Robust deep learning object recognition models rely on low frequency information in natural images

Zhe Li[1,*,‡], Josue Ortega Caro[1,*], Evgenia Rusak[2], Wieland Brendel[2], Matthias Bethge[2], Fabio Anselmi[1], Ankit B. Patel[1,3,4,+], Andreas S. Tolias[1,3,4,+,‡], and Xaq Pitkow[1,3,4,+,‡]

[1]Department of Neuroscience, Baylor College of Medicine, Houston, 77030, USA
[2]University of Tübingen, Germany
[3]Department of Electrical and Computer Engineering, Rice University, Houston, 77005, USA
[4]Center for Neuroscience and Artificial Intelligence, Baylor College of Medicine, Houston, 77030, USA
[*]co-first authors
[+]co-senior authors
[‡]co-corresponding authors

ABSTRACT

Machine learning models have difficulty generalizing to data outside of the distribution they were trained on. In particular, vision models are usually vulnerable to adversarial attacks or common corruptions, to which the human visual system is robust. Recent studies have found that regularizing machine learning models to favor brain-like representations can improve model robustness, but it is unclear why. We hypothesize that the increased model robustness is partly due to the low spatial frequency preference inherited from the neural representation. We tested this simple hypothesis with several frequency-oriented analyses, including the design and use of hybrid images to probe model frequency sensitivity directly. We also examined many other publicly available robust models that were trained on adversarial images or with data augmentation, and found that all these robust models showed a greater preference to low spatial frequency information. We show that preprocessing by blurring can serve as a defense mechanism against both adversarial attacks and common corruptions, further confirming our hypothesis and demonstrating the utility of low spatial frequency information in robust object recognition.

Introduction

Currently, deep neural networks are the state-of-the-art models for numerous computer vision tasks such as object detection[1], image recognition[2], semantic segmentation[3], etc. However, these models are often not robust, as demonstrated by their inability to generalize to new data distributions. For instance, current neural networks are not able to generalize to common corruption noise such as ImageNet-C[4], where network performance is stress tested against 15 different kinds of image corruption applied to the ImageNet dataset. Furthermore, these models seem to be extremely sensitive to targeted noise such as adversarial attacks[5]. In contrast, the human visual system does not seem to suffer from such problems: in particular, recognizing object identity is little affected by the common corruptions[6], and adversarial perturbations that break machine learning models are imperceptible to humans[7]. This difference in behavior between the brains and deep learning algorithms might be explained by differences in inductive bias, i.e. they learn different features from data. Accordingly, natural vision has an inductive bias, a bias towards which fixed network are learned by the optimization algorithm from a class of models given the set of training data, towards robust features, which means insensitivity to perturbations that do not change the perceptual relevant latent variables such as object identities. How can we instill the brain's inductive bias towards robust to targeted and random noise distortions to these deep learning algorithms? Recent work has shown that machine learning models which are encouraged to learn brain-like representations, a paradigm known as neural regularization, are also more robust to certain common corruptions such as Gaussian noise and adversarial attacks[8,9]. Furthermore, other work has shown that models that explain more variation in primate primary visual cortical (V1) responses also tend to be more robust to adversarial attacks[10].

Much recent parallel work has also attempted to produce models that are robust to common corruptions (ANT[11], SIN[12], DeepAugment, AugMix[4]) and to adversarial attacks (PGD Training[13], TRADES[14]). These models achieve significant improvements upon baseline models by employing several other methods, including data augmentation, adversarial training, and anti-aliasing[15]. AutoAugment, a data augmentation method for common corruption robustness, has produced improvements mainly for high frequency common corruption noises[16]. Consistent with this work, we hypothesize that one of the reasons for the success of current robustness methods for both common corruption and adversarial attacks can be explained by a simple computational principle: models that are biased to rely on low spatial frequency information for object recognition are more robust. Here, we tested this hypothesis with several frequency-oriented analyses. Finally, we introduced a simple preprocessing step based on these principles that produced robust models with comparable performance to these more sophisticated methods.

Results

Neural regularization boosts model robustness

There are many ways to bias a machine learning model toward brain-like computations, such as architecture, learning rules, the training dataset or task's objective functions[17]. Among them, methods that use auxiliary loss functions to bias models towards brain-like representations are called neural regularization. We will focus on two particular forms of neural regularization: neural similarity regularization[8, 18] and neural response regularization[9]. Both methods directly encourage a model to learn a brain-like representation of the visual stimuli.

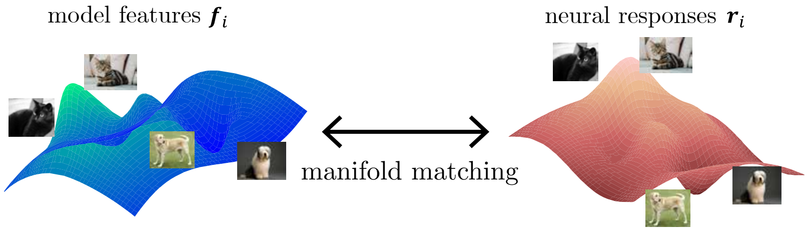
model features $\boldsymbol{f}_i$
neural responses $\boldsymbol{r}_i$
manifold matching

Figure 1. Schematic of neural similarity regularization. Machine learning models are trained so that the stimuli manifold in the model feature space resembles the manifold in the neural response space. The most simple version considers only pairwise relationships among stimuli instances. Two images that are close in the neural response space should also be close in the model feature space.

The neural responses of natural images are located on a low dimension manifold in the response space[19]. Similarly, for any machine learning model, features of images stimuli also form their own manifold embedded in model feature space. Neural similarity regularization adjusts the manifold in the model's hidden layers so that it is biased towards the same geometry as the manifold in the neural response space (Fig. 1), while performing benchmark tasks like image classification at the same time. Li et al.[8] showed that a ResNet18 model regularized to incentivize its representational similarity[20], which is the pair-wise cosine similarity between the representation for a set of images, to be more like the one measured in mouse V1 is more robust against Gaussian noise and adversarial attacks.

Another way to induce brain-like representations is to request the model to predict neural responses of the visual input as an auxiliary task. Safari et al.[9] showed that a VGG19 model co-trained with monkey V1 data, where the VGG19's core was used to predict neural responses directly, is able to improve robustness against common corruptions on Tiny ImageNet. Since the neural readout module is shallow, the VGG19 core has to encode neural features to make the co-training work.

In this study, we expanded both neural regularization, either using mouse's representational similarity matrix or using the monkey responses directly, by testing model robustness against various common corruptions and adversarial attacks.

Since neurally regularized models are trained on grayscale images, we used grayscale versions of the CIFAR10-C and TinyImageNet-C datasets for evaluation. Models regularized with either mouse or monkey V1 representations have higher accuracy at different severity levels of common corruptions compared to baseline models (Fig. 2a, e), though their performance on clean images is slightly worse. Targeted gradient-based boundary attacks[21] were performed to find the minimum perturbations needed to change model predictions to wrong categories. The use of strong attacks is critical in evaluating adversarial robustness, as weak attacks such as FGSM only provide a loose upper bound on the minimum adversarial perturbation size. We verified the effectiveness of boundary attacks by extensive comparison between many off-the-shelf methods, and are confident it characterizes the adversarial robustness of models. Evaluation on the full testing dataset showed that neurally regularized models need larger adversarial perturbations on average (Fig. 2b, f), i.e. they are more robust against adversarial attacks. The attack size that gives 50% success rate is $\varepsilon = 1.25/255$ for the baseline ResNet and $\varepsilon = 2.89/255$ for the mouse regularized one. If we look at the distribution of minimum perturbation size needed for each image, the mean and standard deviation is $\varepsilon = (1.34 \pm 0.70)/255$ for the baseline ResNet, and $\varepsilon = (3.09 \pm 1.61)/255$ for the mouse regularized one. Similarly, the attack size that gives 50% success rate is $\varepsilon = 1.84/255$ for the baseline VGG and $\varepsilon = 2.46/255$ for the monkey regularized one. And the minimum perturbation size for all images is $\varepsilon = (1.93 \pm 0.89)/255$ for the baseline VGG, and $\varepsilon = (2.64 \pm 1.26)/255$ for the monkey regularized one. In the following text, we use the mean value of minimum perturbation sizes to characterize adversarial robustness of the model.

A natural question is: what does the neural regularization do? First, we decided to explore the structure of the neural similarity matrix of mouse V1 responses with a simple decomposition analysis. We calculated the eigenvalues and eigenvectors

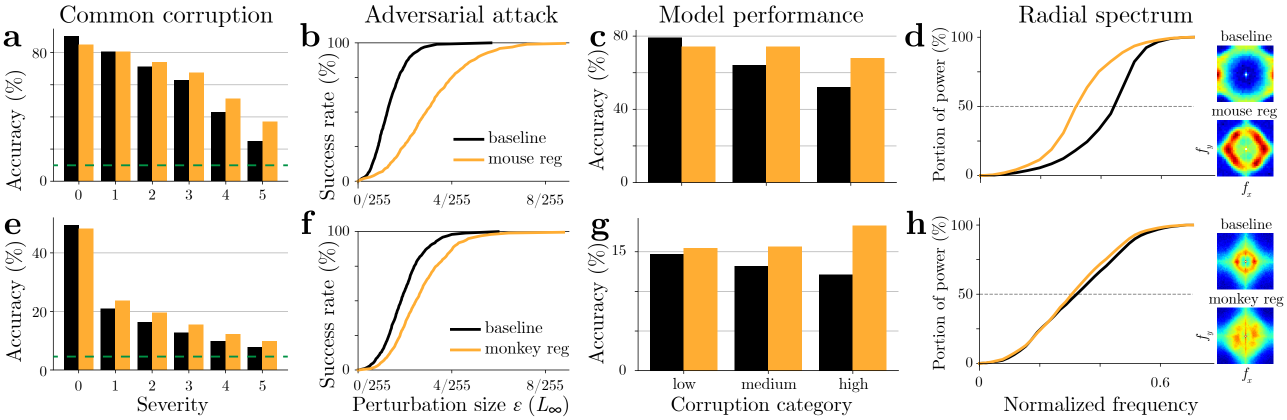

Figure 2. Neural regularization boosts model robustness and makes it less sensitive to high frequency component of the input. ResNet18 models are trained for grayscale CIFAR10 with mouse neural similarity regularization[8], VGG19 models are trained for grayscale TinyImageNet with monkey neural response regularization[9]. (a) Grayscale CIFAR10 classification accuracy against common corruptions at different severity levels. Average accuracy over all corruptions are reported for a baseline ResNet model (black) and a mouse regularized model (orange). (b) Success rate of targeted attacks at different perturbation budget ε, using the boundary attack[21] with an L_∞ metric. (c) Classification accuracy against different types of corruptions, broken down into three groups based on their frequency characteristics (Appendix Tab. 1). Model performance is averaged over all severity levels. (d) Radial profile of the Fourier spectrum of adversarial perturbations. We found the minimal adversarial perturbations of all testing images, and calculated the averaged Fourier spectrum thereof, where blue is minimum and red is maximum values (insets). The portion of power under different frequency thresholds are compared between baseline and neurally regularized models. The abscissa is the absolute value of the spatial frequency, normalized by sampling frequency f_s. (e–h) Same as a–d, except comparing a baseline VGG model with a model co-trained with monkey neural data on the grayscale TinyImageNet dataset[22].

of the similarity matrix and computed the linear approximation of each principal component with respect to the image. We observed that geometry of mouse V1 representation is well approximated by a small number of principal components, and the spatial tuning of them reveals low frequency structure (Appendix Fig. 10). This has led us to ask if the bias towards low frequency features is inherited by neurally regularized models. We divided the 15 common corruptions in the CIFAR10-C and TinyImageNet-C datasets[23] into three categories based on their frequency spectra (Appendix Tab. 1). Low-frequency corruptions: 'snow', 'frost', 'fog', 'brightness', 'contrast'; medium-frequency corruptions: 'motion_blur', 'zoom_blur', 'defocus_blur', 'glass_blur', 'elastic_transform', 'jpeg_compression', 'pixelate'; high-frequency corruptions: 'gaussian_noise', 'shot_noise', 'impulse_noise'. Fourier spectra of different corruptions are shown in Appendix Fig. 7. The biggest performance boost in model classification accuracy comes from the category with the high-frequency corruptions (Fig. 2c, f).

We also compared the adversarial perturbations for baseline models and neurally regularized ones. We performed a Fourier analysis on the minimal adversarial perturbation found through our boundary attack[21], and calculated the average frequency spectrum for different models (insets of Fig. 2d, h). We observed that the mouse and monkey neurally regularized models contain relatively higher low-frequency components than the baseline model. To quantify this frequency shift, we characterized the frequency preference by a radial profile of the spectrum of the adversarial perturbation, *i.e.* the power of all Fourier components whose frequency is smaller than certain values. We found the radial profile of the adversarial perturbation spectrum for the neurally regularized model is shifted toward lower frequencies (Fig. 2d, h). Furthermore, we can quantify this shift by the half power frequency $f_{0.5}$, which is the frequency where half of the Fourier power lies below (marked by the dashed line in Fig. 2d, h). $f_{0.5} = 0.316 \pm 0.015$ for mouse regularized ResNet and $f_{0.5} = 0.442 \pm 0.009$ for baseline, with mean and standard deviation estimated from 1000 images (Fig. 2d). Similarly, $f_{0.5} = 0.306 \pm 0.020$ for monkey regularized VGG and $f_{0.5} = 0.316 \pm 0.034$ for baseline (Fig. 2h). The half power frequency $f_{0.5}$ is smaller for the neurally regularized model compared with baseline, though effect on the monkey regularized model is much weaker than the mouse regularized one.

Hybrid image experiment

To directly probe the frequency bias of models, we next designed a new dataset of hybrid images constructed by mixing the low-frequency components and high-frequency components from two different images. We select two images belonging to two different categories, and examine whether the model prediction on the mixed image is consistent with either component. For a given mixing frequency f_{mix}, we combine the Fourier components of an image whose frequencies are smaller than f_{mix} with

Fourier components of another image whose frequencies are larger than f_{mix}, and then use an inverse Fourier transformation to get a hybrid image (Fig. 3a, see Methods for more details).

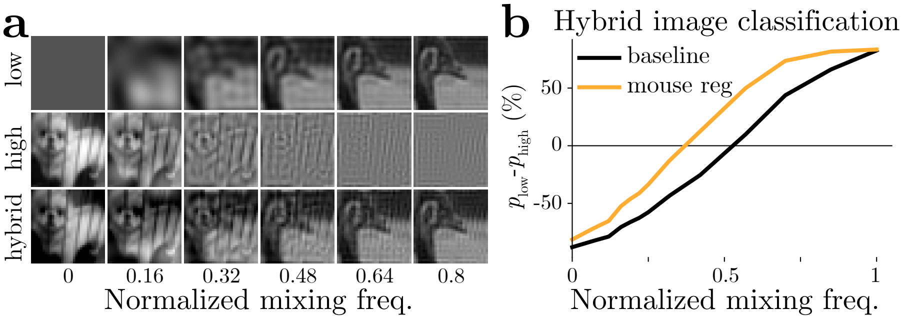

Figure 3. Probing frequency sensitivity of mouse regularized model using hybrid images. (a) Examples of hybrid images at different mixing frequencies. Hybrid images are constructed by mixing the low-frequency component of one image and the high-frequency component of another, while the two seed images belong to different categories. The range of mixing frequency's values are normalized by the Nyquist frequency. (b) Model predictions on hybrid images at different mixing frequencies. As more low-frequencies from one image are included, the probability that a network reports its label p_{low} increases. The reversal frequency f_{rev} where $p_{\text{low}} = p_{\text{high}}$ is smaller for the mouse regularized model ('neural') than for the baseline model ('base').

We denote the probability that a hybrid image is classified as the low-/high-frequency seed image class as p_{low} and p_{high} respectively, and calculate the probability difference $p_{\text{low}} - p_{\text{high}}$ at different mixing frequency values. When the mixing frequency is very small, the hybrid image is close to the high frequency seed image, therefore p_{high} is high for a properly trained model. Likewise, when the mixing frequency is big, p_{low} is high. We are interested in the mixing frequency when $p_{\text{low}} = p_{\text{high}}$, as it characterizes the frequency bias of a model. We term this frequency the reversal frequency f_{rev}. The result shows that f_{rev} is smaller for a mouse regularized model (0.371 ± 0.0006, mean ± standard deviation estimated from 4 randomly permuted hybrid image datasets) than a baseline model (0.528 ± 0.0009), indicating that the neurally regularized model is more likely to classify the hybrid image as the class of the low-frequency component (Fig. 3b). This provides strong evidence that the reason behind the robustness gain by mouse neural regularization could be a bias towards low-frequency features. The same experiments and analysis were done for monkey regularized models as well, but the low-frequency bias is smaller compared to mice. f_{rev} is 0.376 ± 0.003 and 0.426 ± 0.002 for monkey regularized VGG19 and a baseline model, respectively (Appendix Fig. 8).

Frequency analysis on robust models

We then asked if other models that are not regularized with neural data, but engineered to be robust to common corruptions and adversarial attacks also have this low frequency bias. To this end, we analyzed and compared several robust models trained on CIFAR10, most of which were downloaded from RobustBench[24]. Some of the models are trained for adversarial robustness and some are trained for common corruption robustness. A full description of models is listed in Appendix Tab. 2. Since the mouse regularized models in Fig. 2 are trained on grayscale CIFAR10, it is not included in this comparison.

Two additional models were trained and included in this comparison. Both models were constructed by attaching a preprocessing layer before a ResNet18 model. The parameters of the preprocessing are not trainable but fixed beforehand. One is called the 'blur' model as the preprocessing is a convolution with a Gaussian kernel. The other is called the 'PCA' model as the preprocessing keeps only the first a few principal components (PCs) calculated from all training images. The standard deviation σ of the Gaussian kernel and the number of PCs K are chosen such that the classification accuracy on CIFAR10 of both models are around 90%, similar to other robust models. More details are included in the Methods section.

The 'blur' model directly biases the model towards low-frequency features, because the high frequency features are attenuated during the preprocessing step. The filtering of features based on their spatial frequency can be treated as a special form of reweighting of input principal components, since principal component analysis on natural images with translation invariance recovers the Fourier basis, and the variance explained by a feature usually decreases monotonically as frequency increases. Hence a 'PCA' model that explicitly projects the original image onto a low dimensional space spanned by the first PCs is also included in the comparison. The hypothesis behind 'PCA' model is that the directions parallel to the data manifold are robust features, and perturbations that are orthogonal to the manifold can be safely removed. Since natural images have the greatest variance in low frequencies, the distance of any data point to a class boundary are longest along low frequency dimensions, and thus a model that mainly uses these low frequencies should be the more robust. Previous work also found that this kind of PCA data preprocessing increases robustness to adversarial attacks[25].

Similar to the analysis done on neurally regularized models, we calculated the Fourier spectrum of minimum adversarial perturbations for all robust models. A fixed set of 1000 images were selected from testing set, and the incorrect class targets for each image was also fixed. The adversarial perturbations for robust models, including the ones not specifically trained for adversarial robustness, contained more low spatial frequency components, while in the baseline models, adversarial perturbations were dominated by high spatial frequency components (Fig. 4a). To better visualize the differences of adversarial spectrum shape, we plotted the radial spectrum for each model. The portions of the spectral power within different spatial frequencies are compared (Fig. 4b), and the half power frequency $f_{0.5}$ is marked by the dashed line. Compared with the baseline model, all robust models have smaller $f_{0.5}$, indicating the minimum adversarial perturbations for them have lower frequencies. The size of average minimal perturbations are plotted against $f_{0.5}$ in Fig. 4c, revealing a negative correlation between adversarial robustness and the frequency bias of the model.

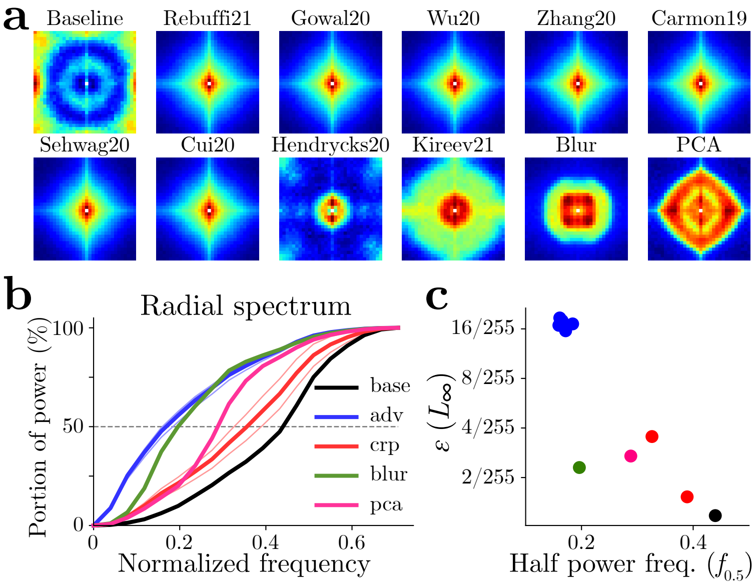

Figure 4. Frequency analysis of adversarial attacks on robust models trained on CIFAR10. (a) The Fourier spectrum of the minimal adversarial perturbations of different models, including one baseline model, seven models trained for adversarial robustness, two models for corruption robustness, one model preprocessing by blurring, and one preprocessing by PCA compression (Appendix Tab. 2). The spectrum is averaged over 1000 images, and color maps are not shared across each panel. (b) Radial profiles of adversarial perturbation spectra. Dotted lines represent each individual model, while thick lines are the average within each robustness method ('adv', 'crp'). The frequency where each line crosses 50% is denoted as half power frequency $f_{0.5}$. (c) Scatter plot of minimum adversarial perturbation size versus $f_{0.5}$ for all models.

We next tested model predictions on hybrid images. Hybrid images for RGB CIFAR10 are constructed similarly as in Fig. 3a. The reversal frequencies f_{rev} for all models are plotted against the classification accuracy on the CIFAR10-C dataset (averaged over all corruptions and severity levels) in Fig. 5b. The result shows that all models are more robust to common corruptions than the baseline, even those trained for adversarial robustness. They all have a low frequency bias characterized by f_{rev}, which suggests that low frequency information is more important to these models. However, one interesting observation is that compared to adversarial attacks, common corruption robustness seems to cap in an specific f_{rev}. For example the adversarially robust models (blue) have smaller corruption accuracy compared to the other robust models. This seems to indicate a trade-off, where using too few frequencies ends up affecting performance. We decided to explore this by using different 'blur' models with increasing blurring kernels' size. In Fig. 5b, we observe that most models except the ones trained for common corruption robustness lie close to the curve (green) fit to 'blur' models. This is an indication that the decrease in performance as a function of f_{rev} can be partly explained through the frequency preference of the models.

Next we explored this relationship between robustness and frequency bias on models trained for ImageNet classification. In addition to a baseline model, we analyzed two adversarially trained models and six models trained for robustness to common corruption (Appendix Tab. 3). In Fig. 6a, we plot the minimum adversarial perturbation size ε with respect to the half-power frequencies, $f_{0.5}$, of adversarial perturbation spectra. We can observe that the minimum perturbation size is negatively correlated

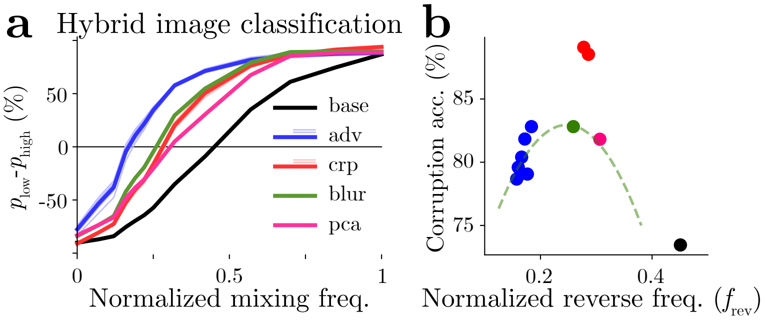

Figure 5. Hybrid CIFAR10 image classification performance of robust models. (a) Difference between probability of choosing the low frequency label vs the high frequency label of hybrid images. Thin and light lines of 'adv' models and 'crp' models are for each individual model in Appendix Tab. 2, and the dark line is the average within either group. (b) Scatter plot of common corruption robustness versus reversal frequency f_{rev} in hybrid image classification. The dashed green line is the performance of a series of 'blur' models, using different degree of low-pass filtering. The left end corresponds to $\sigma = 3$ pixels and the right end corresponds to $\sigma = 1$ pixel, while the green dot is the model with $\sigma = 1.5$ pixel listed in Tab. 2.

with half-power frequencies, this indicates that robustness to adversarial attacks is correlated with adversarial attacks that have more energy in the low frequencies. Furthermore, Fig. 6b shows corruption accuracy versus the reverse frequency, f_{rev}, for the ImageNet-C dataset. Here we observe that the common corruption robust models are more robust than the baseline and have smaller f_{rev}, consistent with our findings in CIFAR10 models. However the adversarially robustness models are less robust than baseline even though they have smaller f_{rev}. This is probably because the clean performance of adversarially robust models on this dataset is too low, therefore they are not comparable to baseline (Appendix Tab. 3).

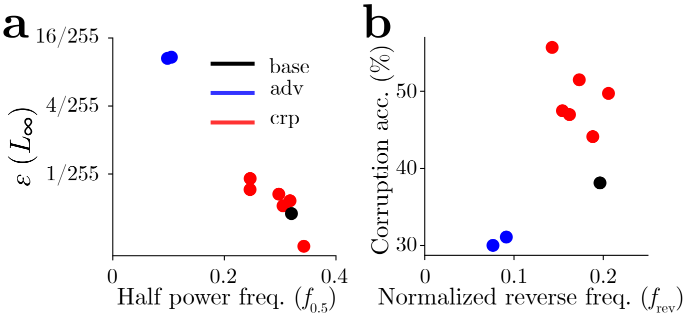

Figure 6. Frequency analysis of models trained on ImageNet. One baseline model ('base'), two models ('adv') trained for adversarial robustness, and six models ('crp') trained for corruption robustness are compared. (a) Minimum adversarial perturbation size ε versus the half-power frequency $f_{0.5}$ calculated from adversarial perturbation spectra. (b) Common corruption robustness versus reverse frequency f_{rev} calculated from hybrid image experiment.

Discussion

Recent studies have linked brain-like representations to robustness of deep network models[8–10]. By introducing a novel hybrid image task, we were able to study the frequency preference of a neural networks, and show that these brain-like models were more robust mainly through a preference towards low spatial frequency features. This work follows in spirit the approach of Geirhos et al.[12], where the authors combined the shape of an input and the texture of another one to produce an input that had features with "conflicting information" towards different classes. This approach of producing cue-conflict style tasks seems to be useful for understanding the model preference towards certain features and not others.

Furthermore, we have shown that this bias is present in an extensive group of machine learning models trained for robustness: we found that robustness to common corruptions and adversarial examples are correlated with this low frequency bias. However, there seems to be a difference between robust models on CIFAR10 vs ImageNet. Adversarially robust models are more robust to common corruptions on CIFAR10 but not on ImageNet. Previous work has also shown that adversarially robust models are not robust to common corruptions[26]. However, they attributed this behavior to both perturbations having qualitative different properties. In our case, we argue that adversarial robustness in ImageNet dataset is more difficult to achieve, and therefore the models we analyzed cannot handle perturbations as large as the ones in common corruptions compared to CIFAR10 trained models. All of this seem to indicate that robustness to different datasets might be achieved trough different methods but low-frequency preference seem to be a key component of current robust models.

In addition, we were able to use the strategy of biasing towards low spatial frequencies and enforced it with a simple preprocessing layer to produce robustness on par with this group of robust machine learning methods. Previous work has also tried to use frequency information to produce robust models for ImageNet. For example, antialising has been shown to improve robustness to common corruptions when introduced before or after the nonlinearities of different models[27,28]. However, this method seems to behave differently than our preprocessing models. For example, these anti-aliasing models decrease in performance as you go into the high frequency corruptions, similar to the baseline models. This is in contrast to the other robust models that perform better on high frequency corruptions compared to low frequency ones. As the authors explained in their work, the anti-aliasing methods focus more on producing shift-invariant models while preserving as much high frequency information as possible. This might be the reason why this model behaves differently than the 'blur' or 'PCA' models which explicitly attenuate high frequency information from the input, and why our models behave more similarly to neurally regularized and data augmented models. This indicates that there can be different approaches to producing robust models to common corruptions with focus on different properties of the frequency spectrum.

However, as previous work has shown, using low frequency bias has its limitations. For example, Yin et al.[16] have shown that training a model on the "Fog" common corruption only using the frequency, but not the spatial information is insufficient to generalize to the same "Fog" common corruption during testing. This makes sense given that the corruption has a very specific spatial structure that does not depend only on the frequency. In addition, as we observed in Fig. 5b and Fig. 6b, the bias towards low frequencies can be too strong, such that the performance deteriorates. This suggests that to reduce the gap between humans and machine learning models in out-of-distribution generalization, we must move beyond this low frequency-based preference found in current robust machine learning models and find a better and more principled inductive bias.

One possibility to achieve this grand goal is to use ideas from neuroscience. As our and previous work has suggested, neuroscience has provided inspiration to machine learning, but specific paradigms to directly translate biological scientific insights and data into a performance improvement in machine learning are largely absent. The neural regularization approach is a demonstration that this direction can help engineer more intelligent algorithms to usher the new field of NeuroAI. Furthermore, given that current machine learning models are still not robust to adversarial attacks and common corruptions, this work is just the start of bridging the gap between natural and artificial intelligence.

Methods

Hybrid images

We randomly select two images I_{low} and I_{high} from two different categories as seed images for low-frequency and high-frequency components respectively. The 2D Fourier transform of these two images are denoted as $\tilde{I}_{\text{low}}$ and $\tilde{I}_{\text{high}}$, and we define a binary mask on frequency domain as

$$M(f) = \begin{cases} 1, & ||f|| \leq f_{\text{mix}} \\ 0. & \text{otherwise} \end{cases} \tag{1}$$

The Fourier transform of the hybrid image is thus the combination of $\tilde{I}_{\text{low}}$ and $\tilde{I}_{\text{high}}$ through M, according to $\tilde{I}_{\text{hybrid}} = \tilde{I}_{\text{low}} \odot M + \tilde{I}_{\text{high}} \odot (1-M)$, where $\odot$ denotes an element-wise product. The hybrid image I_{hybrid} is calculated from $\tilde{I}_{\text{hybrid}}$ using an inverse Fourier transform.

Blur model

We designed a blur model to explicitly implement a frequency bias. Our blur model is simply a ResNet model prepended with a blurring layer. The blurring layer is a linear convolutional layer whose kernel weight is fixed as

$$w(\Delta x, \Delta y) = \frac{1}{Z} \exp\left(-\frac{\Delta x^2 + \Delta y^2}{2\sigma^2}\right) \tag{2}$$

in which Z is the normalization factor. The blurring layer can pass a gradient back to the input image, so gradient-based adversarial attacks can be performed. Clean performance of the 'blur' model decreases as σ increases because more information is discarded. We choose the value of σ such that the trained model has a similar classification accuracy on CIFAR10 testing set compared with other robust methods; $\sigma = 1.5$ pixel is selected.

PCA model

Similar to the blurring model, the PCA model is also a ResNet model with a non-trainable preprocessing layer. Treating the input image as a vector x whose dimension N is the number of pixels, we first performed principal component analysis (PCA) on the training set and obtained the eigenvectors v_i ($1 \leq i \leq N$) sorted by their eigenvalues. We took the first K eigenvectors, denoted by a $K \times N$ matrix W_K, and constructed the filtering matrix

$$W = W_K \cdot {W_K}^\mathsf{T}. \tag{3}$$

The PCA preprocessing layer is a linear layer whose input and output dimensions are both N, with its weights given by the matrix W. This layer projects the original image onto the subspace spanned by the first K eigenvectors, and effectively denoises the image. Similarly to the blur model, the PCA model allows gradient backward passes to the input image, and can be attacked with gradient-based adversarial attacks. The value of K is chosen so that the trained model has a similar clean performance compared with other models, and $K = 512$ was selected for the CIFAR10 dataset.

Contributions

ZL, JOC, AT, XP, AP conceived the conceptualized framework. ZL and JOC trained and analyzed models. ER contributed robust models. ZL, JOC, WB, MB contributed adversarial analyses. ZL wrote the first draft, and ZL, JOC, FA, AT, XP, AP revised and edited. AT, XP, AP supervised the project. All authors provided comments on the final manuscript.

Acknowledgements

It is supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior/Interior Business Center (DoI/IBC) contract number D16PC00003. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoI/IBC, or the U.S. Government. Research reported in this publication was supported by the National Eye Institute of the National Institutes of Health under Award Numbers R01EY026927. This research was also supported by NEI/NIH Core Grant for Vision Research (EY-002520-37).

References

1. Girshick, R. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, 1440–1448 (2015).
2. He, K., Zhang, X., Ren, S. & Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778 (2016).
3. He, K., Gkioxari, G., Dollár, P. & Girshick, R. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, 2961–2969 (2017).
4. Hendrycks, D. *et al.* Augmix: A simple method to improve robustness and uncertainty under data shift. In *International Conference on Learning Representations* (2020).
5. Szegedy, C. *et al.* Intriguing properties of neural networks. *arXiv e-prints* (2013).
6. Geirhos, R. *et al.* Comparing deep neural networks against humans: object recognition when the signal gets weaker. *arXiv e-prints* (2017).
7. Szegedy, C. *et al.* Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199* (2013).
8. Li, Z. *et al.* Learning from brains how to regularize machines. In Wallach, H. *et al.* (eds.) *Advances in Neural Information Processing Systems 32*, 9525–9535 (Curran Associates, Inc., 2019).
9. Safarani, S. *et al.* Towards robust vision by multi-task learning on monkey visual cortex. In *Advances in Neural Information Processing Systems 34* (Curran Associates, Inc., 2021).
10. Dapello, J. *et al.* Simulating a primary visual cortex at the front of cnns improves robustness to image perturbations. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M. F. & Lin, H. (eds.) *Advances in Neural Information Processing Systems 33*, vol. 33, 13073–13087 (Curran Associates, Inc., 2020).
11. Rusak, E. *et al.* A simple way to make neural networks robust against diverse image corruptions. In Vedaldi, A., Bischof, H., Brox, T. & Frahm, J.-M. (eds.) *Computer Vision – ECCV 2020*, 53–69 (Springer International Publishing, Cham, 2020).
12. Geirhos, R. *et al.* Imagenet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. In *International Conference on Learning Representations* (2019).

13. Madry, A., Makelov, A., Schmidt, L., Tsipras, D. & Vladu, A. Towards deep learning models resistant to adversarial attacks. *arXiv e-prints* (2017).

14. Zhang, H. *et al.* Theoretically principled trade-off between robustness and accuracy. In *International Conference on Machine Learning*, 7472–7482 (PMLR, 2019).

15. Zhang, R. Making convolutional networks shift-invariant again. In *International conference on machine learning*, 7324–7334 (PMLR, 2019).

16. Yin, D., Gontijo Lopes, R., Shlens, J., Cubuk, E. D. & Gilmer, J. A fourier perspective on model robustness in computer vision. In *Advances in Neural Information Processing Systems 32*, vol. 32, 13276–13286 (Curran Associates, Inc., 2019).

17. Sinz, F. H., Pitkow, X., Reimer, J., Bethge, M. & Tolias, A. S. Engineering a less artificial intelligence. *Neuron* **103**, 967–979 (2019).

18. Federer, C., Xu, H., Fyshe, A. & Zylberberg, J. Improved object recognition using neural networks trained to mimic the brain's statistical properties. *Neural Networks* **131**, 103–114 (2020).

19. Stringer, C., Pachitariu, M., Steinmetz, N., Carandini, M. & Harris, K. D. High-dimensional geometry of population responses in visual cortex. *Nature* **571**, 361–365, DOI: 10.1038/s41586-019-1346-5 (2019).

20. Kriegeskorte, N., Mur, M. & Bandettini, P. Representational similarity analysis - connecting the branches of systems neuroscience. *Front. Syst. Neurosci.* **2**, 4, DOI: 10.3389/neuro.06.004.2008 (2008).

21. Brendel, W., Rauber, J., Kümmerer, M., Ustyuzhaninov, I. & Bethge, M. Accurate, reliable and fast robustness evaluation. In Wallach, H. *et al.* (eds.) *Advances in Neural Information Processing Systems 32*, 12861–12871 (Curran Associates, Inc., 2019).

22. Le, Y. & Yang, X. Tiny imagenet visual recognition challenge. *CS 231N* **7**, 3 (2015).

23. Hendrycks, D. & Dietterich, T. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations* (2019).

24. Croce, F. *et al.* Robustbench: a standardized adversarial robustness benchmark. *arXiv e-prints* (2020).

25. Bhagoji, A. N., Cullina, D., Sitawarin, C. & Mittal, P. Enhancing robustness of machine learning systems via data transformations. In *2018 52nd Annual Conference on Information Sciences and Systems (CISS)*, 1–5 (IEEE, 2018).

26. Laugros, A., Caplier, A. & Ospici, M. Are adversarial robustness and common perturbation robustness independant attributes? In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, 0–0 (2019).

27. Zhang, R. Making convolutional networks shift-invariant again. In *International conference on machine learning*, 7324–7334 (PMLR, 2019).

28. Vasconcelos, C., Larochelle, H., Dumoulin, V., Roux, N. L. & Goroshin, R. An effective anti-aliasing approach for residual networks. *arXiv e-prints* (2020).

29. Rebuffi, S.-A. *et al.* Fixing Data Augmentation to Improve Adversarial Robustness. *arXiv e-prints* arXiv:2103.01946 (2021).

30. Gowal, S., Qin, C., Uesato, J., Mann, T. & Kohli, P. Uncovering the Limits of Adversarial Training against Norm-Bounded Adversarial Examples. *arXiv e-prints* (2020).

31. Wu, D., Xia, S.-t. & Wang, Y. Adversarial Weight Perturbation Helps Robust Generalization. *arXiv e-prints* arXiv:2004.05884 (2020).

32. Zhang, J. *et al.* Geometry-aware Instance-reweighted Adversarial Training. *arXiv e-prints* arXiv:2010.01736 (2020). 2010.01736.

33. Carmon, Y., Raghunathan, A., Schmidt, L., Liang, P. & Duchi, J. C. Unlabeled Data Improves Adversarial Robustness. *arXiv e-prints* (2019).

34. Sehwag, V., Wang, S., Mittal, P. & Jana, S. HYDRA: Pruning Adversarially Robust Neural Networks. *arXiv e-prints* (2020).

35. Cui, J., Liu, S., Wang, L. & Jia, J. Learnable Boundary Guided Adversarial Training. *arXiv e-prints* (2020).

36. Kireev, K., Andriushchenko, M. & Flammarion, N. On the effectiveness of adversarial training against common corruptions. *arXiv e-prints* (2021).

37. Engstrom, L., Ilyas, A., Salman, H., Santurkar, S. & Tsipras, D. Robustness (python library) (2019).

38. Hendrycks, D. *et al.* The Many Faces of Robustness: A Critical Analysis of Out-of-Distribution Generalization. *arXiv e-prints* (2020).

39. Lee, J. *et al.* Compounding the Performance Improvements of Assembled Techniques in a Convolutional Neural Network. *arXiv e-prints* arXiv:2001.06268 (2020).

40. Sinz, F. *et al.* Stimulus domain transfer in recurrent models for large scale cortical population prediction on video. In Bengio, S. *et al.* (eds.) *Advances in Neural Information Processing Systems 31*, 7199–7210 (Curran Associates, Inc., 2018).

Appendix

Common corruption categorization

Fourier analysis is performed for the perturbations induced by common corruptions in the CIFAR10-C dataset (at severity 5). All 15 corruptions are divided loosely into three categories based on their dominant frequencies (Tab. 1)

category	corruptions
low	snow, frost, fog, brightness, contrast
medium	motion_blur, zoom_blur, defocus_blur, glass_blur, elastic_transform, jpeg_compression, pixelate
high	gaussian_noise, shot_noise, impulse_noise

Table 1. Categorization of common corruptions. 15 types of corruptions[23] are divided into 3 categories based on the average frequency estimated from the Fourier spectrum of the perturbations (Appendix Fig. 7).

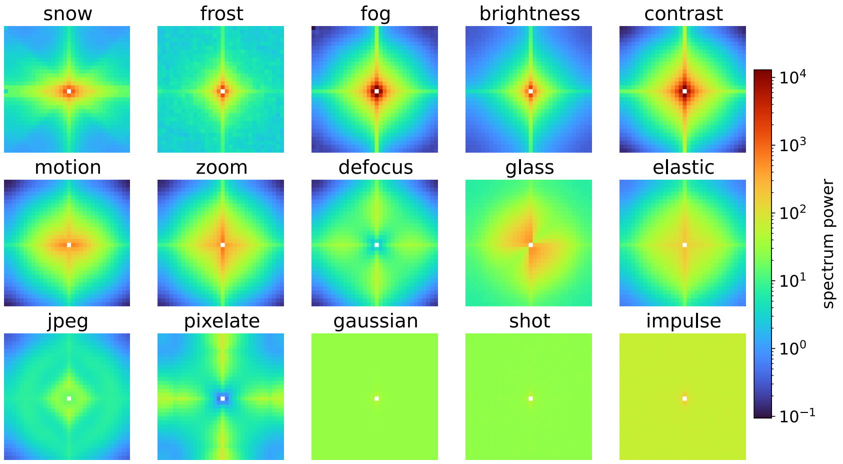

Figure 7. Corruption spectrum for the CIFAR10-C dataset. Fourier power spectra are plotted for all different common corruptions. Color maps are shared across panels.

Hybrid image experiment for monkey regularized model

Frequency bias is compared between a monkey-response-regularized VGG model and a baseline model through the experiment using hybrid images. Though a weaker effect compared with the mouse-regularization result, we found the reversing frequency for 'neural' model is smaller than that of 'base' model, suggesting a low frequency bias induced by neural regularization.

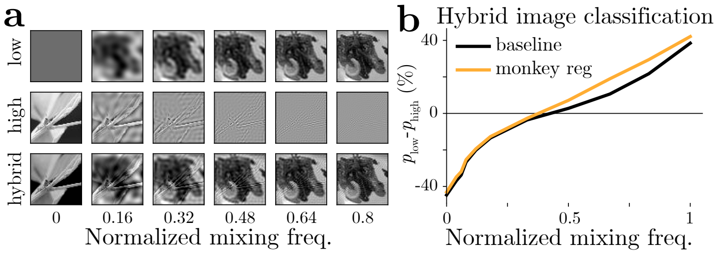

Figure 8. Probing frequency sensitivity of the monkey regularized model using hybrid images. Results are presented similar to Fig. 3.

Details of robust models

Details of models trained for CIFAR10, including one baseline model, six models trained for adversarial robustness, two models trained for common corruption robustness and two models using preprocessing are shown in Tab. 2.

Type	Model name	Architecture	Test accuracy on CIFAR10
baseline	Baseline[24]	WideResNet-28-10	94.78%
adversarial	Rebuff21[29]	WideResNet-70-16	92.23%
	Gowal20[30]	WideResNet-70-16	91.10%
	Wu20[31]	WideResNet-28-10	88.25%
	Zhang20[32]	WideResNet-28-10	89.36%
	Carmon19[33]	WideResNet-28-10	89.69%
	Sehwag20[34]	WideResNet-28-10	88.98%
	Cui20[35]	WideResNet-34-20	88.70%
corruption	Hendrycks20[4]	ResNeXt29-32x4d	95.83%
	Kireev21[36]	PreActResNet-18	94.77%
preprocess	Blur	ResNet-18	90.66%
	PCA	ResNet-18	89.85%

Table 2. Models trained for CIFAR10. One baseline model, 7 models trained for adversarial robustness, 2 models trained for common corruption robustness, and 2 models with simple preprocessing are compared in this study.

Details of models trained for ImageNet, including one baseline model, two models trained for adversarial robustness and six models trained for common corruption robustness are shown in Tab. 3.

Type	Model name	Architecture	Top1 test accuracy on ImageNet
baseline	Baseline[37]	ResNet-50	76.13%
adversarial	L_∞ ($\varepsilon = 4/255$)[37]	ResNet-50	62.42%
	L_2 ($\varepsilon = 3$)[37]	ResNet-50	57.90%
corruption	ANT[11]	ResNet-50	76.07%
	SIN[12]	ResNet-50	74.59%
	AugMix[4]	ResNet-50	77.54%
	DeepAugment[38]	ResNet-50	74.59%
	DeepAug+AugMix[38]	ResNet-50	75.82%
	Assemble[39]	Assemble-ResNet-50	80.81%

Table 3. Models trained for ImageNet. One baseline model, two models trained for adversarial robustness and six models trained for common corruption robustness are compared.

Analysis on neural similarity matrix

Previous work[8] demonstrated that models regularized with neural similarity matrix are more robust against multiple types of pixel noise as well as adversarial attacks. To understand why this type of neural regularization works, we analyzed the neural similarity matrix that characterizes the geometry of mouse V1 representation. We obtain neural responses of natural images through a well trained predictive model[40], and denote the population response to image i as r_i. The dimension of vector r_i is the number of neurons. Neural similarity matrix S^{neural} is defined as the cosine similarity of mean-corrected responses $r_1, \dots, r_N$ for N images,

$$S_{ij}^{\text{neural}} = \frac{\tilde{r}_i \cdot \tilde{r}_j}{\|\tilde{r}_i\| \|\tilde{r}_j\|}, \tag{4}$$

in which $\tilde{r}_i = r_i - \bar{r}$ is the population response to image i subtracted by mean response.

A first thing to notice is that the neural similarity matrix is low rank. For example, the one shown in Fig. 9 is a 5000 × 5000 matrix from 5000 images, but a rank-204 approximation can explain more than 90% of its variance. To account for 99% of the variance, a matrix of rank 1452 is sufficient. The result is not due to a small number of neurons. In fact, the neural response vector r_i used in this example is a union over 8 different scans, containing more than 40,000 recorded units. The low rank nature of S^{neural} shows that the vision system is encoding a small number of features through a highly correlated neuron population.

The next question is, how do these neural features look? Performing eigenvalue decomposition on S^{neural}, we can calculate its eigenvalues $\lambda_1, \lambda_2, \dots, \lambda_N$ ($\lambda_1 > \lambda_2 > \dots > \lambda_N$) and the corresponding eigenvectors $v_1, v_2, \dots, v_N$ ($\|v_i\| = 1$). The i-th neural feature is defined as $f_i = \sqrt{\lambda_i} v_i$. The rank-204 approximation in Fig. 9 is generated using the first 204 neural features, *i.e.* $\hat{S} = \sum_{i=1}^{204} f_i f_i^{\mathsf{T}}$.

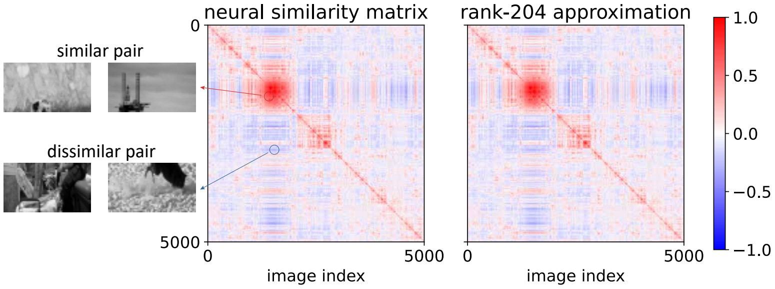

Figure 9. The neural similarity matrix and its low rank approximation. Neural responses of 5000 grayscale images are provided by a well trained brain model[40]. The cosine similarities between all pairs of responses are then calculated after subtracting the mean responses. An eigenvalue decomposition of the matrix shows that the first 204 principal components account for more than 90% of the variance. Therefore, a low rank approximation can be constructed based on these components.

Each neural feature f_i is a vector of the same length as the number of images, and can be treated as a scalar function of images. The first order approximation of f_i is a linear model with respect to the pixel values as input. The linear weight can be easily calculated by solving the regression problem,

$$w_i = \underset{w_i}{\operatorname{argmin}} \left(\left\| f_i - w_i^\mathsf{T} X \right\|^2 + \alpha \|w_i\|^2 \right). \tag{5}$$

Each column of X is a flattened image, and the dimension of w_i vector is the number of pixels. $\alpha\|w_i\|^2$ is a regularization term. The first 16 linear weights w_i are visualized as as spatial maps in Fig. 10.

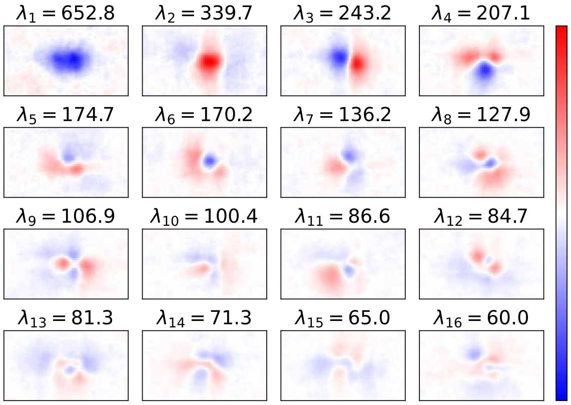

Figure 10. Linear approximation of neural features. Neural features are the eigenvectors v_i of the neural similarity matrix properly scaled by corresponding eigenvalues λ_i. Each neural feature is approximated by a linear function on image pixel values, and the linear weight w_i is displayed as a spatial map.

We further analyzed two properties of the linear approximation of neural features. Treating w_is as spatial maps, we can calculate its dominant spatial frequency via Fourier analysis. The results show that the dominant Fourier component of w_i associated with strong neural features are relatively low frequency (Fig. 11). Though w_i show certain spatial structure (Fig. 10), neural features f_i are nonlinear in general. We quantified the linearity of f_i by how good the linear approximation is, and found that correlation coefficient between f_i and the best linear prediction is high only for the neural features with high eigenvalues (Fig. 11).

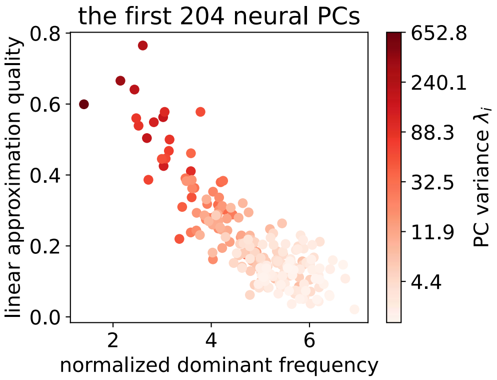

Figure 11. Overview of neural features. Properties of the first 204 neural features are visualized with colors indicating the eigenvalue corresponding to each. Each neural feature is approximated by a linear model. The ordinate is the correlation coefficient of linear approximation and the neural features on a hold-out set of images, characterizing how linear the feature is. The abscissa is the dominant frequency of the linear weights when viewed as spatial maps (Fig. 10). The results show that neural features, with high eigenvalues, are more linear, and contains lower spatial frequencies.