Metamers of neural networks reveal divergence from human perceptual systems

Jenelle Feather[1,2,3] **Alex Durango**[1,2,3] **Ray Gonzalez**[1,2,3] **Josh McDermott**[1,2,3,4]
[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] Speech and Hearing Bioscience and Technology, Harvard University
{jfeather,durangoa,raygon,jhm}@mit.edu

Abstract

Deep neural networks have been embraced as models of sensory systems, instantiating representational transformations that appear to resemble those in the visual and auditory systems. To more thoroughly investigate their similarity to biological systems, we synthesized model metamers – stimuli that produce the same responses at some stage of a network's representation. We generated model metamers for natural stimuli by performing gradient descent on a noise signal, matching the responses of individual layers of image and audio networks to a natural image or speech signal. The resulting signals reflect the invariances instantiated in the network up to the matched layer. We then measured whether model metamers were recognizable to human observers – a necessary condition for the model representations to replicate those of humans. Although model metamers from early network layers were recognizable to humans, those from deeper layers were not. Auditory model metamers became more human-recognizable with architectural modifications that reduced aliasing from pooling operations, but those from the deepest layers remained unrecognizable. We also used the metamer test to compare model representations. Cross-model metamer recognition dropped off for deeper layers, roughly at the same point that human recognition deteriorated, indicating divergence across model representations. The results reveal discrepancies between model and human representations, but also show how metamers can help guide model refinement and elucidate model representations.

1 Introduction

Artificial neural networks now achieve human-level performance on tasks such as image and speech recognition, raising the question of whether they should be taken seriously as models of biological sensory systems [1, 2, 3, 4, 5]. Detailed comparisons of network performance characteristics in some cases reveal human-like error patterns, suggesting computational similarities with humans [6, 7, 8]. Other studies have found that brain responses can be better predicted by features learned by deep neural networks than by those of traditional sensory models [2, 8]. On the other hand, neural network models can typically be fooled by adversarial perturbations that have no effect on humans [9, 10], are in some cases excessively dependent on particular image features, such as texture [11], and do not fully mirror human sensitivity to image distortions [12, 13], suggesting differences with human perceptual systems. However, these discrepancies have primarily been demonstrated using stimuli specifically constructed to induce classification errors. Here, we demonstrate that the divergence between artificial network and human representations occurs generically rather than only in adversarial situations.

33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.

We use "model metamers" to test the similarity between human and artificial neural network representations. Metamers are stimuli that are physically distinct but that are perceived to be the same by an observer. Stimuli that are metameric for humans have long been used to infer the underlying structure of the human perceptual system. Metamers provided some of the original evidence for trichromacy in human color vision, and have also been applied to texture perception [14] and visual crowding [15, 16]. Related ideas can also be used to test models of neural computation [17]. Here we leverage the idea that metamers for a valid model of human perception should also be metamers for humans. Model metamers produce the same activations in a model layer as some other stimulus (here a natural sound or image). Because the activations at all subsequent layers must also be the same, the metamers are classified the same by the model. Here, we approximate model metamers via iterative optimization, producing stimuli that produce nearly the same activations as a natural stimulus, thus leading to the same network prediction. As a test of whether the model accurately reflects human perception, we measure whether humans also correctly classify the model metamers. Although this test is looser than the classical metamer test (which requires metamers to be fully indistinguishable), it is conservative with respect to the goal of testing a model of human recognition. We consider model metamers that are unrecognizable to a human to be a model failure, cognizant that models that do not perfectly match human representations in this way might nonetheless be useful in other respects.

Because the neural network models we consider are trained to classify exemplars of highly variable object or speech classes, and thus to instantiate representations that are invariant to within-class variation, it is expected that metamers from deeper layers will exhibit greater physical variability than those from early layers. The question we sought to answer is whether the nature of the invariances would be similar to those of humans, in which case the model metamers should remain human-recognizable regardless of the stage from which they are generated. We generated model metamers for three image-trained and five sound-trained models that perform well on state-of-the-art tasks and then measured human recognition of the model metamers in psychophysical experiments. We also applied the same method across networks, to ask whether the invariances learned by one network resemble those learned by another. The results establish metamers as a tool to test and understand deep neural networks, with potential uses for multi-task applications, transfer learning, and network interpretability.

2 Related Work

2.1 Visualization of deep networks

Previous neural network visualizations have used gradient descent on the input signals to visualize the representations in neural networks [18], in some cases matching the activations at a given layer [19] as we do here. Natural image priors have been shown to make images reconstructed in this way "look" more natural, and further regularization tools have been proposed with a similar purpose [20, 21]. Although such regularization can generate visually appealing images, the importance of using a natural image prior suggests differences between the network representations and those of humans. Taking this observation as a starting point, we measured the human-recognizability of images or sounds that were matched at different network stages without imposing a separate prior, to quantify the potential divergence in representations and get clues as to its origins.

2.2 Comparing networks with other networks

Prior work on network similarity relates the learned representations via methods such as canonical correlation analysis (CCA) [22, 23, 24]. Other such work has been inspired by the neuroscience technique of representational similarity analysis [25, 26]. Here we also use metamers for model comparison, on the grounds that metamers for one model should also be metamers for another model (as measured here by producing the same class labels, although one could apply more fine-grained methods) if the two models share invariances.

2.3 Metamers applied to averaged features

Metamers have been used to develop models of human perception by pooling features to directly induce invariance across space or time. Work on visual crowding used images that have the

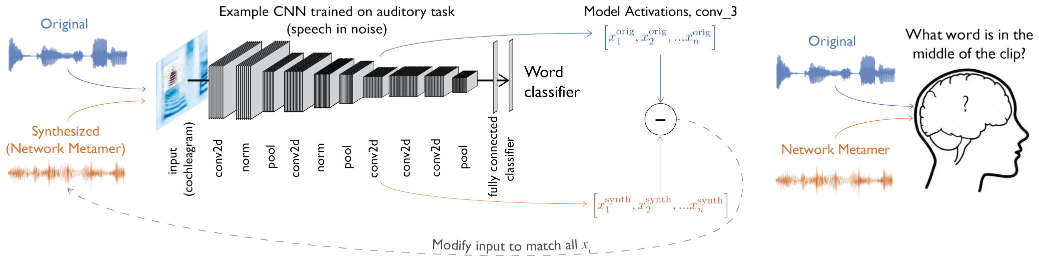


Figure 1: Model metamers are constructed by optimizing a random input signal such that it matches the measured activations of an original signal at a particular network stage. Model metamers are then presented to humans (or other networks) to measure the similarity of internal representations.

same spatially-averaged statistics in the periphery and are indistinguishable from the original in particular viewing conditions [27, 16, 28, 29]. Other work has used time-averaged statistics measured from auditory models, generating auditory textures that are mistaken for the original natural sound [30, 31]. Our work here is a more general instantiation of the metamerism approach, applicable to domains outside of peripheral vision and texture where invariances arise in the service of recognition rather than as a direct consequence of pooling.

3 Methods

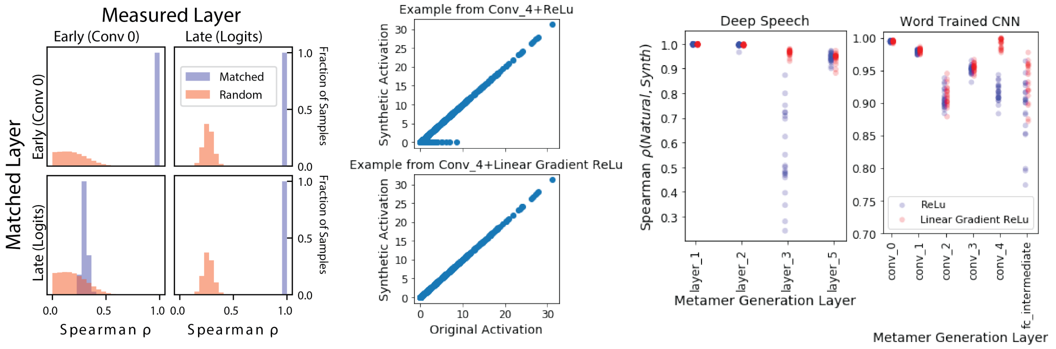
3.1 Metamer generation

Model metamers were generated using an iterative feature visualization technique [19] [1]. We initialized the metamer with noise and then performed gradient descent to minimize the squared error between its network activations and those for a paired natural signal. All models and metamer generation were implemented in TensorFlow [32]. Metamer synthesis used 15000 iterations of the Adam optimizer [33] with a learning rate of 0.001, with the exception of the VGGish Embedding (0.01) and DeepSpeech (0.0001) models.

In order to validate that we had appropriately matched the synthetic signal to the original, we computed the Spearman correlation between the model metamer and corresponding original signal. These correlations were typically close to 1 (Figure 2). Once candidate metamers were generated, the following two conditions had to be true for a model metamer to be included in our experiments: (1) The network predicted the same label for the synthetic metamer and the paired natural image. This is the same classification test we apply to humans and other networks. (2) The Spearman ρ between the metamer and natural image fell outside of a null distribution measured between 1,000,000 randomly chosen image or audio pairs from the training set. We compare to a null distribution rather than applying a strict threshold because the expected correlation varies with the network and layer. Setting hard cutoffs could potentially call samples metameric which are no more matched than chance, and we empirically found this procedure crucial for the random network (Figure S3). Histograms of the null and metamer correlations for all networks and selected layers are included in Tables S4-S5 and Figures S1-S8.

We found empirically that it was difficult to match some layers after a ReLU activation due to the initialized signal producing many activations of zero (Fig 2(b)). To improve the optimization, we modified the gradient through the metamer generation layer ReLU to be 1 for all values, including for values below zero, when generating a metamer for activations immediately following a ReLU. Figure 2(c) shows the matching fidelity (as measured by Spearman's ρ) for 20 example metamers generated with either the normal gradient or the modified gradient. The modified gradient substantially improved the matching on some layers (layer_3 of DeepSpeech, and conv_4 of the Word Trained CNN). We used the modified gradient for all metamers generated after a ReLU.

[1] Example generation code and trained models: https://github.com/jenellefeather/model_metamers



(a)

(b)

(c)

Figure 2: Validation of model metamer optimization (a) The model metamer is intended to produce the same activations as the original stimulus in a particular network layer. We quantified the fidelity of the matching as the Spearman correlation between the activations produced by a model metamer and the corresponding original stimulus, with histograms across stimuli. For a comparison null distribution, we also measured the correlation for randomly chosen pairs of signals from the training set. As intended, metamers generated from an early layer (top row) are well matched to the original in the early layer, with correlations close to 1 (blue distribution, top left), far above the null distribution across stimuli (red). Because the networks used here are deterministic and feedforward, the metamers should also produce the same activations at all subsequent layers, and they do (correlations near 1 in late layers, blue distribution, top right). Because of the many-to-one mapping instantiated by the network, metamers for a late layer (bottom row) do not match the activations in the early layer better than chance (left), but match the late layer as intended (right). (b) Comparison of activation matching with a standard ReLU activation function gradient and with the modified ReLU gradient. Without the modification, many non-zero values in the original activation get matched to zero. (c) Example layer-wise matching fidelity for metamers generated with either the standard ReLU gradient (blue) and the linear gradient ReLU (red) for two audio networks. In both networks there are layers that are significantly better matched using the modified ReLU gradients.

For visual metamers, pixel values were bounded between 0-255 or 0-1 (matching the preprocessing of the trained network), and were initialized with white noise with mean at the center value of the range. No other regularization was employed. For audio metamers, we applied gradient clipping to operations that resulted in problems with the optimization (specifically, logarithms and power operations) which were present in the audio pre-processing (that transformed the waveform to a frequency representation that provided the input to the networks). The audio metamer generation was initialized with pink noise at an RMS value of 0.01.

3.2 Auditory models

Our experiments used a five-layer convolutional network trained on the output of a model of the human ear. This cochlear model consisted of a filterbank of 171 filters spaced between 20Hz-80Hz with bandwidths and spacing modeled on the human ear [34, 30]. The envelope of each resulting audio subband was extracted via the Hilbert transform, downsampled to 200Hz, and passed through a compressive non-linearity. This yielded a ‘cochleagram’ representation, similar to a conventional spectrogram but with frequency resolution based on the human cochlea. We trained an architecture similar to that in [8] (full architecture described in Table S2).

Many neural networks do not obey the sampling theorem (because downsampling occurs without a preceding lowpass filter), and others have suggested that this could yield invariances that do not align with human perception [35, 36, 37]. Motivated by these observations, we constructed a modified architecture to reduce aliasing artifacts (Table S3). The modifications replaced max pooling operations with weighted average pooling using a hanning kernel applied with stride equal to that of the original max pooling. Any convolutional layer with a stride greater than one was replaced with a convolutional layer with a stride of one, followed by a hanning pooling operation with stride equal to the original convolutional stride.

As a demonstration that model metamers could be used to investigate representations in other audio models, we also generated example metamers from the VGGish network, which outputs embeddings used for training an environmental sound classifier and was released with the AudioSet dataset [38]. We also generated metamers for the publicly available DeepSpeech architecture [39].

3.3 Auditory CNN training

The auditory models were trained a word recognition task similar to [8], using segments from the Wall Street Journal [40] and Spoken Wikipedia Corpora [41]. Two-second speech segments were used for training examples, with the word in the middle of the clip assigned as the class label for training. There were 793 word classes sourced from 432 unique speakers, with 230357 unique clips in the training set and 40651 segments in the validation set (full details of the dataset construction are in Section S1.1). During training, the speech segments were randomly shifted in time and superimposed on a subset of 718625 AudioSet examples, spanning 516 AudioSet categories [42]. Some CNN models were trained to predict the AudioSet labels. In order to match performance between multiple models trained on the same task in Section 4.3 and eliminate confounds due to task performance, we used an early stopping criteria on the validation set of 57% correct for the word task and a mean area under the curve (AUC) of 0.83 for the AudioSet task.

3.4 Auditory metamer generation and experiments

We measured human recognition of model metamers using a task similar to that of [8]. A human observer listened to a clip and chose one of 587 possible word labels. Sixteen participants completed the experiment, each completing five trials from each of the included conditions, randomly ordered. Stimuli were generated from a set of 295 speech exemplars from the WSJ corpus (see Table S1.2 for a summary of auditory model metamers, and Figures S1-S5 for full histograms of the null and metamer Spearman ρ). Five sets of CNN metamers were generated for the experiment, one for each of five models: 1) the architecture inspired by [8], trained on the word task, 2) the random initialization of the reduced aliasing architecture, and 3) the reduced aliasing architecture trained on the word task, 4) the reduced aliasing architecture trained on the AudioSet task, and 5) the reduced aliasing architecture trained simultaneously on the AudioSet and word tasks. For each model, we included metamers constructed by matching the representations of the activation following each convolutional layer, fully-connected layer, and the logits (with the exception of the hanning pooling layer in the reduced aliasing networks that immediately followed strided convolutions, to equate the number of features to that for the aliasing networks). We also included metamers for the cochlear representation.

3.5 Image models, metamer generation, and experiments

ImageNet-trained models were obtained from publicly available pretrained checkpoints[2]. We generated metamers from a subset of layers for each of VGG-19 [43], Inception-V3 [44], and ResNet-101-V2 [45]. To compare performance between networks and humans in the visual domain, we used a modified version of the image classification task described in [13]. For each of a set of layers in the three pretrained ImageNet models, we generated metamers of 36 randomly selected natural images across each of the 16 MS-COCO categories (see Supplement Table 4 for a summary of matching the visual model metamers, and Supplement Figures 1-3 for full histograms of the null and metamer histograms). Each of sixteen participants had to classify a subset of these metameric stimuli and their corresponding natural image seeds, choosing the MS-COCO category; each participant classified 10 examples per network-layer metamer condition.

4 Results

4.1 Image network model metamers

For all tested image networks,the metamers became unrecognizable to humans by the final stages of the network (Figure 3a-b). The appearance of the metamers to humans varied depending on the architecture. In Inception-V3 and ResNet-101-V2 (both of which include convolutions with

[2] https://github.com/tensorflow/models/tree/master/research/slim

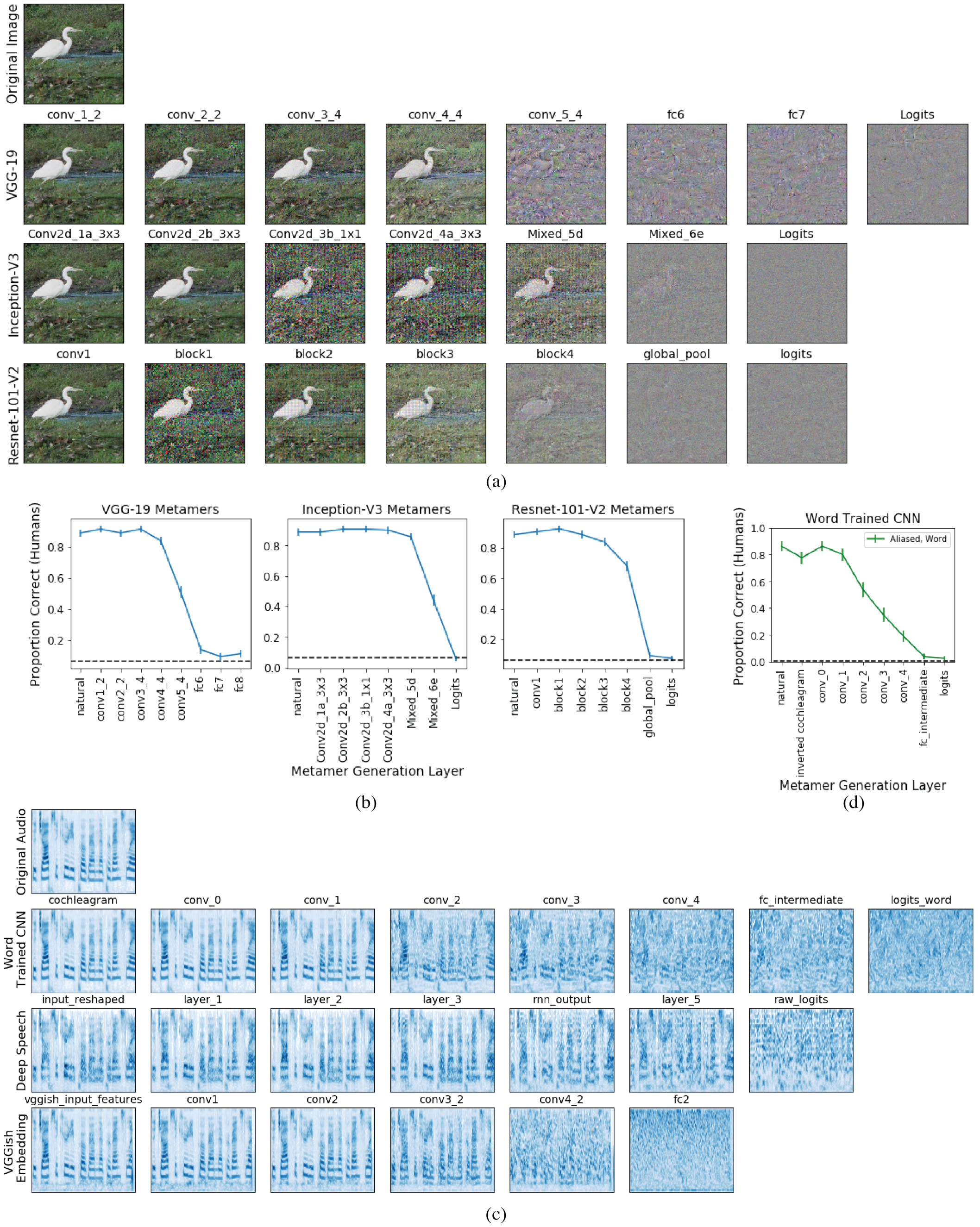


Figure 3: Deep network model metamers and their recognition by human observers. (a) Example visual network model metamers synthesized to produce the same activations at a particular layer of a particular network as the image in the top left. (b) Human recognition of visual network model metamers. Recognition is good for early-layer metamers but poor for deep-layer metamers, implying a divergence from human perceptual representations. Error bars are standard error of the mean (SEM). (c) Example cochleagrams (time-frequency decompositions) for metamers from an audio network trained to recognize words. (d) Human recognition of word-trained CNN model metamers. As for vision-trained models, recognition is good for early-layer metamers but poor for deep-layer metamers. Error bars are SEM.

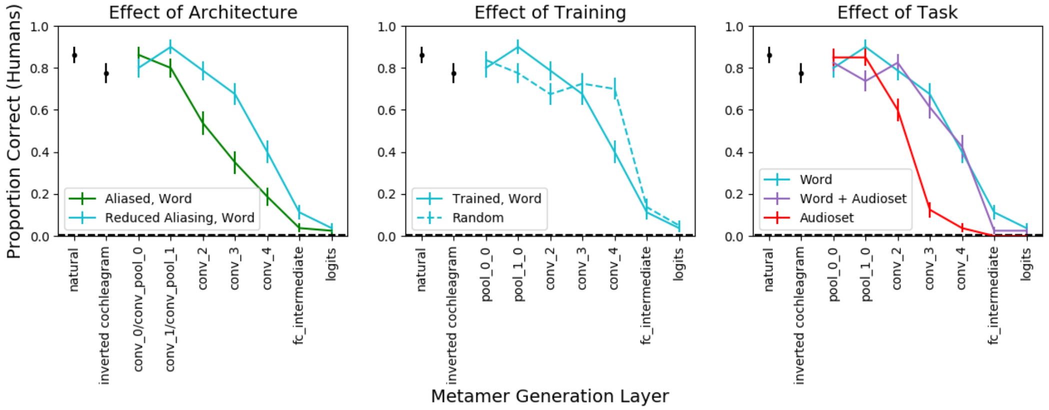


Figure 4: Human recognition of audio network model metamers. Architectural manipulations that reduce aliasing (left), training (middle), and task (right) all altered the recognizability of metamers.

a stride greater than one) there is visible ‘gridding’ in the metamers generated from early layers, plausibly due to aliasing.

4.2 Audio network model metamers

The metamers from the word-trained network with an architecture based on [8] also quickly become unrecognizable to humans (Figure 3c-d). Although not included in the human behavioral experiment, we also generated example metamers from DeepSpeech and the VGGish Embedding Network[3]. All metamers from DeepSpeech sound unnatural due to the input representation (framed MFCCs). The metamers on the VGGish embedding network become difficult to recognize by conv_4 (perhaps unsurprisingly, as we only generated metamers for speech, and the network was not trained for speech recognition).

4.3 Model metamers from audio networks with modified task or architecture

We considered that the decrease in metamerism for humans might be due to aliasing (from convolutional layers with strides greater than 1, and maxpooling layers). Consistent with this idea, the modified architecture that reduces aliasing yielded model metamers that were more recognizable to humans (Figure 4). We also considered the effect of training on metamerism. Unlike, the trained networks, metamers from a random network with reduced aliasing remained recognizable through all convolutional layers, only becoming unrecognizable at the top fully-connected layer. This result suggests that task optimization adds invariances to the network that can in some cases be different than human invariances. However, the human-recognizability of the model metamers was task-specific – the same network architecture trained to classify the AudioSet backgrounds produced metamers that became unrecognizable more quickly than when trained on the word task. Training on the AudioSet classification in addition to the word task did not impair metamerism (Figure 4). In all cases the metamers from deep layers remained unrecognizable to humans, but the effects of these manipulations raise the possibility that appropriate choices of training and architecture might produce a model that better accounts for human perception.

4.4 Metamer comparisons between ImageNet architectures

The metamer test can also be used to compare different architectures. We generated metameric images for one ImageNet-trained network and then presented its metamers to a second network. If the representational spaces between the two networks are the same, then the second network should be able to correctly classify the metamers from the first network. For all three tested networks, we find that the representations diverge from those of the other networks (Figure 5). Further, at late layers the model metamers are generally not even recognizable to the same

[3] Example audio metamers: http://mcdermottlab.mit.edu/jfeather/model_metamers/audio_metamers.html

ImageNet-trained architecture trained with a different initialization (especially evident in the 1000 way classification task). Interestingly, image metamers for one network become non-metameric for another network at roughly the same layer at which human performance diverges.

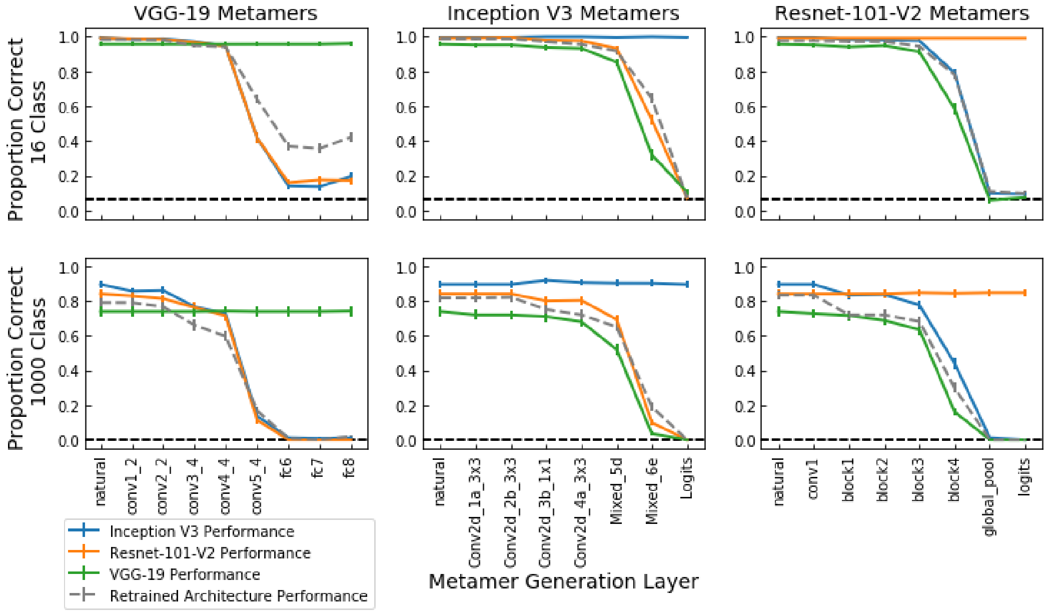


Figure 5: Network recognition of metamers from other networks and for networks with the same architecture but different initializations. All networks were trained on ImageNet. Top row: performance on 16-way classification task with metamers (using groups of the original ImageNet classes, used for human recognition experiment in Figure 2). Bottom row: performance on original ImageNet classification task with metamers.

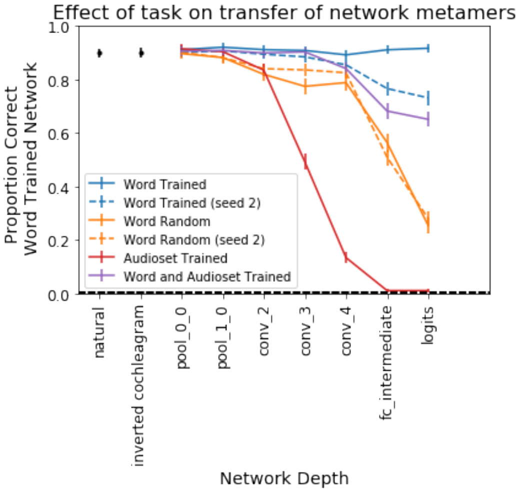


Figure 6: Word-trained network recognition of metamers from other networks. Metamers were generated from networks with the same architecture but trained on different tasks. Error bars are bootstrapped SEM.

4.5 Metamer comparisons between audio networks trained on different tasks

In the audio domain, we tested whether model metamers generalized across training tasks and random seeds (Figure 6). We measured performance of the word-trained network on metamers generated from networks with the same architecture but trained on a different task. Metamers generated from untrained networks were poorly recognized by the word-trained network, providing further evidence that training alters the network invariances. Model metamerism did not transfer between the word-trained network and the AudioSet-trained network, but metamers generated from the network trained on both tasks were only slightly less metameric than metamers from a word-trained network with weights initialized with a different random seed. This latter result provides a proof of concept that it is possible for metamers to be shared across distinct systems.

5 Discussion

Our results show that model metamers generated from deep layers of artificial neural networks are not metameric for humans or other networks. These findings demonstrate a divergence in the invariances learned by neural networks from those present in human perceptual systems. They also highlight the benefits of using model metamers as a network comparison tool. Our results suggest that discrepancies between model and human representations, and between different models, arise in later model stages, identifying those stages as targets for model refinement. Indeed, we were able to modify some aspects of our audio-trained models to reduce aliasing and increase human recognition of the model metamers. We also demonstrate that human recognition of the metamers is dependent on the training task, possibly suggesting that the failure of humans to recognize the model metamers may be a reflection of training on a single task (in this case, recognizing speech but ignoring the background). Future work could investigate this by modifying tasks to be more diverse, or more human-like, and assessing whether the improved models better predict human behavior.

The transfer of metamers with different random seeds was surprisingly different between the image- and audio-trained networks. Further investigation revealed that optimizing the cochleagram representation rather than the audio yielded model metamers that were less recognizable by a network trained on a different random seed (Figure S9). This result raises the possibility that the shared "cochlear" pre-processing (consisting of fixed stages of convolution, pooling, and non-linearities) enforces shared invariances between audio-trained networks with different initializations. Future work could use metamerism to explore the use of shared early-layer representations as a way to unify representations across models and potentially better model human perceptual systems, for instance by adding additional biological constraints on the input representation.

Model metamers are complementary to adversarial examples. Adversarial examples are metameric (perceived similarly) for humans but are not metameric to the network they are derived for, demonstrating that the network lacks some invariances present in humans. Model metamers conversely demonstrate that invariances present in networks are not necessarily invariances for human perception (or other networks). The relationship between adversarial and metameric images was explored recently in [46], who concluded that the cross-entropy loss creates excessive invariance in the final classification layer, leading to adversarial examples. We explore related issues but examined a more diverse set of network layers and explicitly performed human and network-network experiments. Together, these lines of work suggest that techniques for reducing adversarial vulnerability may also improve the transfer of metamers across models. Moreover, metamers could be useful for evaluating the adversarial vulnerability of a model. However, unlike adversarial examples, which are specifically engineered to fool a particular system, model metamers are constrained only to produce the same model activations (rather than to fool humans). The considerable lack of metamer transfer to humans thus arguably represents a more substantial model failure, and a useful measuring stick for models of perceptual systems.

Acknowledgements

We thank Richard McWalter, Alex Kell, and Sam Norman-Haignere for comments on an early draft of this work. We also thank Mark Saddler and Andrew Francl for contributing to a shared codebase used in this project. This work was funded by a McDonnell Scholar Award to J.H.M., NSF grant BCS-1634050, NIH grant R01-DC017970 and a DOE CSGF Fellowship to J.J.F.

References

[1] Nikolaus Kriegeskorte. Deep neural networks: a new framework for modeling biological vision and brain information processing. *Annual Review of Neuroscience*, pages 417–446, 2015.

[2] Daniel LK Yamins and James J DiCarlo. Using goal-driven deep learning models to understand sensory cortex. *Nature neuroscience*, 19(3):356, 2016.

[3] Alex Kell and Josh H. McDermott. Deep neural network models of sensory systems: windows onto the role of task constraints. *Current Opinion in Neurobiology*, 55:121–132, 2019.

[4] David GT Barrett, Ari S Morcos, and Jakob H Macke. Analyzing biological and artificial neural networks: challenges with opportunities for synergy? *Current opinion in neurobiology*, 55:55–64, 2019.

[5] Adam H Marblestone, Greg Wayne, and Konrad P Kording. Toward an integration of deep learning and neuroscience. *Frontiers in computational neuroscience*, 10:94, 2016.

[6] Rishi Rajalingham, Kailyn Schmidt, and James J DiCarlo. Comparison of object recognition behavior in human and monkey. *Journal of Neuroscience*, 35(35):12127–12136, 2015.

[7] Saeed Reza Kheradpisheh, Masoud Ghodrati, Mohammad Ganjtabesh, and Timothée Masquelier. Deep networks can resemble human feed-forward vision in invariant object recognition. *Scientific reports*, 6:32672, 2016.

[8] Alexander JE Kell, Daniel LK Yamins, Erica N Shook, Sam V Norman-Haignere, and Josh H McDermott. A task-optimized neural network replicates human auditory behavior, predicts brain responses, and reveals a cortical processing hierarchy. *Neuron*, 98(3):630–644, 2018.

[9] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *International Conference on Learning Representations(ICLR)*, 2014.

[10] Nicholas Carlini and David Wagner. Audio adversarial examples: Targeted attacks on speech-to-text. In *2018 IEEE Security and Privacy Workshops (SPW)*, pages 1–7. IEEE, 2018.

[11] Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A Wichmann, and Wieland Brendel. Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. *International Conference on Learning Representations (ICLR)*, 2019.

[12] Alexander Berardino, Valero Laparra, Johannes Ballé, and Eero Simoncelli. Eigen-distortions of hierarchical representations. In *Advances in neural information processing systems*, pages 3530–3539, 2017.

[13] Robert Geirhos, Carlos RM Temme, Jonas Rauber, Heiko H Schütt, Matthias Bethge, and Felix A Wichmann. Generalisation in humans and deep neural networks. In *Advances in Neural Information Processing Systems*, pages 7538–7550, 2018.

[14] B. Julesz. Visual pattern discrimination. *IEEE Transactions on Information Theory*, 8:84–92, 1962.

[15] Benjamin Balas, Lisa Nakano, and Ruth Rosenholtz. A summary-statistic representation in peripheral vision explains visual crowding. *Journal of vision*, 9(12):13–13, 2009.

[16] Jeremy Freeman and Eero P Simoncelli. Metamers of the ventral stream. *Nature neuroscience*, 14(9):1195, 2011.

[17] Sam V Norman-Haignere and Josh H McDermott. Neural responses to natural and model-matched stimuli reveal distinct computations in primary and nonprimary auditory cortex. *PLoS biology*, 16(12):e2005127, 2018.

[18] Chris Olah, Alexander Mordvintsev, and Ludwig Schubert. Feature visualization. *Distill*, 2017. https://distill.pub/2017/feature-visualization.

[19] Aravindh Mahendran and Andrea Vedaldi. Understanding deep image representations by inverting them. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5188–5196, 2015.

[20] Jason Yosinski, Jeff Clune, Anh Nguyen, Thomas Fuchs, and Hod Lipson. Understanding neural networks through deep visualization. *International Conference on Machine Learning, Deep Learning Workshop*, 2015.

[21] Alexander Mordvintsev, Christopher Olah, and Mike Tyka. Inceptionism: Going deeper into neural networks. 2015.

[22] Ari Morcos, Maithra Raghu, and Samy Bengio. Insights on representational similarity in neural networks with canonical correlation. In *Advances in Neural Information Processing Systems*, pages 5727–5736, 2018.

[23] Yixuan Li, Jason Yosinski, Jeff Clune, Hod Lipson, and John E Hopcroft. Convergent learning: Do different neural networks learn the same representations? In *FE@ NIPS*, pages 196–212, 2015.

[24] Maithra Raghu, Justin Gilmer, Jason Yosinski, and Jascha Sohl-Dickstein. Svcca: Singular vector canonical correlation analysis for deep learning dynamics and interpretability. In *Advances in Neural Information Processing Systems*, pages 6076–6085, 2017.

[25] Nikolaus Kriegeskorte, Marieke Mur, and Peter A Bandettini. Representational similarity analysis-connecting the branches of systems neuroscience. *Frontiers in systems neuroscience*, 2:4, 2008.

[26] Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. *International Conference on Machine Learning*, 2019.

[27] Thomas SA Wallis, Christina M Funke, Alexander S Ecker, Leon A Gatys, Felix A Wichmann, and Matthias Bethge. Image content is more important than bouma's law for scene metamers. *eLife*, 8:e42512, 2019.

[28] T. S. A. Wallis, C. M. Funke, A. S. Ecker, L. A. Gatys, F. A. Wichmann, and M. Bethge. A parametric texture model based on deep convolutional features closely matches texture appearance for humans. *Journal of Vision*, 17(12), Oct 2017.

[29] Arturo Deza, Aditya Jonnalagadda, and Miguel Eckstein. Towards metamerism via foveated style transfer. *International Conference on Learning Representations*, 2017.

[30] Josh H McDermott and Eero P Simoncelli. Sound texture perception via statistics of the auditory periphery: Evidence from sound synthesis. *Neuron*, 71:926–940, 2011.

[31] Jenelle Feather and Josh H. McDermott. Auditory texture synthesis from task-optimized convolutional neural networks. *Conference on Computational Cognitive Neuroscience*, 2018.

[32] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. Tensorflow: A system for large-scale machine learning. In *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, pages 265–283, 2016.

[33] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *International Conference on Learning Representations*, 2015.

[34] Brian Glasberg and Brian C J Moore. Derivation of auditory filter shapes from notched-noise data. *Hearing Research*, 47:103–138, 1990.

[35] Olivier J Hénaff and Eero P Simoncelli. Geodesics of learned representations. *International Conference on Learning Representations*, 2016.

[36] Richard Zhang. Making convolutional networks shift-invariant again. *International Conference on Machine Learning*, 2019.

[37] Aharon Azulay and Yair Weiss. Why do deep convolutional networks generalize so poorly to small image transformations? *arXiv preprint arXiv:1805.12177*, 2018.

[38] Shawn Hershey, Sourish Chaudhuri, Daniel P. W. Ellis, Jort F. Gemmeke, Aren Jansen, Channing Moore, Manoj Plakal, Devin Platt, Rif A. Saurous, Bryan Seybold, Malcolm Slaney, Ron Weiss, and Kevin Wilson. Cnn architectures for large-scale audio classification. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2017.

[39] Awni Hannun, Carl Case, Jared Casper, Bryan Catanzaro, Greg Diamos, Erich Elsen, Ryan Prenger, Sanjeev Satheesh, Shubho Sengupta, Adam Coates, et al. Deep speech: Scaling up end-to-end speech recognition. *arXiv preprint arXiv:1412.5567*, 2014.

[40] Douglas B Paul and Janet M Baker. The design for the wall street journal-based csr corpus. In *Proceedings of the workshop on Speech and Natural Language*, pages 357–362. Association for Computational Linguistics, 1992.

[41] Arne Köhn, Florian Stegen, and Timo Baumann. Mining the spoken wikipedia for speech data and beyond. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may 2016. European Language Resources Association (ELRA).

[42] Jort F. Gemmeke, Daniel P. W. Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R. Channing Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *Proc. IEEE ICASSP 2017*, New Orleans, LA, 2017.

[43] Karen Simonyan and Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. *International Conference on Learning Representations*, pages 1–14, 2014.

[44] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2818–2826, 2016.

[45] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *ECCV*, 2016.

[46] J.-H. Jacobsen, J. Behrmann, R. Zemel, and M. Bethge. Excessive invariance causes adversarial vulnerability. *International Conference on Learning Representations (ICLR)*, 2019.

[47] Victor W Zue and Stephanie Seneff. -transcription and alignment of the timit database. In *Recent Research Towards Advanced Man-Machine Interface Through Spoken Language*, pages 515–525. Elsevier, 1996.

Supplement: Metamers of neural networks reveal divergence from human perceptual systems

S1.1 Audio CNN training dataset

The auditory models were trained on the word recognition task described in [8], but with an updated training set using segments from the Wall Street Journal [40] and Spoken Wikipedia Corpora [41]. We screened the Wall Street Journal (WSJ) [40], TIMIT [47], and a subset of articles from the Spoken Wikipedia Copora (SWC) [41] for appropriate audio segments (i.e., in which words overlapped the center of a two second segment). Each segment was assigned the word class label of the word occurring at the segment midpoint, and a speaker class label determined by the speaker.

In hopes 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 (i.e., 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 (i.e., 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 230356 unique segments in 432 speaker classes and 793 word classes, with 40650 unique segments in the validation set.

During training, the speech segments were randomly shifted in time and superimposed on AudioSet [42] examples such that models could also be trained on the AudioSet task. We randomly varied the SNR between the source (Speech) and the noise (AudioSet), uniformly distributed between -10dB SNR and 10dB SNR. To minimize ambiguity, we removed any sounds under the "Speech" or "Whispering" branch of the ontology. Since a high proportion of AudioSet clips contain music, we achieved a more balanced set by excluded any clips that were only labeled as the root "Music" with no specific branch labels, and the "Music" label was not used during the AudioSet task. We also removed silent clips by first discarding everything tagged with a "Silence" label then culling clips containing more than 10% zeros. This screening resulted in a training set of 718625 unique background clips spanning 516 categories. During training, we cycled through the sets of speech and AudioSet clips in random order, randomly sampling a two-second segment from the AudioSet clip and adding it to the speech clip to form a training example. Validation performance is reported on data constructed with the same training augmentations (specifically, variable SNR and temporal shifts). CNN models were trained across two NVIDIA GPUs each with 11GB memory.

S1.2 Retrained ImageNet Description

The ImageNet-trained architectures used to generate metamers for the behavioral and network-network experiments were downloaded from the TFSlim repository. The code at this repository was also used to retrain ImageNet architectures for the random seed experiments. Architecture details and preprocessing were matched to the downloaded checkpoints. The batch size, number of GPUs, and learning rate that we used was likely different from that used for training the downloaded checkpoints, which is potentially reflected the slightly worse training accuracy for some of the retrained models S1.2.

ImageNet Network	Top-1 Accuracy	Top-5 Accuracy
VGG-19	72.0	90.6
Inception-V3	75.2	92.5
Resnet-101-V2	73.6	91.5

Table S1: Summary of retrained ImageNet architectures for random seed experiments.

Table S2: Auditory CNN Architecture Definition ([8] with reshaped kernels to account for the modified input size.

Layer	Type	Filters	Size	Stride
0	input	-	[211, 400]	-
1	batch-normalization	-	-	-
2	conv2d	96	[7, 14]	[3, 3]
3	relu (conv_0)	-	-	-
4	max-pooling2d	-	[2, 5]	[2, 2]
5	batch-normalization	-	-	-
6	conv2d	256	[4, 8]	[2, 2]
7	relu (conv_1)	-	-	-
8	max-pooling2d	-	[2, 5]	[2, 2]
9	batch-normalization	-	-	-
10	conv2d	512	[2, 5]	[1, 1]
11	relu (conv_2)	-	-	-
12	conv2d	1024	[2, 5]	[1, 1]
13	relu (conv_3)	-	-	-
14	conv2d	512	[2, 5]	[1, 1]
15	relu (conv_4)	-	-	-
16	avg-pool	-	[2, 5]	[2, 2]
17	flatten	-	-	-
18	fully-connected	4096	-	-
19	relu (fc_intermediate)	-	-	-
20	dropout, 0.5	-	-	-
21	fully-connected classification (logits)	-	-	-

Table S3: Auditory CNN Architecture Definition with Reduced Aliasing

Layer	Type	Filters	Size	Stride
0	input	-	[211, 400]	-
1	batch-normalization	-	-	-
2	conv2d	96	[7, 14]	[1, 1]
3	relu	-	-	-
4	hpool (pool_0_0)	-	[12, 12]	[3, 3]
5	hpool	-	[8, 8]	[2, 2]
6	batch-normalization	-	-	-
7	conv2d	256	[4, 8]	[1, 1]
8	relu	-	-	-
9	hpool (pool_1_0)	-	[8, 8]	[2, 2]
10	hpool	-	[8, 8]	[2, 2]
11	batch-normalization	-	-	-
12	conv2d	512	[2, 5]	[1, 1]
13	relu (conv_2)	-	-	-
14	conv2d	1024	[2, 5]	[1, 1]
15	relu (conv_3)	-	-	-
16	conv2d	512	[2, 5]	[1, 1]
17	relu (conv_4)	-	-	-
18	avg-pool	-	[2, 5]	[2, 2]
19	flatten	-	-	-
20	fully-connected	4096	-	-
21	relu (fc_intermediate)	-	-	-
22	dropout, 0.5 training	-	-	-
23	fully-connected classification (logits)	-	-	-

Network Metamer Generation Layer	**Number Generated Metamers**	**Number features**	**Median Spearman ρ at Layer**	**Median Spearman ρ Null at Layer**	**Median Spearman ρ at Logits**
Natural Sound	295	84400	-	-	-
Inverted Cochleagram	295	84400	0.998674	0.179334	0.999976
Word Trained (Aliased)					
conv_0	292	913344	0.994434	0.265085	0.999764
conv_1	294	156672	0.980023	0.268980	0.998522
conv_2	293	78336	0.918038	0.170268	0.997408
conv_3	294	156672	0.956672	0.233818	0.999491
conv_4	295	78336	0.996275	0.039919	0.999997
fc_intermediate	291	4096	0.944563	0.079779	0.999487
logits	290	794	0.995809	0.139888	0.995809
Word Trained (Reduced Aliasing)					
pool_0	291	913344	0.997652	0.561209	0.999733
pool_1	288	156672	0.989475	0.602778	0.997916
conv_2	292	78336	0.991182	0.205391	0.999851
conv_3	293	156672	0.989225	0.280117	0.999919
conv_4	295	78336	0.972968	0.048382	0.999996
fc_intermediate	290	4096	0.999361	0.147935	0.999813
logits	286	794	0.998158	0.147180	0.998158
Random (Reduced Aliasing)					
pool_0	272	913344	0.997214	0.952567	0.999999
pool_1	278	156672	0.999251	0.962971	0.999997
conv_2	281	78336	0.999756	0.968697	0.999997
conv_3	279	156672	0.999791	0.963797	0.999997
conv_4	285	78336	0.999814	0.959306	0.999997
fc_intermediate	289	4096	0.999683	0.985956	0.999994
logits	293	794	0.999996	0.986279	0.999996
Trained Audioset (Reduced Aliasing)					
pool_0	291	913344	0.998042	0.451898	0.999866
pool_1	290	156672	0.994289	0.454849	0.999089
conv_2	290	78336	0.986702	0.193952	0.999923
conv_3	291	156672	0.964322	0.137700	0.999967
conv_4	292	78336	0.966812	0.134290	0.999972
fc_intermediate	294	4096	0.997083	0.314034	0.999972
logits	292	517	0.999752	0.463126	0.999752
Trained Word and Audioset (Reduced Aliasing)					
pool_0	285	913344	0.997472	0.555888	0.999618
pool_1	282	156672	0.990088	0.560066	0.996624
conv_2	286	78336	0.982321	0.179038	0.999580
conv_3	287	156672	0.976542	0.212300	0.999808
conv_4	288	78336	0.921548	0.047874	0.999943
fc_intermediate	292	4096	0.959105	0.232050	0.999751
logits	289	794	0.998801	0.146840	0.998801

Table S4: Summary of network metamer generation for audio network. The number of generated network metamers varies by layer due to failed optimizations (measured by an overlap with the null or not having the same maximum logit as the original) or due to node time outs during the generation. Null distributions are constructed from 1,000,000 image pairs in the training set. Metamers included in the experiment do not overlap with the null distributions, even in the case of the Random (Reduced Aliasing) network layers where activations are strongly correlated for the null. Metamers were generated on NVIDIA GPUs with 11-12GB of RAM.

Network Metamer Generation Layer	Number Generated Metamers	Number features	Median Spearman ρ at Layer	Median Spearman ρ Null at Layer	Median Spearman ρ at Logits
Natural Image	256	89401	-	-	-
Natural Image Small	256	50176	-	-	-
Inception-V3 [44]					
Conv2d_1a_3x3	256	710432	0.999980	0.724671	1.000000
Conv2d_2b_3x3	256	691488	0.999539	0.510215	0.999994
Conv2d_3b_1x1	244	426320	0.984236	0.335206	0.990008
Conv2d_4a_3x3	253	967872	0.995720	0.592758	0.998891
Mixed_5d	254	352800	0.992983	0.183679	0.999667
Mixed_6e	253	221952	0.950064	0.172391	0.998504
Mixed_7c [4]	240	180224	0.756891	0.064566	0.961890
Logits	255	1001	0.999831	0.040540	0.999831
Resnet-101-V2 [45]					
conv_1	256	1440000	1.000000	0.120331	1.000000
block_1	256	369664	0.999787	0.754825	0.999448
block_2	256	184832	0.999978	0.496263	0.999981
block_3	254	102400	0.999302	0.342142	0.999609
block_4	255	204800	0.994098	0.284898	0.999230
global	254	2048	0.902678	0.214380	0.998909
logits	254	1001	0.999659	0.047858	0.999659
VGG-19 [43]					
conv1_2	256	3211264	0.999961	0.184170	1.000000
conv2_2	256	1605632	0.999152	0.066985	0.999998
conv3_4	256	802816	0.999155	0.108890	0.999995
conv4_4	255	401408	0.994657	0.035149	0.999994
conv5_4	256	100352	0.971722	0.022134	0.999980
fc6	256	4096	0.977821	0.031115	0.999993
fc7	256	4096	0.987343	0.043484	0.999980
fc8 (logits)	255	1000	0.999924	0.187791	0.999924

Table S5: Summary of network metamer generation for visual networks. [1] Although metamers were generated for Mixed_7c, we did not include Mixed_7c metamers for human behavior or model-model comparisons, as the optimization did not succeed to the same extent as the other layers (detailed histogram in Figure S6)

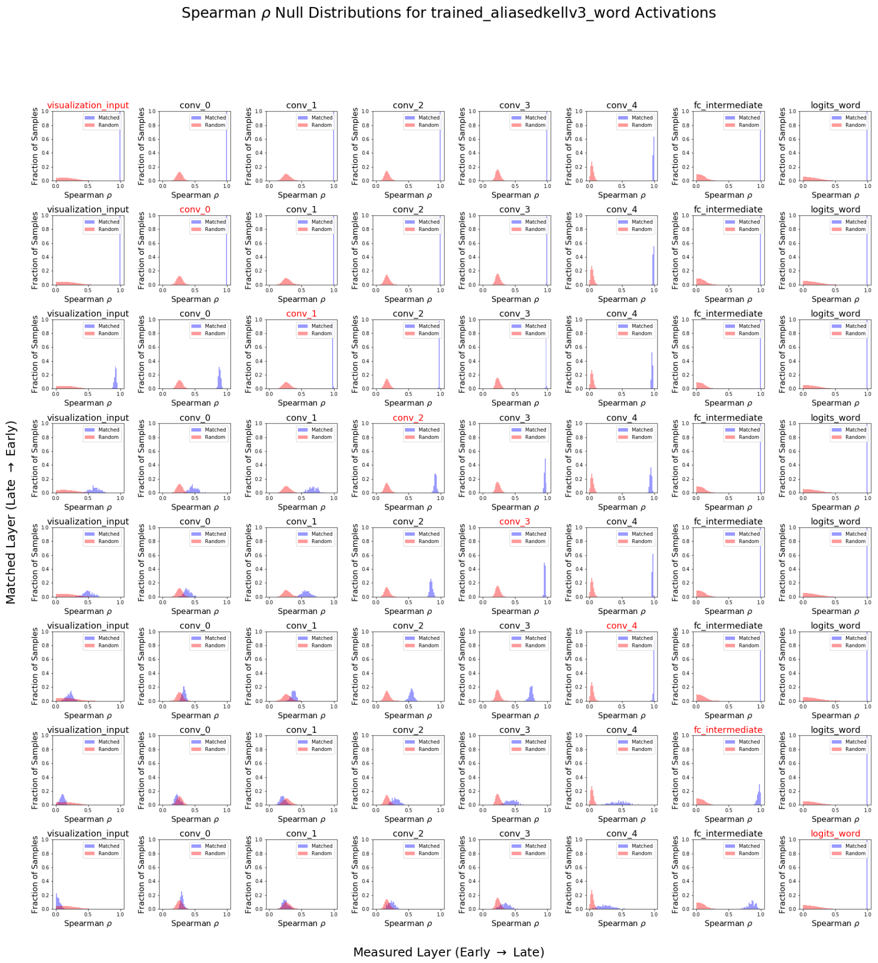


Figure S1: Spearman correlation coefficient for the word task CNN metamer generation compared with a null correlation distribution obtained by correlating 1000000 random speech sounds from the training set. Diagonal elements (with figure titles in red) correspond to the network metamer generation layer. For a given metamer generation layer, metamer Spearman correlations for the later network layers (further to the right) remain far from the null, while for earlier layers the distributions begin to overlap with the null, demonstrating the the generated stimulus is physically distinct from the natural sound.

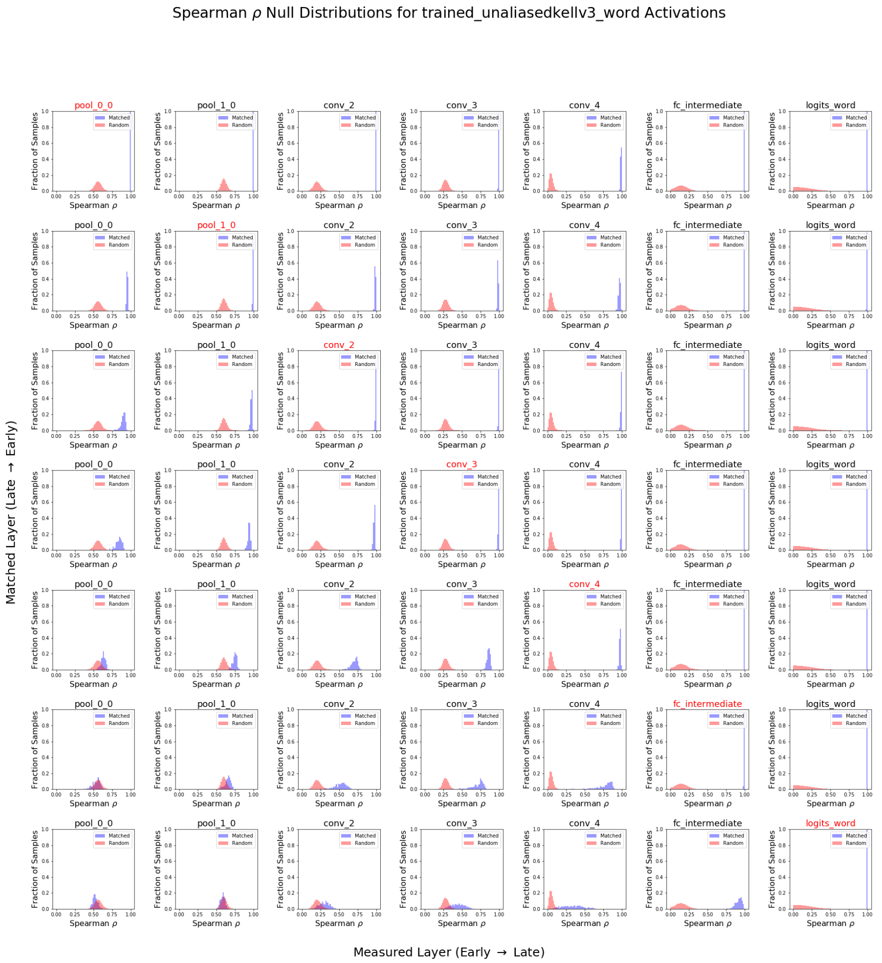


Figure S2: Network metamer Spearman correlation coefficients compared with the null correlation distribution for the Work Task CNN (with reduced aliasing).

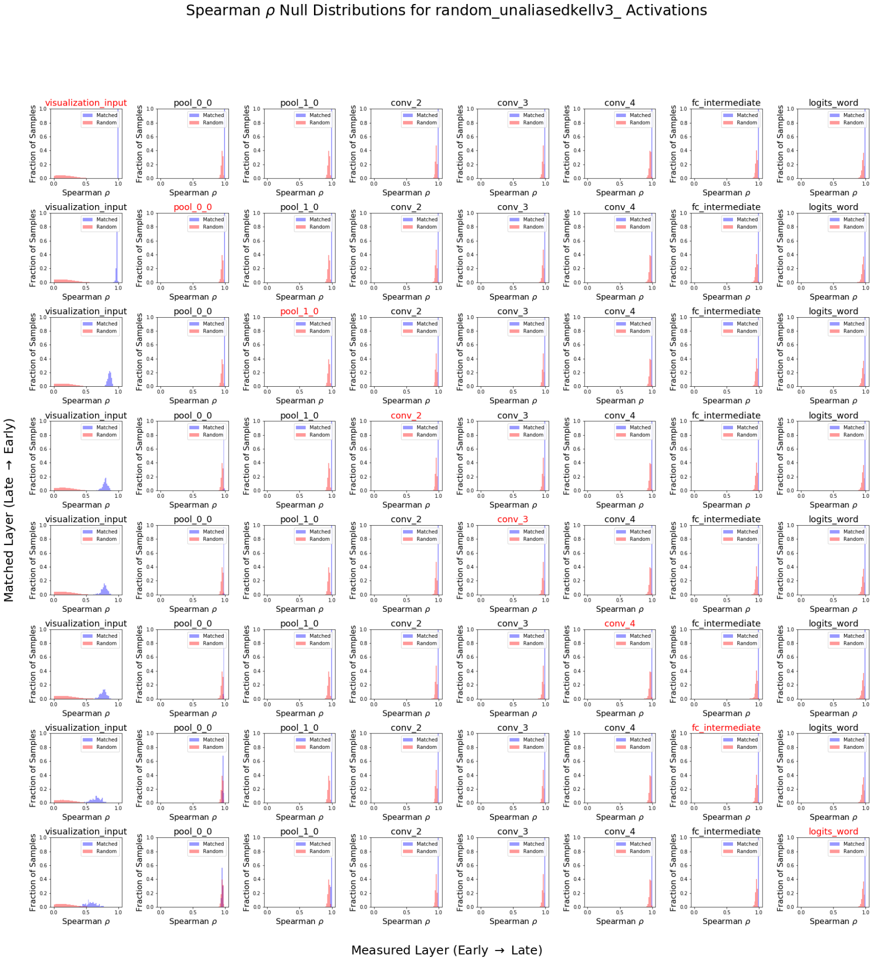


Figure S3: Network metamer Spearman correlation coefficients compared with the null correlation distribution for the Random Word Task CNN (with reduced aliasing). Even though the null distribution correlations are very high for deep layers in this network, there is no overlap between the null distributions and the distribution from model metamers used for the experiments.

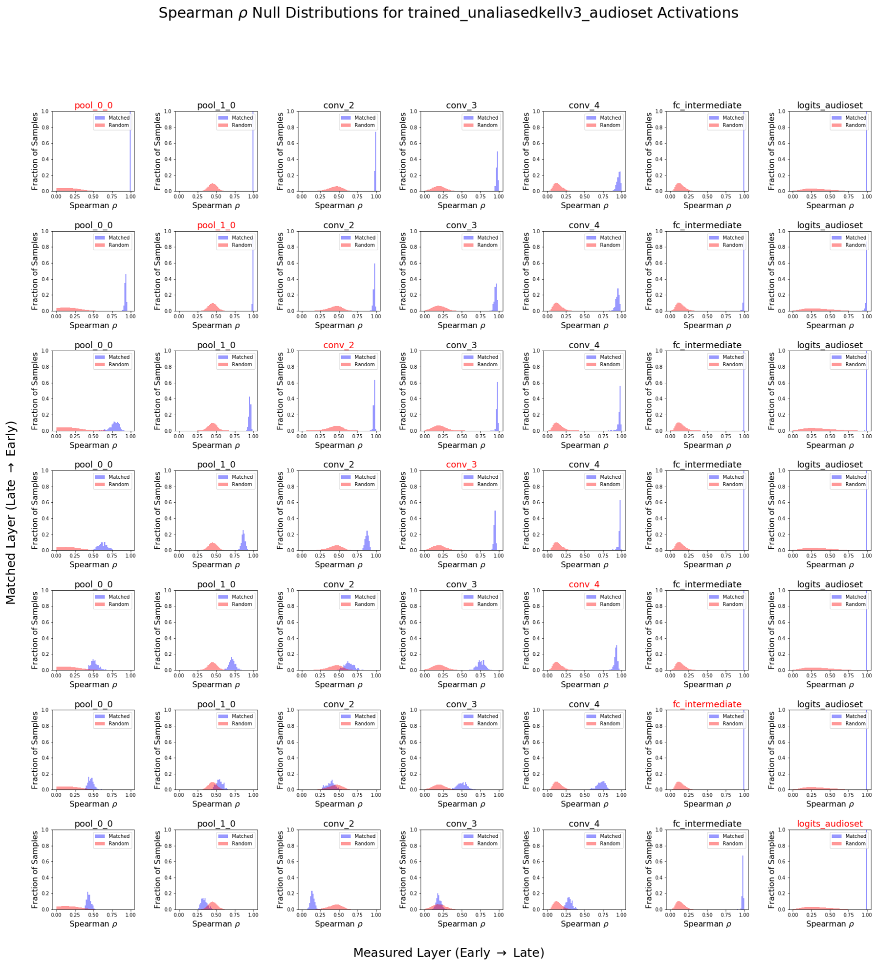


Figure S4: Network metamer Spearman correlation coefficients compared with the null correlation distribution for the Audioset Task CNN (with reduced aliasing).

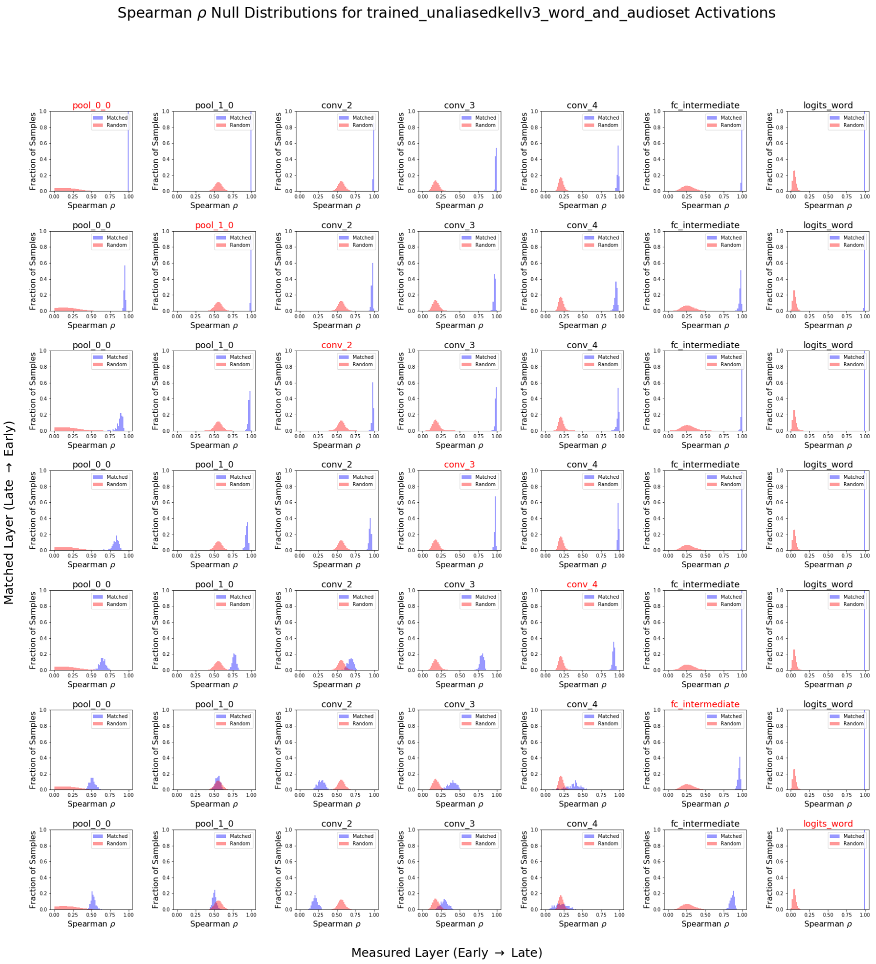


Figure S5: Network metamer Spearman correlation coefficients compared with the null correlation distribution for the Word and Audioset Task CNN (with reduced aliasing).

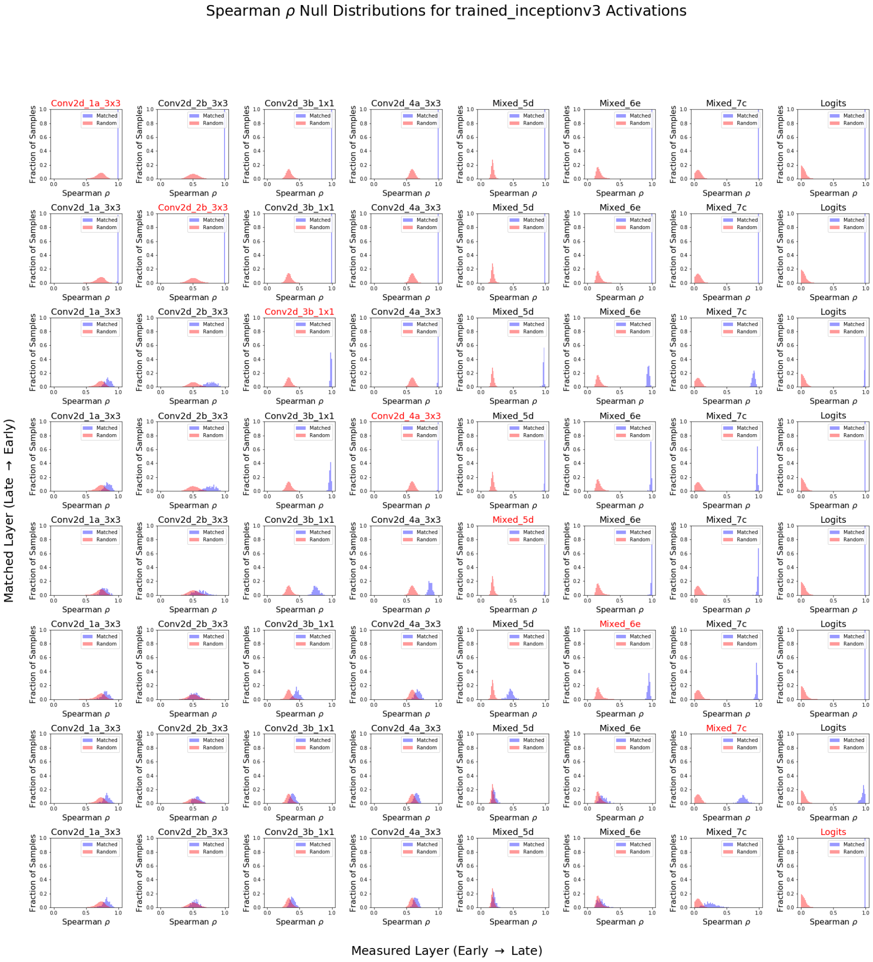


Figure S6: Model metamer Spearman correaltion coefficients compared with the null correlation distribution for Inception-V3. Metamers were generated for layer Mixed_7c, however the optimization did not succeed to the same extent as the other layers (with a median Spearman ρ below 0.9) and we thus do not report behavioral results for this layer.

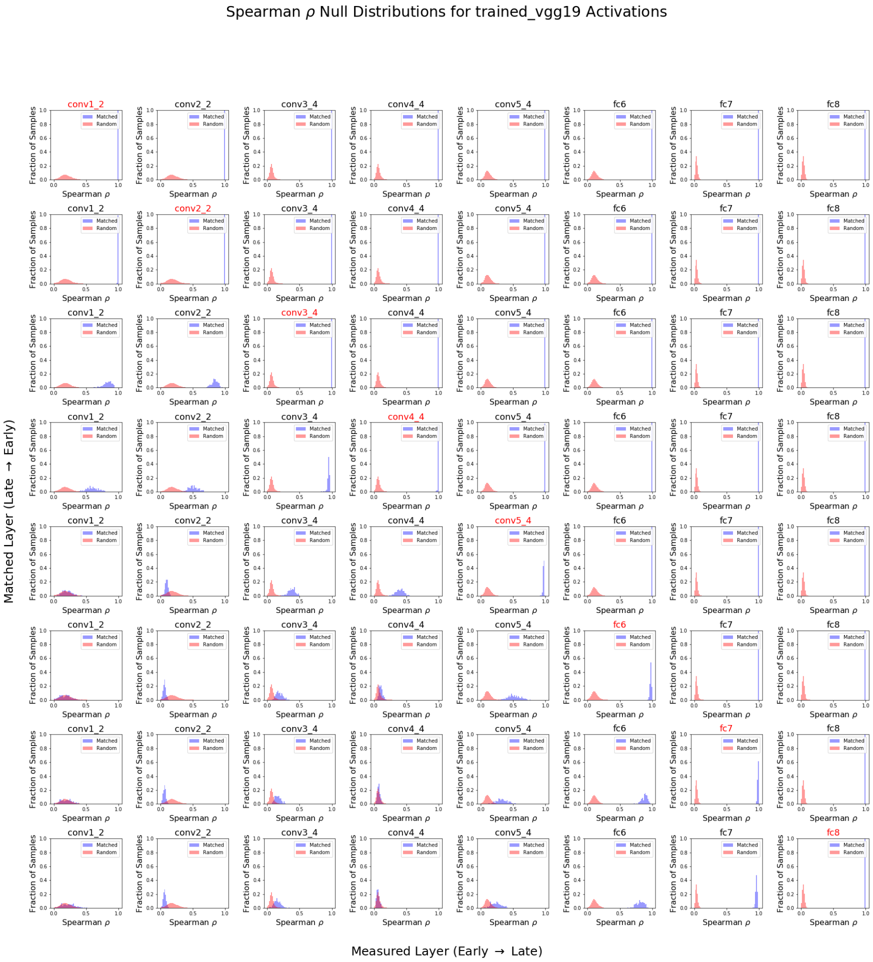


Figure S7: Network metamer Spearman correlation coefficients compared with the null correlation distribution for VGG-19.

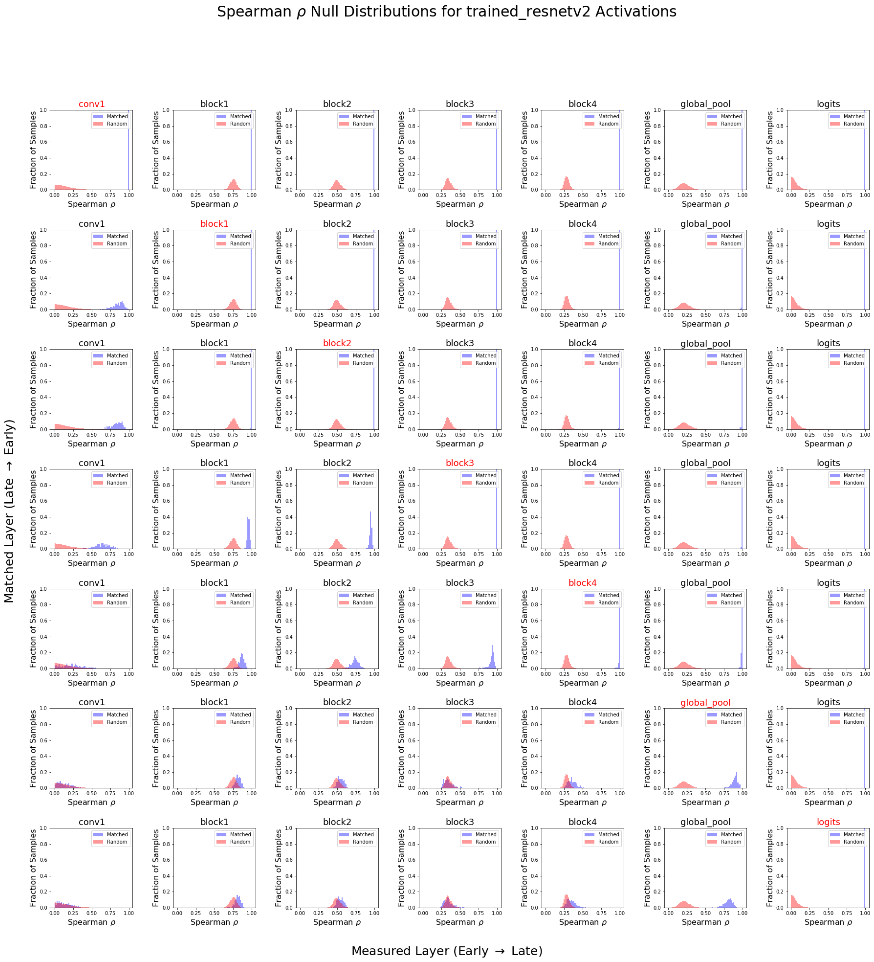


Figure S8: Network metamer Spearman correlation coefficients compared with the null correlation distribution for Resnet-V2-101.

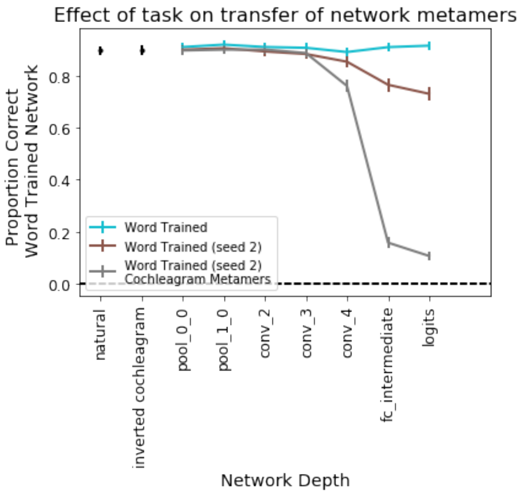


Figure S9: Transfer of metamers between the same architecture and task but different random seeds when generating the metamer by optimizing the waveform (as in all our main experimental conditions, because we needed to present the stimuli as sounds to human listeners) vs. the cochleagram. The audio waveform-generated metamers transfer between two architectures trained on different random seeds, while the cochleagram-generated metamers do not. This suggests that including the cochleagram generation stages in the optimization imposes additional constraints on the audio that restrict the representational capacity, increasing the likelihood of transfer across models. Quality of cochleagram metamer generation is summarized in Table S1.2

	Number		**Median**	**Median**	**Median**
Network Metamer	**Generated**	**Number**	**Spearman ρ**	**Spearman ρ**	**Spearman ρ**
Generation Layer	**Metamers**	**features**	**at Layer**	**Null at Layer**	**at Logits**
Natural Sound					
orig	295	84400	-	-	-
visualization	295	84400	0.998674	0.179334	0.999976
Word Trained (Reduced Aliasing), Cochleagram Metamers					
pool_0	293	913344	0.999119	0.561209	0.999974
pool_1	292	156672	0.998646	0.602778	0.999982
conv_2	293	78336	0.998229	0.205391	0.999989
conv_3	294	156672	0.995979	0.280117	0.999976
conv_4	293	78336	0.983156	0.048382	0.999993
fc_intermediate	290	4096	0.992512	0.147935	0.998165
logits	281	794	0.995095	0.147180	0.995095

Table S6: Summary of network metamer generation for audio metamers generated by optimizing the cochleagram.