2023 MC REU PROJECT PROPOSAL: FOURIER NEURAL BLOCK AS A GENERIC VISION BACKBONE

Jonathan Cui, David A. Araujo & Md Faisal Kabir [*]
The Pennsylvania State University – Harrisburg
Middletown, PA 17057, USA
{jpc6988,daa5724,mpk5904}@psu.edu

ABSTRACT

Background: Keeping pace with the Moore's Law, there has been a recent surge in the investigation of using Transformers and Multi-Layer Perceptrons (MLPs) for generic vision tasks. However, state-of-the-art MLP models still lag behind Vision Transformers (ViTs) and Convolutional Neural Networks (CNNs) in terms of performance, which researchers postulate to be a consequence of the lack of cross-token modeling as compared with Multi-Head Self-Attention in ViTs.

Method: We propose to investigate a novel generic vision backbone architecture: Fourier Neural Block, which learns mappings between feature spaces in the time and the frequency domains simultaneously.

Goal: The goal of the project is to establish a robust vision MLP with the Fourier Neural Block in image classification/image generation.

Significance: Research in this area will provide deeper insight as to identifying the key shortcoming of vanilla vision MLP architectures and to shed light on the mathematical dynamics of MLP training.

1 INTRODUCTION

The recent success of Vision Transformers (ViTs) has raise questions about the widely held notion that visual inductive biases such as locality, weight sharing, and translation equivariance are indispensable to robust performance of neural networks. However, as evidenced by the paradigm shift from CNNs to ViTs in computer vision, explicit architectural biases are *not* necessary to training a robust, high-performance neural network.

Up to date, three main categories of NN architecture dominate the field of Computer Vision: CNNs, ViTs, and Vision MLPs. The two former has been extensively studied, while Vision MLPs have yet to gain traction, largely because of their suboptimal performance compared with other architectural choice. In this work, we expect to push the boundary of Vision MLPs and establish new baseline for future studies in this area.

2 RELATED WORKS

2.1 VISION MLPS

Rosenblatt (1958) first proposed the Perceptron as a mathematical model for neural activities, which failed to gain traction due to its linearity and thus inability to approximate complex functions. However, later works (Rumelhart et al., 1985) demonstrated the vast potential of MLPs when stacked in multiple layers with activation functions, and researchers were able to prove its ability for universal approximation (Hornik et al., 1989).

Subsequently, they have been largely used as general function approximators between vector spaces. Nonetheless, the development of CNNs (Krizhevsky et al., 2017) quickly rendered fully-connected

[*] Primary advisor.

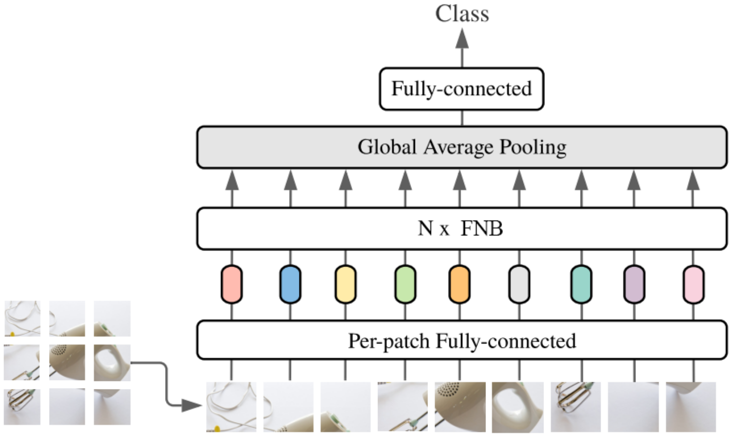

Figure 1: The proposed architecture for the Fourier Neural Block for image classification (adapted from Tolstikhin et al. (2021)).

MLPs obsolete for the following reasons: (I) the number of parameters is $O(n^2)$ for plain Vision MLPs but only $O(1)$ for CNNs, where n is the size of the input image; (II) the CNN architecture explicitly incorporates image-specific inductive biases (e.g., weight sharing and translational invariance) (Raghu et al., 2021).

Beginning in the 2020s, the exponential increase in the computational resources and training data available has enabled researchers to perform experiments with various neural architectures at ever larger scales. Empirical evidence converges to the conclusion that adequate training data and compute are much more central to robust network performance than image-specific inductive biases, as supported by recent state-of-the-art models like ViTs (Dosovitskiy et al., 2021) and MLP-Mixers (Tolstikhin et al., 2021).

2.2 Fourier Transforms in Neural Networks

Fourier Transforms are fundamental to the development of deep learning: they appear in a proof for the universal approximation theorem (Hornik et al., 1989), and they are also used to accelerate the computation of discrete convolutions (Mathieu et al., 2013). The many desirable mathematical properties of Fourier transforms also make it a natural choice to incorporate DFTs into neural architectures (Bengio et al., 2007; Sitzmann et al., 2020). They have also emerged in the latest development in neural PDE solvers, directly learning neural operators between infinite-dimensional Fourier function spaces (Li et al., 2020).

3 Methodology

Notation. Vectors shall be denoted with italicized lowercase letters in bold, e.g., $\boldsymbol{x}$. Matrices shall be denoted with italicized uppercase letters, e.g., X. Counting from 0, the ith entry of a vector $\boldsymbol{x}$ shall be denoted as x_i. Note that the index variable i may be distinguished from the imaginary unit $\mathrm{i} = \sqrt{-1}$ from italicization.

3.1 Patch Embedding

We adopt a patch embedding setup as implemented in ViT (Dosovitskiy et al., 2021) and (Tolstikhin et al., 2021), c.f. Fig. 1.

3.2 Fourier Neural Block

A Fourier Neural Block takes in an input feature representation $X \in \mathbb{R}^{N \times D}$, where N is the number of tokens and D is the token embedding dimension. A token-mixing MLP is first applied to X, as in Tolstikhin et al. (2021):

$$U \coloneqq X + W_2 \cdot \sigma(W_1 \cdot \mathrm{Norm}_1(X) + \boldsymbol{b}_1) + \boldsymbol{b}_2, \tag{1}$$

where $\sigma \colon \tilde{\mathbb{R}} \to \tilde{\mathbb{R}}$ is the GELU activation function (Hendrycks & Gimpel, 2016),[1] Norm_1 is a LayerNorm layer (Ba et al., 2016), and $\boldsymbol{b}_{1,2}$ are "broadcasted" along the channel (column) axis.

Then, a Discrete Fourier Transform (DFT) is applied simultaneously for each token (row transposed) $\boldsymbol{u}^{(i)} \in \mathbb{R}^D$ of U ($i \in \{0, \cdots, N-1\}$):[2]

$$\tilde{u}_n^{(i)} = \sum_{k=0}^{D-1} \tilde{u}_k^{(i)} \cdot \mathrm{e}^{-2\pi \mathrm{i} nk/D}.$$

Since the input signal $\boldsymbol{u}^{(i)}$ is real, its DFT $\tilde{\boldsymbol{u}}^{(i)}$ will be Hermitian-symmetric; i.e., $\tilde{u}_n^{(i)} \equiv \overline{\tilde{u}_{D-n}^{(i)}}$, where the bar denotes complex conjugation. We shall choose an even value for D for the sake of consistency between the FFT and the inverse FFT. We now truncate all entries in $\tilde{\boldsymbol{u}}^{(i)}$ after $D/2$, and concatenate the real and the imaginary parts of each entry, which yields the token-wise Fourier feature map $\tilde{\boldsymbol{u}}'^{(i)}$:

$$\tilde{u}_n'^{(i)} = \begin{cases} \mathrm{Re}\, \tilde{u}_n^{(i)}, & n = 0, 1, \cdots, D/2, \\ \mathrm{Im}\, \tilde{u}_{n-D/2}^{(i)}, & n = D/2 + 1, \cdots, D + 1. \end{cases}$$

Stacking $\left(\tilde{\boldsymbol{u}}'^{(i)}\right)^\top$, where $i = 0, 1, \cdots, n-1$, will yield the full Fourier feature map $\tilde{U} \in \mathbb{R}^{N \times D'}$, where $D' = D + 2$.

We now apply a channel-mixing MLP on U and $\tilde{U}$ simultaneously:

$$Y \coloneqq U + \sigma(\mathrm{Norm}_2(U) \cdot W_3 + \boldsymbol{b}_3) \cdot W_4 + \boldsymbol{b}_4 + \mathrm{IFFT}\{\sigma(\mathrm{Norm}_2(\tilde{U}) \cdot W_5 + \boldsymbol{b}_5) \cdot W_6 + \boldsymbol{b}_6\}, \tag{2}$$

where $\mathrm{Norm}_{2,3}$ represents LayerNorm layers (Ba et al., 2016) and $b_{3,4,5,6}$ are "broadcasted" appropriately along the token (row) axis. The corresponding IFFT is obtained by first reconstructing the full frequency domain from only $\tilde{U}$ (above the Nyquist frequency) and then performing the IFFT for each token (row), defined as

$$\mathrm{IFFT}\{\tilde{\boldsymbol{x}}\} \coloneqq \frac{1}{N} \sum_{k=0}^{N-1} \tilde{x}_k \cdot \mathrm{e}^{2\pi \mathrm{i} nk/N}.$$

The Fourier Neural Block then outputs $Y \in \mathbb{R}^{N \times D}$, which has the same shape as the input $X \in \mathbb{R}^{N \times D}$.

3.3 Model Construction

We will perform preliminary experiments on the CIFAR-10 dataset (Krizhevsky et al., 2009b), which contains a total of 60,000 RGB images of size 32×32. We will use the train–test split provided by the authors. We begin our experiments by modifying a publicly available code source (Balsam, 2023). A preliminary experiment of the CNN baseline yields a top-1 test accuracy of $.9399 \pm 1.509 \cdot 10^{-6}$.

4 Projected Timeline

Week 1

Jonathan C.: Drafting the PyTorch code base for image classification (data processing, image masking, data augmentation, and initial neural network implementation). Our code for image classification will be a modification from this repository on GitHub.

David A. A.: Data downloading, cleaning, and preprocessing. Datasets to use include CIFAR-10 (Krizhevsky et al., 2009a), ImageNet (Krizhevsky et al., 2017), FFHQ-256 (Karras et al., 2019), and MS-COCO (Lin et al., 2014).

[1] Here, $\tilde{\mathbb{R}} \coloneqq \bigcup_{n \in \mathbb{N}} \mathbb{R}^n$ is the union of vector spaces $\mathbb{R}^n$.

[2] An implementation of the Fast Fourier Transform (FFT) algorithm is used in our code.

WEEK 2

Jonathan C.: Drafting the PyTorch code base for image generation (data processing, data augmentation, and unconditional GAN implementation).

David A. A.: Environment setup across computing nodes (setting up cloud computing server, install required dependency packages, establishing version control, and creating Dockerfile/Docker image for training container).

WEEK 3

Jonathan C.: Completing the PyTorch code base for image generation (cGAN implementation, GAN training, and evaluation metrics).

David A. A.: Running preliminary ablation experiments on CIFAR-10 (CNN & ViT baselines and grid search on F-MLP).

WEEK 4

Jonathan C.: Conducting the main experiments for image classification (against EfficientNet, ViT, DeiT, MLP-Mixer, and gMLP) and drafting the Methodology section of the manuscript.

David A. A.: Performing a literature review of CNNs (EfficientNet & ResNeXt), Vision Transformers (ViT, DiT & DeiT), and Vision MLPs (MLP-Mixer & gMLP) and draft the Related Works section of the manuscript.

WEEK 5

Jonathan C.: Completing ablation studies for GANs/LDMs with F-MLPs (across the PSNR, LSIM, and FID metrics), beginning the Analysis/Discussion section of the manuscript, and deriving the mathematical relationship between F-MLPs and CNNs.

David A. A.: Finishing the statistical analyses of the results obtained from experiments for image classification.

WEEK 6

Jonathan C.: Conducting the main experiments for image generation (against StyleGAN2, StyleGAN-T, StyleNAT, and DeiT) and continuing the Analysis/Discussion section.

David A. A.: Beginning the Results section of the manuscript for image classification and running extra experiments ad hoc if necessary.

WEEK 7

Jonathan C.: Completing the abstract and the introduction sections of the manuscript, creating a code base for an online demo (like this one), and testing deployment with Penn State Harrisburg's SUN Lab Computers.

David A. A.: Creating a code base for demo, completing result analysis for image generation, and finishing the Results section of the manuscript.

WEEK 8

Jonathan C.: Preparing for paper submission and journal- or conference-specific formatting instructions.

David A. A.: Revising the final manuscript and preparing code base for open-source release.

References

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

Tysam Balsam. hlb-CIFAR10, 2 2023. URL `https://github.com/tysam-code/hlb-CIFAR10`.

Yoshua Bengio, Yann LeCun, et al. Scaling learning algorithms towards ai. *Large-scale kernel machines*, 34(5):1–41, 2007.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. URL `https://openreview.net/forum?id=YicbFdNTTy`.

Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366, 1989.

Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4401–4410, 2019.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009a.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009b.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017.

Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. *arXiv preprint arXiv:2010.08895*, 2020.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pp. 740–755. Springer, 2014.

Michael Mathieu, Mikael Henaff, and Yann LeCun. Fast training of convolutional networks through ffts. *arXiv preprint arXiv:1312.5851*, 2013.

Maithra Raghu, Thomas Unterthiner, Simon Kornblith, Chiyuan Zhang, and Alexey Dosovitskiy. Do vision transformers see like convolutional neural networks? *Advances in Neural Information Processing Systems*, 34:12116–12128, 2021.

Frank Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386, 1958.

David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning internal representations by error propagation. Technical report, California Univ San Diego La Jolla Inst for Cognitive Science, 1985.

Vincent Sitzmann, Julien Martel, Alexander Bergman, David Lindell, and Gordon Wetzstein. Implicit neural representations with periodic activation functions. *Advances in Neural Information Processing Systems*, 33:7462–7473, 2020.

Ilya O Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, et al. Mlp-mixer: An all-mlp architecture for vision. *Advances in neural information processing systems*, 34:24261–24272, 2021.