Global Filter Networks for Image Classification

Yongming Rao, Wenliang Zhao, Zheng Zhu, Jiwen Lu, Jie Zhou

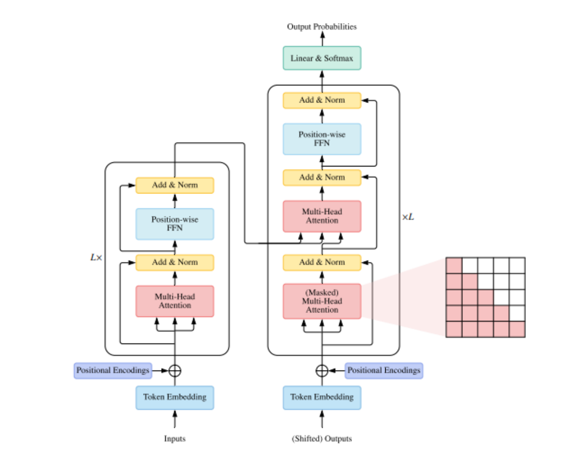
Coby Penso

What we'll see

1. Introduction
2. Preliminaries
3. Global Filter Networks
4. Experiments

Introduction - Transformers

- The Transformer architecture - originally designed for the natural language processing (NLP) tasks, has shown promising performance on various vision problems recently.



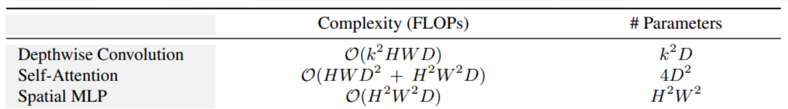
Introduction - Vision Transformers

Recently, there are a large number of works which aim to improve the transformers.

- Better training strategies
- Better architectures (Complexity and Number of Parameters)

Several layer types used:

- Self-Attention
- Depthwise Convolution
- Spatial MLP



	Complexity (FLOPs)	# Parameters
Depthwise Convolution	$\mathcal{O}(k^2HWD)$	k^2D
Self-Attention	$\mathcal{O}(HWD^2 + H^2W^2D)$	$4D^2$
Spatial MLP	$\mathcal{O}(H^2W^2D)$	H^2W^2

H,W, D - input dim. k - kernel size.

Introduction - MLP-like models

Several works propused to use MLP to replace self-attention laye Two drawbacks:

- like SA, Spatial MLP requires computational complexity quadratic to the length of tokens.
- MLP models are hard to scale up to higher resolution since the weights of the spatial MLPs have fixed sizes.

In this paper, GFNet enjoys log-linear complexity and can be easily scaled up to any resolution

Discrete Fourier Transform - 1D DFT

- **Definition:** Given a sequence of N complex numbers $x[n], 0 \leq n \leq N-1$, the 1D DFT defined as:

  $$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j(2\pi/N)kn} = \sum_{n=0}^{N-1} x[n] W_N^{kn}$$

  where j is the imaginary unit and $W_N = e^{-j(2\pi/N)}$
- **Observation:** $X[k]$ repeats on intervals of length N, thus sufficient to take $X[k], k = 0, 1, ..., N-1$
- **Invertible:** DFT is one-to-one transformation. Given DFT $X[k]$, we can recover $x[n]$ by IDFT:

  $$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] e^{j(2\pi/N)kn}$$

Discrete Fourier Transform - 2D DFT

1D DFT can be extended to 2D signals.

- **Definition:**
  Given the 2D signal $x[m,n], 0 \leq m \leq M-1, 0 \leq n \leq N-1$, the 2D DFT defined as:

$$X[u,v] = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x[m,n] e^{-j2\pi\left(\frac{um}{M} + \frac{vn}{N}\right)}$$

- **Property:**
  For real input $x[n]$ or $x[m,n]$, DFT is conjugate symmetric

$$X[N-k] = X^*[k]$$

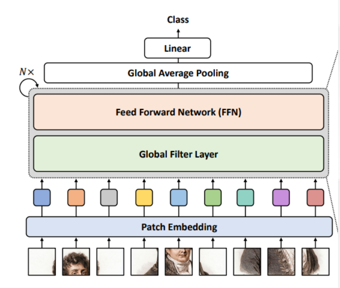
$$X[M-u, N-v] = X^*[u,v]$$

Global Filter Networks - Overall architecture

Motivation: Recent advances in vision transformers demonstrate that models based on self-attention can achieve competitive performance even without the inductive biases associated with the convolutions.

Overall architecture:

- Input HxW of non-overlapping patches and projects the flattened patches into L = HW tokens with dimension D.
- Globa Filter Layer (GFL)
- A Feed Forward network (FFN)



Global Filter Networks - Global filter layer

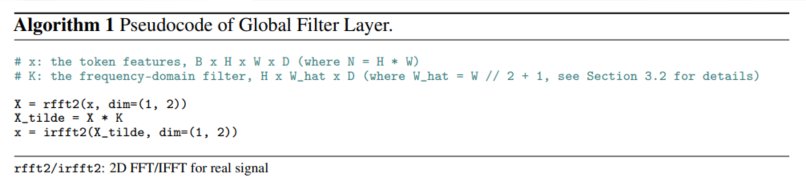
Algorithm:

- Given $x \in \mathbb{R}^{HxWxD}$
- First, perform 2D FFT - $X = \mathcal{F}[x] \in \mathbb{C}^{HxWxD}$
- Modulate the spectrum by multiplying with a learnable filter $K \in \mathbb{C}^{HxWxD}$

$$\tilde{X} = K \odot X,$$

- The filter K is called the global filter - can represent an arbitrary filter in the frequency domain.
- Finally, apply Inverse FFT

$$x \leftarrow \mathcal{F}^{-1}[\tilde{X}]$$



Algorithm 1 Pseudocode of Global Filter Layer.

```
# x: the token features, B x H x W x D (where N = H * W)
# K: the frequency-domain filter, H x W_hat x D (where W_hat = W // 2 + 1, see Section 3.2 for details)

X = rfft2(x, dim=(1, 2))
X_tilde = X * K
x = irfft2(X_tilde, dim=(1, 2))
```

`rfft2`/`irfft2`: 2D FFT/IFFT for real signal

Global Filter Networks - Global filter layer

Observation:
The global filter layer is equivalent to a depthwise global circular convolution with the filter size H × W. Therefore, the global filter layer is different from the standard convolutional layer.

Complexity:
$\mathcal{O}(DLlogL)$ vs the vanilla depthwise global circular convolution in the spatial domain $\mathcal{O}(DL^2)$.

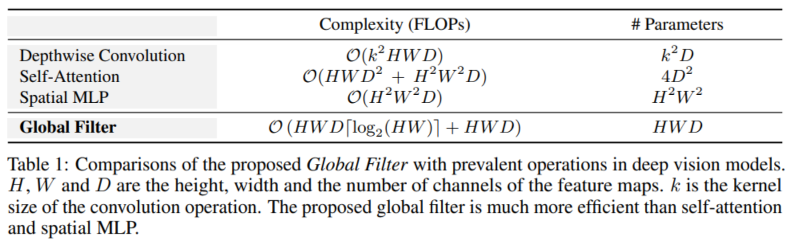
Making it even more efficient:
Using the property of DFT on real signals to reduce redundant computation. Therefore, can take only half of the values in X without losing information.

$$X_r = X[:, 0 : \hat{W}] = \mathcal{F}_r[x],\ \hat{W} = ceil(W/2)$$

Thus, reducing half of the parameters

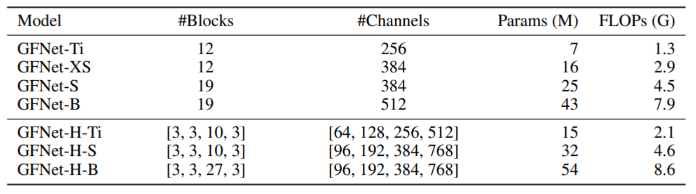
$$K_r \in \mathbb{C}^{Hx\hat{W}xD}$$

Global Filter Networks - Complexity and Variants



	Complexity (FLOPs)	# Parameters
Depthwise Convolution	$\mathcal{O}(k^2HWD)$	k^2D
Self-Attention	$\mathcal{O}(HWD^2 + H^2W^2D)$	$4D^2$
Spatial MLP	$\mathcal{O}(H^2W^2D)$	H^2W^2
Global Filter	$\mathcal{O}\left(HWD\lceil\log_2(HW)\rceil + HWD\right)$	HWD

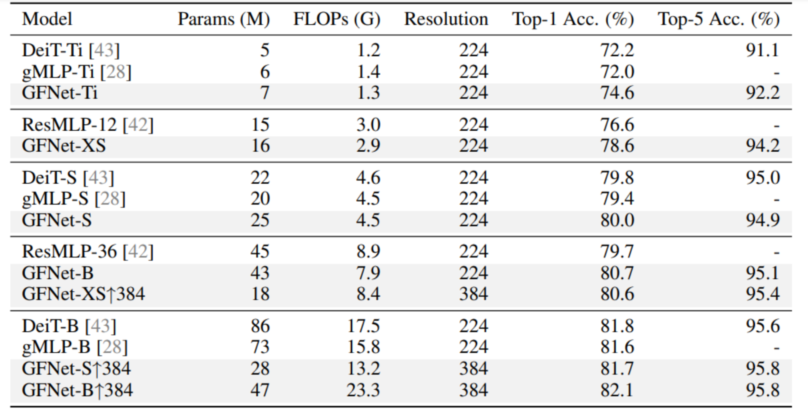
Table 1: Comparisons of the proposed *Global Filter* with prevalent operations in deep vision models. H, W and D are the height, width and the number of channels of the feature maps. k is the kernel size of the convolution operation. The proposed global filter is much more efficient than self-attention and spatial MLP.



Model	#Blocks	#Channels	Params (M)	FLOPs (G)
GFNet-Ti	12	256	7	1.3
GFNet-XS	12	384	16	2.9
GFNet-S	19	384	25	4.5
GFNet-B	19	512	43	7.9
GFNet-H-Ti	[3, 3, 10, 3]	[64, 128, 256, 512]	15	2.1
GFNet-H-S	[3, 3, 10, 3]	[96, 192, 384, 768]	32	4.6
GFNet-H-B	[3, 3, 27, 3]	[96, 192, 384, 768]	54	8.6

ImageNet Classification

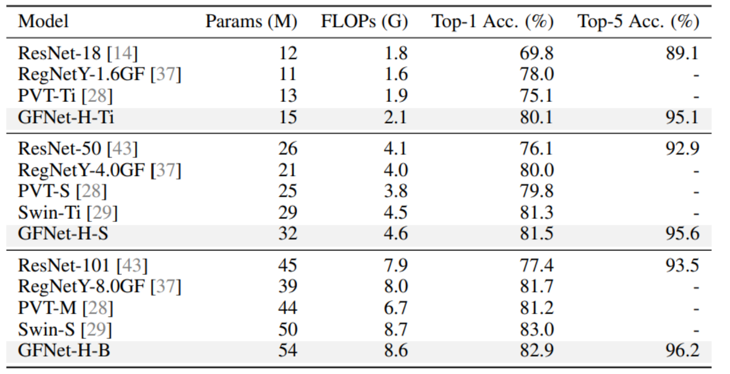
Comparisons with transformer-style architectures on ImageNet:



Model	Params (M)	FLOPs (G)	Resolution	Top-1 Acc. (%)	Top-5 Acc. (%)
DeiT-Ti [43]	5	1.2	224	72.2	91.1
gMLP-Ti [28]	6	1.4	224	72.0	-
GFNet-Ti	7	1.3	224	74.6	92.2
ResMLP-12 [42]	15	3.0	224	76.6	-
GFNet-XS	16	2.9	224	78.6	94.2
DeiT-S [43]	22	4.6	224	79.8	95.0
gMLP-S [28]	20	4.5	224	79.4	-
GFNet-S	25	4.5	224	80.0	94.9
ResMLP-36 [42]	45	8.9	224	79.7	-
GFNet-B	43	7.9	224	80.7	95.1
GFNet-XS↑384	18	8.4	384	80.6	95.4
DeiT-B [43]	86	17.5	224	81.8	95.6
gMLP-B [28]	73	15.8	224	81.6	-
GFNet-S↑384	28	13.2	384	81.7	95.8
GFNet-B↑384	47	23.3	384	82.1	95.8

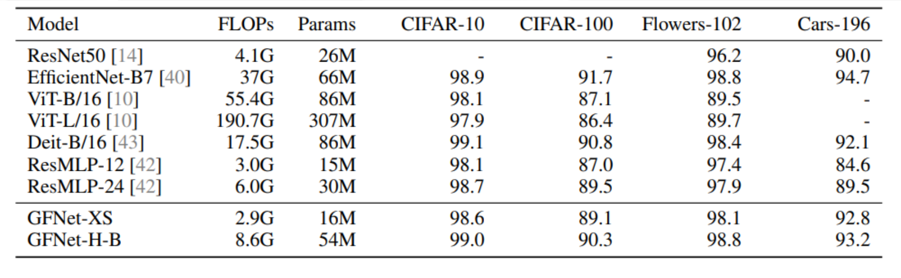
ImageNet Classification

Comparisons with hierarchical architectures on ImageNet:



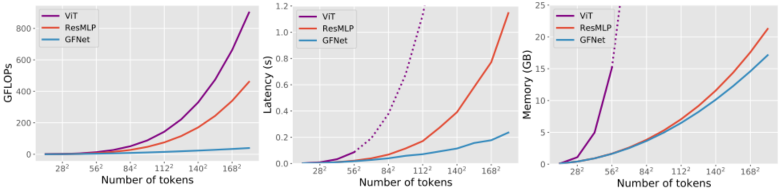
Model	Params (M)	FLOPs (G)	Top-1 Acc. (%)	Top-5 Acc. (%)
ResNet-18 [14]	12	1.8	69.8	89.1
RegNetY-1.6GF [37]	11	1.6	78.0	-
PVT-Ti [28]	13	1.9	75.1	-
GFNet-H-Ti	15	2.1	80.1	95.1
ResNet-50 [43]	26	4.1	76.1	92.9
RegNetY-4.0GF [37]	21	4.0	80.0	-
PVT-S [28]	25	3.8	79.8	-
Swin-Ti [29]	29	4.5	81.3	-
GFNet-H-S	32	4.6	81.5	95.6
ResNet-101 [43]	45	7.9	77.4	93.5
RegNetY-8.0GF [37]	39	8.0	81.7	-
PVT-M [28]	44	6.7	81.2	-
Swin-S [29]	50	8.7	83.0	-
GFNet-H-B	54	8.6	82.9	96.2

Transfer Learning



Model	FLOPs	Params	CIFAR-10	CIFAR-100	Flowers-102	Cars-196
ResNet50 [14]	4.1G	26M	-	-	96.2	90.0
EfficientNet-B7 [40]	37G	66M	98.9	91.7	98.8	94.7
ViT-B/16 [10]	55.4G	86M	98.1	87.1	89.5	-
ViT-L/16 [10]	190.7G	307M	97.9	86.4	89.7	-
Deit-B/16 [43]	17.5G	86M	99.1	90.8	98.4	92.1
ResMLP-12 [42]	3.0G	15M	98.1	87.0	97.4	84.6
ResMLP-24 [42]	6.0G	30M	98.7	89.5	97.9	89.5
GFNet-XS	2.9G	16M	98.6	89.1	98.1	92.8
GFNet-H-B	8.6G	54M	99.0	90.3	98.8	93.2

Flops



(a) (b) (c)

Figure 2: Comparisons among GFNet, ViT [10] and ResMLP [42] in **(a)** FLOPs **(b)** latency and **(c)** GPU memory with respect to the number of tokens (feature resolution). The dotted lines indicate the estimated values when the GPU memory has run out. The latency and GPU memory is measured using a single NVIDIA RTX 3090 GPU with batch size 32 and feature dimension 384.

Accuracy VS. Complexity

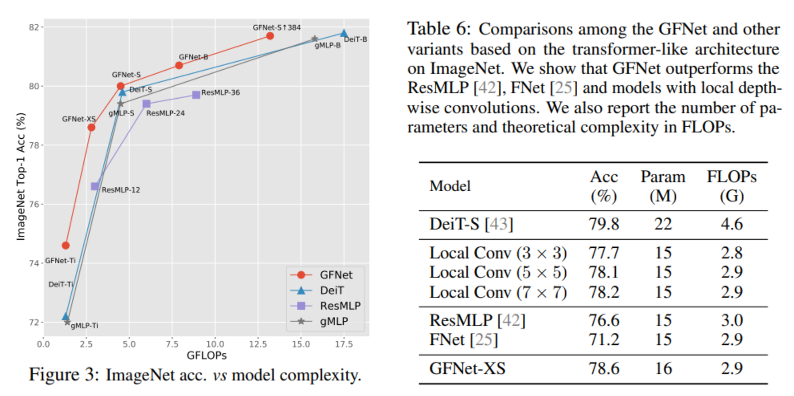
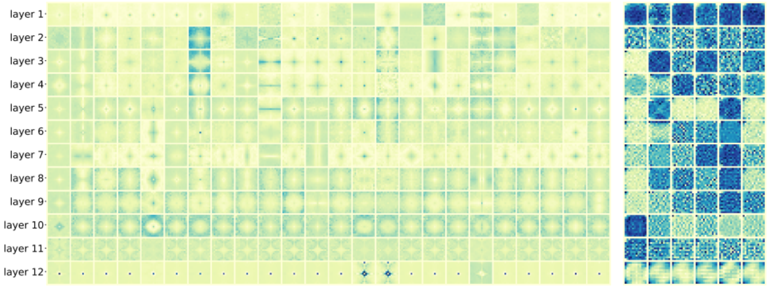


Figure 3: ImageNet acc. *vs* model complexity.

Table 6: Comparisons among the GFNet and other variants based on the transformer-like architecture on ImageNet. We show that GFNet outperforms the ResMLP [42], FNet [25] and models with local depthwise convolutions. We also report the number of parameters and theoretical complexity in FLOPs.

Model	Acc (%)	Param (M)	FLOPs (G)
DeiT-S [43]	79.8	22	4.6
Local Conv (3 × 3)	77.7	15	2.8
Local Conv (5 × 5)	78.1	15	2.9
Local Conv (7 × 7)	78.2	15	2.9
ResMLP [42]	76.6	15	3.0
FNet [25]	71.2	15	2.9
GFNet-XS	78.6	16	2.9

Visualization



(a) Frequency domain

(b) Spatial domain

Figure 4: Visualization of the learned *global filters* in GFNet-XS. We visualize the original frequency domain global filters in (a) and show the corresponding spatial domain filters for the first 6 columns in (b). There are more clear patterns in the frequency domain than the spatial domain.

Questions?

Bibliography

- Global Filter Networks for Image Classification
  https://arxiv.org/pdf/2107.00645.pdf
- A Survey of Transformers
  https://arxiv.org/pdf/2106.04554.pdf