufficient (as shown in Figures 7 and 8). Failing the test also does not imply that the model representations are not present in the brain, only that they are not sufficient to account for the recognition behavior under consideration. For instance, there is considerable evidence for time-averaged auditory statistics in auditory perception (30, 74) even though they do not produce human-recognizable metamers for speech (Figure 4f). We thus argue that a large suite of test metrics, including but not limited to the model metamer test, is needed for model comparison.

Model metamers are generated via iterative gradient-based optimization of a non-convex loss function, and only approximately reproduce the activations of the natural stimulus to which they are matched. We attempted to improve on previous deep network visualization work (60, 61) by setting explicit criteria for the optimization success. For models that perform classification tasks, we verified that the model metamer produced the same class label as the reference stimulus to which it was matched. Failures of the metamer test represent an unambiguous dissociation of model and human behavior for such models. We also verified that the residual error between the metamer and reference activations was much smaller than would be expected by chance. However, the reliance on optimization may be a limitation in some contexts and with some models.

We also cannot fully exclude the possibility that the metamer optimization process does not sample uniformly from the set of a model's metamers. Such non-uniform sampling cannot explain the human-model discrepancies we observed, but could in principle contribute to differences between the magnitude of discrepancies for some models compared to others. For instance, the greater human-recognizability of metamers from some models could in principle be caused by differences in the optimization landscape that make it less likely that the metamer generation process samples along a model's idiosyncratic invariances. We are not aware of any reason to think that this might be the case, but it is not obvious how to fully exclude this possibility.

Adversarial training reduces idiosyncratic invariances

Why does adversarial training produce models with more recognizable metamers? One interpretation is that it prevents models from "cheating" on tasks by using data set artifacts to make decisions (54) and as a consequence develops more human-like invariances. This interpretation is supported by findings that adversarial examples often transfer across models (53, 75). Our results suggest that the effect of adversarial training on metamers likely has a distinct explanation. First, we found that not all model changes that improve adversarial robustness have comparable effects on metamer recognizability (Figure 7), suggesting that it is the adversarial training procedure, rather than adversarial robustness per se, that underlies the increased human recognizability of model metamers. Second, metamers from standard trained models generally did not transfer across models trained on the same data set (unlike adversarial examples). Metamers from adversarially trained models were more recognizable to humans but also transferred substantially better to other models. This finding suggests that adversarial training serves to avoid the model-specific invariances that are otherwise learned by deep neural network models. This set of results highlights the relationship between adversarial robustness and model metamers as a promising direction for future study.

Utility of synthetic stimuli

Most previous work comparing artificial neural network representations to human brain and behavioral representations has compared human and model responses to a large set of natural images or sounds (5–7, 76–79). Although this approach has provided evidence of similarities

between models and biological sensory systems, it is vulnerable to the correlations that typically exist between features in natural stimuli (51, 80). Stimulus correlations make it difficult to separate whether human brain responses are in fact due to the same underlying features instantiated in a model, as opposed to a set of distinct features that are correlated with the model's features within the stimulus set. Synthetic stimuli generated to match particular model features disrupt these correlations because they are constrained only by the model features, and as a result can provide a stronger model test (51, 81). Synthesis methods can also be used to dissociate predictions of different models (82), to test reliance on specific features (83, 84), or to optimize stimuli for a model of a neural response (85–87). Model metamers are another example of this general approach, showing that models can produce similar brain predictions for natural stimuli (Figure 8) despite possessing distinct invariances from those of human observers. The similar predictions for natural stimuli suggest that the discrepancies revealed by metamers may be difficult, if not impossible, to reveal using only natural stimuli.

Future directions

The underlying causes of the divergence between human and model invariances demonstrated here seems important to understand, in part because we will not fully understand biological sensory systems until we can replicate them in a model, and deep neural networks are currently the most promising candidates. In addition, many otherwise promising model applications, such as model-based signal enhancement (88, 89), are likely to be hindered by human-discrepant model invariances.

The supervised models we explored here were trained on a single speech or image recognition task. It is possible that the use of a single task causes models to discard information that is preserved by biological sensory systems. One possibility is that biological sensory systems do not instantiate invariances per se, in the sense of mapping multiple different inputs onto the same representation. Instead, they might learn representations that untangle behaviorally relevant variables. For instance, a system could represent word labels and talker identity, or object identity and pose, via independent directions in a representational space. Such a representational scheme could enable invariant classification without invariant representations. Some model architectures may be able to address this hypothesis, for instance by factoring out different latent variables that, once combined, can reconstruct the input representation. But as of yet we lack methods for building such models that can support human-level recognition at scale. Training on multiple tasks could be another approach that would serve this goal. The results of Figure 9 – showing that the recognition judgments of a set of models for another model's metamers are predictive of human recognition – suggest a way to efficiently test any model for discrepant metamers, which should facilitate evaluation of these and any other new model class that is developed in coming years.

The discrepancies shown here for model metamers contrast with a growing number of examples of compelling human-model similarities for behavioral judgments of natural stimuli. Models optimized for object recognition (90), speech recognition (5), sound localization (91), and pitch recognition (92) all exhibit qualitative and often quantitative similarities to human judgments when run in traditional psychophysical experiments with natural or relatively naturalistic stimuli. These results suggest that neural network models trained in naturalistic conditions often match human behavior for signals that fall within their training distribution, but not for some signals derived from the model that fall outside the distribution of natural sounds and images. Understanding the source of this contrast seems critical to understanding both the successes and failures of deep network models.

Current deep neural network models are overparametrized, such that training produces one of many functions consistent with the training data. From this perspective it is unsurprising that different systems can perform similarly on natural signals while exhibiting different responses to signals outside the training distribution of natural images or sounds. And yet we nonetheless found that sensible engineering modifications to models succeeded in bringing the models into better alignment with human invariances. These results demonstrate that divergence between human and model invariances is not inevitable, and show how metamers can be a useful metric to guide and evaluate the next generation of brain models.

METHODS

Model training and evaluation

Models were trained and evaluated with the PyTorch deep learning library (93), and the Robustness library (94), modified to accommodate metamer generation and audio model training. Model code and checkpoints will be made available online via GitHub upon acceptance of the paper. Model architecture descriptions are provided as Supplemental Information. All models were trained on the OpenMind computing cluster at MIT using NVIDIA GPUs with a minimum of 11GB memory.

Image training dataset

All vision neural network models were trained on the ImageNet Large Scale Visual Recognition Challenge dataset (37). This classification task consists of 1000 classes of images with 1,281,167 images in the training set and 50,000 images in the validation set. All classes were used for training the neural network models. Accuracy on ImageNet task and additional training parameters are reported in Table 1.

Visual model training and evaluation

Unless otherwise described below, visual models consisted of publicly available checkpoints. Standard supervised models used the pretrained PyTorch checkpoints from torchvision.models (documentation https://pytorch.org/vision/stable/models.html, referred to as "pytorch" in Table 1). Visual model performance was evaluated as the model accuracy on the ImageNet validation set, implemented by resizing the images so the smallest dimension was 256 pixels (or 250 in the case of HMAX) and taking a center crop of 224x224 (or 250 in the case of HMAX) pixels of the image. Train, test, and metamer images were all normalized by subtracting channel means of [0.485, 0.456, 0.406] and dividing by channel standard deviations of [0.229, 0.224, 0.225] before being passed into the first layer of the neural network backbone (except in the case of HMAX, where this normalization was not applied).

Self-supervised vision models

Self-supervised models were downloaded from the OpenSelfSup Model Zoo, and the training details that follow are taken from the documentation (https://github.com/open-mmlab/OpenSelfSup, referred to as "openselfsup" in Table 1). Three models, each with a ResNet50 architecture, were used: MoCo_V2, SimCLR and BYOL. MoCo_V2 self-supervised training had a batch size of 256, with data augmentations consisting of random crop (224x224 pixels), random horizontal flip (probability=0.5), random color jitter (brightness=0.4, contrast=0.4, saturation=0.4, hue=0.1, probability=0.8, all values uniformly chosen), random greyscale (probability=0.2), and random gaussian blur (sigma_min=0.1, sigma_max=0.2, probability=0.5). SimCLR self-supervised training had a batch size of 256, with augmentations consisting of random crop (224x224 pixels), random horizontal flip (probability=0.5), random color jitter (brightness=0.8, contrast=0.8, saturation=0.8, hue=0.2, probability=0.8, all values uniformly chosen), random greyscale (probability=0.2), and random gaussian blur (sigma_min=0.1, sigma_max=0.2, probability=0.5). BYOL self-supervised training had a batch size of 4096, with augmentations consisting of random crop (224x224 pixels), random horizontal flip (probability=0.5), random color jitter (brightness=0.4, contrast=0.4, saturation=0.2, hue=0.1, probability=0.8, all values uniformly chosen), random greyscale (probability=0.2), and random gaussian blur (sigma_min=0.1, sigma_max=0.2, probability=0.5). For all self-supervised models, a linear readout consisting of a fully connected layer with 1000 units applied to the average pooling layer of the network was trained using the same augmentations used for supervised training of the other models described above. The linear readout was trained for 100 epochs of ImageNet (while the rest of the model backbone remained unchanged). For MoCo_V2 and SimCLR models, the accuracy was within 1% of that reported on the OpenSelfSup (BYOL average pooling

evaluation was not posted at the time of training). The linear readout served as a check that the downloaded models were instantiated correctly, and was used to help verify the success of the metamer generation optimization procedure, as described below.

HMAX vision model

The hand-engineered HMAX vision model was based off of a publicly availably implementation in PyTorch (https://github.com/wmvanvliet/pytorch_hmax) which follows the model documented in a previous publication (4). A gaussian activation function was used, and boundary handling was added to match the MATLAB implementation provided by the original HMAX authors (https://maxlab.neuro.georgetown.edu/hmax.html). For full comparison to the other models, we trained a linear classifier consisting of 1000 units to perform the ImageNet recognition task on the final C2 output of the HMAX model. This fully connected layer was trained for 30 epochs of the ImageNet training dataset, and the learning rate was dropped after every 10 epochs. Inputs to HMAX during the classifier training consisted of random crops (250x250 pixels), random horizontal flip (p=0.5), random color jitter (brightness=0.1, contrast=0.1, saturation=0.1, probability=1, all values uniformly chosen), and lighting noise (alpha standard deviation of 0.05, an eigenvalue of [0.2175, 0.0188, 0.0045], and channel eigenvectors of [[-0.5675, 0.7192, 0.4009], [-0.5808, -0.0045, -0.8140], [-0.5836, -0.6948, 0.4203]]). HMAX performance was evaluated by measuring the model accuracy on the ImageNet validation set after resizing the images so that the smallest dimension was 250 pixels, taking a center crop of 250x250 pixels of the image, converting to greyscale, and multiplying by 255 to scale the image to the 0-255 range. As expected, the performance on this classifier was low, but it was significantly above chance and could thus be used for the metamer optimization criteria described below.

Adversarial training – vision models

Adversarially trained ResNet50 models were obtained from the robustness library (https://github.com/MadryLab/robustness, referred to as "robustness" in Table 1). Adversarially trained AlexNet architectures and the random perturbation ResNet50 and AlexNet architectures were trained for 120 epochs of the ImageNet dataset, with image pixel values scaled between 0-1. Learning rate was decreased by a factor of 10 after every 50 epochs of training. During training, data augmentation consisted of random crop (224x224 pixels), random horizontal flip (probability=0.5), color jitter (brightness=0.1, contrast=0.1, saturation=0.1, probability=1, all values uniformly chosen), and lighting noise (alpha standard deviation of 0.05, an eigenvalue of [0.2175, 0.0188, 0.0045], and channel eigenvectors of [[-0.5675, 0.7192, 0.4009], [-0.5808, -0.0045, -0.8140], [-0.5836, -0.6948, 0.4203]]). An adversarial or random perturbation was then added. All adversarial examples were untargeted, such that the loss used to generate the adversarial example pushed the input away from the original class by maximizing the cross-entropy loss, but did not push the prediction towards a specific target class. For the L_2-norm ($\epsilon = 3$) network, adversarial examples were generated with a step size of 1.5 and 7 attack steps. For the L_∞-norm ($\epsilon = 8/255$) network, adversarial examples were generated with a step size of 4/255 and 7 attack steps. For both ResNet50 and AlexNet random-perturbation L_2-norm networks, a random sample on the L_2 ball with width $\epsilon = 3$ was drawn and added to the input, independently for each training example and dataset epoch. Similarly, for both Resnet50 and AlexNet random perturbation L_∞-norm networks, a random sample on the corners of the L_∞ ball was selected by randomly choosing a value of ±8/255 to add to each image pixel, independently chosen for each training example and dataset epoch. After the adversarial or random perturbation was added to the input image, the new image was clipped between 0-1 before being passed into the network.

VOneAlexNet vision model

The VOneAlexNet architecture was downloaded from the VOneNet GitHub repository (https://github.com/dicarlolab/vonenet) (57). Modifications were then made to use Gaussian noise

rather than Poisson noise as the stochastic component, as in (58), and to use the same input normalization as in our other models (rather than a mean of 0.5 and standard deviation of 0.5 as used in (57)). The model was trained with stochastic responses (Gaussian noise with standard deviation of 4) in the "VOne" model stage, but for the purposes of metamer generation we fixed the noise by randomly drawing one noise sample when loading the network and using this noise sample for all metamer generation and adversarial evaluation. Although "fixing" the noise reduces the measured adversarial robustness compared to when a different sample of noise is used for each iteration of the adversarial example generation, the model with a single noise draw was still significantly more robust than a standard model, and allowed us to perform the metamer experiments without having to account for the stochastic representation during metamer optimization.

LowpassAlexNet vision model

The LowpassAlexNet architecture was trained for 120 epochs using the same augmentations described for the adversarially trained AlexNet networks (but without adversarial or random perturbations). To approximately equate performance on natural stimuli with the VOneNetAlexNet, we chose an early checkpoint that was closest, but did not exceed, the Top 1% performance of the VOneAlexNet model (to ensure that the greater recognizability of the metamers from LowpassAlexNet could not be explained by higher overall performance of that model). This resulted in a comparison model trained for 39 epochs of the ImageNet dataset.

AlexNet vision model, early checkpoint

We trained an AlexNet architecture for 120 epochs using the same augmentations described for the adversarially trained AlexNet networks (but without adversarial or random perturbations). After training, to approximately equate performance on natural stimuli with the VOneNetAlexNet and LowpassAlexNet, we chose an early checkpoint that was closest, but not lower than, the performance of the VOneAlexNet model. This resulted in a comparison model trained for 51 epochs of the ImageNet dataset.

Adversarial evaluation -- vision models

The adversarial vulnerability of visual models was evaluated with white-box untargeted adversarial attacks (i.e., in which the attacker has access to the model's parameters when determining an attack that will cause the model to classify the image as any category other than the correct one). All 1000 classes of ImageNet were used for the adversarial evaluation. Attacks were computed with L_1, L_2, and L_∞ maximum perturbation sizes (ε) added to the image, with 64 gradient steps each with size ε/4 (pilot experiments suggested that this step size and number of steps were sufficient to produce adversarial examples for most models). We randomly chose images from the ImageNet evaluation dataset to use for adversarial evaluation, applying the evaluation augmentation described above (resizing so that the smallest dimension was 256 pixels, followed by a center crop of 224x224 pixels). Five different subsets of 1024 stimuli were drawn to compute error bars.

For the statistical comparisons between the adversarial robustness of architectures for Figure 7 we performed a repeated measure ANOVA with within-group factors of architecture and perturbation size ϵ. A separate ANOVA was performed for each adversarial attack type. The values of ϵ included in the ANOVA were constrained to a range where the VOneAlexNet and LowPassAlexNet showed robustness over the standard AlexNet (four values for each attack type, $\epsilon_{L_1} \in \{10^{1.5}, 10^{2}, 10^{2.5}, 10^{3}\}$, $\epsilon_{L_2} \in \{10^{-1}, 10^{-0.5}, 10^{0}, 10^{0.5}\}$, $\epsilon_{L_\infty} \in \{10^{-3.5}, 10^{-3}, 10^{-2.5}, 10^{-2}\}$), so that any difference in clean performance did not affect the comparisons. We computed statistical significance for the main effect of architecture by a permutation test, randomly permuting the

architecture assignment, independent for each subset of the data. We computed a p-value by comparing the observed F-statistic to the null distribution of F-statistics from permuted data (i.e., the p-value was one minus the rank of the observed F-statistic divided by the number of permutations). In cases where the maximum possible number of unique permutations was less than 10,000, we instead divided the rank by the maximum number of unique permutations.

Model	Learning Rate (at Start)	Batch Size	Accuracy Top 1	Accuracy Top 5
AlexNet Standard	(pretrained, pytorch)	(pretrained)	56.518	79.070
AlexNet L_2-norm ($\epsilon = 3$)	0.01	256	41.882	65.018
AlexNet random L_2-norm ($\epsilon = 3$)	0.01	256	57.978	79.986
AlexNet L_∞-norm ($\epsilon = 8/255$)	0.01	256	29.702	51.034
AlexNet random L_∞-norm ($\epsilon = 8/255$)	0.01	256	57.738	79.996
ResNet50 Standard	(pretrained, pytorch)	(pretrained)	76.130	92.862
ResNet50 L_2-norm ($\epsilon = 3$)	(pretrained, robustness)	(pretrained)	57.900	80.706
ResNet50 random L_2-norm ($\epsilon = 3$)	0.1	256	76.600	93.136
ResNet50 L_∞-norm ($\epsilon = 4/255$)	(pretrained, robustness)	(pretrained)	62.424	84.060
ResNet50 L_∞-norm ($\epsilon = 8/255$)	(pretrained, robustness)	(pretrained)	47.906	72.484
ResNet50 random L_∞-norm ($\epsilon = 8/255$)	0.1	256	73.062	91.366
ResNet101 Standard	(pretrained, pytorch)	(pretrained)	77.374	93.546
VGG-19 Standard	(pretrained, pytorch)	(pretrained)	72.376	90.876
CORnet-S Standard	(pretrained, cornet)	(pretrained)	73.020	91.116
ResNet50-BYOL	(pretrained, openselfsup) / 30 linear eval	(pretrained) / 256 linear eval	68.706	88.090
ResNet50-MOCO_V2	(pretrained, openselfsup) / 30 linear eval	(pretrained) / 256 linear eval	67.832	88.322
ResNet50-SIMCLR	(pretrained, openselfsup) / 30 linear eval	(pretrained / 256 linear eval	59.204	81.304
HMAX	0.1 linear eval	256	6.101	15.070
VOneAlexNet	0.01	256	47.84	71.57
LowpassAlexNet	0.01	256	47.620	72.416
AlexNet (EarlyCheckpoint)	0.01	256	52.536	76.446

Table 1: Vision architecture training parameters and ImageNet accuracy.

Audio training dataset
All audio neural network models were trained on the Word-Speaker-Noise (WSN) dataset. This dataset was first presented in (36) and was constructed from existing speech recognition and environmental sound classification datasets. The dataset is approximately balanced to enable performance of three tasks on the same training exemplar: (1) recognition of the word at the center of a two second speech clip (2) recognition of the speaker and (3) recognition of environmental sounds, that are superimposed with the speech clips (serving as "background noise" for the speech tasks while enabling an environmental sound recognition task). Although the dataset is constructed to enable all three tasks, the models described in this paper were only trained to perform the word recognition task. The speech clips used in the dataset were excerpted from the Wall Street Journal (95) (WSJ) and Spoken Wikipedia (96) (SWC).

To choose speech clips, we screened WSJ, TIMIT (97) and a subset of articles from SWC for appropriate audio clips (specifically, clips that contained a word at least four characters long and that had one second of audio before the beginning of the word and after the end of the word, to enable the temporal jittering augmentation described below). Some SWC articles were left out of the screen due to a) potentially offensive content for human listening experiments; (29/1340 clips), b) missing data; (35/1340 clips), or c) bad audio quality (for example, due to computer generated voices of speakers reading the article or the talker changing mid-way through the clip; 33/1340 clips). Each segment was assigned the word class label of the word overlapping the segment midpoint and a speaker class label determined by the speaker. With the goal of constructing a dataset with speaker and word class labels that were approximately independent, we selected words and speaker classes such that the exemplars from each class spanned at least 50 unique cross-class labels (e.g., 50 unique speakers for each of the word classes). This exclusion fully removed TIMIT from the training dataset. We then selected words and speaker classes that each contained at least 200 unique utterances, and such that each class could contain a maximum of 25% of a single cross-class label (e.g., for a given word class, a maximum of 25% of utterances could come from the same speaker). These exemplars were subsampled so that the maximum number in any word or speaker class was less than 2000. The resulting training dataset contained 230,356 unique clips in 793 word classes and 432 speaker classes, with 40,650 unique clips in the test set. Each word class had between 200 and 2000 unique exemplars. A "null" class was used as a label when a background clip was presented without the added speech.

The environmental soundtrack clips that were superimposed on the speech clips were a subset of examples from the AudioSet dataset (a set of annotated YouTube video soundtracks) (98). To minimize ambiguity for the two speech tasks, we removed any sounds under the "Speech" or "Whispering" branch of the AudioSet ontology. Since a high proportion of AudioSet clips contain music, we achieved a more balanced set by excluding any clips that were only labeled with the root label of "Music", with no specific branch labels. We also removed silent clips by first discarding everything tagged with a "Silence" label and then culling clips containing more than 10% zeros. This screening resulted in a training set of 718,625 unique natural sound clips spanning 516 categories. Each AudioSet clip was a maximum of 10 seconds long, from which a 2-second excerpt was randomly cropped during training (see below).

Audio model training
During training, the speech clips from the Word-Speaker-Noise dataset were randomly cropped in time and superimposed on random crops of the AudioSet clips. Data augmentations during training consisted of 1) randomly selecting a clip from AudioSet to pair with each labeled speech clip, 2) randomly cropping 2 seconds of the AudioSet clip and 2 seconds of the speech clip, cropped such that the labeled word remained in the center of the clip (due to training pipeline

technicalities, we used a pre-selected set of 5,810,600 paired speech and natural sound crops which spanned 25 epochs of the full set of speech clips and 8 passes through the full set of AudioSet clips), 3) superimposing the speech and the noise (i.e., the AudioSet crop) with a Signal-to-Noise-Ratio (SNR) sampled from a uniform distribution between -10dB SNR and 10dB SNR, augmented with additional samples of speech without an AudioSet background (i.e. with infinite SNR, 2464 examples in each epoch) and samples of AudioSet without speech (i.e. with negative infinite SNR, 2068 examples in each epoch) and 4) setting the root-mean-square (RMS) amplitude of the resulting signal to 0.1. Evaluation performance is reported on one pass through the speech test set (i.e., one crop from each of the 40,650 unique test set speech clips) constructed with the same augmentations used during training (specifically, variable SNR and temporal crops, paired with a separate set of AudioSet test clips, same random seed used to test each model such that test sets were identical across models).

Each audio model was trained for 150 epochs (where an epoch is defined as a full pass through the set of 230,356 speech training clips). The learning rate was decreased by a factor of 10 after every 50 epochs (see Table 2).

Audio model cochlear stage

The first stage of the audio models produced a "cochleagram" – a time-frequency representation of audio with frequency tuning that mimics the human ear, followed by a compressive nonlinearity (44). This stage consisted of the following sequence of operations. First, the 20kHz audio waveform passed through a bank of 211 bandpass filters with center frequencies ranging from 50Hz to 10kHz. Filters were zero-phase with frequency response equal to the positive portion of a single period of a cosine function, implemented via multiplication in the frequency domain. Filter spacing was set by the Equivalent Rectangular Bandwidth (ERB_N) scale (43). Filters perfectly tiled the spectrum such that the summed squared response across all frequencies was flat (four low-pass and four high-pass filters were included in addition to the bandpass filters in order to achieve this perfect tiling). Second, the envelope was extracted from each filter subband using the magnitude of the analytic signal (via the Hilbert transform). Third, the envelopes were raised to the power of 0.3 to simulate basilar membrane compression. Fourth, the compressed envelopes were lowpass-filtered and downsampled to 200Hz (1d convolution with a Kaiser-windowed Sinc filter of size 1001 in the time domain, applied with a stride of 100 and no zero padding, i.e. "valid" convolution), resulting in a final "cochleagram" representation of 211 frequency channels by 390 time points. The first layer of the neural network "backbone" of the auditory models operated on this cochleagram representation. Cochleagram generation was implemented in PyTorch such that the components were differentiable for metamer generation and adversarial training. Cochleagram generation code will be released upon acceptance of the paper.

Spectemp model

The hand-engineered Spectro-Temporal filter model (Spectemp) was based on a previously published model (49). Our implementation differed from the original model in specifying spectral filters in cycles/ERB rather than cycles/octave (because our implementation operated on a cochleagram generated with ERB-spaced filters). The model consisted of a linear filter bank tuned to spectro-temporal modulations at different frequencies, spectral scales, and temporal rates. The filtering was implemented via 2D convolution with zero padding in frequency (211 samples) and time (800 samples). Spectro-temporal filters were constructed with spectral modulation center frequencies of [0.0625, 0.125, 0.25, 0.5, 1, 2] cycles/ERB and temporal modulation center frequencies of [0.5, 1, 2, 4, 8, 16, 32, 64] Hz, including both upward and downward frequency modulations (resulting in 96 filters). An additional 6 purely spectral and 8 purely temporal modulation filters were included for a total of 110 modulation filters. This filterbank operated on

the cochleagram representation (yielding the 'filtered_signal' stage in Figure 4d-f). We squared the output of each filter response at each time step ('power') and took the average across time for each frequency channel ('average'), similar to previous studies (5, 50, 59). To be able to use model classification judgments as part of the metamer generation optimization criteria (see below), we trained a linear classifier after the average pooling layer (trained for 150 epochs of the speech training set with a learning rate that started at 0.01 and decreased by a factor of 10 after every 50 speech epochs, using the same data augmentations as for the neural networks). Although performance on the word recognition task for the Spectemp model was low, it was significantly above chance, and thus could be used to help verify the success of the metamer generation optimization procedure.

Adversarial training – auditory models – waveform perturbations
CochResNet50 and CochCNN9 were adversarially trained with perturbations in the waveform domain. We also included a control training condition in which random perturbations were added to the waveform. For both adversarial and random waveform perturbations, after the perturbation was added, the audio signal was clipped to fall between -1 and 1. As with the adversarially trained vision models, all adversarial examples were untargeted. The L_2-norm ($\epsilon = 0.5$ and $\epsilon = 1.0$) network adversarial examples were generated with a step size of 0.25 and 0.5, respectively, and 5 attack steps. L_∞-norm ($\epsilon = 0.002$) network adversarial examples in the waveform space were generated with a step size of 0.001 and 5 attack steps. For random perturbation L_2-norm networks (both CochResNet50 and CochCNN9), a random sample on the L_2 ball with width $\epsilon = 1.0$ was selected and added to the waveform, independently for each training example and dataset epoch. Similarly, for random perturbation L_∞-norm networks, a random sample on the corners of the L_∞ ball was selected by randomly choosing a value of ±0.002 to add to each image pixel, chosen independently for each training example and dataset epoch.

We estimated the $\mathrm{SNR_{dB}}$ for the perturbations of the waveform using:

$$\mathrm{SNR_{dB}} = 20 \log_{10} \frac{||x||}{||\xi||}$$

where x is the input waveform and ξ is the adversarial perturbation. As described above, the input waveforms, x, to the network were RMS normalized to 0.1, and thus $||x|| = 0.1 * \sqrt{n}$, where n is the number of samples in the waveform (40,000). For L_2-norm perturbations to the waveform, the norm of the perturbation is just the ϵ value, and so $\epsilon = 0.5$ and $\epsilon = 1.0$ correspond to $||\xi|| = 0.5$ and $||\xi|| = 1$, resulting in $\mathrm{SNR_{dB}}$ values of 32.04 and 26.02, respectively. For L_∞-norm perturbations, the worst case (lowest) $\mathrm{SNR_{dB}}$ is achieved by a perturbation that maximally changes every input value. Thus, an L_∞ perturbation with $\epsilon = 0.002$ has $||\xi|| = \sqrt{\sum_{n=1}^{40,000}(0.002^2)}$, corresponding to a $\mathrm{SNR_{dB}}$ value of 33.98. These $\mathrm{SNR_{dB}}$ values do not guarantee that the perturbations were always fully inaudible to humans, but they confirm that the perturbations are relatively minor and unlikely to be salient to a human listener.

Adversarial training – auditory models – cochleagram perturbations
CochResNet50 and CochCNN9 were adversarially trained with perturbations in the cochleagram domain. We also included a control training condition in which random perturbations were added to the cochleagram. Adversarial or random perturbations were added to the output of the cochleagram stage, after which the signal was passed through a ReLU so that no negative values were fed into the neural network backbone. All adversarial examples were untargeted. The L_2-norm ($\epsilon = 0.5$ and $\epsilon = 1.0$) network adversarial examples were generated with a step size of 0.25 and 0.5 respectively, and 5 attack steps. For random perturbation L_2-norm networks (both

CochResNet50 and CochCNN9), a random sample on the L_2 ball with width $\epsilon = 1.0$ was selected, independently for each training example and dataset epoch.

We estimated the $\mathrm{SNR_{dB}}$ of the cochleagram perturbations using the average cochleagram from the test dataset, whose L_2-norm was 40.65. Using this value with the $\mathrm{SNR_{dB}}$ equation yielded estimates of 38.20 and 32.18 dB for cochleagram networks trained with $\epsilon = 0.5$ and $\epsilon = 1.0$, respectively. We again cannot guarantee that the perturbations are inaudible to a human, but they are fairly low in amplitude and thus unlikely to be salient.

Adversarial evaluation – audio networks

As in visual adversarial evaluation, the adversarial vulnerability of audio networks was evaluated with untargeted white-box adversarial attacks. Attacks were computed with L_1, L_2, and L_∞ maximum perturbation sizes (ε) added to the waveform, with 64 gradient steps each with size ε/4 (pilot experiments and previous results (58) suggested that this step size and number of steps were sufficient to attack most audio models). We randomly chose audio samples from the WSN evaluation dataset to use for adversarial evaluation, including the evaluation augmentations described above (additive background noise augmentation with SNR randomly chosen between -10 to 10 dB SNR, and rms normalization to 0.1). Five different subsets of 1024 stimuli were drawn to compute error bars.

Model	Learning Rate (at Start)	Batch Size	Accuracy Top 1	Accuracy Top 5
CochCNN9 Standard	0.01	128	66.672	83.129
CochCNN9 waveform L_2-norm ($\epsilon = 1$)	0.01	128	48.091	67.240
CochCNN9 random waveform L_2-norm ($\epsilon = 1$)	0.01	128	65.710	82.376
CochCNN9 waveform L_∞-norm ($\epsilon = 0.002$)	0.01	128	60.440	78.278
CochCNN9 random waveform L_∞-norm ($\epsilon = 0.002$)	0.01	128	66.283	82.952
CochCNN9 cochleagram L_2-norm ($\epsilon = 1$)	0.01	128	48.002	66.089
CochCNN9 cochleagram L_2-norm ($\epsilon = 0.5$)	0.01	128	57.198	75.001
CochCNN9 random cochleagram L_2-norm ($\epsilon = 1$)	0.01	128	66.706	83.087
CochResNet50 Standard	0.1	256	86.797	95.360
CochResNet50 waveform L_2-norm ($\epsilon = 0.5$)	0.1	256	78.130	90.536
CochResNet50 waveform L_2-norm ($\epsilon = 1$)	0.1	256	70.546	85.474
CochResNet50 random waveform L_2-norm ($\epsilon = 1$)	0.1	256	85.916	94.871
CochResNet50 waveform L_∞-norm ($\epsilon = 0.002$)	0.1	256	83.491	93.658
CochResNet50 random waveform L_∞-norm ($\epsilon = 0.002$)	0.1	256	86.367	95.090
CochResNet50 cochleagram L_2-norm ($\epsilon = 1$)	0.1	256	71.392	85.149
CochResNet50 cochleagram L_2-norm ($\epsilon = 0.5$)	0.1	256	80.435	91.444
CochResNet50 random cochleagram L_2-norm ($\epsilon = 1$)	0.1	256	86.556	95.144
Spectemp (linear eval)	0.01 (linear eval)	128 (linear eval)	5.743	13.780

Table 2: Audio model architecture training parameters and word classification accuracy.

Metamer generation

Optimization of metamers

Gradient descent was performed to minimize the normalized squared error between all activations at a particular network layer (for instance, each *x, y,* and *channel* value from the output of a convolutional layer) for the model metamer and the corresponding activations for a natural signal:

$$\frac{||A - A'||}{\left||A|\right|}$$

where *A* represents the activations from the natural signal and *A'* represents the activations from the model metamer. The weights of the network remained fixed during the optimization. Optimization was performed with gradient descent on the input signal, which was otherwise unconstrained. Each step of gradient descent was constrained to have a maximum L_2 norm of α, where α was initialized at 1 and was dropped by a factor of 0.5 after every 3000 iterations. Optimization was run for a total of 24000 steps for each generated metamer. For all vision models other than HMAX, the input signal was initialized as a sample from a normal distribution with standard deviation of 0.05 and a mean of 0.5 (natural inputs to the models were scaled between 0-1). The size of the input stimuli, range of pixel values, and normalization parameters were matched to the test data for the model. For all audio models other than the spectrotemporal model, the input signal was initialized from a random normal distribution with standard deviation of 10^{-7} and a mean of zero. The perturbation *s* was initialized to 0. Normalization that occurred after data augmentation in the visual models (subtracting channel means and dividing by channel standard deviations) was included as a model component during metamer generation (i.e., gradients from these operations contributed to the metamer optimization along with all other operations in the model).

The two hand-engineered models (HMAX and the Spectemp model) had different initialization settings. For the HMAX model, whose inputs were scaled to fall between 0-255, metamers were initialized with a random normal distribution with a mean of 127.5 and standard deviation of 10. For the Spectemp model, metamers were initialized with a random normal distribution with a mean of zero and standard deviation of 10^{-5}. Empirically, we found that both of these models contained stages that were difficult to optimize with the same strategy used for the neural networks. For both models, we maintained the learning rate schedule of 24000 total optimization steps with learning rate drops of 0.5 after every 3000 iterations (initialized at a learning rate of 1). However, in both cases we found that optimization was aided by selectively optimizing for subsets of the units (channels) in the early iterations of the optimization process. For the HMAX model, the subsets that were chosen depended on the model stage. For the S1 layer, we randomly choose activations from a single Gabor filter channel to include in the optimization. For the C1 layer, we randomly selected a single scale. And for the S2 and C2 layers we randomly chose a single patch size. The random choice of subset was changed after every 50 gradient steps. This subset-based optimization strategy was used for the first 2000 iterations at each learning rate value. All units were then included for the remaining 1000 iterations for that learning rate value.

For the Spectemp model, we observed that the higher frequency modulation channels were hardest to optimize. We set up a coarse-to-fine optimization strategy by initially only including the lowest frequency spectral and temporal modulation filters in the loss function, and adding in the filters with the next lowest modulation frequencies after every 400 optimization steps (with 7 total sets of filters defined by center frequencies in both temporal and spectral modulation, and the remaining 200/3000 steps continuing to include all of the filters from the optimization). The temporal modulation cutoffs for each of the 7 sets were [0, 0.5, 1, 2, 4, 8, 16] Hz and the spectral modulation cutoffs were [0, 0.0625, 0.125, 0.25, 0.5, 1, 2] cycles/ERB; a filter was included in the nth set if it had either a temporal or spectral scale that was equal to or less than the nth temporal or spectral cutoff, respectively. This strategy was repeated for each learning rate.

Criteria for optimization success
Because metamers are derived via a gradient descent procedure, the activations that they produce approach those of the natural signal used to generate them, but never exactly match. It was thus important to define criteria by which the optimization would be considered sufficiently successful for the result to be considered a model metamer and included in the behavioral experiments.

The first criterion was that the models had to produce the same class label for the model metamer and natural signal. For visual models, the model metamer had to result in the same 16-way classification label as the natural signal to which it was matched. For the auditory models, the model metamer had to result in the same word label (out of 794 possible labels, including "null") as the natural speech signal to which it was matched. For models that did not have a classifier stage (the self-supervised models, HMAX, and the spectrotemporal filter model), we trained a classifier as described above for this purpose. The classifier was included to be conservative, but in practice could be omitted in future work, as very few stimuli pass the matching fidelity criterion but not the classifier criterion.

The second criterion was that the activations for the model metamer had to be matched to those for the natural signal better than would be expected by chance. We measured the fidelity of the match between the activations for the natural stimulus and its model metamer at the matched model stage using three different metrics: Spearman ρ, Pearson R^2, and the Signal-To-Noise Ratio:

$$\mathrm{SNR_{dB}} = 10 \log_{10} \frac{\sum(\mathrm{x}^2)}{\sum((\mathrm{x}-\mathrm{y})^2)}$$

where *x* is the activations for the original sound when comparing metamers, or for a randomly selected sound for the null distribution and *y* is activations for the comparison sound (the model metamer or another randomly selected sound). We then ensured that for each of the three measures, the value for the model metamer fell outside of a null distribution measured between 1,000,000 randomly chosen image or audio pairs from the training dataset. Metamers that did not pass the null distribution test for any of the Spearman ρ, Pearson R^2, or Signal-To-Noise Ratio measured at the layer used for the optimization were excluded from the set of experimental stimuli. The only exception to this was the HMAX model, for which we only used the Signal-To-Noise Ratio for the matching criteria (we found empirically that after the S2 layer of the HMAX model, activations from pairs of random images became strongly correlated due to the different offsets and scales in the natural image patch set, such that the correlation measures were not diagnostic of the match fidelity).

Handling gradients through the ReLU operation
Many neural networks use the ReLU nonlinearity, which yields a partial derivative of zero if the input is negative. We found empirically that it was difficult to match ReLU layers due to the initialization producing many activations of zero. To improve the optimization when generating a metamer for activations immediately following a ReLU, we modified the derivative of the metamer generation layer ReLU to be 1 for all input values, including values below zero (36). ReLU layers that were not the metamer generation layer behaved normally, with a gradient of 0 for input values below zero.

Online behavioral experiments
All behavioral experiments presented in the main text were run on Amazon Mechanical Turk. To increase data quality, Amazon Turk qualifications were set to relatively stringent levels: the "HIT Approval Rate for all Requesters' HITs" had to be at least 97% and the "Number of HITs

Approved" had to exceed 1000. Code to run the online experiments will be released via GitHub upon publication. All experiments with human participants (both online and in-lab) were approved by the Committee On the Use of Humans as Experimental Subjects at the Massachusetts Institute of Technology and were conducted with the informed consent of the participants.

Stimuli - image experiments
Each stimulus belonged to one of the 16 entry-level MS COCO (Microsoft Common Objects in Context) categories. We used a mapping from these 16 categories to the corresponding ImageNet categories (where multiple ImageNet categories can map onto a single MS COCO category), used in a previous publication (18). For each of the 16 categories, we selected 25 examples from the ImageNet validation dataset for a total of 400 natural images that were used to generate stimuli. A square center crop was taken for each ImageNet image (with the smallest dimension of the image determining the size) and the square image was rescaled to the necessary input dimensions for each ImageNet trained network. Metamers were generated for each of the 400 images to use for the behavioral experiments.

Stimuli - auditory experiments
A set of two-second speech audio excerpts with no background noise was used to generate stimuli. These clips were randomly chosen from the test set of the Word-Speaker-Noise dataset described above, constrained such that only the WSJ corpus was used. We further ensured that the clips were taken from unique sources within WSJ, and that the sounds were cropped to the middle two seconds of the training set clip such that the labeled word was centered at the one-second mark. To reduce ambiguity about the clip onset and offset (which were helpful in judging whether a word was centered on the clip), we also screened the chosen clips to ensure that the beginning 0.25s or end 0.25s of the clip was no more than -20dB quieter than the full clip. 400 clips were chosen subject to these constraints and such that each clip contained a different labeled word. Metamers were generated for each of the 400 clips to use for the behavioral experiments.

Image behavioral experiment
We created a visual experiment in JavaScript similar to that used in a previous publication (18). Participants were tasked with classifying an image into one of 16 presented categories (airplane, bear, bicycle, bird, boat, bottle, car, cat, chair, clock, dog, elephant, keyboard, knife, oven, truck). Each category had an associated image icon that participants chose from during the experiment. Each trial began with a fixation cross at the center of the screen for 300ms, followed by a natural image or a model metamer presented at the center of the screen for 300ms, followed by a pink noise mask presented for 300ms, followed by a 4x4 grid containing all 16 icons. Participants selected an image category by clicking on the corresponding icon. To minimize effects of network disruptions, we ensured that the image was loaded into the browser cache before the trial began. We note that the precise timing of the image presentation for web experiments via JavaScript is less controlled compared to in-lab experiments, but any variation in timing should be similar across conditions and should average out over trials. To assess whether any timing variation in the online experiment set up might have affected overall performance, we compared recognition performance on natural images to that measured during in-lab pilot experiments (with the same task but different image examples) reported in an earlier conference paper (36). The average online performance across all natural images was on par or higher than that measured in-lab (in-lab proportion correct = 0.888 ± 0.0240) for all experiments.

The experimental session began with 16 practice trials to introduce participants to the task. There was one trial for each category, each presenting a natural image pulled from the ImageNet training set. Participants received feedback for these first 16 trials which included their answer and the

correct response. Participants then began a 12-trial demo experiment which contained some natural images and some model metamers generated from the ImageNet training set. The goal of this demo experiment was two-fold – first, to introduce participants to the types of stimuli they would see in the main experiment and second, to be used as a screening criterion to remove participants who were distracted, misunderstood the task instructions, had browser incompatibilities, or were otherwise unable to complete the task. Participants were only allowed to start the main experiment if they correctly answered 7/12 correct on the demo experiment, which was the minimum that were correctly answered for these same demo stimuli by 16 in-lab participants in a pilot experiment (36). 143/168 participants passed the demo experiment and chose to move onto the main experiment.

There were 6 different main image experiments, each including a set of conditions (model stages) to be compared. The stimuli for each main experiment were generated from a set of 400 natural images. Participants only saw one natural image or metamer for each of the 400 images in the behavioral stimulus set. Participants additionally completed 16 catch trials. These catch trials each consisted of an image that exactly matched the icon for one of the classes. Participant data was only included in the analysis if the participant got 15/16 of these catch trials correct (123/143 participants were included). Of these participants, 53 self-identified as female and 70 as male; mean age=39.9, minimum age=22, maximum age=78. For all but the HMAX experiment, participants completed 416 trials – one for each of the 400 original images, plus the 16 catch trials. The 400 images were randomly assigned to the experiment conditions subject to the constraint that each condition had approximately the same number of trials. Specifically, each condition was initially allocated ceiling(400/N) trials, and then trials were removed at random until the total number of trials was equal to 400. In addition, if the stimulus for a condition did not pass the metamer optimization criteria (and thus, had to be omitted from the experiment), the natural image was substituted for it as a placeholder, and analyzed as an additional trial for the natural condition. These two constraints resulted in the number of trials per condition varying somewhat across participants (see table below). The resulting 416 total trials were then presented in random order across the conditions of the experiment. The HMAX experiment used only 200 of the original 400 images, for a total of 216 trials. The experiment was run with a smaller number of stimuli than other experiments because it contained only 6 conditions (metamer optimization was run for all 400 stimuli and a subset of 200 images was randomly chosen from the subset of the 400 images for which metamer optimization was successful in every layer of the model). HMAX metamers were black and white, while all metamers from all other models were in color.

Model performance on this 16-way classification task was evaluated by measuring the predictions for the full 1000-way ImageNet classification task and finding the maximum probability for a label that was included in the 16-class dataset (231 classes).

Experiment	Models (number of layers)	Total Number of conditions (Includes Natural Image)	Number of trials per condition per participant Average (min, max)	Number of Participants
Visual Experiment 1 (Standard Models)	CORnet-S VGG-19 ResNet50 ResNet101 AlexNet	37	10 (8, 14)	22

Visual Experiment 2 (Self-Supervised Models)	ResNet50 Standard Supervised ResNet50 SimCLR ResNet50 MoCo_V2 ResNe50 BYOL	29	13 (11, 20)	21
Visual Experiment 3 (HMAX)	HMAX	6	33 (31, 34)	20
Visual Experiment 4 (ResNet50 Adversarially Robust)	ResNet50 Standard Supervised, ResNet50 L_2-norm ($\epsilon = 3$) ResNet50 random L_2-norm ($\epsilon = 3$) ResNet50 L_∞-norm ($\epsilon = 4/255$) ResNet50 L_∞-norm ($\epsilon = 8/255$) ResNet50 random L_∞-norm ($\epsilon = 8/255$)	43	9 (6, 13)	20
Visual Experiment 5 (AlexNet Adversarially Robust)	AlexNet Standard Supervised, AlexNet L_2-norm ($\epsilon = 3$) AlexNet random L_2-norm ($\epsilon = 3$) AlexNet L_∞-norm ($\epsilon = 8/255$) AlexNet random L_∞-norm ($\epsilon = 8/255$)	41	9 (7, 14)	20
Visual Experiment 6 (Lowpass AlexNet and VOneAlexNet)	AlexNet Standard Supervised Lowpass AlexNet VOneAlexNet	25	16 (14, 20)	20

Table 3: Conditions and number of trials included in each visual experiment.

Audio behavioral experiment

We developed an audio experiment in JavaScript that was similar to an experiment used in earlier publications from our lab (5, 36). Each human participant listened to a two-second audio clip and had to choose one of 793 word-labels corresponding to the word in the middle of the clip (centered at the one-second mark of the clip). Each trial began with the participant hearing the audio clip and typing the word they thought they heard into a response box. As participants typed, words matching the letter string they were typing appeared below the response box to help participants identify words that were in the response list. Once a word was typed that matched one of the 793 responses, participants could move onto the next trial.

To increase data quality, participants first completed a short experiment (six trials) that screened for the use of headphones (99). If participants scored 5/6 or higher on this screen (224/377 participants), they moved onto a practice experiment consisting of 10 natural audio trials with feedback (drawn from the training set), designed to introduce the task. This was followed by a demo experiment of 12 trials without feedback. These 12 trials contained both natural audio and model metamers, using the same set of 12 stimuli as in a demo experiment used for earlier in-lab experiments (36). As with the visual demo experiment, the goal of the audio demo experiment

was to introduce participants to the type of stimuli they would hear in the main experiment and to screen out poorly performing participants. A screening criteria was set at 5/12, which was the minimum for 16 in-lab participants in earlier work (36). 154/224 participants passed the demo experiment and chose to move onto the main experiment. We have repeatedly found that online auditory psychophysical experiments qualitatively and quantitatively reproduce in-lab results, provided that steps such as these are taken to help ensure good audio presentation quality and attentive participants (100–103). The average online performance on natural stimuli was comparable to in-lab performance reported in (36) on natural stimuli using the same task with different audio clips (in-lab proportion correct = 0.863 ± 0.0340).

There were 6 different main auditory experiments, each including a set of conditions (multiple models and multiple model stages) to be compared. The design of these experiments paralleled the image experiments. The stimuli for each main experiment were generated from the set of 400 natural speech behavioral stimuli described above. Participants only heard one natural speech or metamer stimulus for each of the 400 excerpts in the behavioral stimulus set. Participants additionally completed 16 catch trials. These catch trials each consisted of a single word corresponding to one of the classes. Participant data was only included in the analysis if the participant got 15/16 of these trials correct (this inclusion criterion removed 8/154 participants). As the audio experiment was long, some participants chose to leave the experiment early and their data was excluded from analysis (23/154). An additional 3 participants were excluded due to self-reported hearing loss, for a total of 120 participants across all audio experiments. Of these participants, 45 self-identified as female, 68 as male, and 7 chose not to report; mean age=39.0, minimum age=22, maximum age=77. For all but the Spectemp experiment, participants completed 416 trials – one for each of the 400 original excerpts, plus the 16 catch trials. The 400 excerpts were randomly assigned to the experiment conditions subject to the constraint that each condition had approximately the same number of trials. As in the visual experiments, each condition was initially allocated ceiling(400/N) trials, and then trials were removed at random until the total number of trials was equal to 400. If the chosen network-layer condition pair corresponded to a metamer that did not pass the metamer optimization criteria (and thus, was omitted from experiment stimuli), the natural audio was used for that condition as a placeholder, but was not included in the analysis. As in the visual experiments, these two constraints resulted in the number of trials per condition varying somewhat across participants (see Table 4). The resulting 416 total trials were then presented in random order across the conditions of the experiment. The Spectemp experiment used only 200 of the original 400 excerpts, for a total of 216 trials. This experiment was run with a smaller number of stimuli because it contained only 6 conditions (metamer optimization was run for all 400 stimuli and a subset of 200 was randomly chosen from the subset of the 400 original excerpts for which metamer optimization was successful in every layer of the model). We collected online data in batches until we reached the target number of participants for each experiment.

Experiment	Models (number of layers)	Total Number of conditions (Includes Natural Image)	Number of trials per condition per participant Average (min, max)	Number of Participants
Audio Experiment 1 (Standard Models)	CochResNet50 Standard CochCNN9 Standard	18	22 (18, 23)	20
Audio Experiment 2 (Spectemp Model)	Spectemp Model	6	33 (30, 34)	20

Audio Experiment 3 (CochResNet50 Waveform Adversarial Training)	CochResNet50 Standard Supervised, CochResNet50 waveform L_2-norm ($\epsilon = 0.5$) CochResNet50 waveform L_2-norm ($\epsilon = 1$) CochResNet50 random waveform L_2-norm ($\epsilon = 1$) CochResNet50 waveform L_∞-norm ($\epsilon = 0.002$) CochResNet50 random waveform L_∞-norm ($\epsilon = 0.002$)	49	8 (5, 9)	20
Audio Experiment 4 (CochCNN9 Waveform Adversarial Training)	CochCNN9 Standard Supervised, CochCNN9 waveform L_2-norm ($\epsilon = 1$) CochCNN9 random waveform L_2-norm ($\epsilon = 1$) CochCNN9 waveform L_∞-norm ($\epsilon = 0.002$) CochCNN9 random waveform L_∞-norm ($\epsilon = 0.002$)	46	8 (6, 9)	20
Audio Experiment 5 (CochResNet50 Cochleagram Adversarial Training)	CochResNet50 Standard Supervised, CochResNet50 cochleagram L_2-norm ($\epsilon = 0.5$) CochResNet50 cochleagram L_2-norm ($\epsilon = 1$) CochResNet50 random cochleagram L_2-norm ($\epsilon = 1$) CochResNet50 waveform L_2-norm ($\epsilon = 1$)	41	9 (7, 10)	20
Audio Experiment 6 (CochCNN9 Cochleagram Adversarial Training)	CochCNN9 Standard Supervised, CochCNN9 cochleagram L_2-norm ($\epsilon = 0.5$) CochCNN9 cochleagram L_2-norm ($\epsilon = 1$) CochCNN9 random cochleagram L_2-norm ($\epsilon = 1$) CochCNN9 waveform L_2-norm ($\epsilon = 1$)	46	8 (6, 9)	20

Table 4: Conditions and number of trials included in each audio experiment.

Statistical tests -- difference between human and model recognition accuracy
All human recognition experiments involving neural network models were analyzed by comparing human recognition of a generating model's metamers to the generating model's recognition of the same stimuli (its own metamers). Each human participant was run on a distinct set of model metamers; we presented each set to the generation model and measured its recognition performance for that set. Thus, if N human participants performed an experiment, we obtained N model recognition curves. We ran mixed model repeated measures ANOVAs with a within-group factor of metamer generation model stage and a between-group factor of observer (human or model observer), testing for both a main effect of observer and an interaction between observer and model stage. Data were non-normal due to a prevalence of values close to 1 or 0 depending on the condition, and so we evaluated statistical significance non-parametrically, using permutation tests in which we compared the observed F-statistic to that obtained after randomly permuting the data labels. To test for main effects, we permuted observer labels (model vs. human). To test for interactions of observer and model stage, we randomly permuted both the observer labels and the model stage labels, independently for each participant. In each case we used 10,000 random permutations and computed a p value by comparing the observed F-statistic to the null distribution of F-statistics from permuted data (i.e., the p value was one minus the rank of the observed F-statistic divided by the number of permutations).

Because the classical models did not themselves perform recognition judgments, rather than comparing human and model recognition as in the experiments involving neural network models, we instead tested for a main effect of model stage on human observer recognition. We performed a single-factor repeated measure ANOVA using a within-group factor of model stage, again evaluating statistical significance non-parametrically. We randomly permuted the model stage labels of the recognition accuracy data, independently for each participant (with 10,000 random permutations).

Statistical tests -- difference between human recognition of metamers generated from different models
To compare human recognition of metamers generated from different models, we ran a repeated measures ANOVA with within-group factors of model stage and generating model. This type of comparison was only performed in cases where the generating models had the same architecture (so that the model stages were shared between models). We again evaluated statistical significance non-parametrically, by comparing the observed F-statistic to a null distribution of F-statistics from permuted data (10,000 random permutations). To test for a main effect of generating model, we randomly permuting the generating model label, independently for each participant. To test for an interaction between the generating model and model stage, we permuted both the generating model label and the model stage label, independently for each participant.

Power analysis to determine sample sizes
We planned to run ANOVAs examining the interaction between human performance and model performance, the interaction between model and layer, the interaction between standard models and adversarially trained models, the interaction between different types of adversarial training, and the interaction between standard supervised models and self-supervised models. To estimate the number of participants necessary to be well powered for these analyses, we ran a pilot experiment comparing the standard versus adversarially trained ResNet50 and CochResNet50 models, as this experiment included the largest number of conditions and we expected that the

differences between different types of adversarially trained models would be subtle, and would put an upper bound on the sample sizes that would be needed across experiments.

For the vision experiment, we ran 10 participants in a pilot experiment on Amazon Mechanical Turk. The format of the pilot experiment was identical to that of the main experiments in this paper, with the exception that we used a screening criteria of 8/12 correct for the pilot, rather than the 7/12 correct used for the main experiment. In this pilot experiment, the smallest effect size out of those we anticipated analyzing in the main experiments was the comparison between the L_∞-norm ($\epsilon = 8/256$) adversarially trained ResNet50 and the L_2-norm ($\epsilon = 3$) adversarially trained Resnet50, with a partial eta squared value of 0.10 for the interaction. A power analysis was performed with g*power (104); 18 participants were needed to have a 95% chance of seeing an effect of this size at a p<.01 significance level. We thus set a target of 20 participants for each main vision experiment.

For the auditory experiments we ran 14 participants in a pilot experiment on Amazon Mechanical Turk. The format of the pilot experiment was identical to that of the main experiments in this paper with the exception that 8 of the 14 participants only received 6 original audio trials with feedback, while in the main experiment 10 trials with feedback were used. As in the vision pilot experiment, in this pilot the smallest effect size of interest was that for the comparison between the L_∞-norm ($\epsilon = 0.002$) adversarially trained CochResNet50 and the L_2-norm ($\epsilon = 1$) waveform adversarially trained CochResNet50, yielding a partial eta squared value of 0.37 for the interaction. A power analysis was performed with g*power; this indicated that 12 participants were needed to have a 95% change of seeing an effect for this size at a p<0.01 significance level. To match the image experiments, we set a target of 20 participants for each main auditory experiment.

Experiment testing effect of optimization algorithm (Supplementary Figure 2)
An in-lab visual experiment was conducted to analyze whether differences in the optimization strategy would have an effect on the human-recognizability of model metamers. This experiment used one example visual model (a ResNet50 architecture). The optimization code and techniques used for this small in-lab experiment differed from any of the experiments described in the main text of this paper (they followed the methods used in our previous work (36)) but showed a similar main effect of layer for standard networks, where model metamers generated from late stages of ImageNet task-optimized ResNet50 networks were unrecognizable to humans.

Visual models included in the experiment
The models and metamer generation for this experiment were implemented in TensorFlow v1.12 (105). For these experiments, we converted the ResNet50 model available via the PyTorch Model Zoo to TensorFlow using ONNX (version 1.6.0).

The experiment was run together with an unrelated pilot experiment comparing four other models, the data for which are not analyzed here. To reduce the number of conditions, the layer corresponding to ResNet50 “layer1” was not included in metamer generation or in the experiment (from ad-hoc visual inspection of a few examples, the model metamers for conv1_relu1, layer1, and layer2 looked similar enough to the natural image that we expected ceiling performance for all conditions).

Model metamer optimization
Metamers were optimized using TensorFlow code based on that used for a previous conference paper (36). Unless otherwise noted the methods were identical to those used in the main experiments of this paper.

We tested two different optimization schemes for generating model metamers. The first used stochastic gradient descent with 15000 iterations, where each step of gradient descent was constrained to have an L_2 norm of 1. The second method used the Adam optimizer (106), which uses an adaptive estimation of first and second order moments of the gradient. Metamers were generated with 15000 iterations of the Adam optimizer with an exponentially decaying learning rate (initial learning rate of 0.001, 1000 decay steps, and a decay rate of 0.95).

Stimuli – Image dataset for optimization strategy experiment
Similar to the other visual experiments in this paper, each stimulus belonged to one of the 16 entry-level MS COCO categories. For each of the 16 categories, we randomly selected 16 examples from the ImageNet training dataset using the list of images provided by (18) for a total of 256 natural images that were used to generate stimuli. Metamers were generated for each of the 256 images to use for the behavioral experiments.

In-lab visual experiment
The experimental setup was the same as the experiment described for our main experiments (but implemented in MATLAB Psychotoolbox rather than JavaScript), in which participants had to classify an image into one of the 16 presented categories. Each trial began with a fixation cross at the center of the screen for 300ms, followed by a natural image or a model metamer presented at the center of the screen for 200ms, followed by a pink noise mask presented for 200ms, followed by a 4x4 grid containing all 16 icons. Stimuli were presented on a 20" ACER LCD (backlit LED) monitor with a spatial resolution of 1600x900 and a refresh rate of 60Hz. Stimuli spanned 256x256 pixels and were viewed at a distance of approximately 62 cm (set by the chair position, which was fixed; participants were free to position themselves in the chair as was most comfortable, which introduced minor variation in viewing distance).

Before the experiment, each participant was shown a printout of the 16 category images with labels and the experimenter pointed to and read each category. This was followed by a demo experiment with 12 trials without feedback (same stimuli as in the main experiments, but performance was not used to exclude participants).

Each participant saw 6 examples from each condition, chosen such that each natural image or metamer was from a unique image from the 256-image behavioral set. Ten participants completed the experiment.

fMRI Analysis
The auditory fMRI analysis closely followed that of a previous publication (5) using the fMRI dataset collected in another previous publication (59).The key components of the dataset and analysis methods are replicated here, but for additional details see (5, 59). The text from sections "*fMRI data acquisition and preprocessing*" and "*Voxel and ROI selection"* is replicated from a previous publication with minor edits (5).

Natural sound stimuli
The stimulus set was composed of 165 two-second natural sounds spanning eleven categories (instrumental music, music with vocals, English speech, foreign speech, non-speed vocal sounds, animal vocalization, human-non-vocal sound, animal non-vocal sound, nature sound, mechanical sound, or environment sound). The sounds were presented in a block design with five presentations of each two-second sound. To prevent sounds from being played at the same time as scanner noise, a single fMRI volume was collected following each sound presentation ("sparse scanning"). This resulted in a 17-second block. Blocks were grouped into 11 runs with 15 stimuli each and four blocks of silence. Silence blocks were the same duration as the stimulus blocks

and were used to estimate the baseline response. Participants performed a sound-intensity discrimination task to increase attention. One sound in the block of five was presented 7dB lower than the other four (the quieter sound was never the first sound) and participants were instructed to press a button when they heard the quieter sound. Sounds were presented with MR-compatible earphones (Sensimetrics S14) at 75 dB SPL for the louder sounds and 68dB SPL for the quieter sounds.

fMRI data acquisition and preprocessing

MR data was collected on a 3T Simens Trio scanner with a 32-channel head coil at the Athinoula A. Martinos Imaging Center of the McGovern Institute for Brain Research at MIT. Data was first published in (59), and was re-analyzed for this paper. Each functional volume consisted of fifteen slices oriented parallel to the superior temporal plane, covering the portion of the temporal lobe superior to and including the superior temporal sulcus. Repetition time (TR) was 3.4 s (acquisition time was only 1 s due to sparse scanning), echo time (TE) was 30 ms, and flip angle was 90 degrees. For each run, the five initial volumes were discarded to allow homogenization of the magnetic field. In-plane resolution was 2.1 x 2.1 mm (96 x 96 matrix), and slice thickness was 4 mm with a 10% gap, yielding a voxel size of 2.1 x 2.1 x 4.4 mm. iPAT was used to minimize acquisition time. T1-weighted anatomical images were collected in each subject (1mm isotropic voxels) for alignment and surface reconstruction.

Functional volumes were preprocessed using FSL and in-house MATLAB scripts. Volumes were corrected for motion and slice time. Volumes were skull-stripped, and voxel time courses were linearly detrended. Each run was aligned to the anatomical volume using FLIRT and BBRegister. These preprocessed functional volumes were then resampled to vertices on the reconstructed cortical surface computed via FreeSurfer, and were smoothed on the surface with a 3mm FWHM 2D Gaussian kernel to improve SNR. All analyses were done in this surface space, but for ease of discussion we refer to vertices as ‘‘voxels’’ in this paper. For each of the three scan sessions, we estimated the mean response of each voxel (in the surface space) to each stimulus block by averaging the response of the second through the fifth acquisitions after the onset of each block (the first acquisition was excluded to account for the hemodynamic lag). Pilot analyses showed similar response estimates from a more traditional GLM (59). These signal-averaged responses were converted to percent signal change (PSC) by subtracting and dividing by each voxel’s response to the blocks of silence. These PSC values were then downsampled from the surface space to a 2mm isotropic grid on the FreeSurfer-flattened cortical sheet.

Voxel and ROI selection

We used the same voxel selection criterion as Norman-Haignere et al. (2015), selecting voxels with a consistent response to sounds from a large anatomical constraint region encompassing the superior temporal and posterior parietal cortex. Specifically, we used two criteria: (1) a significant response to sounds compared with silence ($p < 0.001$); and (2) a reliable response to the pattern of 165 sounds across scans. The reliability measure was as follows:

$$r = 1 - \frac{||v_{12} - proj_{v_3} v_{12}||^2}{||v_{12}||^2}$$

$$proj_{v_3} v_{12} = \left(\frac{v_3^T}{||v_3||^2} v_{12} \right) v_3$$

where v_{12} is the response of a single voxel to the 165 sounds averaged across the first two scans (a vector), and v_3 is that same voxel’s response measured in the third. The numerator in the second term in the first equation is the magnitude of the residual left in v_{12} after projecting out

the response shared with v_3. This ''residual magnitude'' is divided by its maximum possible value (the magnitude of v_{12}). The measure is bounded between 0 and 1, but differs from a correlation in assigning high values to voxels with a consistent response to the sound set, even if the response does not vary substantially across sounds. We found that using a more traditional correlation-based reliability measure excluded many voxels in primary auditory cortex because some of them exhibit only modest response variation across natural sounds. We included voxels with a value of this modified reliability measure of 0.3 or higher, which when combined with the sound responsive t-test yielded a total of 7694 voxels across the eight participants (mean number of voxels per subject: 961.75; range: 637-1221).

We localized four regions of interest (ROIs) in each participant, consisting of voxels selective for (1) frequency (i.e., tonotopy), (2) pitch, (3) speech, and (4) music. In each case we ran a ''localizer'' statistical test and selected the top 5% most significant individual voxels in each subject and hemisphere (including all voxels identified by the sound-responsive and reliability criteria described above). We excluded voxels that were identified in this way by more than one localizer. The frequency, pitch, and speech localizers required acquiring additional imaging data, and were collected either during extra time during the natural sound stimuli scan sessions or on additional sessions on different days. Scanning acquisition parameters were identical to those used to acquire the natural sounds data. Throughout this paper we refer to voxels chosen by these criteria as ''selective,'' for ease and consistency.

To identify frequency-selective voxels, we measured responses to pure tones in six different frequency ranges (center frequencies: 200, 400, 800, 1600, 3200, 6400 Hz) (107, 108). For each voxel, we ran a one-way ANOVA on its response to each of these six frequency ranges and selected voxels that were significantly modulated by pure tones (top 5% of all selected voxels in each subject, ranked by p values). Although there was no spatial contiguity constraint built into our selection method, in practice most selected voxels were contiguous and centered around Heschl's gyrus.

To identify pitch-selective voxels, we measured responses to harmonic tones and spectrally-matched noise (108). For each voxel we ran a one-tailed t-test evaluating whether the response to tones was greater than that to noise. We selected the top 5% of individual voxels in each subject that had the lowest p values for this contrast.

To identify speech-selective voxels, we measured responses to German speech and to temporally scrambled (''quilted'') speech stimuli generated from the same German source recordings (109). We used foreign speech to identify responses to speech acoustical structure, independent of linguistic structure. Note that two of the participants had studied German in school and for one of these participants we used Russian utterances instead of German. The other subject was tested with German because the Russian stimuli were not available at the time of the scan. For each voxel we ran a one-tailed t-test evaluating whether responses were higher to intact speech than to statistically matched quilts. We selected the top 5% of all selected voxels in each subject.

To identify music-selective voxels, we used the music component derived by (59). We inferred the ''voxel weights'' for each voxel to all six of the components from its response to the 165 sounds:

$$w = C'v$$

where w contains the inferred voxel weights (a vector of length 6), C' is the Moore-Penrose pseudoinverse of the ''response components'' (a 6 by 165 matrix), and v is the measured response of a given voxel (vector of length 165). We assessed the significance of each voxel's music component weight via a permutation test. During each iteration, we shuffled all the

component elements, recomputed this new matrix's pseudoinverse, and recomputed each voxel's weights via the matrix multiply above. We performed this procedure 10,000 times, and fit a Gaussian to each voxel's null distribution of music weights. We then calculated the likelihood of the empirically observed voxel weight from this null distribution, and took the top 5% of voxels with the lowest likelihood under this null distribution.

Voxelwise encoding analysis

Each of the 165 sounds from the fMRI experiment was resampled to 20,000Hz (the sampling rate of the training dataset) and passed through each auditory model. We measured the model response at each layer of the network for each sound. To compare the network responses to the fMRI response we averaged over the time dimension for all units that had a temporal dimension (all layers except fully connected layers).

In the CochCNN9 architecture, this resulted in the following number of regressors for each layer: cochleagram (211), relu0 (6818), relu1 (4608), relu2 (4608), relu3 (9216), relu4 (4608), avgpool (2560), relufc (4096), final (794). In the CochResNet50 architecture, this resulted in the following number of regressors for each layer: cochleagram (211), conv1_relu1 (6784), layer1 (13568), layer2 (13824), layer3 (14336), layer4 (14336), avgpool (2048), final (794). The spectrotemporal model consisted of only the time-averaged power of the spectrotemporal filters (23421 activations).

We used the features sets extracted from model layers to predict the fMRI response to the natural sound stimuli. Each voxel's time-averaged responses were modeled as a linear combination of a layer's time-averaged unit responses. Ten random train-test splits (83/82) were taken from the 165 sound dataset. For each split we estimated a linear mapping using L2-regularized ("ridge") linear regression using RidgeCV from the scikit learn library version 0.23.1 (110). Ridge regression places a zero-mean gaussian prior on the regression coefficients, corresponding to solving the regression problem given by:

$$w = (X^T X + n\lambda I)^{-1} X^T y$$

where n is the number of stimuli used for estimation (here equal to 83), w is the d-length column vector the length of the regressors (number of extracted layer features), y is an n-length column vector containing the voxels response to each sound (length n), X is a matrix containing the regressors (n stimuli by d regressors, the extracted layer features for each sound), I is the identity matrix, and λ is the ridge regression regularization parameter. The mean of each column was subtracted from the regressor matrix before fitting the model.

The best ridge regression parameter for each voxel was independently selected using leave-one-out cross validation across the 83 training sounds in each split sweeping over 81 logarithmically spaced values (each power of 10 between $10^{-40} – 10^{40}$). For each held-out sound in the training set, the mean squared error of the prediction was computed using regression weights from the other 82 training set sounds, for each of the regularization coefficients. The regularization parameter that minimized the mean squared error on the held-out sounds was used to fit a linear mapping between model responses to all 83 training set sounds and the voxel responses. This mapping was used to predict the voxel response to the held-out 82 test sounds and fitting fidelity was evaluated with the squared Pearson correlation (r^2) as a metric of explained variance of the predicted voxel response and the observed voxel response.

This measured explained variance was corrected for the effects of measurement noise by computing the reliability of the voxel responses and of the predicted voxel response. We use the correction for attenuation (111), which estimates the correlation between two variables

independent of measurement noise, resulting in an unbiased estimator of the correlation coefficient that would be observed from noiseless data. The corrected variance explained is:

$$r_{v,\hat{v}}^{2*} = \frac{r(v_{123}, \hat{v}_{123})^2}{r_v' r_{\hat{v}}'}$$

where v_{123} is the voxel response to the 82 test sounds averaged across all three scans, $\hat{v}_{123}$ is the predicted response to the 82 test sounds using the 83 training sounds to learn the regression weights, r is a function that computes the Pearson correlation coefficient, r_v' is the correction for reliability of the voxel responses, and $r_{\hat{v}}'$ is the correction for reliability of the predicted voxel response. r_v' is computed as the median Spearman-Brown corrected Pearson correlation between the 3 pairs of scans (scan 0 with scan 1, scan 1 with scan 2, scan 0 with scan 2), where the Spearman-Brown correction accounts for increased reliability expected from tripling the amount of data (112). $r_{\hat{v}}'$ is similarly computed by using the training data each of the three pairs of scans to predict the test data from the same scan, and calculating the median Spearman-Brown corrected correlation for the predicted voxel responses. If voxels and/or predictions are very unreliable, this can lead to large corrected variance explained measures (113). We set a minimum value of 0.182 for r_v' (corresponding to the value at which the correlation of two 83-dimensional random variables reaches significance at a threshold of $p<.05$; 83 being the number of training data values) and a minimum value of 0.183 for $r_{\hat{v}}'$ (the analogous value for 82-dimensional random variables, corresponding to the number test data values).

The above computation of corrected variance explained was computed for each voxel using each layer response for each of 10 train/test splits of data. We took the median variance explained across the 10 splits of data. To provide one summary metric across each of the ROIs (all auditory voxels, tonotopic voxels, pitch voxels, music voxels, speech voxels) we chose the best layer for each ROI using a hold-one-participant out analysis. A summary measure for each participant and model layer was computed by taking the median across all voxels of the voxel-wise corrected variance explained values within the ROI. Holding out one participant, we averaged across the remaining participant values to find the layer with the highest variance explained within the given ROI. We measured the corrected variance explained for this layer in the held-out participant and repeated the procedure holding out one participant at a time. This cross validation avoids issues of non-independence when selecting the layer. We report the mean corrected variance explained across the participants (Figure 8). In the spectrotemporal filter model, we only analyzed the power of the spectrotemporal filter responses, as this is the model stage that is standardly used when analyzing fMRI data in this way. One sided paired sample t-tests were performed to test for a difference in the variance explained summary metrics between different networks. Significance values were computed by constructing a null-distribution of t-statistics by randomly permuting the network assignment independently for each participant (10,000 permutations). The p-value is reported as one minus the rank of the observed t-statistic divided by the number of permutations. In cases where the maximum possible number of unique permutations was less than 10,000, we instead divided the rank by the maximum number of unique permutations

Model recognition of metamers generated from other models

To measure the recognition of a model's metamers by other models, we took the generated image or audio that was used for the human behavioral experiments, provided it as input to a "recognition" neural network, and measured the 16 way image classification (for ImageNet models) or the 763-way word classification (for the audio models).

The plots in Figure 9b show the average recognition by other models of metamers generated from a particular type of ResNet50 model. We used one "Standard" generation model (the standard supervised ResNet50). This curve is the average across all other vision recognition models (as

shown in Figure 9a). The curve for self-supervised models shows results averaged across the three self-supervised generation models (SimCLR, MoCo_V2, and BYOL); the curve for adversarially trained models shows results averaged across the three adversarially trained ResNet50 models (trained with L_2-norm ($\epsilon = 3$), L_∞-norm ($\epsilon = 4/255$), and L_∞-norm ($\epsilon = 8/255$) perturbations, respectively). For these latter two curves, we first computed the average curve for each recognition model across all three generation models, omitting the recognition model from the average if it was the same as the generation model (in practice, this meant that there was one fewer value included in the average for the recognition models that are part of the generation model group). We then averaged across the curves for each recognition model. The error bars on the resulting curves are the SEM computed across the recognition models.

Figure 9c was generated in an analogous fashion. We used one "Standard" generation model (the standard supervised CochResNet50). The curve for the waveform adversarially trained models shows results averaged across the three such CochResNet50 models (trained with L_2-norm ($\epsilon = 0.5$), L_2-norm ($\epsilon = 1$), and L_∞-norm ($\epsilon = 0.002$) perturbations, respectively). The curve for the cochleagram adversarially trained models shows results averaged across the two such CochResNet50 models (trained with L_2-norm ($\epsilon = 0.5$), and L_2-norm ($\epsilon = 1$) perturbations, respectively). The group averages and error bars were computed as in Figure 9b.

Acknowledgements
We thank Ray Gonzalez for help constructing the Word-Speaker-Noise dataset used for training. We also thank Ray Gonzalez and Alex Durango for help running in-lab experiments and Joel Dapello for guidance on the VOneNet models. We thank Andrew Francl and Mark Saddler for advice on model training and evaluation, Alex Kell and Sam Norman-Haignere for help with the fMRI data analysis, and Malinda McPherson for help with Amazon Turk experiment design and statistics decisions. This work was supported by NSF grant no. BCS-1634050 to J.H.M., National Institutes of Health grant no. R01DC017970 to J.H.M., a DOE CSGF fellowship under grant no. DE-FG02-97ER25308 to J.F., and a Friends of the McGovern Institute Fellowship to J.F.

References

1. D. H. Hubel, T. N. Wiesel, Receptive fields and functional architecture in two nonstriate visual areas (18 and 19) of the cat. *J. Neurophysiol.* **28**, 229–289 (1965).
2. D. J. Felleman, D. C. Van Essen, Distributed hierarchical processing in the primate cerebral cortex. *Cereb. Cortex* **1**, 1–47 (1991).
3. K. Fukushima, Neocognitron: a self organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biol. Cybern.* **36**, 193–202 (1980).
4. T. Serre, A. Oliva, T. Poggio, A feedforward architecture accounts for rapid categorization. *Proc. Natl. Acad. Sci. U. S. A.* **104**, 6424–6429 (2007).
5. A. J. E. Kell, D. L. K. Yamins, E. N. Shook, S. V. Norman-Haignere, J. H. McDermott, A task-optimized neural network replicates human auditory behavior, predicts brain responses, and reveals a cortical processing hierarchy. *Neuron* **98**, 630-644.e16 (2018).
6. N. Kriegeskorte, Deep neural networks: A new framework for modeling biological vision and brain information processing. *Annu. Rev. Vis. Sci.* **1**, 417–446 (2015).
7. D. L. K. Yamins, J. J. DiCarlo, Using goal-driven deep learning models to understand sensory cortex. *Nat. Neurosci.* **19**, 356–365 (2016).
8. D. G. Barrett, A. S. Morcos, J. H. Macke, Analyzing biological and artificial neural networks: challenges with opportunities for synergy? *Curr. Opin. Neurobiol.* **55**, 55–64 (2019).
9. A. H. Marblestone, G. Wayne, K. P. Kording, Toward an integration of deep learning and neuroscience. *Front. Comput. Neurosci.* **10**, 94 (2016).
10. Y. Xu, M. Vaziri-Pashkam, Limits to visual representational correspondence between convolutional neural networks and the human brain. *Nat. Commun.* **12**, 2065 (2021).
11. B. M. Lake, T. D. Ullman, J. B. Tenenbaum, S. J. Gershman, Building machines that learn and think like people. *Behav. Brain Sci.* **40**, e253 (2017).
12. A. Tacchetti, L. Isik, T. A. Poggio, Invariant recognition shapes neural representations of visual input. *Annu. Rev. Vis. Sci.* **4**, 403–422 (2018).
13. S.-M. Khaligh-Razavi, N. Kriegeskorte, Deep supervised, but not unsupervised, models may explain IT cortical representation. *PLoS Comput. Biol.* **10**, e1003915 (2014).
14. I. Goodfellow, H. Lee, Q. Le, A. Saxe, A. Ng, Measuring Invariances in Deep Networks in *Advances in Neural Information Processing Systems*, Y. Bengio, D. Schuurmans, J. Lafferty, C. Williams, A. Culotta, Eds. (Curran Associates, Inc., 2009), pp. 646–654.
15. J. J. DiCarlo, D. D. Cox, Untangling invariant object recognition. *Trends Cogn. Sci.* **11**, 333–341 (2007).
16. M. Riesenhuber, T. Poggio, Hierarchical models of object recognition in cortex. *Nat. Neurosci.* **2**, 1019–1025 (1999).
17. N. C. Rust, J. J. Dicarlo, Selectivity and tolerance (“invariance”) both increase as visual information propagates from cortical area V4 to IT. *J. Neurosci.* **30**, 12978–12995 (2010).
18. R. Geirhos, C. R. M. Temme, J. Rauber, Generalisation in humans and deep neural networks. *Adv. Neural Inf. Process. Syst.* (2018).
19. A. Berardino, J. Ballé, V. Laparra, E. P. Simoncelli, Eigen-Distortions of Hierarchical Representations. *arXiv [cs.CV]* (2017).

20. H. Jang, D. McCormack, F. Tong, Noise-trained deep neural networks effectively predict human vision and its neural responses to challenging images. *PLoS Biol.* **19**, e3001418 (2021).

21. R. Zhang, Making convolutional networks shift-invariant again. *arXiv [cs.CV]* (2019).

22. A. Azulay, Y. Weiss, Why do deep convolutional networks generalize so poorly to small image transformations? *arXiv [cs.CV]* (2018).

23. A. Nguyen, J. Yosinski, J. Clune, Deep neural networks are easily fooled: High confidence predictions for unrecognizable images in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (IEEE, 2015) https:/doi.org/10.1109/cvpr.2015.7298640.

24. C. Szegedy, *et al.*, Intriguing properties of neural networks in *2nd International Conference on Learning Representations, ICLR 2014*, (2014) (February 9, 2022).

25. B. Biggio, *et al.*, Evasion Attacks against Machine Learning at Test Time in *Machine Learning and Knowledge Discovery in Databases*, (Springer Berlin Heidelberg, 2013), pp. 387–402.

26. N. Carlini, D. Wagner, Audio Adversarial Examples: Targeted Attacks on Speech-to-Text in *2018 IEEE Security and Privacy Workshops (SPW)*, (2018), pp. 1–7.

27. B. A. Wandell, *Foundations of vision* (Sinauer Associates, 1995).

28. G. Wyszecki, W. S. Stiles, *Color science* (Wiley New York, 1982).

29. B. Julesz, Visual Pattern Discrimination. *IEEE Trans. Inf. Theory* **8**, 84–92 (1962).

30. J. H. McDermott, M. Schemitsch, E. P. Simoncelli, Summary statistics in auditory perception. *Nat. Neurosci.* **16**, 493–498 (2013).

31. C. M. Ziemba, E. P. Simoncelli, Opposing effects of selectivity and invariance in peripheral vision. *Nat. Commun.* **12**, 4597 (2021).

32. J. M. Hillis, M. O. Ernst, M. S. Banks, M. S. Landy, Combining sensory information: mandatory fusion within, but not between, senses. *Science* **298**, 1627–1630 (2002).

33. H. Sohn, M. Jazayeri, Validating model-based Bayesian integration using prior–cost metamers. *PNAS* (2021) (March 20, 2022).

34. B. Balas, L. Nakano, R. Rosenholtz, A summary-statistic representation in peripheral vision explains visual crowding. *J. Vis.* **9**, 13.1-18 (2009).

35. J. Freeman, E. P. Simoncelli, Metamers of the ventral stream. *Nat. Neurosci.* **14**, 1195–1201 (2011).

36. J. Feather, A. Durango, R. Gonzalez, J. McDermott, Metamers of neural networks reveal divergence from human perceptual systems in *Advances in Neural Information Processing Systems*, (2019) (March 16, 2021).

37. J. Deng, *et al.*, ImageNet: A large-scale hierarchical image database in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, (IEEE, 2009) https:/doi.org/10.1109/cvpr.2009.5206848.

38. M. Schrimpf, *et al.*, Brain-Score: Which Artificial Neural Network for Object Recognition is most Brain-Like? *bioRxiv* (2018) https:/doi.org/10.1101/407007.

39. K. Simonyan, A. Zisserman, Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv [cs.CV]* (2014).

40. K. He, X. Zhang, S. Ren, J. Sun, Identity mappings in deep residual networks. *arXiv [cs.CV]* (2016).

41. A. Krizhevsky, I. Sutskever, G. E. Hinton, ImageNet classification with deep convolutional neural networks in *Advances in Neural Information Processing Systems*, (2012), pp. 1097–1105.

42. M. Schrimpf, *et al.*, Integrative benchmarking to advance neurally mechanistic models of human intelligence. *Neuron* **108**, 413–423 (2020).

43. B. R. Glasberg, B. C. J. Moore, Derivation of auditory filter shapes from notched-noise data. *Hear. Res.* **47**, 103–138 (1990).

44. J. H. McDermott, E. P. Simoncelli, Sound texture perception via statistics of the auditory periphery: evidence from sound synthesis. *Neuron* **71**, 926–940 (2011).

45. G. W. Lindsay, Convolutional Neural Networks as a Model of the Visual System: Past, Present, and Future. *J. Cogn. Neurosci.*, 1–15 (2020).

46. T. Chen, S. Kornblith, M. Norouzi, G. Hinton, A Simple Framework for Contrastive Learning of Visual Representations in *Proceedings of the 37th International Conference on Machine Learning*, Proceedings of Machine Learning Research., H. D. Iii, A. Singh, Eds. (PMLR, 13--18 Jul 2020), pp. 1597–1607.

47. X. Chen, H. Fan, R. Girshick, K. He, Improved Baselines with Momentum Contrastive Learning. *arXiv [cs.CV]* (2020).

48. J.-B. Grill, *et al.*, Bootstrap your own latent: A new approach to self-supervised Learning. *arXiv [cs.LG]* (2020).

49. T. Chi, P. Ru, S. A. Shamma, Multiresolution spectrotemporal analysis of complex sounds. *J. Acoust. Soc. Am.* **118**, 887–906 (2005).

50. R. Santoro, *et al.*, Encoding of natural sounds at multiple spectral and temporal resolutions in the human auditory cortex. *PLoS Comput. Biol.* **10**, e1003412 (2014).

51. S. V. Norman-Haignere, J. H. McDermott, Neural responses to natural and model-matched stimuli reveal distinct computations in primary and nonprimary auditory cortex. *PLoS Biol.* **16**, e2005127 (2018).

52. I. Goodfellow, J. Shlens, C. Szegedy, Explaining and Harnessing Adversarial Examples in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio, Y. LeCun, Eds. (2015).

53. A. Madry, A. Makelov, L. Schmidt, D. Tsipras, A. Vladu, Towards deep learning models resistant to adversarial attacks. *arXiv [stat.ML]* (2017).

54. A. Ilyas, *et al.*, Adversarial examples are not bugs, they are features. *arXiv [stat.ML]* (2019).

55. L. Engstrom, *et al.*, Adversarial robustness as a prior for learned representations. *arXiv [stat.ML]* (2019).

56. O. J. Hénaff, E. P. Simoncelli, Geodesics of learned representations in *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, Y. Bengio, Y. LeCun, Eds. (2016).

57. J. Dapello, *et al.*, Simulating a primary visual cortex at the front of CNNs improves robustness to image perturbations. *Adv. Neural Inf. Process. Syst.* **33**, 13073–13087 (2020).

58. J. Dapello, *et al.*, Neural population geometry reveals the role of stochasticity in robust perception. *Adv. Neural Inf. Process. Syst.* **34** (2021).

59. S. Norman-Haignere, N. G. Kanwisher, J. H. McDermott, Distinct cortical pathways for music and speech revealed by hypothesis-free voxel decomposition. *Neuron* **88**, 1281–1296 (2015).

60. C. Olah, A. Mordvintsev, L. Schubert, Feature Visualization. *Distill* **2** (2017).

61. A. Mahendran, A. Vedaldi, Understanding deep image representations by inverting them in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (IEEE, 2015) https:/doi.org/10.1109/cvpr.2015.7299155.

62. J. Yosinski, J. Clune, A. Nguyen, T. Fuchs, H. Lipson, Understanding neural networks through deep visualization. *arXiv [cs.CV]* (2015).

63. A. Shafahi, *et al.*, Poison Frogs! Targeted Clean-Label Poisoning Attacks on Neural Networks in *Advances in Neural Information Processing Systems*, (Curran Associates, Inc., 2018).

64. J.-H. Jacobsen, J. Behrmann, R. Zemel, M. Bethge, Excessive Invariance Causes Adversarial Vulnerability in *7th International Conference on Learning Representations, (ICLR)*, (2019).

65. J.-H. Jacobsen, J. Behrmannn, N. Carlini, F. Tramèr, N. Papernot, Exploiting excessive invariance caused by norm-bounded adversarial robustness. *arXiv [cs.LG]* (2019).

66. J. Mehrer, C. J. Spoerer, N. Kriegeskorte, T. C. Kietzmann, Individual differences among deep neural network models. *Nat. Commun.* **11**, 5725 (2020).

67. J. Portilla, E. P. Simoncelli, A Parametric Texture Model Based on Joint Statistics of Complex Wavelet Coefficients. *Int. J. Comput. Vis.* **40**, 49–70 (2000).

68. A. Deza, A. Jonnalagadda, M. Eckstein, Towards Metamerism via Foveated Style Transfer in *International Conference on Learning Representations*, (2019) (March 16, 2021).

69. T. S. Wallis, *et al.*, Image content is more important than Bouma's Law for scene metamers. *Elife* **8** (2019).

70. C. Zhuang, *et al.*, Unsupervised Neural Network Models of the Ventral Visual Stream. *Cold Spring Harbor Laboratory*, 2020.06.16.155556 (2020).

71. K. R. Storrs, B. L. Anderson, R. W. Fleming, Unsupervised learning predicts human perception and misperception of gloss. *Nat Hum Behav* **5**, 1402–1417 (2021).

72. T. Konkle, G. A. Alvarez, A self-supervised domain-general learning framework for human ventral stream representation. *Nat. Commun.* **13**, 491 (2022).

73. R. Geirhos, *et al.*, On the surprising similarities between supervised and self-supervised models. *arXiv [cs.CV]* (2020).

74. R. McWalter, J. H. McDermott, Adaptive and Selective Time Averaging of Auditory Scenes. *Curr. Biol.* **28**, 1405-1418.e10 (2018).

75. F. Tramèr, N. Papernot, I. Goodfellow, D. Boneh, P. McDaniel, The Space of Transferable Adversarial Examples. *arXiv [stat.ML]* (2017).

76. U. Güçlü, M. A. J. van Gerven, Deep Neural Networks Reveal a Gradient in the Complexity of Neural Representations across the Ventral Stream. *J. Neurosci.* **35**, 10005–10014 (2015).

77. R. M. Cichy, A. Khosla, D. Pantazis, A. Torralba, A. Oliva, Comparison of deep neural networks to spatio-temporal cortical dynamics of human visual object recognition reveals hierarchical correspondence. *Sci. Rep.* **6**, 27755 (2016).

78. M. Eickenberg, A. Gramfort, G. Varoquaux, B. Thirion, Seeing it all: Convolutional network layers map the function of the human visual system. *Neuroimage* **152**, 184–194 (2017).

79. N. A. Ratan Murty, P. Bashivan, A. Abate, J. J. DiCarlo, N. Kanwisher, Computational models of category-selective brain regions enable high-throughput tests of selectivity. *Nat. Commun.* **12**, 5540 (2021).

80. L. S. Hamilton, A. G. Huth, The revolution will not be controlled: natural stimuli in speech neuroscience. *Lang Cogn Neurosci* **35**, 573–582 (2020).

81. A. Landemard, *et al.*, Distinct higher-order representations of natural sounds in human and ferret auditory cortex. *Elife* **10** (2021).

82. T. Golan, P. C. Raju, N. Kriegeskorte, Controversial stimuli: Pitting neural networks against each other as models of human cognition. *Proc. Natl. Acad. Sci. U. S. A.* **117**, 29330–29337 (2020).

83. K. Hermann, T. Chen, S. Kornblith, The Origins and Prevalence of Texture Bias in Convolutional Neural Networks in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, H. Lin, Eds. (Curran Associates, Inc., 2020), pp. 19000–19015.

84. R. Geirhos, *et al.*, ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness in *International Conference on Learning Representations*, (2019).

85. P. Bashivan, K. Kar, J. J. DiCarlo, Neural population control via deep image synthesis. *Science* **364** (2019).

86. C. R. Ponce, *et al.*, Evolving Images for Visual Neurons Using a Deep Generative Network Reveals Coding Principles and Neuronal Preferences. *Cell* **177**, 999-1009.e10 (2019).

87. E. Y. Walker, *et al.*, Inception loops discover what excites neurons most using deep predictive models. *Nat. Neurosci.* **22**, 2060–2065 (2019).

88. N. A. Lesica, *et al.*, Harnessing the power of artificial intelligence to transform hearing healthcare and research. *Nature Machine Intelligence* **3**, 840–849 (2021).

89. M. R. Saddler, A. Francl, J. Feather, J. H. McDermott, Speech Denoising with Auditory Models in *Interspeech 2021*, (unknown, 2021), pp. 2681–2685.

90. R. Rajalingham, K. Schmidt, J. J. DiCarlo, Comparison of object recognition behavior in human and monkey. *J. Neurosci.* **35**, 12127–12136 (2015).

91. A. Francl, J. H. McDermott, Deep neural network models of sound localization reveal how perception is adapted to real-world environments. *Nat Hum Behav* **6**, 111–133 (2022).

92. M. R. Saddler, R. Gonzalez, J. H. McDermott, Deep neural network models reveal interplay of peripheral coding and stimulus statistics in pitch perception. *Nat. Commun.* **12**, 7278 (2021).

93. A. Paszke, *et al.*, “PyTorch: An Imperative Style, High-Performance Deep Learning Library” in *Advances in Neural Information Processing Systems 32*, H. Wallach, *et al.*, Eds. (Curran Associates, Inc., 2019), pp. 8024–8035.

94. L. Engstrom, A. Ilyas, H. Salman, S. Santurkar, D. Tsipras, Robustness (Python Library) (2019).

95. D. B. Paul, J. M. Baker, The design for the wall street journal-based CSR corpus in *Proceedings of the Workshop on Speech and Natural Language - HLT '91*, (Association for Computational Linguistics, 1992) https:/doi.org/10.3115/1075527.1075614.

96. A. Köhn, F. Stegen, T. Baumann, Mining the Spoken Wikipedia for Speech Data and Beyond in *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, N. C. (conference Chair), *et al.*, Eds. (European Language Resources Association (ELRA), 2016).

97. V. W. Zue, S. Seneff, "Transcription and alignment of the TIMIT database" in *Recent Research Towards Advanced Man-Machine Interface Through Spoken Language*, (Elsevier, 1996), pp. 515–525.

98. J. F. Gemmeke, *et al.*, Audio Set: An ontology and human-labeled dataset for audio events in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, (IEEE, 2017) https:/doi.org/10.1109/icassp.2017.7952261 (January 31, 2022).

99. K. J. P. Woods, M. H. Siegel, J. Traer, J. H. McDermott, Headphone screening to facilitate web-based auditory experiments. *Atten. Percept. Psychophys.* **79**, 2064–2072 (2017).

100. K. J. P. Woods, J. H. McDermott, Schema learning for the cocktail party problem. *Proc. Natl. Acad. Sci. U. S. A.* **115**, E3313–E3322 (2018).

101. M. J. McPherson, J. H. McDermott, Time-dependent discrimination advantages for harmonic sounds suggest efficient coding for memory. *Proc. Natl. Acad. Sci. U. S. A.* **117**, 32169–32180 (2020).

102. J. Traer, S. V. Norman-Haignere, J. H. McDermott, Causal inference in environmental sound recognition. *Cognition* **214**, 104627 (2021).

103. M. J. McPherson, R. C. Grace, J. H. McDermott, Harmonicity aids hearing in noise. *Atten. Percept. Psychophys.* (2022) https:/doi.org/10.3758/s13414-021-02376-0.

104. F. Faul, E. Erdfelder, A.-G. Lang, A. Buchner, G*Power 3: a flexible statistical power analysis program for the social, behavioral, and biomedical sciences. *Behav. Res. Methods* **39**, 175–191 (2007).

105. M. Abadi, *et al.*, TensorFlow: A system for large-scale machine learning. *arXiv [cs.DC]* (2016).

106. D. P. Kingma, J. Ba, Adam: A Method for Stochastic Optimization. *arXiv [cs.LG]* (2014).

107. C. Humphries, E. Liebenthal, J. R. Binder, Tonotopic organization of human auditory cortex. *Neuroimage* **50**, 1202–1211 (2010).

108. S. Norman-Haignere, N. Kanwisher, J. H. McDermott, Cortical pitch regions in humans respond primarily to resolved harmonics and are located in specific tonotopic regions of anterior auditory cortex. *J. Neurosci.* **33**, 19451–19469 (2013).

109. T. Overath, J. H. McDermott, J. M. Zarate, D. Poeppel, The cortical analysis of speech-specific temporal structure revealed by responses to sound quilts. *Nat. Neurosci.* **18**, 903–911 (2015).

110. F. Pedregosa, *et al.*, Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011).

111. C. Spearman, The proof and measurement of association between two things. *Am. J. Psychol.* **15**, 72–101 (1904).

112. C. Spearman, Correlation calculated from faulty data. *Br. J. Psychol.* **3**, 271–295 (1910).

113. A. G. Huth, W. A. de Heer, T. L. Griffiths, F. E. Theunissen, J. L. Gallant, Natural speech reveals the semantic maps that tile human cerebral cortex. *Nature* **532**, 453–458 (2016).

114. D. Tsipras, S. Santurkar, L. Engstrom, A. Turner, A. Madry, Robustness may be at odds with accuracy. *arXiv [stat.ML]* (2018) (March 16, 2022).

115. J. Kubilius, *et al.*, Brain-Like Object Recognition with High-Performing Shallow Recurrent ANNs in *Advances in Neural Information Processing Systems*, (Curran Associates, Inc., 2019).