Unifying the Harmonic Analysis of Adversarial Attacks and Robustness

Shishira R Maiya[1]
shishira@umd.edu
Max Ehrlich[1]
maxehr@umd.edu
Vatsal Agarwal[1]
vatsalag@umd.edu
Ser-Nam Lim[2]
sernamlim@meta.com
Tom Goldstein[1]
tomg@umd.edu
Abhinav Shrivastava[1]
abhinav@cs.umd.edu

[1] University of Maryland
[2] Meta

Abstract

Adversarial examples pose a unique challenge for deep learning systems. Despite recent advances in both attacks and defenses, a consensus on their true nature eludes the community. A deep understanding of these can provide new insights towards the development of more effective attacks and defenses. Driven by the common misconception that adversarial examples are high-frequency noise, we present a frequency-based understanding of adversarial examples, supported by theoretical and empirical findings. Our analysis shows that adversarial examples are neither in high-frequency nor in low-frequency components, but are simply dataset dependent. Particularly, we highlight the glaring disparities between models trained on CIFAR-10 and ImageNet-derived datasets. Utilizing this framework, we analyze many intriguing properties of training robust models with frequency constraints, and propose a frequency-based explanation for the commonly observed *accuracy vs robustness* trade-off.

1 Introduction

Since the introduction of adversarial examples by [10], there has been a curiosity in the community around the nature and mechanisms of adversarial vulnerability. There exists an ever-growing body of work focused on attacking neural networks starting with the simple FGSM [2], followed by the advanced PGD [3], a stronger C&W attack [4], the sparser Deep Fool [5] and recently even a parameter free Auto-Attack [6]. These methods and algorithms are consistently countered by the adversarial defense community, starting with distillation-based methods [7], logit-based approaches [8], then moving on to the simple, yet powerful PGD training [3], ensemble-based methods [9] and various other schemes [10, 11]. Despite the immense progress made by the field, there exist many unanswered questions and ambiguities regarding these methods and adversarial examples themselves. Several works [6, 12, 13, 14]

have raised doubts about the efficacy of many methods and have made appeals to the research community to be more vigilant and skeptical with new defenses. Meanwhile, there exists a thriving research corpus dedicated to deeply studying and understanding adversarial examples themselves. [15] presented a feature-based analysis of adversarial examples, while [16] presented preliminary work on PCA-based analysis of adversarial examples, which was followed up with [17] offering a nuanced view of the same through the lens of SVD. [18] proposed to derive insights from the margins of classifiers. Given the intriguing nature of adversarial examples, another way of examining them is through the signal processing perspective of frequencies. [19] first proposed a frequency framework by studying the sensitivity of CNN's for different Fourier bases. [20] then pursued a related direction where they explored the frequency properties of neural networks with respect to additive noise. [21] explore how the frequency properties of the image itself affect the model's outputs and robustness. [22] studied whether convolution operations themselves have an intrinsic frequency bias. [23] came up with the first variant of adversarial attacks which target the low frequencies and [24] strengthened this line of thought by showing that such attacks had a high success rate against adversarially defended models. [25] proposed a method of generating adversarial attacks in the frequency domain itself. Complementary to these, there have been efforts by [26] and [27] in detecting or mitigating adversarial examples by training in the frequency domain. These works also analyzed the nature of adversarial examples under the purview of frequencies and tried to arrive at an explanation for their nature. [28] hypothesized how CNNs exploit high frequency components, leading to less robust models, which is also the primary argument for a class of pre-processing based defenses, e.g., those based on JPEG. [29] also had arguments in support of this conjecture, based on their analysis on CIFAR-10 [30]. It is confounding that these results are at odds with the successful low frequency adversarial attacks by [24] and raises the pertinent question: *What is the true nature of adversarial examples in the frequency domain?* Our work challenges some pre-existing notions about the nature of adversarial examples in the frequency domain and arrives at a more nuanced understanding that is well rooted in theory and backed by extensive empirical observations spanning multiple datasets. We observe that there exists a relation between adversarial robustness and the frequency properties of the particular dataset. This particular observation was also made in concurrent work [31] and we offer additional evidence in this ongoing debate. The authors in [31] use frequency filtering on input images for analysis. Though this is helpful, it does not give us a clear picture of how perturbations in different individual frequencies affect the model. Hence, in our work we instead re-formulate the PGD itself to incorporate frequency masking, enabling us to make interesting observations about frequency based adversarial training as well. Based on these observations, we arrive at a new framework that explains many properties of adversarial examples, through the lens of frequency analysis. Our key contributions can be summarized as follows:

- We show that adversarial examples are neither high frequency nor low frequency phenomena. It is more nuanced than this dichotomous explanation.
- We propose variations of adversarial training by coupling it with frequency-space analysis, leading us to some intriguing properties of adversarial examples.
- We propose a new framework of frequency-based robustness analysis that also helps explain and control the accuracy vs robustness trade-off during adversarial training.

The rest of the paper is organized as follows: we first start off with basic notations and preliminaries. Then we introduce our main findings about adversarial examples in frequency domain and subsequently present a detailed analysis about their properties, complemented by extensive experiments.

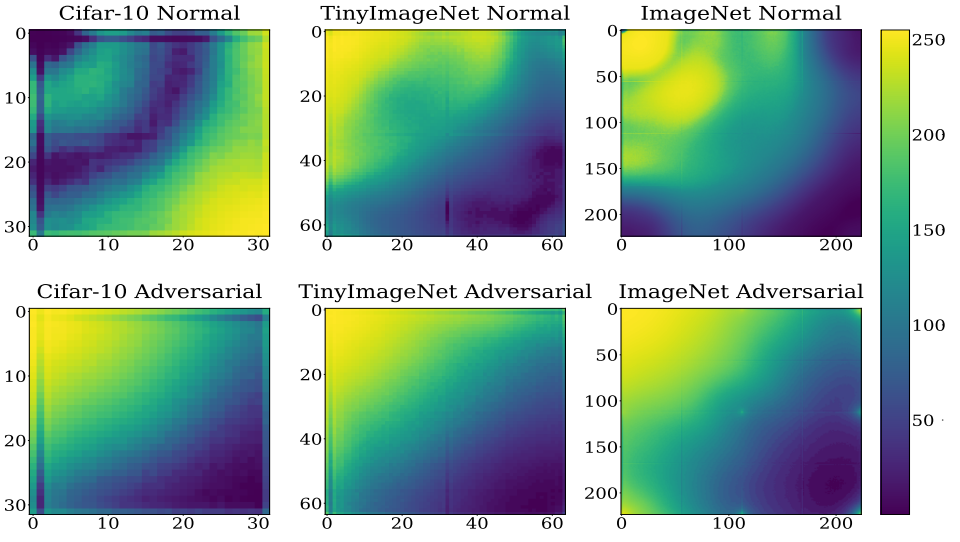

Figure 1: The **DCT of Perturbation Gradients** from ResNet-18 model, averaged across the validation sets, visualized with histogram equalization.

2 Preliminaries

We denote a neural network with parameter θ by $y = h(x;\theta)$, which takes in an input image $x \in \mathbb{R}^{H\times W}$ (omitting the channel dimension for brevity) and outputs $y \in \mathbb{R}^C$ where C is the number of classes. Let D and D^{-1} represent the forward Type-II DCT (Discrete Cosine Transform) [32] and its corresponding inverse. The DCT breaks down the input signal and expresses it as a linear combination of cosine basis functions. Its inverse recovers the input signal from this representation. For a 1-D signal, the k^{th}-freq of $x \in \mathbb{R}^N$ and its corresponding inverse is given by

$$D(x)[k] = g[k] = \sum_{n=0}^{N-1} x_n \lambda_k \cos\frac{(2n+1)k\pi}{2N}, \tag{1}$$

$$D^{-1}(x) = x[n] = \sum_{k=0}^{N-1} g[k] \lambda_k \cos\frac{(2n+1)k\pi}{2N}, \tag{2}$$

$$\text{where } k = \{0,1,\ldots,N-1\} \text{ and } \lambda_k = \begin{cases} \sqrt{\frac{1}{N}} \text{ for } k=0 \\ \sqrt{\frac{2}{N}} \text{ else.} \end{cases} \tag{3}$$

We denote an adversarial attack that is bound by budget ε by

$$\max_{||\delta||_p \le \varepsilon} \mathcal{L}(h(x+\delta;\theta), y) \tag{4}$$

where $\mathcal{L}$ is the loss associated with the network and δ is the adversarial noise bounded under a defined L_p norm to be less than perturbation budget ε. We perform a standard PGD-style update [4] to solve this maximization problem via gradient ascent and for an attack bounded by an L_p norm and step size α, the adversarial noise is given by

$$\delta = \underset{||V||_p \le \alpha}{\arg\max}\, V^T \nabla_x \mathcal{L}(h(x;\theta), y) \tag{5}$$

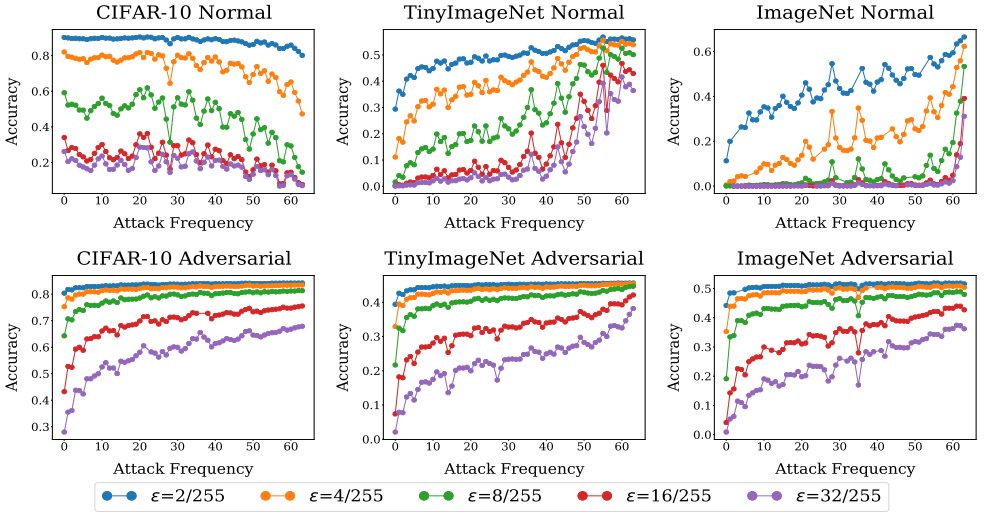

Figure 2: Vulnerability scores (Accuracy under attack) visualized per frequency across datasets. Notice that the trends are reversed from normal training to adversarial training in the case of CIFAR-10.

where V is the direction of steepest normalized descent. Now, to generate an adversarial example that consists of certain frequencies, we restrict its adversarial noise δ to a subspace S defined by $S = \text{Span}\{f_1, f_2, \ldots, f_k\}$, where f_i are orthogonal DCT modes and $k \leq N$,

$$\delta_f = \underset{||V||_p \leq \alpha}{\arg\max}\, V^T D^{-1}(D(\nabla_x \mathcal{L}(h(x;\theta), y)) \odot M) \quad \text{where } M_z(X) \in [0,1] \text{ is the mask} \tag{6}$$

In our work, we consider the L_∞ and L_2 norms, solving for which gives us the update steps:

$$\delta_f = \alpha \cdot \text{Sgn}(D^{-1}(D(\nabla_x \mathcal{L}) \odot M)) \text{ for } L_\infty \text{ and} \tag{7}$$

$$\delta_f = \alpha \cdot D^{-1}\left(D\left(\frac{\nabla_x \mathcal{L}}{||\nabla_x \mathcal{L}||_2}\right) \odot M\right) \text{ for } L_2 \tag{8}$$

$$\hat{x} = x + \delta_f \tag{9}$$

$$\hat{x} = \text{clip}(\hat{x}; -\varepsilon, +\varepsilon) \tag{10}$$

We refer to this method as *DCT-PGD* in the rest of the paper. Since the manual step size selection of standard PGD is not always accurate, leading to discrepancies in robustness measures as illustrated in [26], we provide our results and observations with a DCT version of Auto-Attack as well. Unless mentioned otherwise, we utilize the ResNet-18 architecture for all models. We use the term *adversarial training* to refer to the method by [3] for all models, except for ImageNet models where we use *Adversarial training for free* method [33]. We utilize L_∞ norm with ε of 4/255 for TinyImageNet and ImageNet datasets and ε of 8/255 for CIFAR-10 in all our experiments. Exact training details are included in the Appendix.

3 Nature of Adversarial Samples in Frequency Space

We describe the methods used to analyze frequency response of adversarial examples.

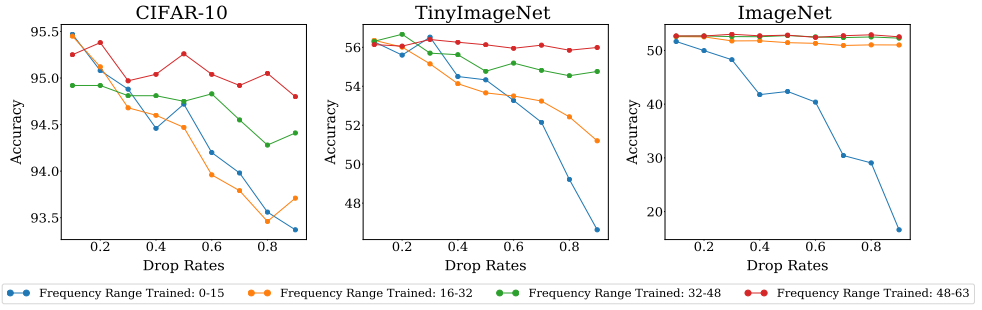

Figure 3: Accuracy of models trained with varying drop rates, for different frequency ranges.

3.1 Perturbation gradients

Measuring the change of output with respect to the input is a fundamental aspect of system design. Whether it is a controls circuit or a mathematical model, the measure $\frac{dy}{dx}$ gives us valuable information about the working of the model. When the model in question is a black box, like a neural network, the measure is invaluable as often it is our only insight into the inner mechanisms of the model. In the case of a classifier, the measure $\frac{dy}{dx}$ is a tensor that is the same size as the input, which tells us about the impact of each pixel in input x on the resulting output y. [34] first applied this concept on neural networks and called them *input gradients*. Over the recent years, this measure and its variants have found a new home in the model interpretability community [35, 36], where it forms the bedrock for various improvements. Taking a cue from this, we propose to measure $\frac{dy}{d\delta}$ or ***Perturbation Gradients***, which inform us about the regions of noise which have maximal impact on the output y. In our work, we are more interested in the frequency properties of adversarial examples, and hence take this one step further and propose to measure the **DCT of Perturbation Gradients**, i.e., $D\left(\frac{dy}{d\delta}\right)$ or $D(\nabla_\delta Y)$. In a sense, we are measuring the model's *reaction* to different frequency components in the adversarial input. This tensor $D(\nabla_\delta Y)$ (which has same shape as input) will point us towards the specific frequencies that affect the output y of the model. To analyze the adversarial frequency properties of a given dataset, we calculate the *Average Perturbation gradients* (over validation set) with respect to the model, under both normal training and adversarial training paradigms. Once computed, it will paint a picture about the interplay of adversarial noise and frequencies.

3.1.1 Analysis of Perturbation Gradients

We define the quantity $D(\nabla_\delta Y)_f$ as the Perturbation gradient at frequency f. Note that this quantity is useful because it differs from $D(\delta)$ by at most a constant multiple, i.e., $D(\delta) \propto D(\nabla_\delta Y)$. Please refer to Appendix sections A1 for the full proofs. We see that the term $D(\nabla_\delta Y)$ corresponds to the frequencies that are affected by adversarial noise. We compute the average DCT of Perturbation gradients over validation sets of TinyImageNet, CIFAR-10, and ImageNet datasets for models with normal and adversarial training under attack from a PGD-based L_∞ adversary. The resulting tensors are visualized in Fig 1. It shows the path taken by the PGD attack in the frequency domain under different scenarios for different datasets. We see that for normally trained CIFAR-10 models, the DCT of Perturbation gradient activations are towards the higher frequencies and they gradually shift towards lower frequencies once the model is adversarially trained. Whereas for TinyImageNet and ImageNet models, we observe that the activations are already in lower-mid frequencies and adversarial training further concentrates them. These results clearly establish the following:

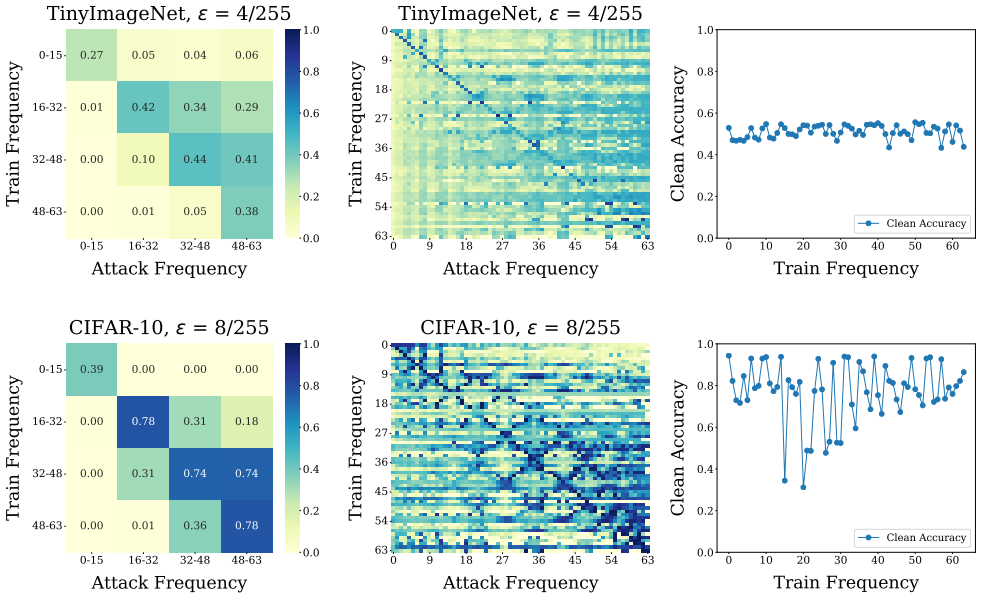

Figure 4: Frequency-based adversarial training across datasets. In the first column we show the results of adversarially training and testing for different frequency ranges. Next, we show results of the same experiments across individual frequencies. The last column shows clean accuracy for each frequency.

- The DCT content of PGD attacks is highly dataset-dependent and it is not possible to make statements about their frequency nature based on training.
- The notion that adversarial training *shifts* the model focus from higher to lower frequencies is not entirely true. In many datasets, the model is already biased towards the lower end of the spectrum even before adversarial training.
- To verify that this phenomenon is attributed to the dataset alone, we also observe similar behaviour across other architectures, across different image sizes and for different attacks like L_2 and Auto-Attack. (Shown in Appendix).

4 Measuring Importance of Frequency Components

To examine the properties and behaviour of adversarial examples in the frequency domain, we also craft various empirical metrics that measure the *importance* of frequency components under various paradigms.

4.1 Importance by Vulnerability

We measure the importance of a frequency component by measuring the attack success rate when an adversarial attack is constrained to frequency f, by quantifying expected vulnerability of each frequency. This amounts to measuring the accuracy of $h(x+\delta_f)$, where δ_f is the adversarial perturbation that is constrained to frequency f, obtained using the aforementioned DCT-PGD method. A lower accuracy of the model for a particular δ_f indicates a more *important* frequency f. In Figure 2, we visualize the accuracy of models with both normal training and adversarial training across different datasets under this setting.We see that only in the case of CIFAR-10, the trends for normal training and adversarial training are reversed, indicating that attacks constrained to higher frequencies are more successful for normal models, while lower frequency attacks are more effective on the adversarially trained models. In TinyImageNet and ImageNet datasets, we see that the overall trend remains same

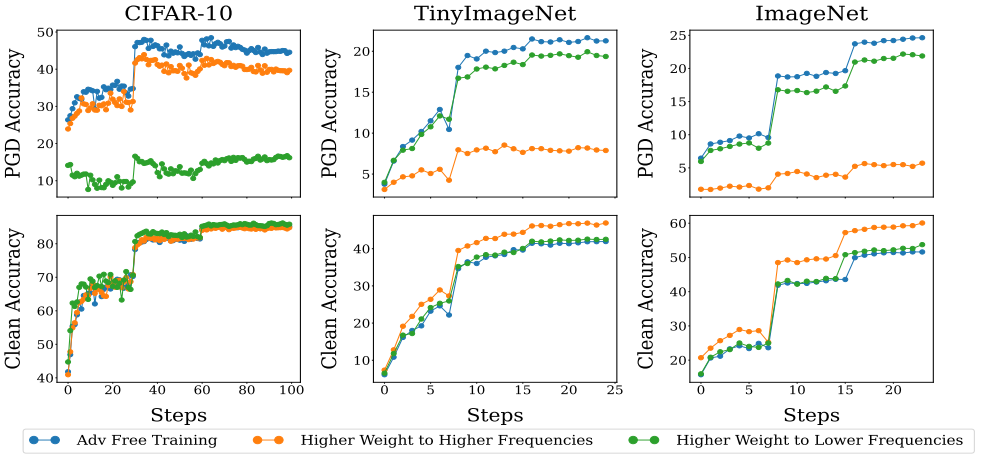

Figure 5: Unequal epsilon distribution: Here we see that models where low frequency perturbations are favoured ends up with higher robustness, but lower clean accuracy.

across the two training paradigms with adversarial training improving robustness across the spectrum. To obtain a high level view, we design another set of experiments where instead of attacking individual frequency components, we restrict the attack to frequency ranges (or bands, set of 16 equal divisions of the spectrum). In their work, [28] had claimed that low frequency perturbations cause visible changes in the image, thus defeating the purpose of *imperceptibility* clause of adversarial examples. However, we find that for larger datasets, such perturbations are imperceptible to a human. Example images have been shown in Appendix.

4.2 Importance during training

To understanding the relative importance of frequency components while training, we formulate an experiment models are trained by masking out frequency components of the input in a probabilistic manner and then using the trained model for normal inference. Example images when certain frequency bands are dropped is shown in appendix. We train four types of models, where the frequency masking is restricted to four equal frequency bands and the amount of masking/dropping is controlled by a parameter p. This translates to training

$$\arg\min_{\theta} \mathcal{L}(h(x_{\hat{f}};\theta),y) \tag{11}$$

$$\text{where } x_{\hat{f}} = D^{-1}(M \odot D(x)) \tag{12}$$

$$M_z = \begin{cases} 1 & z \sim \mathcal{U}_p \wedge z \in [f_1, f_2 ..., f_k] \\ 0 & \text{else} \end{cases} \tag{13}$$

is the Mask generated using p, where $x_{\hat{f}}$ is the input constrained to a particular frequency band within the range $[f_1, f_2, \cdots f_k]$. While training, we select the frequencies to be dropped using a random uniform distribution $\mathcal{U}$, with the percentage of dropping controlled by parameter p. A value of $p = 1$ indicates all frequencies in the specified band are set to zero. We train a total of 36 models per dataset, encompassing 9 different drop rates (p values) and 4 frequency bands. The experiment is repeated across datasets and the results are shown in Figure 3. As expected, we observe that a higher drop rate leads to lower accuracy. We also see that across datasets, high drop rates in low frequency band of 0-15 affects the model

more. This behaviour is expected as lower frequencies have a strong relation with the labels [28] and their extreme dropping leaves the model with little information to learn from. But if we observe the degree to which it affects the performance, we see disparities between the datasets. For example, the model trained on CIFAR-10 experiences a mere ∼2% drop even when 90% of frequencies in the low band (frequencies 0-15) are dropped. Under the same condition, the model on TinyImageNet experiences ∼10% drop and the model on ImageNet experiences a whopping ∼35% drop in accuracy, highlighting the relative importance of these frequency bands. Also, note how very high drop rates in the highest frequency bands (frequencies 48-63) have little to no effect in non CIFAR-10 models.

5 Adversarial Training with frequency perturbations

Till now, we have analyzed the frequency properties of the model across datasets. In all experiments so far, we merely observed how the model *reacts* to adversarial perturbations under various frequency constraints. To further understand the properties of robustness in the frequency domain, we propose to train models with adversarial perturbations restricted to these frequency subspaces, a first of its kind. The training follows

$$\min_{\theta} \max_{||\delta_f||_p \leq \varepsilon} \mathcal{L}(h(x+\delta_f;\theta),y) \tag{14}$$

where δ_f is adversarial noise restricted to a frequency subspace defined by f. To obtain a high-level view of the process, we first train models adversarially with frequencies restricted to four equal frequency bands, ranging from low to high. Predictably, the models perform well when adversarial PGD attack is also restricted to the same frequency bands. The resulting robustness heatmap of attacks across the spectrum is shown in first column of Figure 4. For a more fine-grained view of the same, we adversarially train 64 models for each dataset, by perturbing each individual frequency. Then we adversarially attack these models in every frequency to produce a robustness heatmap, shown in the second column of Figure 4. In their work, [20] had claimed that training with low-frequency perturbations did not help the model to be robust against those frequencies. Their analysis was not based on adversarial perturbations, but their claim was generalized. This effect was not observed in our experiments. We see that the model has good robustness when trained and tested against low-frequency perturbations, across datasets. The diagonals of the robustness heatmaps tell us that models perform well against an adversary constrained to the same frequency used for training. Moreover, we also see that models trained with perturbations restricted to mid/higher frequencies can withstand attacks from a fairly broad range of frequencies compared to models trained with lower frequency perturbations. Now that we have established this new training paradigm, we explore its various nuances and intriguing properties.

5.1 The unequal epsilon distribution

Do all frequencies have the same impact in adversarial training? To answer this question, we modify the construction of adversarial perturbation δ by weighing contributions from different frequency components and manipulating the value of ε they receive. It follows

$$\delta = \sum_{i=0}^{K} \eta_i \cdot \text{sgn}(\nabla_x \mathcal{L})_i \text{ for } L_\infty \text{ norm where } \eta_i = \frac{\varepsilon}{K-i} \tag{15}$$

where K is the number of equal frequency bands (four in our case) and η is a linear scaling parameter. This setting effectively translates to giving more importance to perturbation in

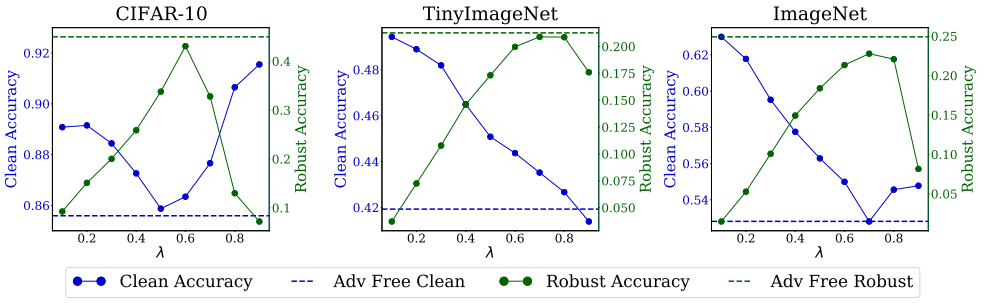

Figure 6: Clean Accuracy vs Robustness across datasets, compared with standard adversarial training for free method. Note that the Y-axis scales are different. Here λ controls the weight of adversarial perturbation towards lower frequencies.

one frequency space over the other. We train 2 models: One as described by equation 15, favoring lower frequency bands and then its complement, by reversing η and favoring higher frequency bands. For these experiments, we employ Free adversarial training by [33]. The plot of PGD and clean accuracy during training are shown in Figure 5. We observe that the model where lower frequencies are favoured acts similar to PGD based training, showing that training across spectrum is not a requirement for robustness. But at the same time, we also observe that models favoring high frequency perturbations show superior clean accuracy in all datasets except CIFAR-10. These results show that - Not all frequencies require the same amount of perturbation while training. We explore this in detail in the next section.

5.2 Accuracy vs Robustness: an alternative perspective

Building on top of previous results, we design an experiment to examine the accuracy vs robustness trade-off that is commonplace while training robust models. We introduce a parameter λ that controls the weight given to frequency components in the perturbation during adversarial training. The update step for PGD under L_∞-norm now looks like:

$$\delta = \lambda \cdot \left[\alpha \cdot \text{sgn}(\nabla_x \mathcal{L}_{\text{LF}})\right] + (1-\lambda) \cdot \left[\alpha \cdot \text{sgn}(\nabla_x \mathcal{L}_{\text{HF}})\right] \tag{16}$$

where $\nabla_x \mathcal{L}_{\text{LF}}$ and $\nabla_x \mathcal{L}_{\text{HF}}$ are gradients restricted to low (frequencies 0-31) and high frequencies (frequencies 32-63) respectively. We adversarially train ten different models by varying the value of λ and show their clean and robust accuracy in Figure 6. We see that in the case of TinyImageNet and ImageNet, the clean accuracy decreases when we train with low frequency perturbations, while increasing robustness. In case of CIFAR-10, we see that there is an initial increase in robustness followed by a steep fall. This is because higher frequencies have a significant role in adversarial robustness for this dataset, which is not achieved when λ values are high. We also observe a steep fall in robustness for ImageNet at λ of 0.9. This is because the frequency importance is distributed in the low-mid range for ImageNet (Figure 1) and very high λ values tend to ignore the 32-48 frequency bands. These results establish that robustness and clean accuracy of an adversarially trained model are dependent on the frequencies we perturb. The λ parameter gives us control over the trade-off, enabling us to be more prudent while designing architectures and training regimes that demand a mix of clean accuracy and robustness. Note that this is different from [10] which introduces a new training strategy, while we analyze and exploit the existing PGD training for gaining a frequency based understanding of the trade-off.

6 Conclusion

In this paper, we analyze adversarial robustness through the perspective of spatial frequencies and show that adversarial examples are not just a high frequency phenomenon, but are in fact dataset-dependent. Then we propose and study the properties of adversarial training using specific frequencies, which can be used to understand the accuracy-robustness trade-off. These results can be utilized to train robust models more quickly by focusing on the frequencies that matter most. We hope that our findings will resolve some misconceptions about the frequency content of adversarial examples and aid in creating more robust architectures.

7 Acknowledgements

This project was partially funded by independent grant from DARPA GARD (HR00112020007) and Facebook AI.

References

[1] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. Intriguing properties of neural networks. *CoRR*, abs/1312.6199, 2014.

[2] I. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. *CoRR*, abs/1412.6572, 2015.

[3] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018.

[4] N. Carlini and D. A. Wagner. Towards evaluating the robustness of neural networks. *CoRR*, abs/1608.04644, 2016.

[5] J. Su, D. V. Vargas, and K. Sakurai. One pixel attack for fooling deep neural networks. *IEEE Transactions on Evolutionary Computation*, 23:828–841, 2019.

[6] F. Croce and M. Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. *ArXiv*, abs/2003.01690, 2020.

[7] N. Papernot, P. Mcdaniel, X. Wu, S. Jha, and A. Swami. Distillation as a defense to adversarial perturbations against deep neural networks. *2016 IEEE Symposium on Security and Privacy (SP)*, pages 582–597, 2016.

[8] H. Kannan, A. Kurakin, and I. Goodfellow. Adversarial logit pairing. *ArXiv*, abs/1803.06373, 2018.

[9] F. Tramèr, A. Kurakin, N. Papernot, D. Boneh, and P. Mcdaniel. Ensemble adversarial training: Attacks and defenses. *ArXiv*, abs/1705.07204, 2018.

[10] H. Zhang, Y. Yu, J. Jiao, E. Xing, L. Ghaoui, and M. I. Jordan. Theoretically principled trade-off between robustness and accuracy. In *ICML*, 2019.

[11] C. Xie, Y. Wu, L. V. D. Maaten, A. Yuille, and K. He. Feature denoising for improving adversarial robustness. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 501–509, 2019.

[12] A. Athalye, N. Carlini, and D. A. Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *ICML*, 2018.

[13] J. Z. Kolter and E. Wong. Provable defenses against adversarial examples via the convex outer adversarial polytope. In *ICML*, 2018.

[14] N. Carlini and D. A. Wagner. Adversarial examples are not easily detected: Bypassing

ten detection methods. *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, 2017.

[15] A. Ilyas, S. Santurkar, D. Tsipras, L. Engstrom, B. Tran, and A. Madry. Adversarial examples are not bugs, they are features. In *NeurIPS*, 2019.

[16] M. Jere, S. Herbig, C. H. Lind, and F. Koushanfar. Principal component properties of adversarial samples. *ArXiv*, abs/1912.03406, 2019.

[17] M. Jere, M. Kumar, and F. Koushanfar. A singular value perspective on model robustness. *ArXiv*, abs/2012.03516, 2020.

[18] G. Ortiz-Jimenez, A. Modas, S.-M. Moosavi-Dezfooli, and P. Frossard. Hold me tight! Influence of discriminative features on deep network boundaries. In *Advances in Neural Information Processing Systems 34*. Dec. 2020.

[19] Y. Tsuzuku and I. Sato. On the structural sensitivity of deep convolutional networks to the directions of fourier basis functions. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 51–60, 2019.

[20] D. Yin, R. G. Lopes, J. Shlens, E. D. Cubuk, and J. Gilmer. A fourier perspective on model robustness in computer vision. *CoRR*, abs/1906.08988, 2019.

[21] A. A. Abello, R. Hirata, and Z. Wang. Dissecting the high-frequency bias in convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 863–871, June 2021.

[22] J. O. Caro, Y. Ju, R. Pyle, S. Dey, W. Brendel, F. Anselmi, and A. Patel. Local convolutions cause an implicit bias towards high frequency adversarial examples, 2021.

[23] C. Guo, J. S. Frank, and K. Q. Weinberger. Low frequency adversarial perturbation. In *UAI*, 2019.

[24] Y. Sharma, G. W. Ding, and M. A. Brubaker. On the effectiveness of low frequency perturbations. *ArXiv*, abs/1903.00073, 2019.

[25] Y. Deng and L. J. Karam. Frequency-tuned universal adversarial attacks. *CoRR*, abs/2003.05549, 2020.

[26] P. Lorenz, P. Harder, D. Strassel, M. Keuper, and J. Keuper. Detecting autoattack perturbations in the frequency domain. 2021.

[27] H. Wang, C. Cornelius, B. Edwards, and J. Martin. Toward few-step adversarial training from a frequency perspective. *Proceedings of the 1st ACM Workshop on Security and Privacy on Artificial Intelligence*, 2020.

[28] H. Wang, X. Wu, Z. Huang, and E. P. Xing. High-frequency component helps explain the generalization of convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[29] Z. Wang, Y. Yang, A. Shrivastava, V. Rawal, and Z. Ding. Towards frequency-based explanation for robust cnn. *ArXiv*, abs/2005.03141, 2020.

[30] A. Krizhevsky. Learning multiple layers of features from tiny images. Technical report, 2009.

[31] R. Bernhard, P. Moëllic, M. Mermillod, Y. Bourrier, R. Cohendet, M. Solinas, and M. Reyboz. Impact of spatial frequency based constraints on adversarial robustness. *CoRR*, abs/2104.12679, 2021.

[32] N. Ahmed, T. Natarajan, and K. Rao. Discrete cosine transform. *IEEE Transactions on Computers*, C-23(1):90–93, 1974.

[33] A. Shafahi, M. Najibi, A. Ghiasi, Z. Xu, J. P. Dickerson, C. Studer, L. Davis, G. Taylor, and T. Goldstein. Adversarial training for free! In *NeurIPS*, 2019.

[34] H. Drucker and Y. Le Cun. Double backpropagation increasing generalization performance. In *IJCNN-91-Seattle International Joint Conference on Neural Networks*, volume ii, pages 145–150 vol.2, 1991.

[35] R. R. Selvaraju, A. Das, R. Vedantam, M. Cogswell, D. Parikh, and D. Batra. Grad-cam: Why did you say that? visual explanations from deep networks via gradient-based localization. *CoRR*, abs/1610.02391, 2016.

[36] H. Wang, M. Du, F. Yang, and Z. Zhang. Score-cam: Improved visual explanations via score-weighted class activation mapping. *CoRR*, abs/1910.01279, 2019.