Machine Learning (2022) 111:3925–3951
https://doi.org/10.1007/s10994-022-06222-8

Aliasing and adversarial robust generalization of CNNs

Julia Grabinski[1] · **Janis Keuper**[3,4] · **Margret Keuper**[1,2]

Received: 10 December 2021 / Revised: 13 May 2022 / Accepted: 2 July 2022 /
Published online: 26 August 2022

Abstract

Many commonly well-performing convolutional neural network models have shown to be susceptible to input data perturbations, indicating a low model robustness. To reveal model weaknesses, adversarial attacks are specifically optimized to generate small, barely perceivable image perturbations that flip the model prediction. Robustness against attacks can be gained by using adversarial examples during training, which in most cases reduces the measurable model attackability. Unfortunately, this technique can lead to robust overfitting, which results in non-robust models. In this paper, we analyze adversarially trained, robust models in the context of a specific network operation, the downsampling layer, and provide evidence that robust models have learned to downsample more accurately and suffer significantly less from downsampling artifacts, aka. aliasing, than baseline models. In the case of robust overfitting, we observe a strong increase in aliasing and propose a novel early stopping approach based on the measurement of aliasing.

Keywords Adversarial robustness · Aliasing · Roubst overfitting · CNNs

Editor: Krzysztof Dembczynski and Emilie Devijver.

Janis Keuper and Margret Keuper these authors contributed equally to this work.

✉ Julia Grabinski
julia.grabinski@uni-siegen.de

Janis Keuper
janis.keuper@hs-offenburg.de

Margret Keuper
margret.keuper@uni-siegen.de

[1] Visual Computing, University of Siegen, Siegen, Germany

[2] Max Planck Institute for Informatics, Saarland Informatics Campus, Saarbrücken, Germany

[3] Institute for Machine Learning and Analytics, Offenburg University, Offenburg, Germany

[4] Competence Center High Performance Computing, Fraunhofer ITWM, Kaiserslautern, Germany

1 Introduction

Convolutional Neural Networks (CNNs) provide highly accurate predictions in a wide range of applications. Yet, to allow for practical applicability, CNN models should not be fooled by small image perturbations, as they are realized by adversarial attacks (Goodfellow et al., 2015a; Moosavi-Dezfooli et al., 2016; Rony et al., 2019). Such attacks are optimized to find image perturbations such that the network makes incorrect predictions. Since the perturbations are small, human observers can usually still easily recognize the correct class label. Susceptibility to such perturbations is prohibitive for the applicability of CNN models in real world scenarios, as it indicates limited reliability and generalization of the model (Fig. 1).

To establish adversarial robustness many sophisticated methods have been developed (Goodfellow et al., 2015a; Rony et al., 2019; Kurakin et al., 2017). Some can defend only against one specific attack (Goodfellow et al., 2015a) while others propose more general defenses against diverse attacks. However, even those defenses suffer from a phenomenon called robust overfitting, the model confronted with adversarial examples overfits on the already seen examples and loses its general robustness against different/stronger attacks like PGD (Kurakin et al., 2017).

Another way to protect CNNs against adversarial examples is to detect them. Harder et al. (2021) as well as Lorenz et al. (2021) detect adversarial examples by inspecting each input image and its feature maps in the frequency domain. Similarly, Yin et al. (2020) showed that natural images and adversarial examples differ significantly in their frequency spectra.

In fact, when considering the architecture of commonly employed CNN models, one could wonder why these models perform so well although they ignore basic sampling theoretic foundations. Concretely, most architectures sub-sample feature maps without ensuring to sample above the Nyquist rate (Shannon 1949), such that, after each down-sampling operation, spectra of sub-sampled feature maps may overlap with their replica. This is called *aliasing* and implies that the network should be genuinely unable to fully restore an image from its feature maps. One can only hypothesize that common CNNs learn to

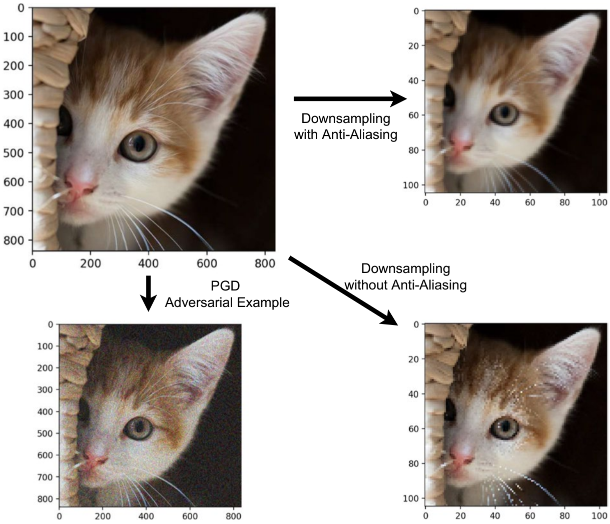


Fig. 1 Illustration of down-sampling, with (top right) and without anti-aliasing filter (bottom right) as well as an adversarial example (bottom left). The top left image shows the original. In the top right, the image is correctly down-sampled with an anti-aliasing filter. In the bottom right, no filter is applied, leading to aliasing. The adversarial example (bottom left) shows visually similar artifacts. In this paper, we investigate the role of aliasing for adversarial robustness

(partially) compensate for this effect by learning appropriate filters. Following this line of thought, recently, several publications suggest to improve CNNs by including anti-aliasing techniques during down-sampling in CNNs (Zhang 2019; Zou et al., 2020; Li et al., 2021b; Hossain et al., 2021). They aim to make the models more robust against image-translations, such that the class prediction does not suffer from small vertical or horizontal shifts of the content.

In this paper, we further investigate the relationship between adversarial robustness and aliases. While previous works (Yin et al., 2020; Harder et al., 2021; Lorenz et al., 2021) focused on adversarial examples, we systematically analyze potential aliasing effects inside CNNs. Specifically, we compare several recently proposed adversarially robust models to conventionally trained models in terms of aliasing. We inspect intermediate feature maps before and after the down-sampling operation at inference. Our first observation is that these models indeed fail to sub-sample according to the Nyquist Shannon Theorem (Shannon 1949): we observe severe aliasing. Further, our experiments reveal that adversarially trained networks exhibit less aliasing than standard trained networks, indicating that adversarial training (AT) encourages CNNs to learn how to properly down-sample data without severe artifacts. Next, we visualize the frequency spectra of adversarial attacks on baseline models as well as on adversarially trained ones. Our experiments show that attacks behave in a less characteristic spectrum when attacked models are adversarially robust. This indicates that adversarial attacks might employ network aliasing as a backdoor, such that high frequency changes can flip the network decision, while attacks on adversarially robust networks have to hamper with the low-frequency components of the image, i.e. the coarse details. Finally, we investigate the behavior during training and observe a strong correlation between robust overfitting during training and the amount of aliasing in the network's downsampling operations. Specifically, the amount of aliasing increases significantly as the model overfits to AT. Based on these findings we propose a new early stopping criterion based on the measurement of aliasing, to prevent robust overfitting during AT.

In summary, our contributions are:

- We introduce a novel measure for aliasing and show that common CNN down-sampling layers fail to sub-sample their feature maps in a Nyquist-Shannon conform way.
- We analyze various adversarially trained models, that are robust against a strong ensemble of adversarial attacks, AutoAttack (Croce and Hein 2020), and show that they exhibit significantly less aliasing than standard models.
- We show strong evidence that robust overfitting coincides with an increased amount of aliasing for several network architectures.
- We introduce a new early stopping criterion for FGSM AT based on our aliasing measurement.

2 Related work

2.1 Downsampling attack

Xiao et al. (2017) demonstrated the power of down-sampling attacks. These attacks modify the images such that their original size is too big for the network and they need to be down-sampled in the pre-processing step. Thereby, one can hide a completely new image in the bigger one. This new image is only visible after down-sampling and will determine the predicted class label. Later, Lohn (2020) re-raised this issue and proposed a defense for such attacks, since they can still be realized due to vulnerable down-scaling in common python libraries like Tensorflow and OpenCV.

2.2 Adversarial attacks

While CNNs are known for their excellent performance on image classification tasks, they are susceptible to adversarial attacks (Moosavi-Dezfooli et al., 2016; Goodfellow et al., 2015a; Szegedy et al., 2014), i.e. to intentional image perturbations. Recently, many different adversarial attacks as well as defenses have been developed. One of the earliest attacks is the Fast Gradient Sign Method (FGSM) by Goodfellow et al. (2015a), followed by more sophisticated methods like Projected Gradient Descent (PGD) (Kurakin et al., 2017), DeepFool (DF) (Moosavi-Dezfooli et al., 2016), Carlini and Wagner (CW) (Carlini and Wagner 2017) or Decoupling Direction and Norm (DDN) (Rony et al., 2019). While single step adversarial examples, like FGSM, take the full possible perturbation step in the range of ϵ in one step, PGD iteratively searches for the best step over a maximal number of iterations. Yet, instead of creating each adversarial example with the same strength, PGD adapts the amount of perturbed pixels by iteratively checking the current class prediction. Recently, AutoAttack (Croce and Hein 2020), an ensemble of different attacks including an adaptive version of PGD, has become the baseline for adversarial robustness.

2.2.1 Adversarial training

Most proposed attacks come with a dedicated defense, to counter their adversarial examples (Goodfellow et al., 2015a; Rony et al., 2019). There are many more adversarial training (AT) schemes which typically consist of either adding a second loss term to be more robust against a special type of adversarial noise (Engstrom et al., 2019; Zhang et al., 2019) or add additional data (Carmon et al., 2019; Sehwag et al., 2021). Some approaches also combine both (Wang et al., 2020c). RobustBench (Croce et al., 2020) evaluates of a variety of models w.r.t. their adversarial robustness.

2.2.2 Robust overfitting

Rice et al. (2020) showed that simple defense methods for examples based on FGSM adversarial examples suffer from robust overfitting, the phenomenon that the model overfits to the seen adversarial examples and shows decreased robust accuracy against other attacks like PGD (Kurakin et al., 2017). Therefore they introduce early stopping based on more expensive PGD adversarial examples to find a good trade-off between the model's performance and robustness. Further, Chen et al. (2021) suggest to prevent overfitting by forcing the network to more learned smoothing during AT. Therefore they perform stochastic weight averaging as well as smoothing of the logits.

2.3 Frequency analysis

Several recent works considered the frequency spectra of images at the input of CNNs and deeper layers, which we briefly summarize below.

2.3.1 Robustness, attack detection and image generation

Yin et al. (2020) could show that conventional CNNs are sensitive to changes in the high frequencies of an image, like Gaussian noise, while most CNNs are robust against changes in the low frequencies, i.e. in the coarse structures. In contrast, when models are trained using additional data augmentation techniques, they are less sensitive to high frequency changes but sacrifice their robustness in the low frequency domain (Yin et al., 2020). Recently, et al. Hossain et al. (2021) analyzed the frequency spectrum of adversarial examples and observed that adversarial perturbations are not exclusively affecting high frequencies, which was assumed before (Wang et al., 2020a). Hossain et al. (2021) observe that the perturbations spectrum highly depends on the dataset used, i.e. CIFAR-10 or ImageNet. These works aim to improve CNNs by reducing aliasing. However, they do not systematically investigate the effects on adversarial robustness and they do not provide an actual measure for aliasing. Also Bernhard et al. (2021) highlight the fact that adversarial attacks do not exclusively hurt the high frequency components by incorporating a frequency constraint which needs to be adapted on the frequency features of the data. Harder et al. (2021) use the spectrum of the adversarial examples to detect them, i.e. Lorenz et al. (2021) train a classifier to detect adversarial examples and defend CNNs. These works indicate that there is a severe domain shift between the frequency distribution of genuine images and adversarial attacks.

Durall et al. (2020) observed that CNN generated images fail to reproduce the spectral distribution of original images, making them easy to detect in practice. Similar to the here addressed aliasing in classifier models during downsampling, Frank et al. (2020) indicate that generative models inherently suffer from aliasing during their upsampling operation. This observation has recently lead to a vast amount of research in the area of fake image detection (Frank et al., 2020; Chandrasegaran et al., 2021; Durall et al., 2020; Dzanic et al., 2020; Wang et al., 2020b) and especially face forgery detection (Li et al., 2021a; Luo et al., 2021). Both benefit from frequency domain representations. He et al. (2021) employ a systematic domain shift in the frequency domain to propose better generalizable deep fake detectors.

2.3.2 Frequency biased models

Saikia et al. (2021) proposed a method to boost adversarial robustness by training a low and a high frequency expert. They suggest training two different models, one that should only use low frequencies for prediction and one for only high frequencies. Both models give a joint prediction and can achieve much higher robustness than standard trained models. This way they bypass the time and computational resources consuming AT. But still they need to train both experts separately.

2.3.3 Domain adaptation and generalization

Yang and Soatto (2020) showed that the Fourier phase and amplitude of an image can be adapted for data augmentation. Thus, they trained a better generalizing model in the context of domain adaptation. Similarly, Xu et al. (2021) use the Fourier phase and shuffled the amplitude of images to train for higher domain generalization.

2.3.4 Anti-aliasing

Azulay and Weiss (2018) discussed the question of why CNNs can not learn invariances to small image transformations (such as shifts) from training data and argue that aliasing during downsampling is causing this behavior. Since then, Anti-Aliasing filters are becoming more and more important for the Deep Learning community. Zhang (2019) established Anti-Aliasing filter in CNNs for shift-invariance for classification tasks. This approach has been further improved by Zou et al. (2020) by adaptive Anti-Aliasing filters depending on the image patch. Li et al. (2021b) use the low frequency components of wavelets in their pooling to increase robustness against common image corruptions by suppressing the effects of aliasing. Here, we show that aliasing is not only relevant for robustness to common corruptions but also affects adversarial robustness. Hossain et al. (2021) propose not only a depth adaptive blurring filter before pooling but also an anti-aliasing activation function. This activation function is inspired by C-ReLu but uses a smooth roll-off phase instead of the sharp cutoff at threshold t. Also, Karras et al. (2021) achieve aliasing free generators for GANs by blurring before sampling and non-linearities, like ReLu, whereas Jung and Keuper (2021) address aliasing in GANs by employing a frequency space discriminator. In contrast to previous work which focused on fixing shift-invariance and model robustness by incorporating anti-aliasing techniques, we are the first focusing on the analysis of aliasing, to obtain a distinct aliasing measurement.

3 Aliasing in CNNs

CNNs usually have a pyramidal structure in which the data is progressively sub-sampled in order to aggregate spatial information while the number of channels increases. During sub-sampling, no explicit precautions are taken to avoid aliases, which arise from under-sampling. Specifically, when sub-sampling with stride 2, any frequency larger than $N/2$, where N is the size of the original data, will cause pathological overlaps in the frequency spectra (Fig. 2). Those overlaps in the frequency spectra cause ambiguities such that high frequency components appear as low frequency components. Hence, local image perturbations can become indistinguishable from global manipulations.

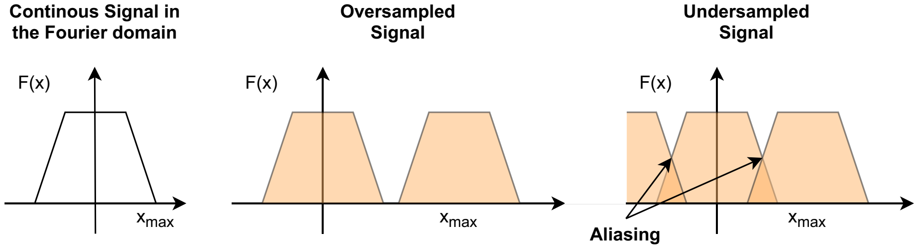


Fig. 2 Abstract example of aliasing in the frequency domain. Left a 1D signal with the x_{max} as maximal width is shown in the frequency domain. When this signal is down-sampled, the original signal is replicated and placed next to each other, depending on the sampling rate. If we sample at a sufficiently large sampling rate, the distance between the replica is large and the signals will not overlap (middle). If the sampling rate is too small we under-sample the signal and get aliases due to the overlapping replica (right)

3.1 Aliasing metric

To measure the possible amount of aliasing appearing after down-sampling we compare each down-sampled feature map in the Fourier domain with its aliasing-free counterpart. To this end, we consider a feature map *f*(*x*) of size $2N \times 2N$ before down-sampling. We compute an "aliasing-free" down-sampling by extracting the *N* lowest frequencies along both axes in Fourier space. W.l.o.G., we consider specifically down-sampling operations by strided convolutions, since these are predominantly used in adversarially robust models (Zagoruyko and Komodakis 2017).

In each strided convolution, the input feature map *f*(*x*) is convolved with the learned weights *w* and downsampled by strides, thus potentially introducing frequency replica (i.e. aliases) in the downsampled signal $\hat{f}_{s2}$.

$$\hat{f}_{s2} = f(x) * g(w, 2) \tag{1}$$

To measure the amount of aliasing, we explicitly construct feature map frequency representations without such aliases. Therefore, the original feature map *f*(*x*) is convolved with the learned weights *w* of the strided convolution without applying the stride *g*(*w*, 1) to obtain $\hat{f}_{s1}$.

$$\hat{f}_{s1} = f(x) * g(w, 1) \tag{2}$$

Afterwards the 2D FFT of the new feature maps $\hat{f}_{s2}$ is computed, which we denote F_{s2}.

$$F_{s2}(k, l) = \frac{1}{N^2} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} \hat{f}_{s2}(m, n) e^{-2\pi j(\frac{k}{M}m + \frac{l}{N}n)}, \tag{3}$$

for $k, l = 0, \dots, N-1$. For the non-down-sampled feature maps $\hat{f}_{s1}$, we proceed similarly and compute for $k, l = 0, \dots, 2 \cdot N - 1$

$$F^{\uparrow}_{s1}(k, l) = \frac{1}{4N^2} \sum_{m=0}^{2N-1} \sum_{n=0}^{2N-1} \hat{f}_{s1}(m, n) e^{-2\pi j(\frac{k}{2M}m + \frac{l}{2N}n)}. \tag{4}$$

The aliasing free version F_{s1} can be obtained by setting all frequencies above the Nyquist rate to zero before down-sampling,

$$F^{\uparrow}_{s1}(k, l) = 0 \tag{5}$$

for $k \in [N/2, 3N/2]$ and for $l \in [N/2, 3N/2]$. Then the down-sampled version in the frequency domain corresponds to extracting the four corners of $F^{\uparrow}_{s1}$ and reassembling them as shown in Fig. 3,

$$\begin{aligned}
F_{s1}(k, l) &= F^{\uparrow}_{s1}(k, l) && \text{for} \quad k, l = 0, \dots, N/2 \\
F_{s1}(k, l) &= F^{\uparrow}_{s1}(k + N, l) && \text{for} \quad k = N/2, \dots, N \\
& && \text{and} \quad l = 0, \dots, N/2 \\
F_{s1}(k, l) &= F^{\uparrow}_{s1}(k, l + N) && \text{for} \quad k = 0, \dots, N/2 \\
& && \text{and} \quad l = N/2, \dots, N \\
F_{s1}(k, l) &= F^{\uparrow}_{s1}(k + N, l + N) && \text{for} \quad k, l = N/2, \dots, N
\end{aligned} \tag{6}$$

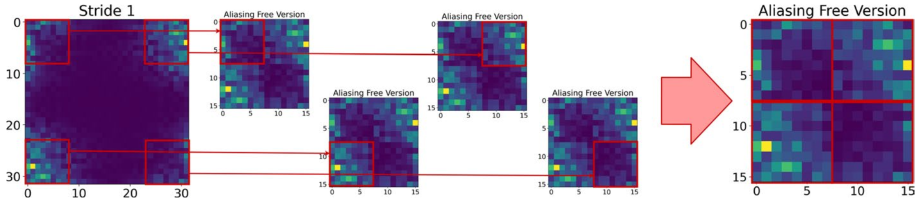


Fig. 3 Step by step computation of the aliasing free version of a feature map. The left image shows the magnitude of the Fourier representation of a feature map with the zero-frequency in the upper left corner, i.e. high frequencies are in the center. Alias-free downsampling suppresses high frequencies prior to sampling. This can be implemented efficiently in the Fourier domain by cropping and reassembling the low-frequency regions of the Fourier representations, i.e. its four corners. Aliasing would correspond to folding the deleted high frequency components into the constructed representation

This way we guarantee that there are no overlaps, i.e. aliases, in the frequency spectra. Figure 3 illustrates the computing process of the aliasing free down-sampling in the frequency domain. The aliasing free feature map can be compared to the actual feature map in the frequency domain to measure the degree of aliasing. The full procedure is shown in Fig. 4, where we start on the left with the original feature map. Then we obtain the two down-sampled versions (with and without aliases) and compute their L_1 difference.

The overall aliasing metric AM for a down-sampling operation is calculated by the L_1 distance between downsampled and alias-free feature maps f_k in the Fourier domain, averaged over K generated feature maps,

$$AM = \frac{1}{K}\sum_{k=0}^{K} \|F_{s1,k} - F_{s2,k}\|. \tag{7}$$

The proposed AM measure is zero if aliasing is visible in none of the down-sampled feature maps, i.e. if sampling has been performed above the Nyquist rate. Whenever AM is greater than 0, this is not the case and we should, from a theoretic point of view, expect the model to be easy to attack since it can not reliably distinguish between fine details and coarse input structures.

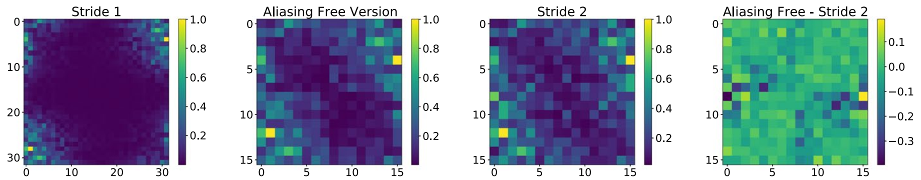


Fig. 4 FFT (Fast Fourier Transformation) of a feature map in the original resolution (left). This feature map is downsampled by striding with a factor of two after aliasing suppression (middle left) and with aliasing (middle right). The difference between the original and aliasing-free FFT of the down-sampled feature map (right)

4 Experiments

4.1 Aliasing in existing models

We conducted an extensive analysis of already existing adversarially robust models trained on CIFAR-10 (Krizhevsky 2012) with two different architectures, namely WideResNet-28-10 (WRN-28-10) (Zagoruyko and Komodakis 2017) and Preact ResNet-18 (PRN-18) (He et al., 2016). Both architectures are commonly supported by many AT approaches. As baseline, we trained a plain WRN-28-10 and PRN-18, both with similar training schemes. Each model is trained with 200 epochs, a batch size of 512, CrossEntropy loss and stochastic gradient descent (SGD) with an adaptive learning rate starting at 0.1 and reducing it at 100 and 150 epochs by a factor of 10, a momentum of 0.9 and a weight-decay of 5e-4. All adversarially trained networks are pre-trained models provided by RobustBench (Croce et al., 2020).

The WRN-28-10 networks have four operations in which down-sampling is performed. These operations are located in the second and third block of the network. In comparison, the PRN-18 networks have six down-sampling operations, located in the second, third and fourth layers of the network.

Both architectures have similar building blocks and the key operations including down-sampling are shown abstractly in Fig. 5. Each block starts with a convolution with stride two followed by additional operations like ReLu and convolutions with stride one. The characteristic skip connection of ResNet architectures also needs to be implemented with stride two if down-sampling is applied in the according block. Consequently, we need to analyze all down-sampling units and skip connections before they are summed up to form the output feature map.

4.1.1 WideResNet 28-10

In the following, differently trained WRN-28-10 networks are compared in terms of their robust accuracy against AutoAttack (Croce and Hein 2020) and the amount of aliasing in their down-sampling layer.

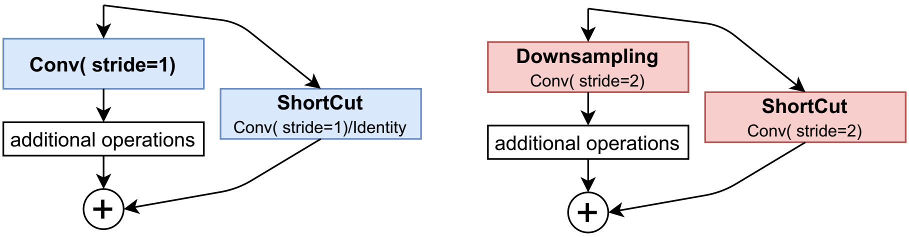


Fig. 5 Abstract Illustration of a building block in PRN-18 and WRN-28-10. The first operation in a block is a convolution, executed with a stride of either one or two. For a stride of one (left) the shortcut simply passes the identity of the feature maps forward. If the first convolution is done with a stride of two, the shortcut needs to have a stride of two (right) too, to guarantee that both representations can be added at the end of the building block

Figure 6 indicates significant differences between adversarially trained and standard trained networks. First, the networks trained without AT are not able to reach any robust accuracy, meaning their accuracy under adversarial attacks is equal to zero. Second, and this is most interesting for our investigation, standard trained networks exhibit much more aliasing during their down-sampling layers than adversarially trained networks. Through all layers and operations in which down-sampling is applied, the adversarially trained networks (blue dots Fig. 6) have much higher robust accuracy and much less aliasing compared to the standard trained networks. We indicate the Pearson correlation *r* between aliasing and robust accuracy above each scatter plot in Fig. 6, indicating a significant negative correlation.

When comparing the conventionally trained network against each other it can be seen that also the specific training scheme used for training the network can have an influence on the amount of aliasing of the network. Concretely, the standard baseline model provided by RobustBench (Croce et al., 2020) exhibits less aliasing than the one trained by us. Unfortunately, there is no further information about the exact training schedule from RobustBench, such that we can not make any assumptions on the interplay between model hyperparameters and aliasing.

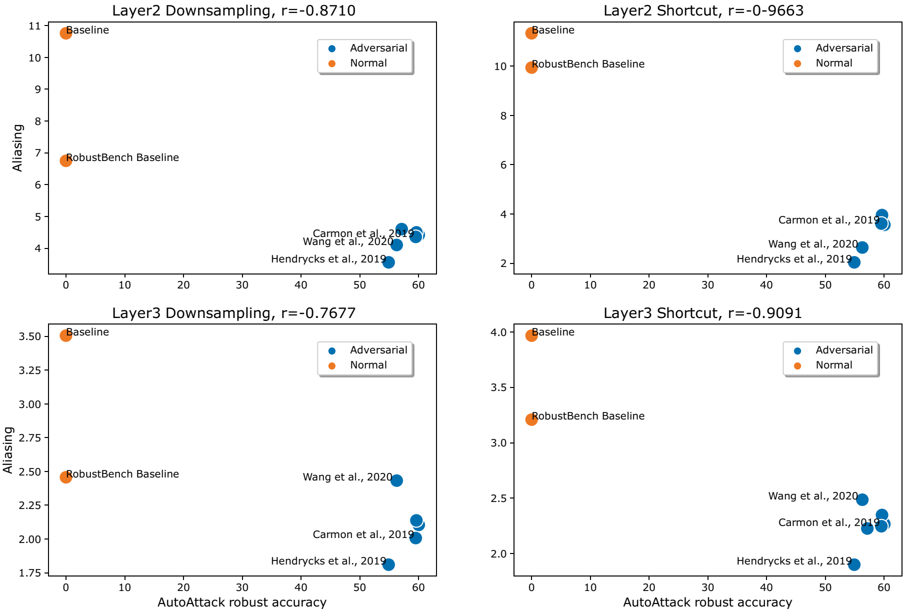


Fig. 6 Adversarial robustness versus aliasing and the according correlation *r*, evaluated on different pre-trained WRN-28-10 models from RobustBench (Croce et al., 2020) as well as two baseline models without AT, one from RobustBench (RobustBench Baseline) and one trained by us (Baseline). All blue dots represent adversarially trained networks for the purpose of clarity we marked three popular models from Carmon et al. (2019), Wang et al. (2020c) and Hendrycks et al. (2019) by name (Color figure online)

4.1.2 Preact ResNet-18

We conducted the same measurements for the PRN-18 as we did for the WRN-28-10 and used the same training procedure. Additionally, we needed to account for one more layer with two additional down-sampling operations.

The overall results, presented in Fig. 7, are similar to the ones for the WRN-28-10 networks, most adversarially trained networks exhibit significantly less aliasing and higher robustness than conventionally trained ones. Yet, the additional down-sampling layer allows one further observation. While the absolute aliasing metric is overall lower, the robust networks reduce the aliasing predominantly in the earlier layers, the second and third layers. The aliasing in the fourth layer of adversarially robust models is not significantly different from the aliasing in conventionally trained models in the same layer. This phenomenon might be explained by the sparsity of the deeper layers. While the earlier feature maps represent the spatial properties of input images, deeper layers rather encode

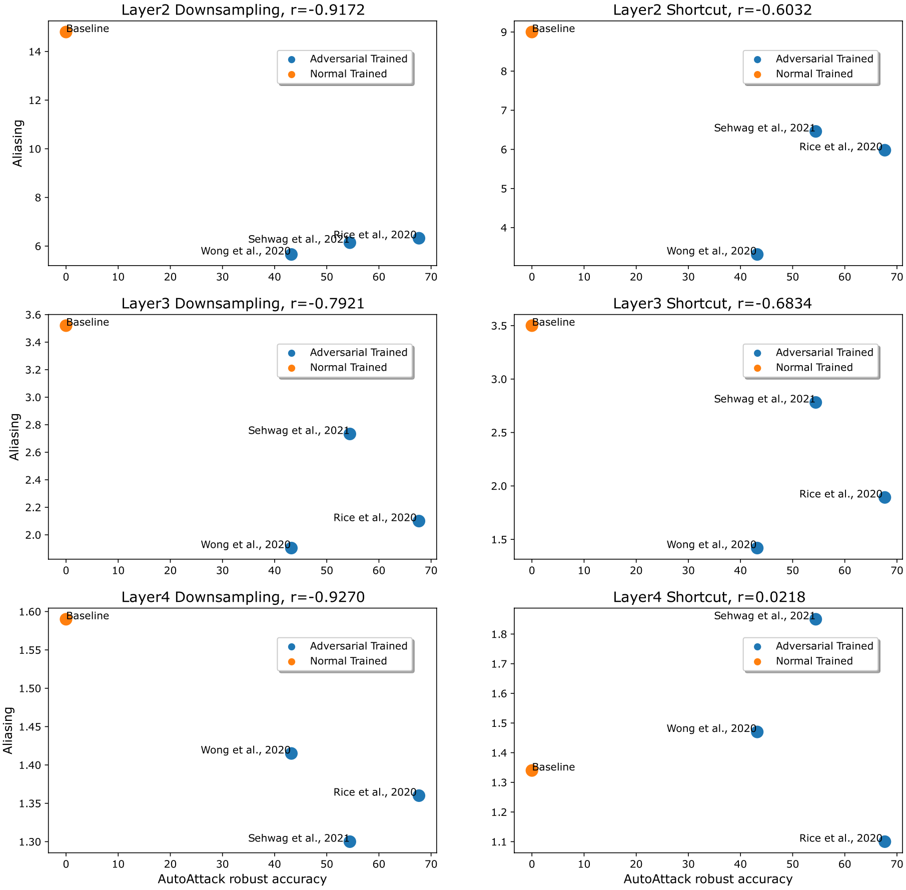


Fig. 7 Adversarial robustness versus aliasing and the according correlation *r*, exemplary evaluated on different pre-trained PRN-18 models. The blue dots represent adversarial trained networks, trained with the training schemes of Wong et al. (2020), Rice et al. (2020) and Sehwag et al. (2021) provided by RobustBench (Croce et al. 2020). The orange dot is the baseline, trained by us without AT (Color figure online)

semantic properties that are sparsely encoded and therefore might be harder to affect. Yet, this aspect needs further investigation for a better understanding, which goes beyond the scope of this work.

4.1.3 Spectrum of adversarial perturbations

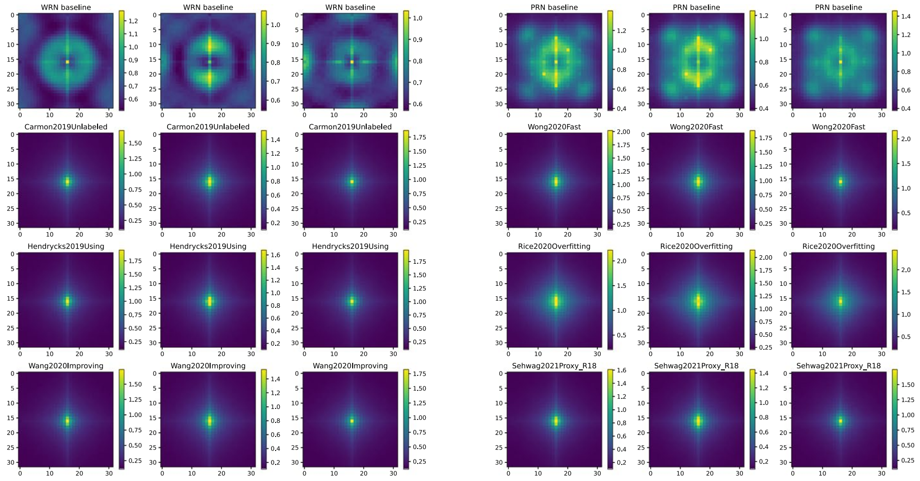
Further we analyze the spectrum of the perturbations created by adversarial examples, i.e. perturbations created by AutoAttack (Croce and Hein 2020).

Firstly, we compare the spectrum of the perturbations created by the AutoAttack standard attack on our baseline model as well as on the robust models which we already evaluated in Sect. 4.1. We compute the perturbations as differences between adversarial and clean images. Afterwards we transform each perturbation into the Fourier space and take each of the three channels, RGB. The results are shown in Fig. 8.

We can see that the frequencies in which adversarial attacks like AutoAttack attack, vary and this is in line with (Maiya et al., 2021). Here we can see that the spectrum of the attack, not only varies w.r.t. the dataset but also w.r.t. the specific model architecture.

4.2 Aliasing during training

Next, we consider the amount of aliasing during training of adversarially robust and conventional models. We trained five PRN-18 models with different training schemes and one PRN-18 without AT as baseline, using the same training parameters described in Sect. 4.1. The training schemes provided by Wong et al. (2020) and Rice et al. (2020) were used for the AT. During each training run, we computed the amount of aliasing in each downsampling and shortcut layer for each epoch from 100 randomly picked CIFAR-10 training samples.



(a) WRN

(b) PRN

Fig. 8 Spectrum of the RGB perturbations created by AutoAttack (Croce and Hein 2020) on the baseline model (top row) and three different robust models from RobustBench Croce et al. (2020)

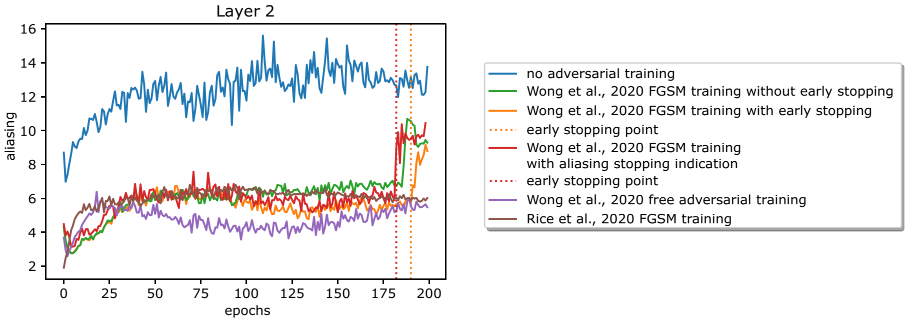


Fig. 9 Amount of aliasing in the second layer of different PRN-18 (an overview over all layers can be found in appendix 1, Fig. 16) during training over 100 random images of the CIFAR-10 training set for the baseline and AT training provided by Wong et al. (2020) and Rice et al. (2020). The aliasing measure is the mean value between the aliasing of the down-sampling and the shortcut layer. All networks are trained for 200 epochs, except FGSM training including early stopping. There the training is stopped earlier based on the evaluation of the model on PGD or our aliasing measure. Still, we record the epochs after early stopping and mark the point of early stopping by the dashed lines, to demonstrate the relationship between aliasing and early stopping.

Table 1 Clean and robust accuracy against PGD (higher is better) and the amount of aliasing in the second layer (lower is better) of the baseline and adversarial trained networks with the training scheme provided by Wong et al. (2020) and Rice et al. (2020) as well as FGSM with the training schedule of Wong et al. (2020) including early stopping criteria based on our aliasing measurement

Method	Clean Acc ↑	PGD Acc ↑	Aliasing ↓
Baseline	**93.29**	0.00	12.12
FGSM (Wong et al., 2020)	<u>90.85</u>	7.05	9.31
early-stopping FGSM (Wong et al., 2020)	80.16	39.76	6.14
Free (Wong et al., 2020)	83.86	48.10	5.62
PGD (Wong et al., 2020)	85.06	**56.37**	6.30
Robust Overfitting (Rice et al., 2020)	84.58	46.70	**3.99**
Aliasing FGSM (ours)	82.91	<u>52.43</u>	<u>5.78</u>

Bold values highlight the low/highest value for each column

Underlined values highlight the second best value for each column

Figure 9 shows the amount of aliasing for the second layer in total, which is computed as the mean between the aliasing of the down-sampling and shortcut layer for the different AT training schemes as well as the baseline. We can observe that the adversarial trained networks produce a lower amount of aliasing during the entire training procedure. The final results for all adversarial trained models for different training schemes are presented in Table 1. All models trained with AT exhibit less aliasing and higher robust accuracy than the baseline.

Further, we can observe that early stopping for FGSM plays a crucial role for the robust accuracy as well as for the aliasing. Table 1 shows that FGSM without early stopping performs nearly as poorly as training without any adversaries. Figure 9 indicates that the model trained with FGSM without early stopping has a high increase in aliasing at the end of the training. The models which include early stopping stop before and thus exhibit no increased aliasing. The training without early stopping continues and

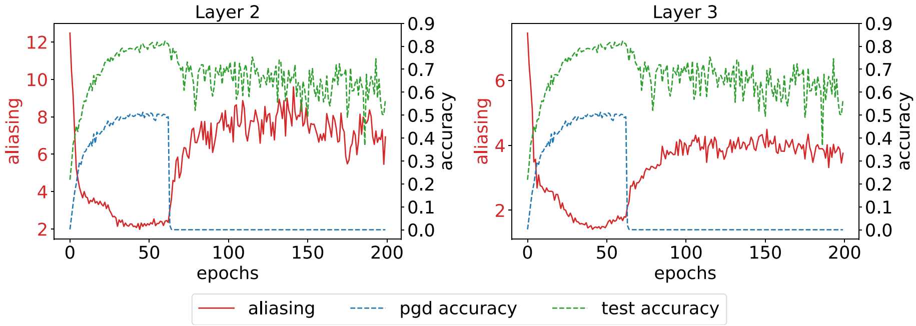


Fig. 10 Aliasing, clean accuracy and PGD accuracy during training of a WRN-28-10 with FGSM AT and cycling learning rate. The model starts to exhibit robust overfitting in epoch 70, i.e. the PGD accuracy drops to zero and the amount of aliasing increases significantly

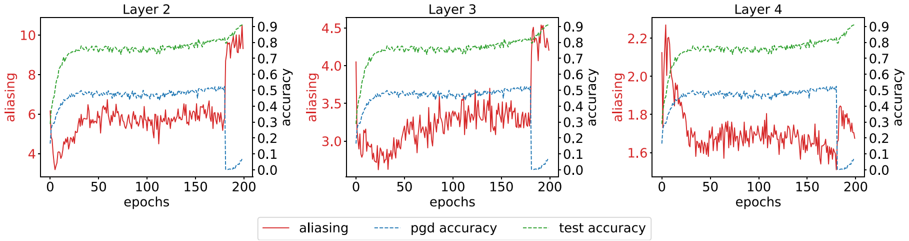


Fig. 11 Aliasing, clean accuracy and PGD accuracy during training of a PRN-18 with FGSM AT and cycling learning rate. The model starts to exhibit robust overfitting in epoch 180, i.e. the PGD accuracy drops to zero and the amount of aliasing increases significantly

increases the amount of aliasing while losing its robust accuracy. This indicates that aliasing and adversarial robustness are highly related.

This behavior of FGSM without early stopping is commonly referred to as robust overfitting. The standard way to overcome this effect is to evaluate the model after each epoch with PGD attacks and to compare the robust accuracy to the previous robust accuracy to note if the performance drops in this respect. Hence, the training is not as expensive as a full training with PGD but almost as good as PGD training.

4.3 Robust overfitting

Following, we aim to explore the relation between robust overfitting and aliasing further.

4.3.1 Aliasing versus PGD accuracy

To emphasize our findings we investigate the direct correlation between aliasing and the PGD attackability of the model during training. Figure 10 for a WRN-28-10 and Fig. 11 for a PRN-18 show the amount of aliasing during training as well as the PGD accuracy for each training epoch. For both networks, we can see that at the point at which robust overfitting occurs, i.e. PGD accuracy drops to zero, the amount of aliasing increases significantly and remains high.

Figure 10 as well as Fig. 11 show that as soon as the robust accuracy against PGD drops, i.e. robust overfitting takes place, the amount of aliasing increases significantly. While the WRN-28-10 suffers from robust overfitting already at epoch ca 60 the PRN-18 needs to be trained at least 180 epochs to exhibit robust overfitting and an increased amount of aliasing. While the clean accuracy for the WRN is highest right before the increase in aliasing, i.e. the robust overfitting, and stays further below, the PRN clean accuracy increases right after the increase in aliasing.

4.3.2 Spectrum of adversarial perturbations during training

Next, we visualize the spectrum of attacks on our robust models before and after robust overfitting. The results for the attacks are shown in Fig. 13. While the perturbations of the robust models, before robust overfitting, exhibit the same spectral characteristics like the robust models from RobustBench (Croce et al., 2020), the perturbations for the models after robust overfitting lie more in the higher frequency spectrum. While they do not lie in the middle frequency spectrum like it is for the baseline model.

4.3.3 Effect on Network Confidences

To investigate further into the co-occurrence of aliasing and robust overfitting, we take a look at the predicted confidence of the network. Therefore we calculate the overall confidence of the network predictions on the clean and PGD perturbed data as well as the confidence on the false predictions caused by PGD which we call *bad pgd confidence*.

Figure 12 shows the networks' confidences in the clean and PGD data as well as the confidence in the wrong predictions caused by PGD and the amount of aliasing. We can observe for both, WRN-28-10 and PRN-18, the models confidence in the clean and adversarial test data increases significantly when aliasing increases, i.e. robust overfitting takes place. Interestingly, the false confidence on PGD perturbations is relatively low before the increase in aliasing but gets highest after the increase. We assume that the network is not only not robust but much more confident with its false predictions after robust overfitting (Fig. 13).

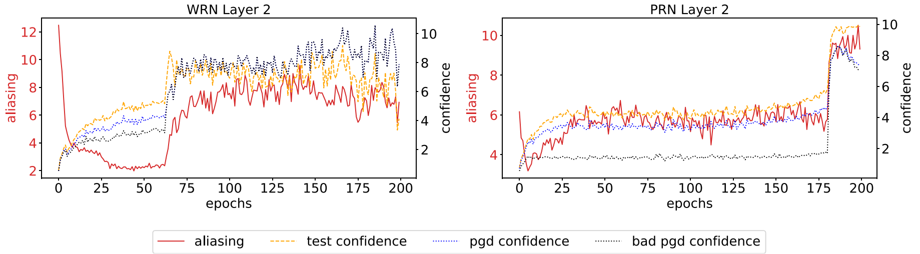


Fig. 12 Aliasing of the second layer (red) and confidence of a WRN-28-10 (left) and PRN-18 (right) model trained with FGSM AT. In red the amount of aliasing, the dotted lines represent the confidences overall in the clean (yellow) and adversarial data (blue) as well as the confidence in the false predictions caused by the adversaries (black) (Color figure online)

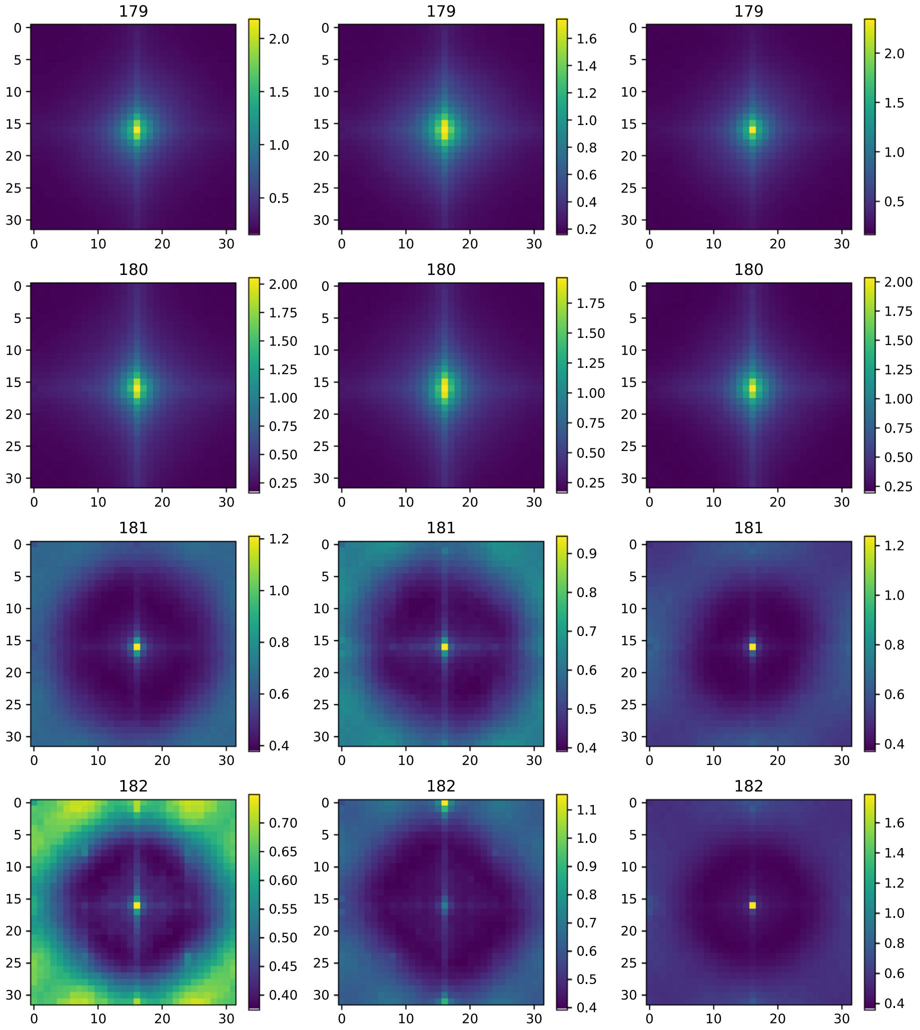


Fig. 13 Spectrum of the RGB perturbations created by AutoAttack at different epochs of a model training, before (two top rows, epoch 179 and 180) and after robust overfitting (bottom rows, epoch 181 and 182) (Color figure online)

5 Aliasing early stopping

Algorithm 1 Aliasing Early Stopping

```
Require: bat
Ensure: the model weights $m_{best}$, stopping point $p_{stop}$
  $t \leftarrow 0.33$
  $AM_{hist} \leftarrow []$, history of the aliasing measurements
  $AM_d, AM_s \leftarrow$ ALIASING METRIC$(2, m, X)$    ▷ Aliasing Metric will give the aliasing for the down-sampling layer $AM_d$ and the shortcut $AM_s$
  $AM \leftarrow (AM_d + AM_s)/2$
  $AM_{hist} \leftarrow AM_{hist}$.append$(AM)$
  for $e$ in $N$ do
      do network training
      $AM_d, AM_s \leftarrow$ ALIASING METRIC$(2, m, X)$
      $AM \leftarrow (AM_d + AM_s)/2$
      $AM_{hist} \leftarrow AM_{hist}$.append$(AM)$
      if $AM > MEDIAN(AM_{hist}) * (1 + t)$ then
          $p_{stop} \leftarrow e$
          return $m_{best}, p_{stop}$
      else
          $AM_{hist}$.append$(AM)$
          $m_{best} = m$
      end if
  end for
```

In Sect. 4.3, we showed that the model's robust overfitting with FGSM training (Goodfellow et al., 2015a) coincides with a sudden increase in aliasing in the models' feature maps. Now, we investigate whether we can exactly determine this overfitting point using the aliasing measure. Robust overfitting mainly occurs for AT with FGSM consequently we will perform our early stopping criteria with aliasing on FGSM AT.

We investigate whether we can exactly determine the point of robust overfitting using the aliasing measure. Thus, we define a threshold for the gap in aliasing between two epochs. This is similar to the procedure proposed by Wong et al. (2020) who determine such a threshold for the gap in the robust accuracy of PGD (Kurakin et al., 2017) between two epochs.

In this experiment, we only employ the aliasing measure computed from the features maps in the second layer. One PRN-18 layer with down-sampling includes two down-sampling operations, so we build the mean between their aliasing measures as done before. However, the aliasing also depends on the images in each specific batch, i.e.aliasing will be low for feature maps computed on very smooth input images while it will be larger for textured input data. To reduce noise, we apply a median filter to the aliasing measurements. This median filtered version is represented by the orange line in Fig. 14.

On the median filtered aliasing curves, we simply compare each new aliasing measure to the median and predict a high loss in PGD accuracy when the aliasing measure increases by more than 33% (see Algorithm 1). Figure 15 evaluates, for five different FGSM training runs, the early stopping points computed by Algorithm 1 for varying thresholds *t*. We report the distance (in epochs) of our predicted stopping point to the best early stopping point predicted using PGD. For thresholds *t* between 0.3 and 0.35, the PGD stopping point is correctly predicted for all training runs. It could even be used to determine the early stopping point in FGSM "on the fly" without explicit robustness tests. The computation of the early stopping point with our aliasing measurement takes only around 903.44 milliseconds per epoch while PGD takes around 1315.89 milliseconds per epoch. The results on FGSM training can be seen in Table 1. When compared with FGSM and FGSM with early stopping based on PGD provided by Wong et al. (2020) we can see that our aliasing early stopping is able to find the best trade-off between clean accuracy and robust accuracy, while keeping the amount of aliasing low.

6 Discussion

Our experiments reveal that common CNNs fail to sub-sample their feature maps in a Nyquist-Shannon conform way and consequently introduce aliasing artifacts. Further, we can give strong evidence that aliasing and adversarial robustness are highly related. All evaluated robust models exhibit significantly less aliasing than standard trained models.

We also gave an example use case for this finding, i.e.we showed that an aliasing based measure could replace the explicit evaluation of network robustness as an early stopping criterion in FGSM (Goodfellow et al., 2015a). We will discuss both aspects in the following.

6.1 Aliasing in pre-trained models

After the application of down-sampling operations in standard CNNs all feature maps suffer from aliasing artifacts occurring due to insufficient sub-sampling.

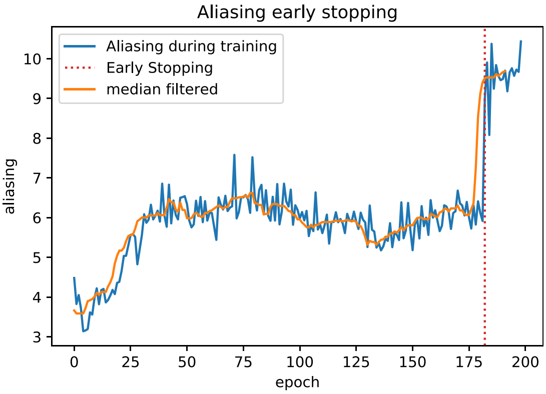
Adversarially trained networks exhibit significantly less aliasing in their feature maps than standard trained networks with the same architecture. As shown in Sect. 4.1 this is valid for different model architectures and training schemes. It raises the question whether models with a low amount of aliasing are necessarily more robust.

We already observed in Sect. 4.1 that low aliasing is especially important in the earlier layers. This can likely be explained by the fact that information is spatially more and more compressed as it is propagated to deeper layers. Therefore, deep layers require sparsity in the feature maps to be expressive. Thus we hypothesize that deeper layers are less vulnerable to aliasing and early layers are more vulnerable. Hence, the difference in aliasing between robust and non-robust models is most visible in the early layers.

6.2 Aliasing and robust overfitting

We provide strong evidence that robust overfitting in FGSM AT is negatively correlated with the amount of aliasing after down-sampling. Whenever a model experiences robust overfitting during training, the amount of aliasing increases significantly, i.e.the increase in aliasing marks the point at which the model loses it's robust generalization ability.

Fig. 14 Aliasing metric in the second layer during training of a PRN-18 with FGSM AT. In epoch 182, the PDG robustness measure as well as the proposed aliasing measure predict the best early stopping point



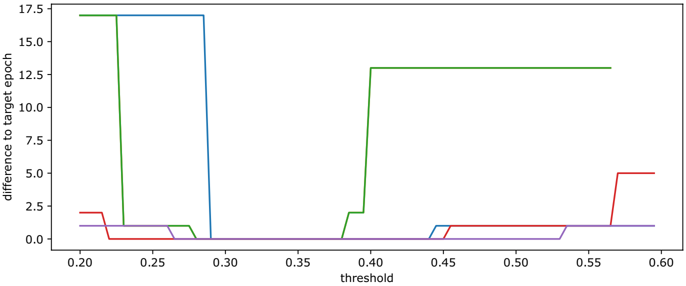


Fig. 15 Evaluation of the threshold for our aliasing metric over five independent training runs. Each line represents the aliasing for one run. The difference of epochs is the L_1 difference between the early stopping epoch and the epoch calculated by the aliasing metric with the corresponding threshold

6.3 Aliasing as early stopping indication

We could show that our aliasing metric can be an indication for the early stopping point, which is needed to prevent robust overfitting on single-step AT schemes like FGSM. Thereby, we choose our hyperparameter, the relative increase in aliasing t at which to stop to be 33%. This threshold is chosen by comparing different trained networks and their aliasing measurements during network training. With this setting we can not guarantee transferability of the approach to different network architectures and datasets.

Yet, the same issue exists for previous approaches where some threshold had to be determined (Wong et al., 2020). In contrast to the explicit robustness evaluation in Wong et al. (2020), the aliasing base indicator does not depend on specific, externally computed perturbations but it can be evaluated during each training iteration on the training batch.

6.4 Spectrum of adversarial perturbations

We could show that adversarial perturbations of robust models dominantly lie in the low frequency spectrum, while the perturbations of non-robust models can lie in the low as well as in the middle or high frequencies. For models that exhibit robust overfitting the generated perturbations dominantly lie in the high frequent spectrum. These findings go in line with Maiya et al. (2021) in such a way that we suggest that the frequency spectrum in which the perturbations can lie are not only depending on the dataset, but also on the specific model architecture as well as the specific training scheme. Further we assume that aliasing is one of the main backdoors which enables adversarial attacks to succeed.

7 Conclusion

Concluding, we provide strong evidence that aliasing and adversarial robustness of CNNs are highly correlated. In particular, we can show that the increase in aliasing is correlated with the decrease of robustness against PGD, when robust overfitting during FGSM AT takes place. Further, we are able to tackle the problem of robust overfitting via early stopping based on our aliasing measurement. We hypothesize that aliasing is one of the main underlying factors that lead to the vulnerability of CNNs. Recent methods to increase model robustness rather heal the symptoms of the underlying problem than investigate its origins. To overcome this challenge we might need to start thinking about CNNs in a more signal processing manner and account for basic principles from this field, like the Nyquist-Shannon theorem, which gives us clear instructions on how to prevent aliasing. Besides downsampling, also padding can lead to unwanted aliasing effects. Still, it is not straightforward to incorporate this knowledge into the architecture and structure of common CNN designs as we have many components to account for. We aim to give a new and more traditional perspective on CNNs to help improve their performance and reliability to enable their application in real world use cases.

Appendix 1

Aliasing during training

Figure 16 shows the amount of aliasing in each layer of the network during baseline training as well as during different AT training schemes.

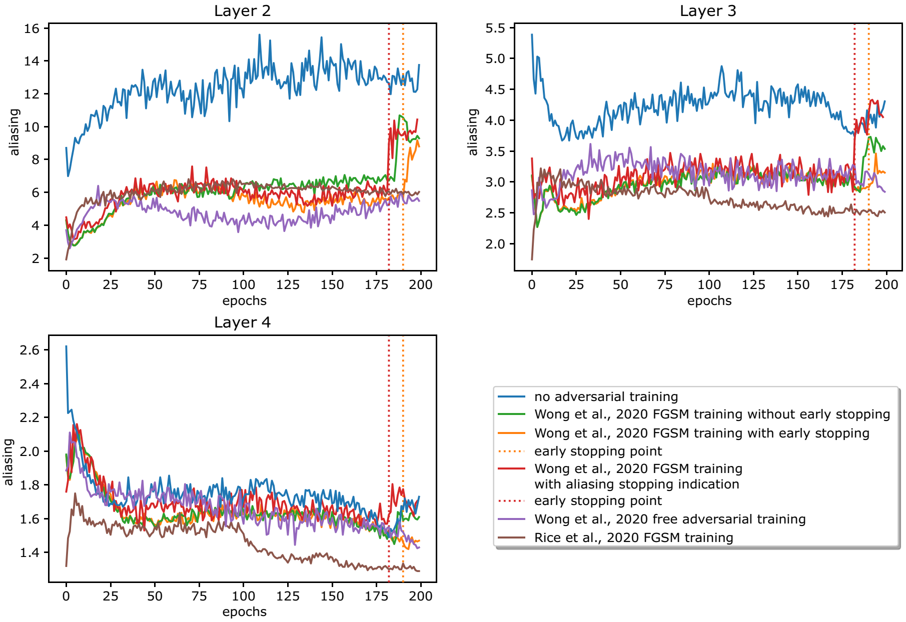


Fig. 16 Amount of aliasing in all layers of PRN-18 during training over 100 random images of CIFAR-10 training set for the baseline and AT training provided by Wong et al. (2020) and Rice et al. (2020). The aliasing measure is the mean value between the aliasing of the down-sampling and the shortcut layer. All networks are trained for 200 epochs, except FGSM training including early stopping. There the training is stopped earlier based on the evaluation of the model on PGD or our aliasing measure. Still, we record the epochs after early stopping and mark the point of early stopping by the dashed lines, to demonstrate the relationship between aliasing and early stopping

Appendix 2

Pooling variation

Additionally to our observation on robust and non-robust networks on CIFAR-10, we conducted a small experiment on MNIST to inspect the influence of different pooling methods. Therefore we trained six small CNNs that all have the same architecture and only differ in the downsampling operation. Either we downsample by using Max- or AvgeragePooling, or we use a convolution with stride two, as it was done for the models provided by RobustBench Croce et al. (2020). We trained each network for 10 epochs with the Adam optimizer and a cycling learning rate and a maximal learning rate of $5e-3$. As criterion we use CrossEntropy and batch size is chosen to be 100. For the adversarial training, we used FGSM adversaries with $\epsilon = 0.3$ and $\alpha = 3.75$.

MaxPooling exhibits the strongest aliasing and also struggles the most with the adversarial attack. While the downsampling via convolution and average pooling is much more able to prevent the attack. AveragePooling can be interpreted as blurring before downsampling, thus helping against aliasing.

Appendix 3

CNN versus FCN

Fully connected neural network classifiers (FCN) as well as CNN-classifiers map the input data into some latent space and finally onto a lower dimensional class label. Thereby, the spatial resolution is usually compressed along the network - not only in the final decision layer. Thereby, dense architectures do not offer an intuitive understanding of the spatial information represented by a hidden variable - nonetheless, spatial compression is implied whenever mapping from a (high dimensional) image space to a semantic label. In contrast, in convolutional networks with systematically defined spatial compression (i.e. sampling), we can systematically measure sampling artifacts (aliasing). For dense networks, this is harder to measure. To shed more light onto the different behavior of convolutions and dense neural networks (FCNs), we conducted a small experiment on the MNIST dataset. We trained a fully connected network without convolutional layers with the same amount of layers, three, and approximate the same number of parameters, 40000, clean and with FGSM adversarial training. The cleanly trained network can achieve a clean accuracy of 97.68% and no robust accuracy (0%). These results are similar to the CNN results (reported in table 2).

Attack structures

Further, we visualize the adversarial examples created on the CNN compared to the FCN shown in Figs. 17 and 18. In Figure 17 we randomly picked six samples from MNIST to investigate into the difference of the perturbations on FCN and CNN. We can see that the

Table 2 Clean and robust accuracy against AutoAttack Croce and Hein (2020) with $\epsilon = 0.3$ for different pooling variations in the same network architecture on MNIST. As well as the amount of aliasing encountered in the downsampling layers

Pooling	Training	Clean Acc ↑	PGD Acc ↑	Aliasing ↓
Convolution	Clean	99.09	0.00	7.18
MaxPooling	Clean	99.39	0.00	27.06
AvgPooling	Clean	99.33	0.00	7.92
Convolution	FGSM	98.94	83.27	5.21
MaxPooling	FGSM	98.20	37.69	12.06
AvgPooling	FGSM	98.82	78.45	4.36

perturbations for the CNNs are more in fine structures organized, like artifacts. While, the adversaries for the FCN are more organized in blocks. Figure 18 shows the mean over all MNIST samples split into the separate classes. While the mean over each class for the perturbations for the FCN are much more centered on the objects/numbers which should be recognized. The mean perturbations for the CNN are much more distributed in the whole image and look much more noisy.

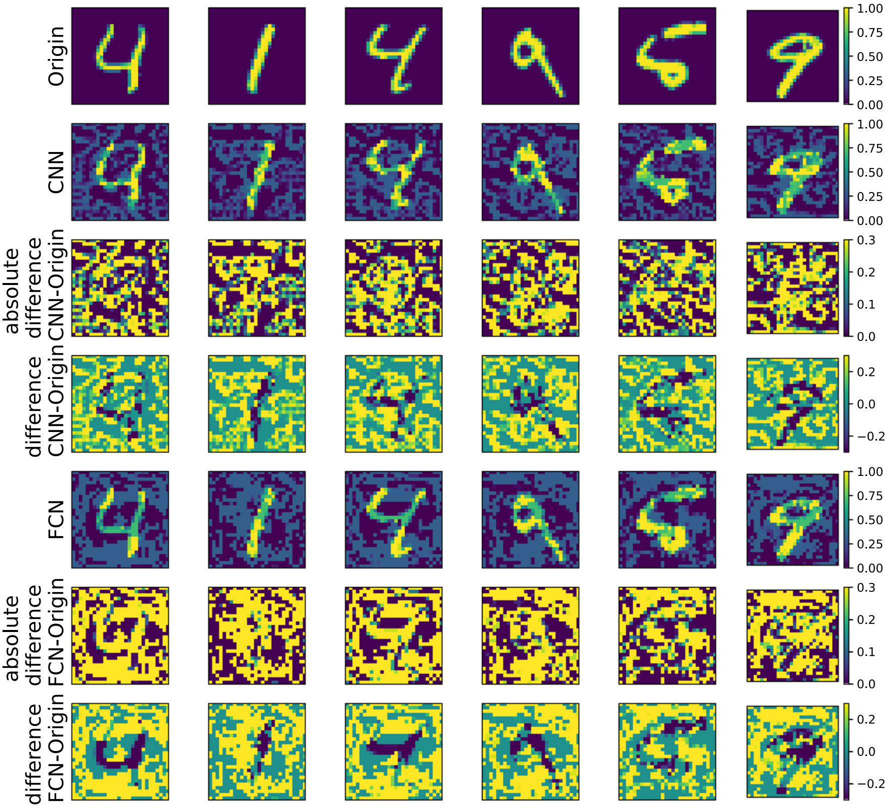


Fig. 17 Comparison of adversarial examples generated for a CNN and FCN. The top row shows the original image without perturbations. The second,third and fourth row show the adversaries generated for the CNN as well as the absolute and normal difference between the original and the adversaries generated for the CNN, the pure perturbations. The bottom three rows show the adversaries generated for the FCN as well as the absolute and normal difference between the original and the adversaries generated for the FCN

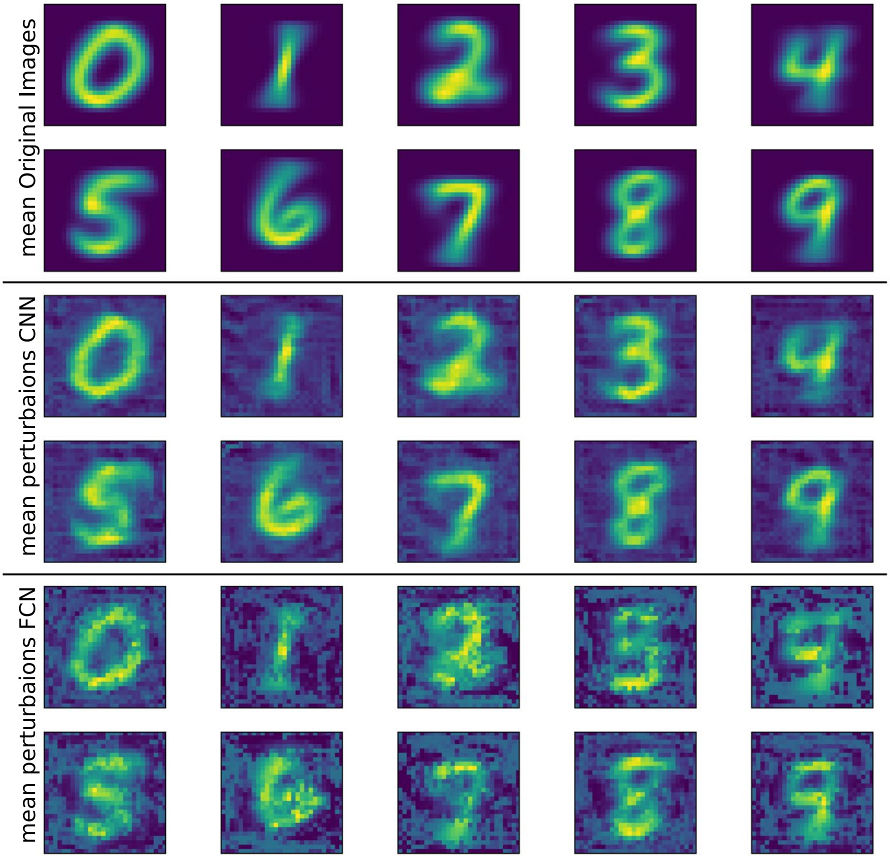
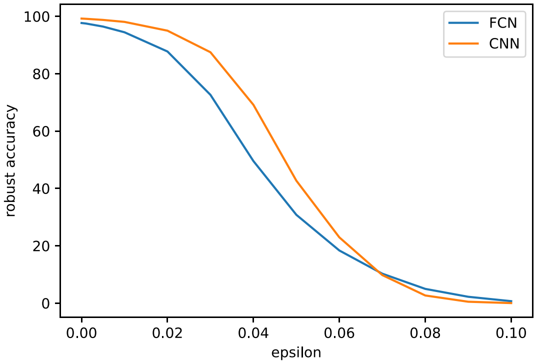


Fig. 18 Mean images over each class on MNIST for the original images as well as the perturbed images created for the FCN and CNN. The top two rows show the clean images, the middle two rows the mean adversarial images on the CNN and the last two rows the mean adversarial images on the FCN. While the numbers are still visible for the mean original images as well as the perturbations on the CNN, the perturbations on the FCN seem much more related to the objects/numbers that should be recognized in the MNIST task

Intrinsic robustness

Finally, we measure the robustness of CNN and FCN with respect to varying epsilons. We compare the standard trained FCN with the CNN. The results are reported in Figure 19. For small epsilons, the CNN is more robust, but after an epsilon of 0.07 this trend switches and the FCN is more robust. Both networks can approximately be completely fooled when $\epsilon > 0.1$.

Fig. 19 Comparison of robust accuracy against AutoAttack Croce and Hein (2020) with varying epsilon values for the FCN and CNN



Author contributions All authors contributed to the manuscript by jointly developing research questions and empirical evaluation setup as well as the analysis of results. Julia Grabinski conducted the experiments.

Funding Open Access funding enabled and organized by Projekt DEAL. No funding was received for conducting this study.

Data availability All data is publicly available through cited prior work.

Code availability We provide source code for the proposed aliasing measure as supplementary material.

Declarations

Conflict of interest The authors are affiliated with the University of Mannheim, University of Siegen, the Max Planck Institute for Informatics, Saarland Informatics Campus, Fraunhofer ITWM, Kaiserslautern and the University of Offenburg.

Ethical approval Some of the experiments have been accepted for presentation as a non-archival short workshop paper at the AAAI workshop on Adversarial Machine Learning.

Consent for publication All authors consent for publication.

References

Azulay, A., & Weiss, Y. (2018). *Why do deep convolutional networks generalize so poorly to small image transformations?*arXiv:1805.12177.

Bernhard, R., Moellic, P.A., & Mermillod, M., et al. (2021). Impact of spatial frequency based constraints on adversarial robustness.

Carlini, N., & Wagner, D. (2017). Towards evaluating the robustness of neural networks.

Carmon, Y., Raghunathan, A., & Schmidt, L., et al. (2019). Unlabeled data improves adversarial robustness.

Chandrasegaran, K., Tran, N.T., & Cheung, N.M. (2021). A closer look at fourier spectrum discrepancies for cnn-generated images detection.

Chen, T., Zhang, Z., & Liu, S., et al. (2021). Robust overfitting may be mitigated by properly learned smoothening. In: *International Conference on Learning Representations*, https://openreview.net/forum?id=qZzy5urZw9

Croce, F., & Hein, M. (2020). Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In: ICML

Croce, F., Andriushchenko, M., & Sehwag, V., et al. (2020). Robustbench: a standardized adversarial robustness benchmark. arXiv preprint arXiv:2010.09670

Durall, R., Keuper, M., & Keuper, J. (2020). Watch your up-convolution: Cnn based generative deep neural networks are failing to reproduce spectral distributions.

Dzanic, T., Shah, K., & Witherden, F. (2020). Fourier spectrum discrepancies in deep network generated images.

Engstrom, L., Ilyas, A., & Salman, H., et al. (2019). Robustness (python library). https://github.com/MadryLab/robustness

Frank, J., Eisenhofer, T., & Schönherr, L., et al. (2020). Leveraging frequency analysis for deep fake image recognition.

Goodfellow, I.J., Shlens, J., & Szegedy, C. (2015). Explaining and harnessing adversarial examples.

Harder, P., Pfreundt, F.J., & Keuper, M., et al. (2021). Spectraldefense: Detecting adversarial attacks on cnns in the fourier domain.

He, K., Zhang, X., & Ren, S., et al. (2016). Identity mappings in deep residual networks.

He, Y., Yu, N., & Keuper, M., et al. (2021). Beyond the spectrum: Detecting deepfakes via re-synthesis. In: *30th International Joint Conference on Artificial Intelligence (IJCAI)*, https://cispa.saarland/group/fritz/wp-content/blogs.dir/13/files/2021/05/ijcai21.pdf

Hendrycks, D., Lee, K., & Mazeika, M. (2019). Using pre-training can improve model robustness and uncertainty.

Hossain, M.T., Teng, S.W., & Sohel, F., et al. (2021). Anti-aliasing deep image classifiers using novel depth adaptive blurring and activation function.

Jung, S., & Keuper, M. (2021). Spectral distribution aware image generation. In: AAAI

Karras, T., Aittala, M., & Laine, S., et al. (2021). Alias-free generative adversarial networks.

Krizhevsky, A. (2012). Learning multiple layers of features from tiny images. University of Toronto

Kurakin, A., Goodfellow, I., & Bengio, S. (2017). Adversarial machine learning at scale.

Li, J., Xie, H., & Li, J., et al. (2021). Frequency-aware discriminative feature learning supervised by single-center loss for face forgery detection.

Li, Q., Shen, L., Guo, S., et al. (2021). Wavecnet: Wavelet integrated cnns to suppress aliasing effect for noise-robust image classification. *IEEE Transactions on Image Processing, 30,* 7074–7089. https://doi.org/10.1109/tip.2021.3101395.

Lohn, A.J. (2020). Downscaling attack and defense: Turning what you see back into what you get.

Lorenz, P., Harder, P., & Straßel, D., et al. (2021). Detecting autoattack perturbations in the frequency domain. In: *ICML 2021 Workshop on Adversarial Machine Learning*, https://openreview.net/forum?id=8uWOTxbwo-Z

Luo, Y., Zhang, Y., & Yan, J., et al. (2021). Generalizing face forgery detection with high-frequency features.

Maiya, S.R., Ehrlich, M., Agarwal, V., et al. (2021). A frequency perspective of adversarial robustness.

Moosavi-Dezfooli, S.M., Fawzi, A., & Frossard, P. (2016). Deepfool: a simple and accurate method to fool deep neural networks

Rice, L., Wong, E., & Kolter, J.Z. (2020). Overfitting in adversarially robust deep learning.

Rony, J., Hafemann, L.G., & Oliveira, L.S., et al. (2019). Decoupling direction and norm for efficient gradient-based l2 adversarial attacks and defenses

Saikia, T., Schmid, C., & Brox, T. (2021). Improving robustness against common corruptions with frequency biased models.

Sehwag, V., Mahloujifar, S., & Handina, T., et al. (2021). Improving adversarial robustness using proxy distributions.

Shannon, C. (1949). Communication in the presence of noise. *Proceedings of the IRE, 37*(1), 10–21. https://doi.org/10.1109/JRPROC.1949.232969.

Szegedy, C., Zaremba, W., & Sutskever, I., et al. (2014). Intriguing properties of neural networks. In: International Conference on Learning Representations, arXiv:1312.6199

Wang, H., Wu, X., & Huang, Z., et al. (2020). High frequency component helps explain the generalization of convolutional neural networks.

Wang, S.Y., Wang, O., & Zhang, R., et al. (2020). Cnn-generated images are surprisingly easy to spot... for now. In: CVPR

Wang, Y., Zou, D., & Yi, J., et al. (2020). Improving adversarial robustness requires revisiting misclassified examples. In: *International Conference on Learning Representations*, https://openreview.net/forum?id=rklOg6EFwS

Wong, E., Rice, L., & Kolter, J.Z. (2020). Fast is better than free: Revisiting adversarial training.

Xiao, Q., Li, K., & Zhang, D., et al. (2017). Wolf in sheep's clothing—the downscaling attack against deep learning applications.

Xu, Q., Zhang, R., & Zhang, Y., et al. (2021). A fourier-based framework for domain generalization.

Yang, Y., & Soatto, S. (2020). Fda: Fourier domain adaptation for semantic segmentation.

Yin, D., Lopes, R.G., & Shlens, J., et al. (2020). A fourier perspective on model robustness in computer vision.

Zagoruyko, S., & Komodakis, N. (2017). Wide residual networks.

Zhang, H., Yu, Y., & Jiao, J., et al. (2019). Theoretically principled trade-off between robustness and accuracy.

Zhang, R. (2019). Making convolutional networks shift-invariant again.

Zou, X., Xiao, F., & Yu, Z., et al. (2020). Delving deeper into anti-aliasing in convnets. In: BMVC