Exploring Frequency Domain Interpretation of Convolutional Neural Networks

Zhongfan Jia[1] Chenglong Bao[2*] Kaisheng Ma[1*]
[1]Institute for Interdisciplinary Information Sciences, Tsinghua University
[2]Yau Mathematical Sciences Center, Tsinghua University
lgg@pku.edu.cn, {clbao, kaisheng}@mail.tsinghua.edu.cn

Abstract

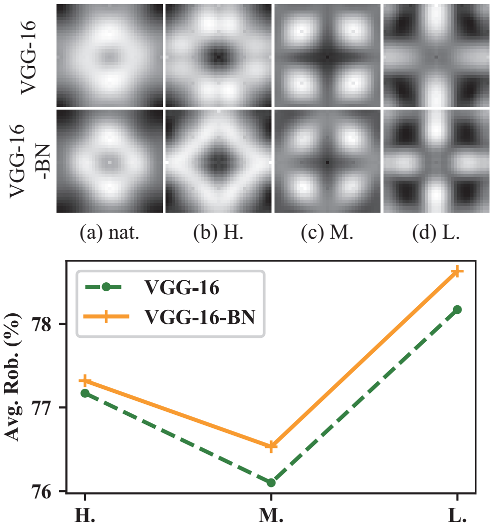
Many existing interpretation methods of convolutional neural networks (CNNs) mainly analyze in spatial domain, yet model interpretability in frequency domain has been rarely studied. To the best of our knowledge, there is no study on the interpretation of modern CNNs from the perspective of the frequency proportion of filters. In this work, we analyze the frequency properties of filters in the first layer as it is the entrance of information and relatively more convenient for analysis. By controlling the proportion of different frequency filters in the training stage, the network classification accuracy and model robustness is evaluated and our results reveal that it has a great impact on the robustness to common corruptions. Moreover, a learnable modulation of frequency proportion with perturbation in power spectrum is proposed from the perspective of frequency domain. Experiments on CIFAR-10-C show 10.97% average robustness gains for ResNet-18 with negligible natural accuracy degradation.

1. Introduction

Successful Convolutional Neural Network (CNN) architectures, such as VGG [24], ResNet [8] and DenseNet [12], can reach sub-human or super-human accuracy in image classification tasks. However, the robustness and interpretability of these models are obstacles to the reality.

One of the problems of CNNs is robustness, which can be defined as the performance of models on altered but similar-semantic datasets to the training set. Hendrycks and Dietterich [9] propose corrupted datasets (*e.g.* CIFAR-10-C), where "common" corruption exists in images (*e.g.* brightness changing or blurring). When testing on these datasets, many models experience severe performance degradation which implies the generalization gap to reality.

Besides robustness, the interpretability of networks remains a mystery. Many existing interpretation methods are closely related to visualization techniques [18], thus it is naturally to make interpretation analysis in spatial domain. However, frequency domain is an alternative perspective to spatial domain, and the explanation of CNNs in frequency domain is rarely studied. Meanwhile, convolution kernel is one form of Finite Impulse Response (FIR) filters, so as the 2D kernels used in CNNs. These filters have distinct frequency response characteristics and can be used to generate



(a) nat. (b) H. (c) M. (d) L.

(e) Average Robustness on CIFAR-10-C

Figure 1. We show the proportion of different frequency filters in the first layer can have an impact on the robustness. All results are averaged over 4 runs. **Frequency response histogram** (a-d). We plot the frequency response histogram of the first layer in VGG-16 (top) and VGG-16-BN (bottom) on CIFAR-10. In histograms, the lowest frequency is in the center and brightness of pixels relates to the response strength. (a) Naturally trained model (nat.). (b-d) Models trained by our *Attended Power Suppression* (APD) that have a "high-" (H.), "medium-" (M.) and "low-frequency dominated" (L.) first layer, where the main components are high-, medium- and low-frequency-response filters. BN: Batch-Normalization [13]. **Robustness** (e). Average accuracy on CIFAR-10-C [9] is tested to measure the robustness of models in (b-d), where exists about 2.1% undulation to verify the influence of frequency proportion. Rob.: Robustness.

[*] Corresponding Authors.

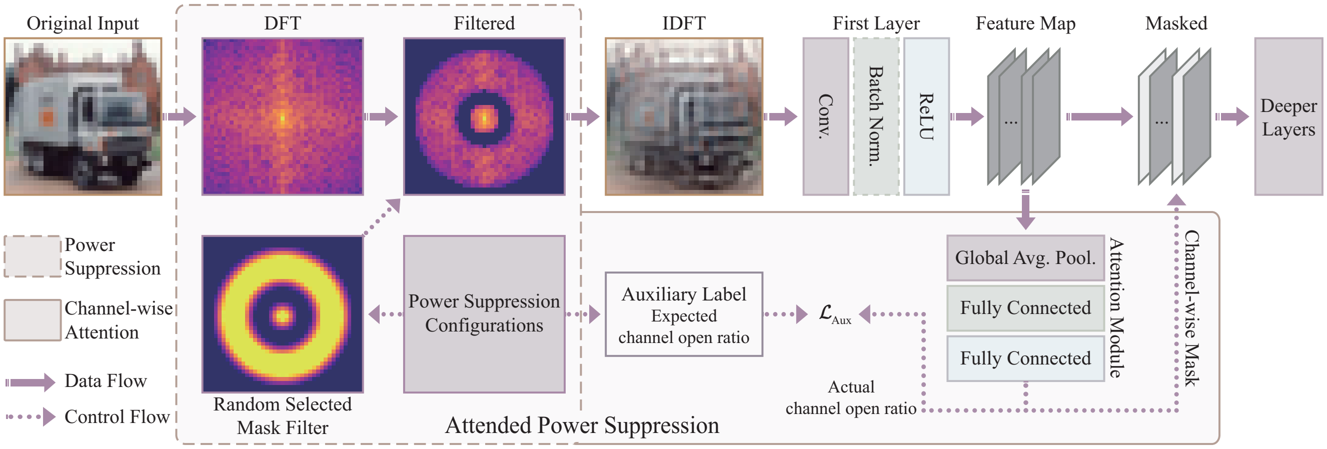


Figure 2. Framework of *Attended Power Suppression* (APS), where *Power Suppression* is only used in training with a certain probability to suppress some frequency components, and channel-wise attention is the part of inference in both training and testing. We hide the situation that input images are not power suppressed for clarity. Channel-wise Attention can constrain the number of opened channels in the first layer guided by the auxiliary label, which is given by power suppression and is corresponding to the mask filter. In this way, we can limit the proportion of first layer filters which are in the unmasked frequencies. If the image is not power suppressed, no auxiliary label is generated. DFT: Discrete Fourier Transform; IDFT: Inverse DFT; $\mathcal{L}_{\text{Aux}}$: Additional auxiliary loss term to the classification loss.

a frequency response histogram of the layer, where the frequency proportion can be reflected. Thus, this frequency proportion is a basic property and may exert the influence on accuracy and robustness to common corruptions, which motivates our research in this paper.

In many CNN architectures, the first convolution layer is the only direct information source of images that can extract and refine useful features for deeper layers. From another point of view, it filtrates different frequency components of input images. We mainly aim at exploring frequency proportion interpretability of the first convolution layer because visualization in image space is more intelligible than in feature space, and it is convenient to analyze the first layer instead of deeper layers. To show the proportion of different frequency filters in the first layer can do have an effect, we train models with a "high-", "medium-" and "low-frequency dominated" first layer, where high-, medium- and low-frequency-response filters occupy the mainstream. The accuracy and robustness of these models to common corruptions are shown in Figure 1 (e), where exists a range of about 2.1% average robustness undulation. To control the frequency proportion of the first layer, we propose *Attended Power Suppression* (APS) which will be explained below.

Figure 2 shows the framework of APS which combines self-supervision and channel-wise attention [2, 11, 27]. Briefly, we suppress some frequency components in the input image (*i.e. power suppression*). The channel-wise attention module after the initial layer can use a soft 0-1 mask to open or close output channels. The information of suppressed frequencies is recorded and used to generate auxiliary labels which are the expected ratio of open channels in the mask. Thus, the number of filters that have response to the unsuppressed frequencies can be limited. By changing suppression configurations, this limitation can be further modulated. Different configurations on VGG-16 and VGG-16-BN (Batch Normalization [13]) is tested, and we find that the proportion of different frequency filters in the first layer has little influence on accuracy on CIFAR-10 [14], but a relatively large impact on robustness to common corruptions on CIFAR-10-C. We further design *Noise in Frequency Domain* (NFD) which adds random noise on the power spectrum of images to improve robustness. On ResNet-18, we combine APS with NFD to achieve about 10.97% average robustness gains on CIFAR-10-C while only dropping 0.22% natural accuracy on CIFAR-10. To the best of our knowledge, it should be noticed that NFD and power suppression in our APS is novel data augmentation methods working in frequency domain which is rarely studied before.

Our main contributions can be summarized as:

- We prove that altering the frequency proportion of filters in the first layer exerts little influence on classification accuracy but much impact on robustness to common corruptions.
- We propose *Attended Power Suppression* (APS) which combines self-supervision and channel-wise attention mechanism to control the proportion of different frequency filters in the first layer and thus enhance frequency domain interpretability.
- Our *Power Suppression* in APS and *Noise in Frequency Domain* (NFD) directly apply data augmen-

tation in frequency domain, which is rarely studied among augmentation methods. We further extend APS with NFD to achieve 10.97% average robustness gains with only 0.22% natural accuracy drops.

2. Related work

Interpretability. Visualization methods and interpretability researches are strongly interacted [18]. Zeiler and Fergus [30] visualize the highest response of each filter with corresponding inputs, while Yosinski *et al.* [29] maximize the activation of filters to visualize their preferred patterns. Mopuri *et al.* [19] and Selvaraju [21] both aim at identifying interested regions in the input image of the model for prediction. These methods all interpret CNNs in spatial domain, in the meanwhile, interpretability of CNNs in frequency domain has been rarely studied while our method has revealed the importance of it.

Data augmentation. Data augmentation is commonly used from AlexNet architecture [15]. Various methods are proposed [23], *e.g.* Patch Gaussian [16], GAN data augmentation [1, 6] and more generally, adversarial training [17, 22]. Cubuk *et al.* [3] propose AutoAugment using reinforcement learning to search augmentation strategies, and Geirhos *et al.* [7] use style transfer to randomize texture information in images from ImageNet [4]. They can improve both accuracy and robustness. However, our *NFD* and *power suppression* works in frequency domain which is relatively unnoticed in data augmentation, and power suppression is designed for interpretability, not accuracy nor robustness.

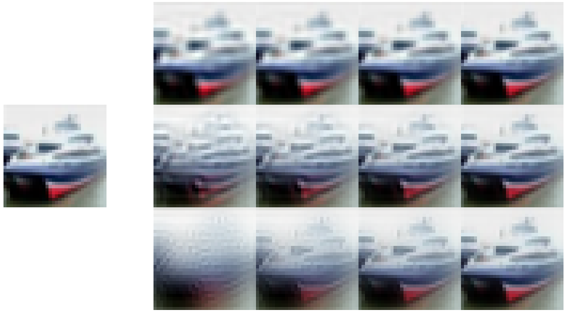
Self-supervised learning. Auxiliary tasks are used in self-supervised learning where labels are pseudo ones generated from data. Self-supervised learning has been used in many applications [5, 26, 31] to improve accuracy, and Hendrycks *et al.* [10] find self-supervised learning can also improve model robustness. In our APS, attention module is trained in a self-supervised framework, but it aims to enrich frequency interpretability which is different from these works.

3. Methods

3.1. Notations

We denote one image by $\mathbf{I} \in [0,1]^{d_1 \times d_2}$. 2D Discrete Fourier Transform (DFT) is represented by the denotation $\mathcal{F} : \mathbb{R}^{d_1 \times d_2} \to \mathbb{C}^{d_1 \times d_2}$ and reverse DFT by $\mathcal{F}^{-1}$. We use index $[i,j]$ to denote matrix elements, where $i, j \in \mathbb{N}$. If the position of index $[0,0]$ in matrix $\mathbf{M}$ is given, i, j can be negative, thus for all elements in $\mathbf{M}$ we denote the set of their indices by $\langle \mathbf{M} \rangle$.

We let the zero frequency component in $\mathcal{F}(\mathbf{I})$ lay at the zero index $[0,0]$. For frequency component $\mathcal{F}(\mathbf{I})[i,j]$, we denote the distance of this frequency to zero by $r(i,j) = \|(i,j)\|_2$. We define r_I to be the largest distance in $\langle \mathcal{F}(\mathbf{I}) \rangle$, and $r_I \leq \left\| \left(\left\lfloor \frac{d_1}{2} \right\rfloor, \left\lfloor \frac{d_2}{2} \right\rfloor \right) \right\|_2$.


(a) Original

(b) Suppressed

Figure 3. Power suppression examples in CIFAR-10. (a) Image $\mathbf{I}$ from CIFAR-10 test set. (b) Suppression results $\boldsymbol{D}(\mathbf{I}, \mathbf{M}, \alpha)$, where mask filter $\mathbf{M} = \mathbf{M}_{\mathrm{L}}$, $\mathbf{M}_{\mathrm{M}}$, $\mathbf{M}_{\mathrm{H}}$ in each row, and suppression ratio $\alpha = 0$, $1/4$, $1/2$, $3/4$ in each column. Results of each row show the low-, band- and high-pass characteristics of $\mathbf{M}_{\mathrm{L}}$, $\mathbf{M}_{\mathrm{M}}$ and $\mathbf{M}_{\mathrm{H}}$.

For a function like $m : [0, r_I] \to [0,1]$, we name it as "mask curve", and define the corresponding mask matrix $\mathbf{M}_m \in [0,1]^{d_1 \times d_2}$ by:

$$\mathbf{M}_m[i,j] = m\big(r(i,j)\big), \forall (i,j) \in \langle \mathbf{M} \rangle. \tag{1}$$

Given a non-zero complex number $c \in \mathbb{C}$, we use the notation $\operatorname{Dir} c$ to represent the normalized c, *i.e.* $\operatorname{Dir} c = \frac{c}{|c|}$. $\operatorname{Dir} 0$ can be arbitrary complex number which modulus is 1.

3.2. Attended Power Suppression (APS)

Method overview. The goal of APS is to control the proportion of different frequency filters in the first layer. To achieve it, *power suppression* and *Channel-wise Attention* is used in APS. For channel-wise attention, it is activated both in training and testing; for power suppression, however, it is disabled in testing, and may be disabled with a certain probability in training, thus normal images and power-suppressed images are mixed up in the same mini-batch.

Some frequency components will be randomly suppressed in power suppression. We assume that only low-frequency information of the input image is left. An auxiliary label will be generated which is a real number in $[0,1]$, *e.g.* 0.2. This label is the expected open ratio of channel-wise attention module, *i.e.* about 20% of output channels in the first layer are expected to be opened. A loss term is added to constrain the actual channel open ratio closing to 0.2.

Since only low-frequency information remains, the first layer filters that mainly in low frequencies can have response. To improve accuracy, attention module has to choose output channels that are in low-frequencies (and have response) to open, and only these selected filters can enforce their low-frequency properties. The low-frequency filters without this property strengthening can be more easily moved to other frequency bands in the following training process. Thus, APS applies loose guidance to the frequency proportion of the first layer filters.

Power suppression. Explanation of how the power suppression works will be presented in this part. We first define three mask curves "L", "M" and "H" as m_{L}, m_{M} and m_{H}:

$$m_{\text{step}}(s) = \begin{cases} 1, & \text{if } s \leq 0, \\ 1 - \frac{s}{3}, & \text{if } 0 < s \leq 3, \\ 0, & \text{else;} \end{cases} \tag{2}$$

$$m_{\text{hill}}(s) = \begin{cases} \frac{s}{3}, & \text{if } 0 < s \leq 3, \\ 1, & \text{if } 3 < s \leq 8, \\ 1 - \frac{s-8}{3}, & \text{if } 8 < s \leq 11, \\ 0, & \text{else;} \end{cases} \tag{3}$$

$$\begin{aligned} m_{\mathrm{L}}(s) &= m_{\text{step}}(s-1) + m_{\text{hill}}(s-1), \\ m_{\mathrm{M}}(s) &= m_{\text{step}}(s-1) + m_{\text{hill}}(s-7), \\ m_{\mathrm{H}}(s) &= m_{\text{step}}(s-1) + m_{\text{step}}(s-13). \end{aligned} \tag{4}$$

The corresponding mask matrices of above curves defined by Equation (1) are $\mathbf{M}_{\mathrm{L}}$, $\mathbf{M}_{\mathrm{M}}$ and $\mathbf{M}_{\mathrm{H}}$. These matrices are roughly a low-pass, band-pass and high-pass filter in frequency domain. Extremely low-frequency contents are kept by the $m_{\text{step}}(s-1)$ terms in Equation (4) because these components always have very high power and can greatly change the average value of the whole image. Medium- and high-frequency components are relatively low-powered, so applying power suppression to them will not alter the first moment of image pixels very much.

For an original image $\mathbf{I}$, the power suppression procedure $\boldsymbol{D}$ can be written in:

$$\boldsymbol{D}(\mathbf{I}, \mathbf{M}, \alpha) = \mathcal{F}^{-1}\Big(\mathcal{F}(\mathbf{I}) * \big(1 - \alpha(1 - \mathbf{M})\big)\Big), \tag{5}$$

where "$*$" is element-wise multiplication, $\alpha \in [0, 1]$ is the suppression ratio (*i.e.* degree of suppression), and the mask filter $\mathbf{M} \in \{\mathbf{M}_{\mathrm{L}}, \mathbf{M}_{\mathrm{M}}, \mathbf{M}_{\mathrm{H}}\}$. In this procedure, the more α is, the more power of suppressed frequencies are saved. Figure 3 is an illustration of images after power suppression.

Constraint configurations. In this part, we explain the generation procedures of auxiliary label β from suppression information $\langle \mathbf{M}, \alpha \rangle$. These procedures define the relationship between suppression information and expected open ratio of channel-wise attention.

(a) **Suppressed Configuration.** In this configuration, for a suppression result $\boldsymbol{D}(\mathbf{I}, \mathbf{M}, \alpha)$, the auxiliary label function β_0 is defined as:

$$\beta_0(\mathbf{M}, \alpha) := \beta_0(\alpha) = \sigma(7\alpha - 3) + 1 - \sigma(4), \tag{6}$$

where $\sigma(\cdot)$ is sigmoid function. The function $\beta_0(\alpha)$ is monotone and $\beta_0(1) = 1$.

(b) **Fully-Open Configuration.** In this configuration, for a suppression result $\boldsymbol{D}(\mathbf{I}, \mathbf{M}, \alpha)$, the auxiliary label function β_1 is always 1, thus:

$$\beta_1(\mathbf{M}, \alpha) = 1. \tag{7}$$

(c) **Learnable Configuration.** In this configuration, for a suppression result $\boldsymbol{D}(\mathbf{I}, \mathbf{M}, \alpha)$, the auxiliary label function β_{L} is defined as:

$$\beta_{\mathrm{L}}(\mathbf{M}, \alpha) = (1 - \alpha) \cdot b(\mathbf{M}) + \alpha, \tag{8}$$

where $b(\mathbf{M})$ is the basic open ratio of mask filter $\mathbf{M}$. In more detail, a parameter $\boldsymbol{\theta}_b = (\theta_{b,\mathrm{L}}, \theta_{b,\mathrm{M}}, \theta_{b,\mathrm{H}})$ is used to determine basic open ratios following this procedure:

$$\big(b(\mathbf{M}_{\mathrm{L}}), b(\mathbf{M}_{\mathrm{M}}), b(\mathbf{M}_{\mathrm{H}})\big) = 1.1 \cdot \frac{\boldsymbol{\sigma}(\boldsymbol{\theta}_b)}{\sum_* \boldsymbol{\sigma}(\boldsymbol{\theta}_b)}, \tag{9}$$

and the denotation $\sum_* \boldsymbol{\sigma}(\boldsymbol{\theta}_b)$ means the summation of elements in vector $\boldsymbol{\sigma}(\boldsymbol{\theta}_b)$.

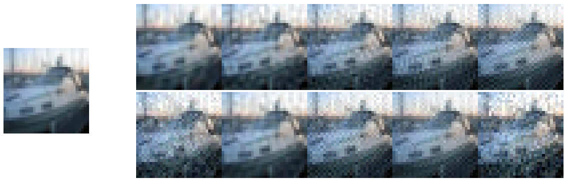
For (a) and (b), we can apply them to mask matrices $\mathbf{M}_{\mathrm{L}}, \mathbf{M}_{\mathrm{M}}, \mathbf{M}_{\mathrm{H}}$ with combinations. For example, We denote that (a) is applied to $\mathbf{M}_{\mathrm{L}}$ (low frequency), and (b) to the rests, by "LMH=011", which is subscripts of constraint functions in order. Therefore, we can get eight combinations in total and denote them by "LMH=111, 011, 101, 110, 001, 010, 100, 000". It should be noticed that though Fully-Open Configuration release the proportion of target frequency filters, it also forces the filters to learn power-invariance of constrained frequencies.

Channel-wise attention and self-supervision. In this part, the details of the attention module will be explained. The channel-wise attention module in our method consists of one global average pooling layer and two subsequent fully-connected layers. The input is the feature map $\mathbf{m} \in \mathbb{R}^{c \times d_1 \times d_2}$ of initial "semantic" layer which typically includes one convolution layer and optional batch-normalization. The output is a soft mask vector $\mathbf{v} \in (0, 1)^c$, and to apply it to the feature map $\mathbf{m}$, the i-th element of mask vector $v[i]$ will multiply to the i-th channel of feature map $\mathbf{m}[i]$. We denote the average value of $\mathbf{v}$ by $\bar{\mathbf{v}}$, which is the actual channel-open ratio of the attention module.

If the input has auxiliary label β which is the expected open ratio of channels, the L_2 loss can be added to the classification loss, making $\mathbf{v}$ meet the expected open ratio after training. The added loss term is

$$\mathcal{L}_{\text{Aux}}(\mathbf{v}, \beta) = \lambda \|\bar{\mathbf{v}} - \beta\|_2^2, \tag{10}$$

where λ is the coefficient. In self-supervised learning, auxiliary tasks are trained by generated pseudo labels, while β



(a) Original (b) Noised

Figure 4. NFD examples in CIFAR-10. (a) Image $\mathbf{I}$ from CIFAR-10 test set. (b) First row: NFD results $\boldsymbol{N}_\sigma(\mathbf{I},\mathbf{M})$, where $\mathbf{M} = \mathbf{M}_0$ to $\mathbf{M}_4$ (*i.e.* low frequency to high frequency noises, Section 3.3), . Second row: NFD results with random $\mathbf{M}$ (Section 3.3). The strength σ in a) and b) is 3 to enhance visual effects.

is generated from input data $\boldsymbol{D}(\mathbf{I},\mathbf{M},\alpha)$ to train the attention module. Therefore, this additional optimization target $\mathcal{L}_{\text{Aux}}$ obeys self-supervised learning framework.

3.3. Noise in Frequency Domain (NFD)

Methods. The details of applying random noise in frequency domain will be described in this section. It should be mentioned that NFD is orthometric to APS, where APS alters the power spectrum but keeps relative strength in altered part, while NFD perturbs in a zero noise expectation.

We denote a noise matrix by $\mathbf{R}_\sigma(\mathbf{M})$, where σ is the maximum variance of noise, and $\mathbf{M} \in [0,1]^{d_1 \times d_2}$ is a control matrix. In more detail, we have

$$\mathbf{R}_\sigma(\mathbf{M})[i,j] \sim \mathcal{N}\big(0, \sigma \cdot (1 - \mathbf{M}[i,j])\big), \tag{11}$$

where zero in $\mathbf{M}$ means the largest variance of noise.

Given image $\mathbf{I}$ and control matrix $\mathbf{M}$, the power perturbation result $\boldsymbol{N}_\sigma(\mathbf{I},\mathbf{M})$ is defined by:

$$\boldsymbol{N}_\sigma(\mathbf{I},\mathbf{M}) = \mathcal{F}^{-1}\big(\mathcal{F}(\mathbf{I}) + \mathbf{R}_\sigma(\mathbf{M}) * \operatorname{Dir}\mathcal{F}(\mathbf{I})\big), \tag{12}$$

where $*$ is element-wise multiplication, and Dir is defined in Section 3.1. This perturbation procedure will not change the semantic of original images significantly, since the unaltered phase spectrum keeps the semantic contents rather than power spectrum [20].

Control matrix and frequency bands of NFD. Based on some ordinary assumptions, we define the concrete range of each frequency bands, which are used to generate the control matrix in Section 3.3. Some images after NFD are shown in Figure 4.

We firstly define a "hill" mask curve by

$$m_{a,b}(s) = \begin{cases} s - a + 1, & \text{if } a - 1 \le s < a, \\ 1, & \text{if } a \le s < b, \\ b - s + 1, & \text{if } b \le s < b + 1, \\ 0, & \text{else.} \end{cases} \tag{13}$$

For image $\mathbf{I}$ in CIFAR-10 with 4 pixels padding on each edge, the image size is 40×40, so $\langle \mathcal{F}(\mathbf{I}) \rangle = \{(i,j) \mid i,j \in [-20,20] \cap \mathbb{Z}\}$, and $\lceil r_I \rceil = \lceil 20\sqrt{2} \rceil = 29$. We define five instances of "hill" mask: $m_0 := m_{0,9}$, $m_1 := m_{10,13}$, $m_2 := m_{14,17}$, $m_3 := m_{18,20}$ and $m_4 := m_{21,29}$; the corresponding mask matrices are $\mathbf{M}_0$ to $\mathbf{M}_4$. To create the control matrix $\mathbf{M}$, an array $\boldsymbol{\alpha}_{\text{r}} \in [0,1]^5$ is generated, where at least one element is randomly sampled from uniform distribution $\mathcal{U}(0,1)$, and the rest elements have equal chance to be 0 or 1. So the control matrix is

$$\mathbf{M} = \sum_{0 \le i \le 4} \boldsymbol{\alpha}_{\text{r}}[i] \cdot \mathbf{M}_{\mathbf{i}}. \tag{14}$$

Some background is needed to explain the parameter of m_0 to m_4. It's known in previous researches a "$1/f$" principle, which means the average amplitude spectrum of natural images falls with a form $1/f$, or $1/f^2$ for average power spectrum [25]. Thus for a frequency component $\mathcal{F}(\mathbf{I})[i,j]$ whose distance to zero $s = r(i,j)$, the amplitude of this frequency, $f(s)$, can be formulated by

$$f(s) = \frac{\alpha}{1+s}, \tag{15}$$

where α is a constant. $V(s)$, the number of minimum bits to represent $\mathcal{F}(\mathbf{I})[i,j]$, can be approximately expressed as

$$V(s) \simeq \log_2 \beta\pi f^2(s) = \log_2 \beta\pi \left(\frac{\alpha}{1+s}\right)^2. \tag{16}$$

If we apply an ideal low-pass filter whose distance of cut-off frequency is s, the proportion of information retained is

$$H(s) = \frac{\sum_{\substack{(i,j)\in\langle\mathcal{F}(\mathbf{I})\rangle \\ r(i,j)\le s}} V\big(r(i,j)\big)}{\sum_{(i,j)\in\langle\mathcal{F}(\mathbf{I})\rangle} V\big(r(i,j)\big)}. \tag{17}$$

This curve is not smooth, and we can sample this curve using natural numbers from zero to $\lceil r_I \rceil$ to get a sequence $\mathbf{H}_s$. Its first order difference is denoted by $\mathbf{H}'_s$, and the element $\mathbf{H}'_s[i]$ corresponds to the proportion of information represented by frequency components of which distance is i. In our method, we set $\alpha = 1600$ and $\beta = 1$ in function V.

Finally, We define the coverage degree C_v of a mask curve m by the following equation:

$$C_v(m) = \sum_{0 \le i \le \lceil r_I \rceil} m(i) \cdot \mathbf{H}'_s[i]. \tag{18}$$

In curve m_0 to m_4 defined above, all the coverage degree is around $1/5$ to keep their equality of information contents.

Attention module and constraint configuration. If APS is used with NFD, it should be mentioned that an extended attention module is used in APS since NFD adds more input disturbances. In more detail, a convolution and batch-normalization layer is added before global average pooling (Section 3.2). Learnable Configuration in Section 3.2 (c) can also be used to find the optimal proportion of filters in each frequency band.

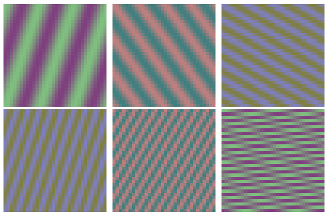


Figure 5. Channel Point Detection Map (cPDM) examples. The strength t in these cPDMs are $4\sqrt{2}$. Best view in color.

3.4. Point Detection Map (PDM)

Since the first "semantic" layer may contain batch normalization and other components, it's hard to directly apply Fourier analysis. By measuring the average response to "Point Detection Map" (PDM) of different frequencies, we can infer the frequency response of this system.

The construction of PDM follows similar ideas to NFD which applies perturbations to power spectrum. Briefly, we denote the image whose pixel is 0.5 by $\mathbf{I}_{0.5}$. A point perturbation $\boldsymbol{P}_{\mathrm{p}}(t,i,j)$ of strength t at frequency index $[i,j]$ is an image that $\|\operatorname{vectorize}\boldsymbol{P}_{\mathrm{p}}(t,i,j)\|_2 = t$, and $\mathcal{F}\big(\boldsymbol{P}_{\mathrm{p}}(t,i,j)\big)$ only has non-zero elements at index $[i,j]$ (and its centrosymmetric index $[-i,-j]$). PDM is then defined as $\boldsymbol{P}(t,i,j) = \mathbf{I}_{0.5} + \boldsymbol{P}_{\mathrm{p}}(t,i,j)$. For PDM of RGB images, only one channel c is PDM $\boldsymbol{P}(t,i,j)$, and other channels are $\mathbf{I}_{0.5}$. This is called as "channel PDM" (cPDM) $\boldsymbol{P}_c(t,i,j)$. Some cPDM examples are shown in Figure 5.

To measure the response at frequency index $[i,j]$ of one specific output channel in the first layer, we can feed $\boldsymbol{P}_c(t,i,j)$ as input, and the average value of this output channel is response strength. In this way, we can get the response of all $\langle c,i,j\rangle$ combinations which can form a frequency response map of this output channel. After normalized to $[0,1]$, response maps of each output channel can "vote" to get a response histogram.

It should be noted that this method is only useful for the first "semantic" layer because the subsequent convolution layer can't match any meaningful patterns from the feature map of the first layer if PDMs are fed into the network.

4. Results

This section can be divided into two parts. First, APS is used to reveal the frequency interpretability of robustness. Second, final robustness gains with APS combined with NFD will be demonstrated.

4.1. Results under different constraints

Experiment setting. We test on CIFAR-10 and CIFAR-10-C using VGG-16 [24] and VGG-16-BN (Batch Normalization [13]). We train in 90 epochs with initial learning rate at 0.01 and decrease it by 0.1 every 30 epochs. Optimizer SGD uses the same hyper-parameter as Simonyan and Zisserman [24] except that the batch-size is 128.

VGG-16	nat.	aug.	att.	aug.+att.
Accuracy	(90.41)	+0.23	+0.00	-0.04
Robust.	(72.80)	+5.68	+0.10	+4.97
mCE	100	79	100	82
VGG-16-BN	nat.	aug.	att.	aug.+att.
Accuracy	(91.62)	+0.32	+0.04	+0.09
Robust.	(72.07)	+5.12	+0.38	+4.29
mCE	100	80	98	83

Table 1. The reference accuracy (%) on CIFAR-10, reference average accuracy on corruptions (Robust., %) and mCE (mean Corruption Error [9]) on CIFAR-10-C, of models that are: (1) naturally trained (nat.); (2) power suppression augmented (aug.); (3) with attention module (att.); (4) combines 2) and 3) (aug.+att.). We test VGG-16 and VGG-16-BN architecture, where "BN" means additional batch-normalization [13] layer inserted. Naturally trained model is our baseline, so it's represented in absolute value for accuracy and robustness where all the rest models are relative to it. All results are averaged over 4 runs. The results show that both used methods and their combination have little affect on accuracy, while they can vary their robustness to common corruptions. Power suppression augmentation can also improve clean accuracy slightly and make robustness gains at the same time.

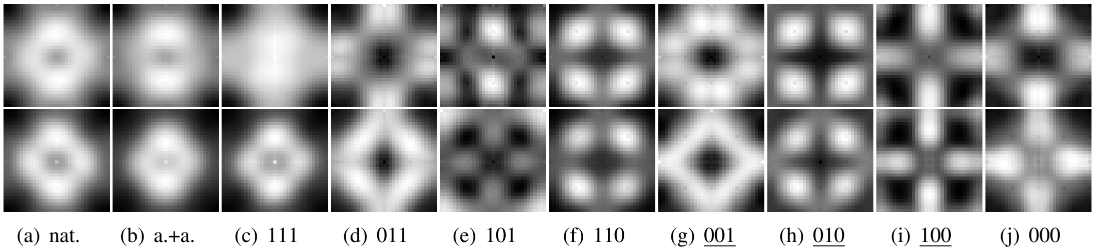
The probability is 1/3 of applying power suppression (Section 3.2) to input images when sampling training data, and mask filter is randomly selected in $\mathbf{M}_{\mathrm{L}}$, $\mathbf{M}_{\mathrm{M}}$ and $\mathbf{M}_{\mathrm{H}}$ when applying suppression. Random crop and horizontal flip are applied before images are fed into the network. We use attention module in Section 3.2 whose number of hidden neurons is 512. λ in Equation (10) is 100, which is the coefficient of additional loss term to cross-entropy loss. Only power suppressed inputs have auxiliary labels β, thus not all input samples have this additional loss term.

Reference accuracy and robustness. We show the accuracy on CIFAR-10, average accuracy on corruptions and mCE [9] on CIFAR-10-C, of VGG-16 and VGG-16-BN (Batch-Normalization) in Table 1. We test models that are: (1) naturally trained; (2) power suppression augmented; (3) only with attention in Section 3.2; (4) both augmented and with attention. For accuracy, we find that power augmented models are slightly better than baseline, and more generally, power suppression and adding attention module are both having little influence. However, power suppression can make much robustness gains, *i.e.* enlarge average accuracy on corruptions and decrease mCE (mean corruption error). The result implies models with similar accuracy can vary their robustness enormously, which agrees with the conclusion of Hendrycks and Dietterich's [9].

Impacts of frequency proportion. We then take different constraint configurations in Section 3.2 into account, which

	LMH	nat.	111	011	101	110	001	010	100	000	variance
VGG-16	Acc.	(90.41)	-0.90	-0.72	-0.70	-0.85	**-0.55**	-0.94	-1.17	-0.89	0.03
	Rob.	(72.80)	**+7.57**	+4.60	+5.45	+4.72	+4.37	+3.30	+5.37	+4.10	1.41
	mCE	100	**74**	83	80	83	84	88	81	85	-
VGG-16-BN	Acc.	(91.62)	**+0.05**	-0.79	-0.66	-1.11	-0.68	-0.91	-1.19	-0.88	0.13
	Rob.	(72.07)	+5.08	+4.83	+5.19	+6.31	+5.25	+4.46	**+6.56**	+5.79	0.46
	mCE	100	81	83	81	79	81	84	**78**	80	-

Table 2. Accuracy gain (Acc., %) on CIFAR-10, average accuracy gain on corruptions (Rob., %) and mCE (mean Corruption Error) on CIFAR-10-C, under different APS configurations. In the title row, for example, "011", where "0" is at the same index to "L" in "LMH", means the proportion of low-frequency-response filters ("L") is limited ("0" in "011") in the first layer, while medium- and high-frequency-response filters ("M" and "H") have no limitations ("1" in "011"). All results are averaged over 4 runs and relative to naturally trained models (nat.). Configuration of best performance for each network is marked in **bold**. Results of the variance of clean and corruption accuracy gain among configurations, imply that the proportion of filters in the first layer that belonging to low-, medium- and high-frequency, can't enormously influence the accuracy of models, but their impacts on robustness can be relatively large. It agree with the results Yin *et al.* [28] that model can achieve high accuracy regardless the frequency components it uses. The latter half of this conclusion about robustness can be more clearly observed when comparing the average robustness gain of "LMH=001, 010, 100" (underlined). A small number of clean accuracy degradation is normal since regularization coefficient λ is large (Section 3.2 and 4.1).



(a) nat. (b) a.+a. (c) 111 (d) 011 (e) 101 (f) 110 (g) 001 (h) 010 (i) 100 (j) 000

Figure 6. 2D-histogram of frequency response map of the first layers in naturally trained model (a, nat.), power suppression augmented model with attention (b, a.+a.), and models under different APS configurations (c-j, "LMH=xxx"). All the models are trained on CIFAR-10. For all histograms, the lowest frequency is in the center, and the brightness of each pixel corresponds to the occurrence of this frequency component among all filter response maps in the first layer. For the top row, response maps come from VGG-16 models, and the bottom row is VGG-16-BN (Batch Normalization). The details of the generation of these histograms are shown in Section 4.1. The underlined configurations ("LMH=001, 010, 001", Section 3.2) can result in "high-", "medium-" and "low-frequency-dominated" first layers (Section 4.1). The suppressed frequencies (*e.g.* medium and high frequencies in "LMH=100") in the corresponding response histograms show a very low occurrence frequency, which can verify that our APS method can actually control the proportion of different frequency filters in the first layer.

can control the frequency proportion more flexibly. We recall that the configuration "LMH=abc" mentioned in Section 3.2 means that auxiliary label generator β_a, β_b and β_c is used for mask filter $\mathbf{M}_\text{L}$, $\mathbf{M}_\text{M}$ and $\mathbf{M}_\text{H}$ respectively. The results of gains on accuracy and robustness are shown in Table 2. We can infer that the proportion itself doesn't exert a very strong influence on model accuracy, and similar results have been found by Yin *et al.* ([28], Figure 1). They use images that applied high-pass (or low-pass) filters to be the network input, thus the first layer is high-frequency dominated (or low-frequency, respectively). They find these two experiment settings can reach similar accuracy, which agrees with our observations.

However, this proportion can exert a relatively large impact on model robustness to common corruptions. In Table 2, the underlined items are the average robustness gain of configuration "LMH=001, 010, 100". These configurations all have limitations on two frequency bands, while the rest one does not. Thus, they can result in "high-", "medium-" and "low-frequency dominated" first layer after training, and lead to very different robustness performances. The undulation of underlined items shows clear evidence of our conclusion about frequency proportion and robustness.

Frequency response. To show that our method can constrain the proportion of different frequencies in the first layer, we can test its frequency response (Section 3.4). To generate the response histogram, the response of each output channel is normalized to $[0, 1]$ and only a small number of extremely large and small pixels are ignored in normalization to keep visual clarity. Due to the capacity limitation, the visualization results of response maps are plotted in supplementary materials. After normalization, each response

	nat.	sup.	NFD	both	both+att.
Acc.	(94.58)	-0.15	-0.29	-0.18	-0.25
Rob.	(72.62)	+4.73	+6.60	+10.16	+10.33
mCE	100	78	80	63	63

Table 3. The reference accuracy (Acc., %) on CIFAR-10, reference average accuracy on corruptions (Rob., %) and mCE (mean Corruption Error) on CIFAR-10-C, of ResNet-18 models that are: (1) naturally trained (nat.); (2) power suppression augmented (sup.); (3) NFD augmented (NFD); (4) combining 2) and 3) (both); (5) 4) with extended version of attention module added (Section 3.3). Naturally trained model is our baseline and its accuracy and robustness are shown in absolute value. All the rest models are relative to baseline, and all results are averaged over 4 runs. The results show that, NFD itself can also achieve high robustness gains, and power suppression augmentation is highly non-overlapping to NFD on robustness gains.

map can make a "vote" to the interested frequencies, and thus the histogram can display the frequency tendentiousness of the first layer.

The response histograms of some reference models and models under different constraint configurations of APS are shown in Figure 6, where the lowest frequency in histogram is placed in the center. For one of the underlined configurations "LMH=100", suppressed medium- and high-frequency components have very sparse occurrence in their per-channel response maps because of the low brightness of corresponding pixels in histograms (Figure 6 (i)). Other configurations also follow the same rule. This verifies that our APS method can actually constrain the proportion of different frequency filters in the first layer as expected.

4.2. Power suppression with NFD

The power suppression augmented model can achieve high robust gains to common corruption in Table 1, which implies other data augmentations in frequency domain may also improve robustness. Thus power suppression (Section 3.2) is combined with NFD to make further exploration.

Experiment setting. The results of ResNet-18 on CIFAR-10 and CIFAR-10-C is tested. The initial learning rate is 0.1, and we use the extended attention module in Section 3.3. For NFD, random method in Section 3.3 is used for control matrix, and the noise variance σ is 1. We apply NFD after power suppression if they're used as data augmentation in the same model. All the rest settings are the same as Section 4.1 if no extra annotations are provided.

Reference accuracy and robustness. For ResNet-18, we show the accuracy on CIFAR-10, average accuracy on corruptions and mCE on CIFAR-10-C in Table 3, where we test the models that are specially trained by: (1) power suppression; (2) NFD; (3) both of them; (4) further combined with extended version of attention module (Section 3.3) basing on 3). For robustness gains, the results show that power suppression and NFD take effects in orthometric ways, and the combination of them can obtain a stacked robustness promotion, which agrees with the analysis of their orthometric perturbation schemes in power spectrum.

	λ	100	50	10	2
Fixed	Acc.	-0.61	-0.61	-0.41	**-0.40**
	Rob.	**+11.20**	+11.13	+10.32	+10.34
	mCE	64	64	**63**	**63**
Learn.	Acc.	-0.31	**-0.22**	-0.24	-0.27
	Rob.	+10.78	**+10.97**	+10.35	+10.26
	mCE	63	**62**	**62**	63

Table 4. Accuracy gain (Acc., %) on CIFAR-10, average accuracy gain on corruptions (Rob., %) and mCE (mean Corruption Error) on CIFAR-10-C, under different λ in APS (Section 3.2). We test models of two constraint configurations (Section 3.2): (1) "LMH=000" (Fixed); (2) Learnable Configuration (Learn.). All results are averaged over 4 runs and all but mCE are relative to accuracy of naturally trained models in Table 3. Best configurations among λ are marked in **bold**. Result shows that Learnable Configuration can keep the balance between robustness gain and natural accuracy drop to reach preferable status.

4.3. Learned frequency configuration

Since power suppression and NFD can make further robustness gains, we apply the Learnable Configuration (Section 3.2) in APS and combine it with NFD. The result is shown in Table 4 and is compared to a fixed configuration ("LMH=000"). The results imply that Learnable Configuration can reach the balance point between robustness gain and natural accuracy loss, while fixed configuration has stronger constraints and it is more likely to improve the robustness and hurt clean accuracy. The Learned frequency proportion is placed in supplementary materials.

5. Conclusion

We aim at the frequency interpretability of CNNs, and in particular, the proportion of different frequency filters in the first layer is explored. Attended Power Suppression (APS) is proposed to reach this goal, and the results show that the influence of this proportion on clean accuracy is relatively small, but it can affect the robustness to common corruptions greatly. It's reasonable that an extremely tendentious first layer can result in high accuracy because of the optimization target, however, this performance can also be more volatile due to this tendentiousness. We further use learnable APS with Noise in Frequency Domain (NFD) to improve robustness of ResNet-18 on CIFAR-10-C with little accuracy drop. All the results inspire us to pay more attention to the frequency interpretability of models.

References

[1] Antreas Antoniou, Amos J. Storkey, and Harrison Edwards. Data augmentation generative adversarial networks. *ArXiv*, abs/1711.04340, 2017.

[2] Long Chen, Hanwang Zhang, Jun Xiao, Liqiang Nie, Jian Shao, and Tat-Seng Chua. Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6298–6306, 2016.

[3] Ekin Dogus Cubuk, Barret Zoph, Dandelion Mané, Vijay Vasudevan, and Quoc V. Le. Autoaugment: Learning augmentation policies from data. *ArXiv*, abs/1805.09501, 2018.

[4] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[5] Basura Fernando, Hakan Bilen, Efstratios Gavves, and Stephen Gould. Self-supervised video representation learning with odd-one-out networks. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5729–5738, 2016.

[6] Maayan Frid-Adar, Idit Diamant, Eyal Klang, Michal Amitai, Jacob Goldberger, and Hayit Greenspan. Gan-based synthetic medical image augmentation for increased cnn performance in liver lesion classification. *Neurocomputing*, 321:321–331, 2018.

[7] 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. *ArXiv*, abs/1811.12231, 2018.

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2015.

[9] Dan Hendrycks and Thomas G. Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *ArXiv*, abs/1903.12261, 2019.

[10] Dan Hendrycks, Mantas Mazeika, Saurav Kadavath, and Dawn Song. Using self-supervised learning can improve model robustness and uncertainty. *ArXiv*, abs/1906.12340, 2019.

[11] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[12] Gao Huang, Zhuang Liu, Kilian Q. Weinberger, and Laurens van der Maaten. Densely connected convolutional networks. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2261–2269, 2016.

[13] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *ArXiv*, abs/1502.03167, 2015.

[14] Alex Krizhevsky. Learning multiple layers of features from tiny images. 2009.

[15] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. *Commun. ACM*, 60:84–90, 2012.

[16] Raphael Gontijo Lopes, Dong Yin, Ben Poole, Justin Gilmer, and Ekin Dogus Cubuk. Improving robustness without sacrificing accuracy with patch gaussian augmentation. *ArXiv*, abs/1906.02611, 2019.

[17] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *ArXiv*, abs/1706.06083, 2017.

[18] Pablo Navarrete Michelini, Hanwen Liu, Yunhua Lu, and Xingqun Jiang. A tour of convolutional networks guided by linear interpreters. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.

[19] Konda Reddy Mopuri, Utsav Garg, and R. Venkatesh Babu. Cnn fixations: An unraveling approach to visualize the discriminative image regions. *IEEE Transactions on Image Processing*, 28:2116–2125, 2017.

[20] A. V. Oppenheim and J. S. Lim. The importance of phase in signals. *Proceedings of the IEEE*, 69:529–541, 1980.

[21] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 618–626, 2016.

[22] Uri Shaham, Yutaro Yamada, and Sahand N. Negahban. Understanding adversarial training: Increasing local stability of supervised models through robust optimization. *Neurocomputing*, 307:195–204, 2018.

[23] Connor Shorten and Taghi M. Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of Big Data*, 6:1–48, 2019.

[24] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

[25] Antonio Torralba and Aude Oliva. Statistics of natural image categories. *Network*, 14 3:391–412, 2003.

[26] Hsiao-Yu Fish Tung, Hsiao-Wei Tung, Ersin Yumer, and Katerina Fragkiadaki. Self-supervised learning of motion capture. In *NIPS*, 2017.

[27] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *The European Conference on Computer Vision (ECCV)*, September 2018.

[28] Dong Yin, Raphael Gontijo Lopes, Jonathon Shlens, Ekin Dogus Cubuk, and Justin Gilmer. A fourier perspective on model robustness in computer vision. *ArXiv*, abs/1906.08988, 2019.

[29] Jason Yosinski, Jeff Clune, Anh Mai Nguyen, Thomas J. Fuchs, and Hod Lipson. Understanding neural networks through deep visualization. *ArXiv*, abs/1506.06579, 2015.

[30] Matthew D. Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. *ArXiv*, abs/1311.2901, 2013.

[31] Xiaohua Zhai, Avital Oliver, Alexander Kolesnikov, and Lucas Beyer. S4l: Self-supervised semi-supervised learning. *ArXiv*, abs/1905.03670, 2019.