FrequencyLowCut Pooling
Plug & Play against Catastrophic Overfitting

Julia Grabinski [1,2,3] Steffen Jung [4] Janis Keuper [2,3] Margret Keuper [1,4]

[1]University of Siegen [2]CC-HPC, Fraunhofer ITWM, Kaiserslautern [3]IMLA, Offenburg University [4]Max Planck Institute for Informatics, Saarland Informatics Campus

Introduction

From an image and signal processing point of view, CNNs success might be a bit surprising as the inherent spatial pyramid design of most **CNNs** is apparently **violating basic signal processing laws**, i.e. *Sampling Theorem* in their down-sampling operations. Our recent work [4] in the context of adversarial attacks and distribution shifts showed that there is a strong **correlation** between the **vulnerability of CNNs** and **aliasing artifacts** induced by poor down-sampling operations. This paper builds on these findings and introduces an **aliasing-free down-sampling operation** which can easily be plugged into any CNN architecture: **FrequencyLowCut pooling**. Our experiments show that in combination with simple and Fast Gradient Sign Method (FGSM) adversarial training (AT), our **hyperparameter-free** operator substantially **improves model robustness and avoids catastrophic overfitting**.

Objectives

- We introduce **FrequencyLowCut pooling**, ensuring **aliasing-free down-sampling**.
- Through extensive experiments with various datasets and architectures, we show empirically that **FLC pooling prevents** single-step AT from **catastrophic overfitting**, while this is not the case for other recently published improved pooling operations (e.g. [12]).
- FLC pooling is substantially **faster**, around five times, and easier to integrate than previous AT or defence methods. It provides a **hyperparameter-free plug & play module** for increased model robustness.

Method

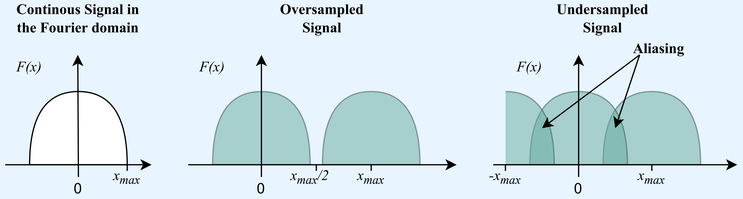

Figure 1. Aliasing is apparent in the frequency domain. Left: The frequency spectrum of a 1D signal with maximal frequency $x_{\max}$. After down-sampling, replica of the signal appear at a distance proportional to the sampling rate. Center: The spectrum after sampling with a sufficiently large sampling rate. Right: The spectrum after under-sampling with aliases due to overlapping replica.

We aim to perfectly remove aliases in CNNs' down-sampling operations without adding additional hyperparameters. Therefore, we directly address the down-sampling operation in the frequency domain, where we can sample according to the Nyquist rate, i.e. remove all frequencies above $\frac{\text{samplingrate}}{2}$ and thus discard aliases.

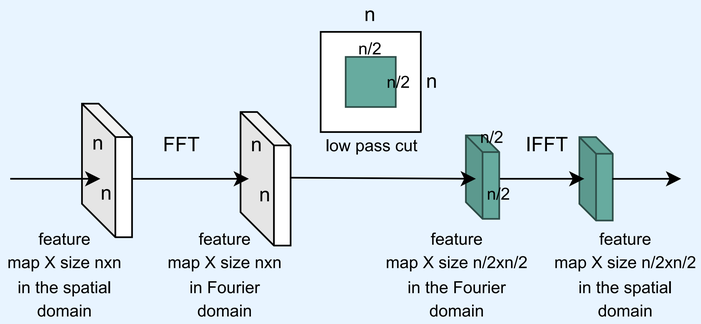

Figure 2. FrequencyLowCut pooling, the proposed, guaranteed alias-free pooling operation. We first transform feature maps into frequency space via FFT, then crop the low frequency components. The result is transformed back into the spatial domain. This corresponds to a sinc-filtered and down-sampled feature map and is fed into the next convolutional layer.

Prediction Results

Table 1. FGSM AT of PRN-18 and Wide-ResNet-28-10 (WRN-28-10) architectures on CIFAR-10. Comparison of clean and robust accuracy (high is better) against PGD [5] and AutoAttack [2] on the full dataset with L_{inf} with $\epsilon = 8/255$ and L_2 with $\epsilon = 0.5$. FGSM test accuracies indicate catastrophic overfitting on the AT data, hence this column is set to gray.

Method	Clean	FGSM $\epsilon=\frac{8}{255}$	PGD L_{inf} $\epsilon=\frac{8}{255}$	AA L_{inf} $\epsilon=\frac{8}{255}$	AA L_2 $\epsilon=0.5$	AA L_{inf} $\epsilon=\frac{1}{255}$
Preact-ResNet-18						
Baseline: FGSM training	**90.81**	90.37	0.16	0.00	0.01	53.10
Baseline & early stopping	82.88	61.71	11.82	3.76	17.44	72.95
BlurPooling [12]	86.24	78.36	1.33	0.06	1.96	66.88
Adaptive BlurPooling [13]	90.35	77.39	0.23	0.00	0.07	39.00
Wavelet Pooling [6]	85.02	64.16	12.13	5.92	19.65	10.08
FLC Pooling (ours)	84.81	58.25	**38.41**	**36.69**	**55.58**	**80.63**
WRN-28-10						
Baseline: FGSM training	86.67	83.64	1.64	0.09	1.47	59.39
Baseline & early stopping	82.29	56.36	31.26	28.54	46.03	76.87
Blurpooling [12]	91.40	89.44	0.22	0.00	0.00	38.45
Adaptive BlurPooling [13]	91.10	89.76	0.00	0.00	0.00	7.42
Wavelet Pooling [6]	**92.19**	90.85	0.00	0.00	0.00	10.08
FLC Pooling (ours)	84.93	53.81	**39.48**	**38.37**	**52.89**	**80.27**

Table 2. Comparison of ResNet-50 models clean and robust accuracy against AutoAttack [2] on ImageNet. We compare against models reported on RobustBench [1].

Method	Clean	PGD L_{inf} $\epsilon=\frac{4}{255}$
Standard [1]	76.52	0.00
FGSM & FLC Pooling (ours)	63.52	27.29
Wong et al., 2020 [9]	55.62	26.24
Robustness lib, 2019 [3]	62.56	29.22
Salman et al., 2020 [7]	64.02	34.96

Compared to other down sampling methods, FLC Pooling is able to obtain state-of-the-art robust accuracy.

Time Consumption Results

In terms of efficiency, FLC pooling is able to provide a good trade-off between training time and model performance. Results show that models with comparable accuracy, generally need several factors of training time.

Table 3. Runtime of AT in seconds per epoch over 200 epochs and a batch size of 512 trained with a PRN-18 for training on the original CIFAR-10 dataset without additional data. Experiments are executed on one Nvidia Tesla V100. Evaluation for clean and robust accuracy, higher is better, on AutoAttack [2] with our trained models. The models reported by the original authors may have different numbers due to different hyperparameter selection. The top row reports the baseline without AT.

Method	Seconds per epoch (avg)	Clean Acc	AA Acc
Baseline	14.6 ± 0.1	95.08	0.00
FGSM & early stopping [9]	$\mathbf{27.3 \pm 0.1}$	82.88	11.82
FGSM & FLC Pooling (Ours)	34.5 ± 0.1	**84.81**	38.41
PGD [5]	115.4 ± 0.2	83.11	40.35
Robustness lib [3]	117 ± 19.0	76.37	32.10
AWP [10]	179.4 ± 0.4	82.61	**49.43**
MART [8]	180.4 ± 0.8	55.49	8.63
TRADES [11]	219.4 ± 0.5	81.49	46.91

Visual Results

In Figure 3, we visualize AutoAttack adversarial attacks. Perturbations created for the baseline trained with FGSM differ substantially from those created for FLC pooling trained with FGSM. While perturbations for the baseline model exhibit high frequency structures, attacks to FLC pooling rather affect the global image structure.

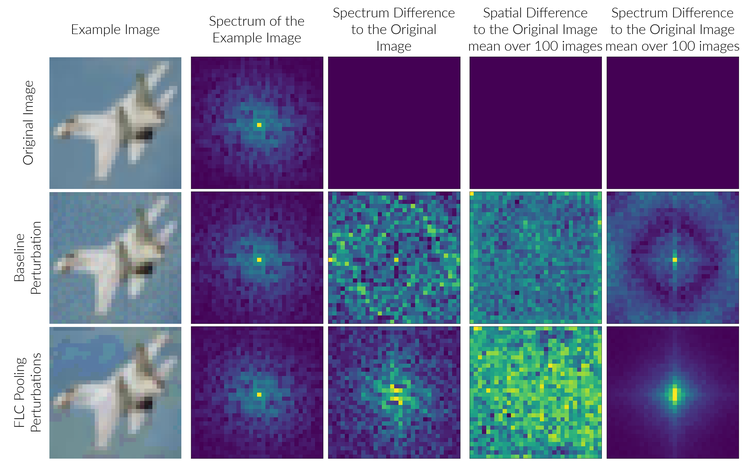

Figure 3. Spatial and spectral differences of adversarial perturbations created by AutoAttack with $\epsilon = \frac{8}{255}$ on the baseline model as well as our FLC Pooling. On the left side for one specific example of an airplane and on the right side the average difference over 100 images.

Conclusion

We developed a fully **aliasing-free down-sampling layer** that can be plugged into any down-sampling operation. Previous attempts in this direction are based on blurring before down-sampling. However, those can only reduce aliasing but can not eliminate it. With **FLC pooling** we developed a **hyperparameter-free** and **easy plug & play down-sampling** which supports CNNs native robustness. Thereby, we can **overcome the issue of catastrophic overfitting** in single-step AT and provide a path to reliable and fast adversarial robustness. We hope that FLC pooling will be used to evolve to fundamentally improved CNNs which do not need to account for aliasing effects anymore.

References

[1] Francesco Croce, Maksym Andriushchenko, Vikash Sehwag, Nicolas Flammarion, Mung Chiang, Prateek Mittal, and Matthias Hein. Robustbench: a standardized adversarial robustness benchmark. *arXiv preprint arXiv:2010.09670*, 2020.

[2] Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *ICML*, 2020.

[3] Logan Engstrom, Andrew Ilyas, Hadi Salman, Shibani Santurkar, and Dimitris Tsipras. Robustness (python library), 2019.

[4] Julia Grabinski, Janis Keuper, and Margret Keuper. Aliasing coincides with CNNs vulnerability towards adversarial attacks. In *The AAAI-22 Workshop on Adversarial Machine Learning and Beyond*, 2022.

[5] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial machine learning at scale, 2017.

[6] Qiufu Li, Linlin Shen, Sheng Guo, and Zhihui Lai. Wavelet integrated cnns for noise-robust image classification, 2020.

[7] Hadi Salman, Andrew Ilyas, Logan Engstrom, Ashish Kapoor, and Aleksander Madry. Do adversarially robust imagenet models transfer better? *Advances in Neural Information Processing Systems*, 33:3533–3545, 2020.

[8] Yisen Wang, Difan Zou, Jinfeng Yi, James Bailey, Xingjun Ma, and Quanquan Gu. Improving adversarial robustness requires revisiting misclassified examples. In *International Conference on Learning Representations*, 2020.

[9] Eric Wong, Leslie Rice, and J. Zico Kolter. Fast is better than free: Revisiting adversarial training, 2020.

[10] Dongxian Wu, Shu-Tao Xia, and Yisen Wang. Adversarial weight perturbation helps robust generalization. *Advances in Neural Information Processing Systems*, 33:2958–2969, 2020.

[11] Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric P. Xing, Laurent El Ghaoui, and Michael I. Jordan. Theoretically principled trade-off between robustness and accuracy, 2019.

[12] Richard Zhang. Making convolutional networks shift-invariant again, 2019.

[13] Xueyan Zou, Fanyi Xiao, Zhiding Yu, and Yong Jae Lee. Delving deeper into anti-aliasing in convnets. In *BMVC*, 2020.

GitHub