NeurIPS 2022(Oral)

MLCF

Inception Transformer

Chenyang Si[1*] Weihao Yu[1,2*] Pan Zhou[1] Yichen Zhou[1,2] Xinchao Wang[2] Shuicheng Yan[1]

[1]Sea AI Lab [2]National University of Singapore

{sicy,yuweihao,zhoupan,zhouyc,yansc}@sea.com, xinchao@nus.edu.sg

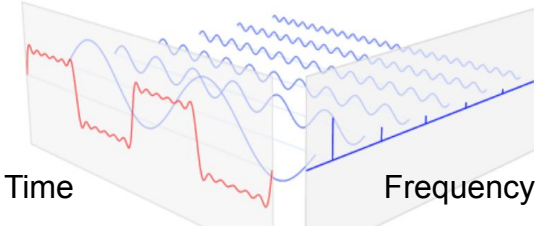
Reviewed by Susang Kim

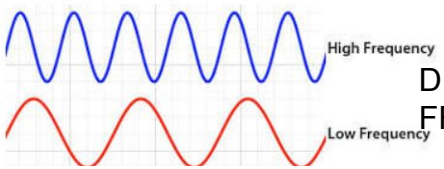
MLCF

Contents

1.Introduction - Fourier Transform

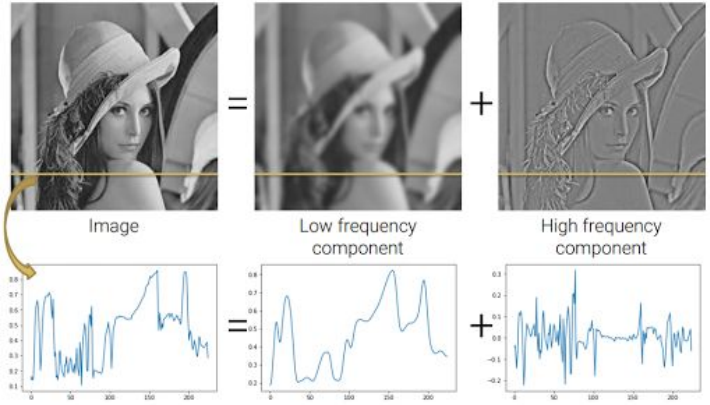
Images are discrete signals, not continuous. And it is a signal defined in a finite interval.

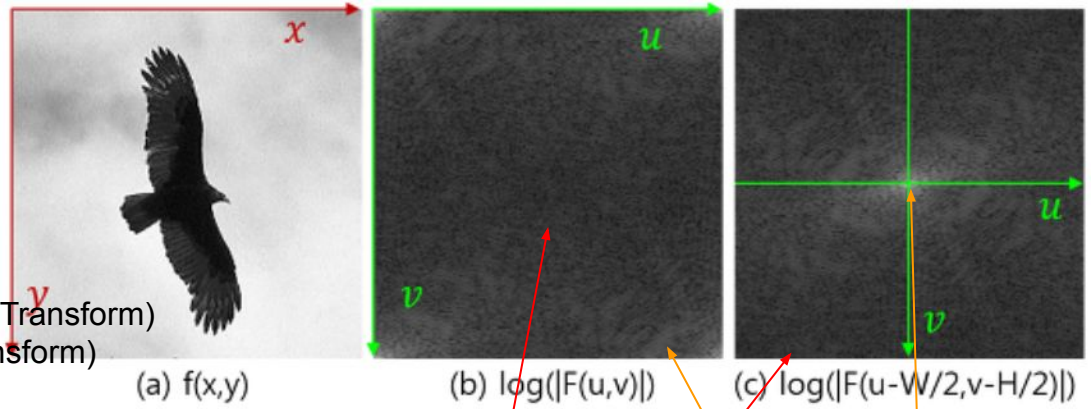




DFT (Discrete Fourier Transform)
FFT (Fast Fourier Transform)

$$F(\omega) = \int_{-\infty}^{\infty} f(x) \exp(-i2\pi\omega x)\, dx$$





https://darkpgmr.tistory.com/171

High Frequency
Local Information

Low Frequency
Global Information

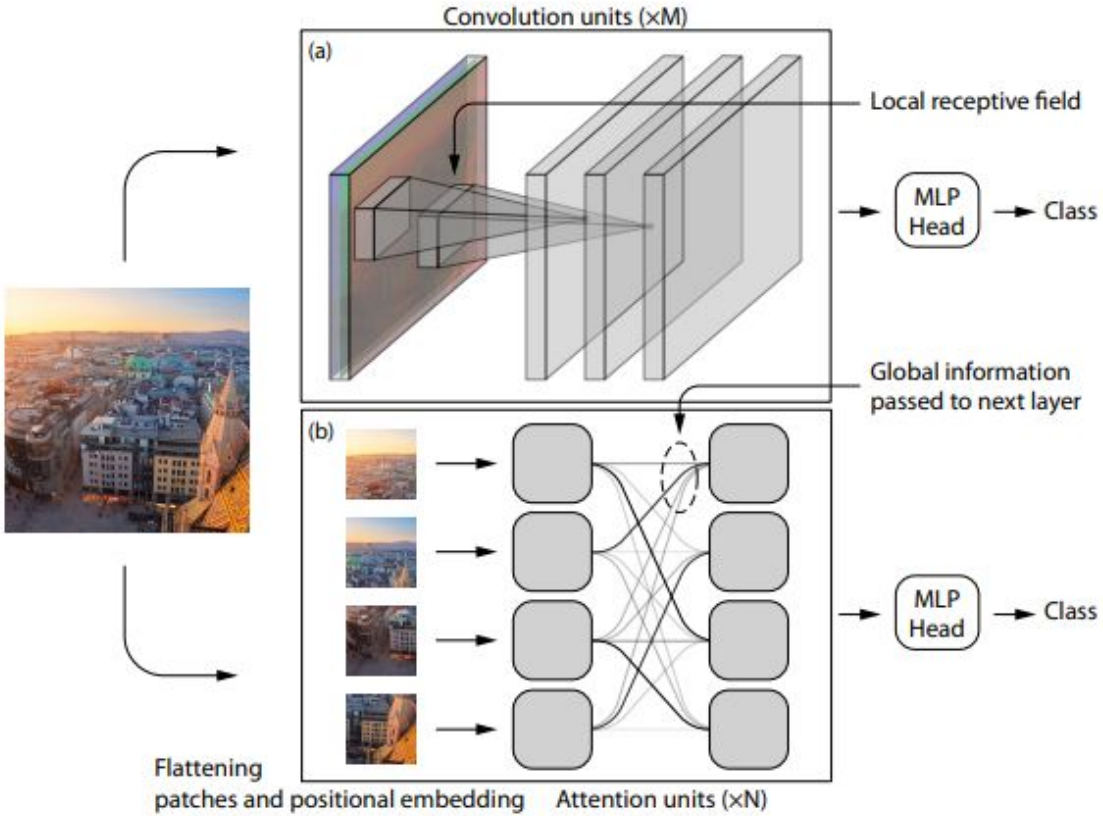
The Fourier Transform converts the image from the spatial domain (i.e., the image's pixel values) to the frequency domain (i.e., a representation of the image's frequencies).

In image processing, the frequency domain is a way to represent an image as a sum of sine waves of different frequencies. The frequency domain is often used for analyzing and processing images because it allows us to better understand the **image's underlying structure and content.**

https://sonsnotation.blogspot.com/2020/12/4-image-filtering.html

1.Introduction - Convolution vs. Transformer

It is difficult for ConvNets to capture long-term dependencies, while self-attention layers are global.

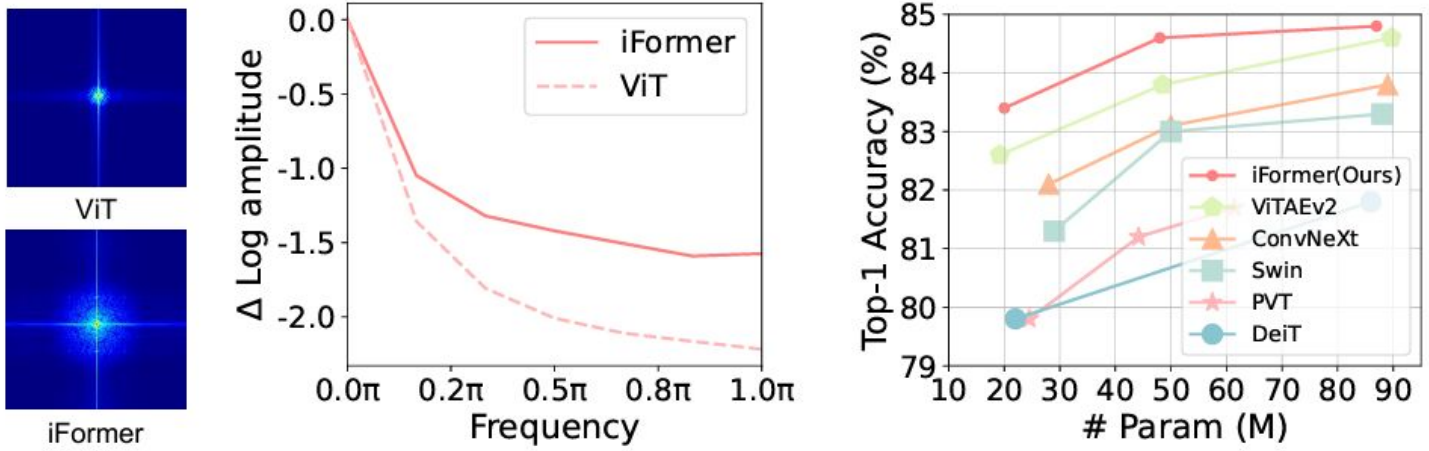


Convolution is efficient in memory and compute.
Local connectivity can lead to loss of global context.
Bad at long sequences(Need to stack many conv layers for outputs to “see” the whole sequence, Static weight).
High Frequency details (Texture biased)

Transformers are flexible and attend to information at various distances away from Patch.
Good at long sequences
- output sees “all” inputs.
Dynamic w.r.t input
- output “sees” inputs adaptively.
Very memory-intensive
Low Frequency global information (Shape Biased)

S Tuli et al., “Are Convolutional Neural Networks or Transformers more like human vision?,” CogSci, 2021.

2.Motivation - Inception Transformer



(a) (b) (c)

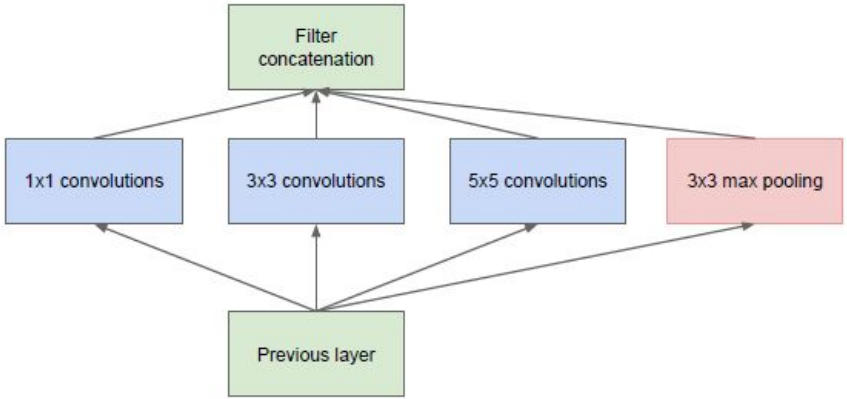
Figure 1: **(a) Fourier spectrum of ViT [18] and iFormer. (b) Relative log amplitudes of Fourier transformed feature maps. (c) Performance of models on ImageNet-1K validation set.** (a) and (b) show that iFormer captures more high-frequency signals.

Effectively learns comprehensive features with both high- and low-frequency information in visual data.
capturing both high and low frequencies

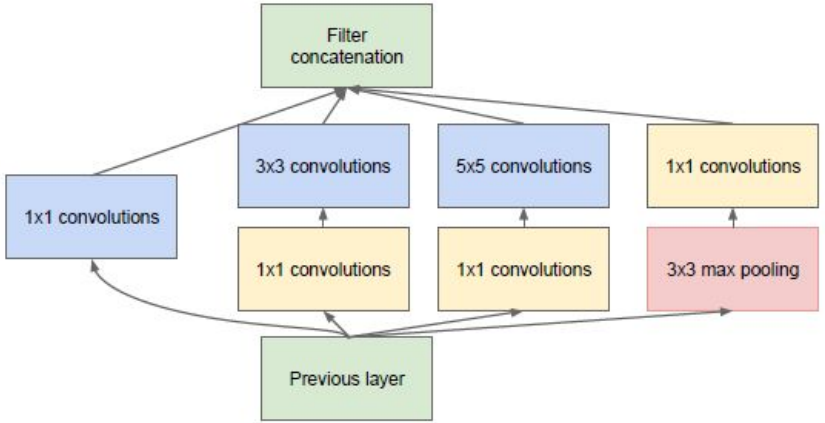
ViT mainly including global shapes and structures of a scene or object, but are not very powerful for learning high-frequencies,mainly including local edges and textures.

3.Related Works - Inception(Going Deeper with Convolutions) (CVPR 2015)

MLCF



(a) Inception module, naïve version



(b) Inception module with dimension reductions

Team	Year	Place	Error (top-5)	Uses external data
SuperVision	2012	1st	16.4%	no
SuperVision	2012	1st	15.3%	Imagenet 22k
Clarifai	2013	1st	11.7%	no
Clarifai	2013	1st	11.2%	Imagenet 22k
MSRA	2014	3rd	7.35%	no
VGG	2014	2nd	7.32%	no
GoogLeNet	2014	1st	6.67%	no

AlexNet

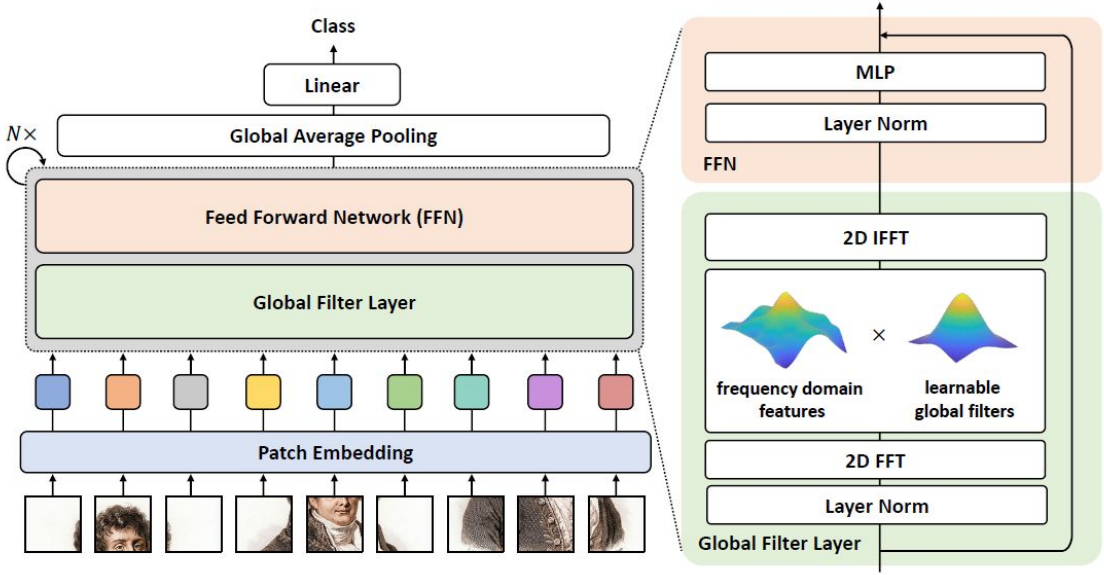
2015 ReNet
2016 GoogleNet v4
2017 SENet

(a) Siberian husky

(b) Eskimo dog

necessary to distinguish between fine-grained visual categories like those in ImageNet

3.Related Works - Global Filter Networks for Image Classification (NeurIPS 2021)




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

$$\boldsymbol{X} = \mathcal{F}[\boldsymbol{x}] \in \mathbb{C}^{H \times W \times D},$$

$$\tilde{\boldsymbol{X}} = \boldsymbol{K} \odot \boldsymbol{X}, \quad \boldsymbol{x} \leftarrow \mathcal{F}^{-1}[\tilde{\boldsymbol{X}}].$$

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

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

GFNet replaces the self-attention layer in vision transformers with three key operations: a 2D discrete Fourier transform, an element-wise multiplication between frequency-domain features and learnable global filters, and a 2D inverse Fourier transform.

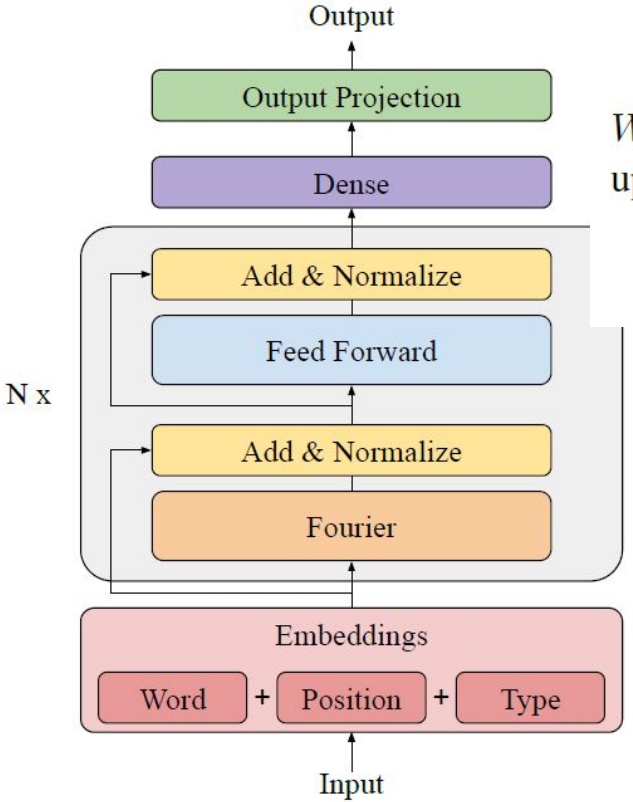
	Complexity (FLOPs)	# Parameters
Depthwise Convolution	$\mathcal{O}(k^2HWD)$	k^2D
Self-Attention	$\mathcal{O}(HWD^2 + H^2W^2D)$	$4D^2$
Spatial MLP	$\mathcal{O}(H^2W^2D)$	H^2W^2
Global Filter	$\mathcal{O}(HWD\lceil\log_2(HW)\rceil + HWD)$	HWD

3.Related Works - FNet:Mixing Tokens with Fourier Transforms(NAACL 2022)

MLCF

Attention may not be the principal component driving the performance of Transformers.
FNet, that uses the Fourier Transform as a mixing mechanism.

$\mathcal{O}(N \log N)$.



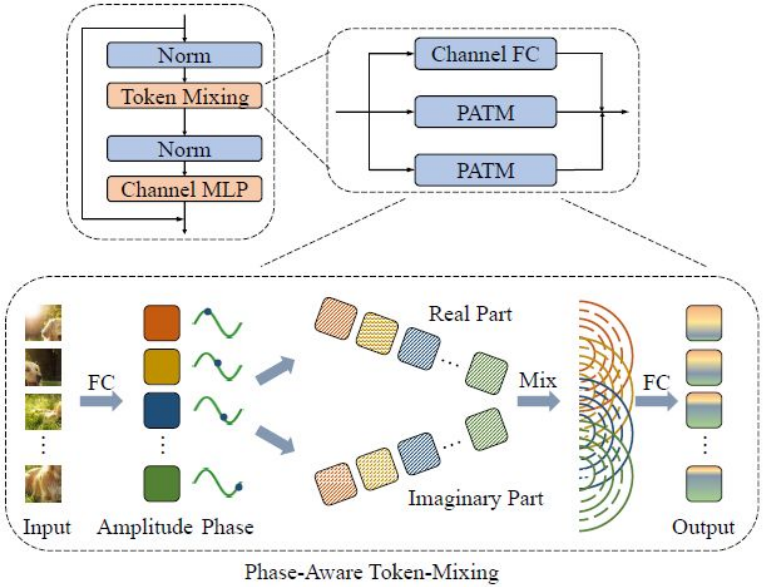
discrete Fourier Transform $X_k = \sum_{n=0}^{N-1} x_n e^{-\frac{2\pi i}{N} nk}, \quad 0 \le k \le N-1.$

W, is a Vandermonde matrix for the roots of unity up to a normalization factor:

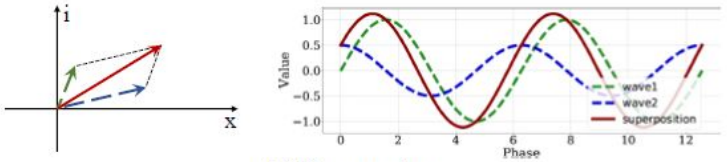
$$W_{nk} = \left(e^{-\frac{2\pi i}{N} nk} / \sqrt{N}\right), \quad V = V(x_0, x_1, \cdots, x_m) = \begin{bmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^n \\ 1 & x_1 & x_1^2 & \cdots & x_1^n \\ 1 & x_2 & x_2^2 & \cdots & x_2^n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_m & x_m^2 & \cdots & x_m^n \end{bmatrix}$$

Model	Mixing layer ops (per layer)	Model params Base	Large
BERT	$2n^2 d_h + 4nd_h^2$	112M	339M
Linear	$n^2 d_h + nd_h^2$	94M	269M
FNet (mat)	$n^2 d_h + nd_h^2$	83M	238M
FNet (FFT)	$nd_h \log(n) + nd_h \log(d_h)$	83M	238M
Random	$n^2 d_h + nd_h^2$	83M	238M
FF-only	0	83M	238M

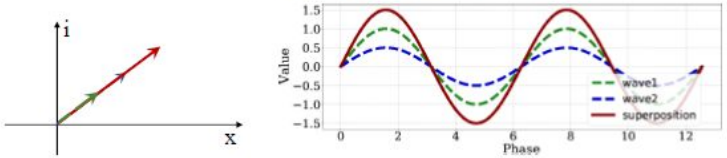
3.Related Works - An Image Patch is a Wave: Phase-Aware Vision MLP (CVPR 2022)



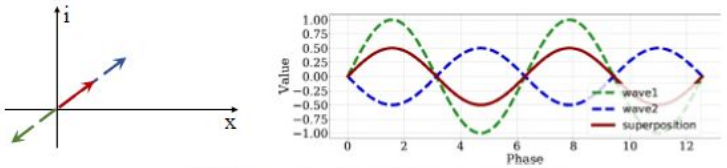
Phase-Aware Token-Mixing



(a) The general case.



(b) Two waves have the same phase.



(c) Two waves have the opposite phase.

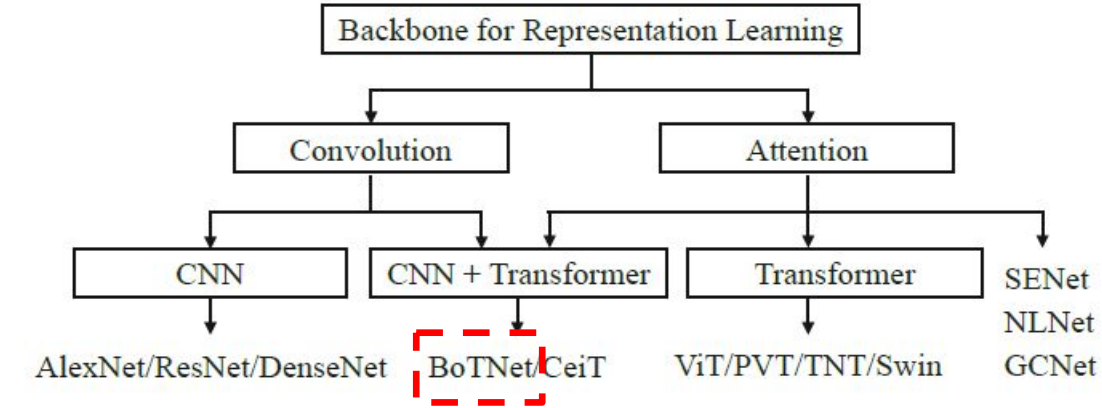
amplitude is a real-value feature representing each token.

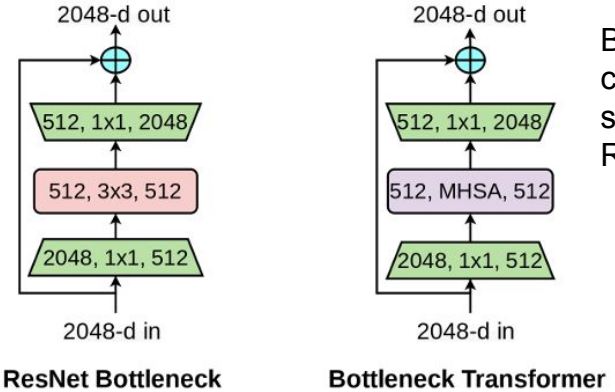
$$|z_r| = \sqrt{|z_i|^2 + |z_j|^2 + 2|z_i| \odot |z_j| \odot \cos(\theta_j - \theta_i)},$$

$$\theta_r = \theta_i + \operatorname{atan2}(|z_j| \odot \sin(\theta_j - \theta_i), \; |z_i| + |z_j| \odot \cos(\theta_j - \theta_i)),$$

Wave-MLP architecture for vision tasks, which takes each token as a wave with both amplitude and phase information. Amplitude is the original real-value feature and the phase modulates relationship between the varying tokens and fixed weights in MLP. With the dynamically produced phase, the tokens are aggregated according to their varying contents from different input images.

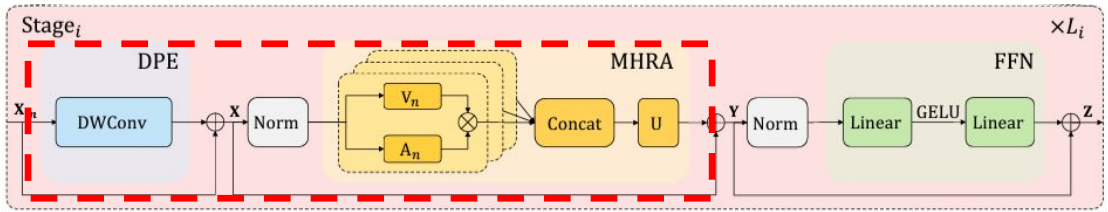
3.Related Works - Hybrid Architecture / BoTNet (CVPR 2021)





By just replacing the spatial convolutions with global self-attention in the final blocks of a ResNet and no other changes.

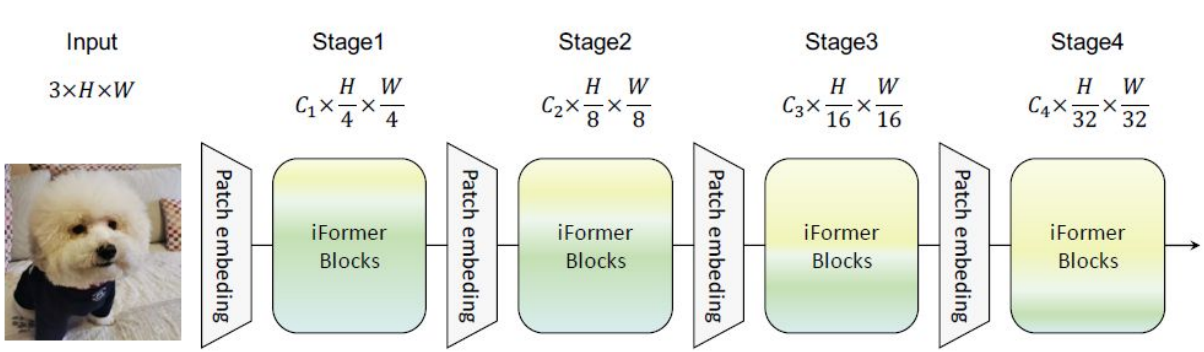
stage	output	ResNet-50	BoTNet-50
c1	512 × 512	7×7, 64, stride 2	7×7, 64, stride 2
c2	256 × 256	3×3 max pool, stride 2	3×3 max pool, stride 2
		[1×1, 64; 3×3, 64; 1×1, 256] ×3	[1×1, 64; 3×3, 64; 1×1, 256] ×3
c3	128 × 128	[1×1, 128; 3×3, 128; 1×1, 512] ×4	[1×1, 128; 3×3, 128; 1×1, 512] ×4
c4	64 × 64	[1×1, 256; 3×3, 256; 1×1, 1024] ×6	[1×1, 256; 3×3, 256; 1×1, 1024] ×6
c5	32 × 32	[1×1, 512; 3×3, 512; 1×1, 2048] ×3	[1×1, 512; MHSA, 512; 1×1, 2048] ×3
# params.		25.5×10^6	20.8×10^6
M.Adds		85.4×10^9	102.98×10^9
TPU steptime		786.5 ms	1032.66 ms

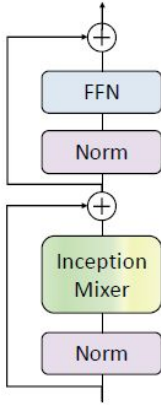


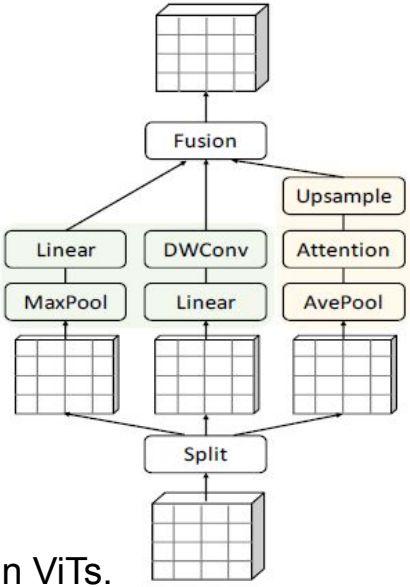
Uniformer (CNN+ViT) (ICLR 2022)

Srinivas, Aravind, et al. "Bottleneck transformers for visual recognition." *CVPR 2021.*

4.Method - The overall architecture of iFormer







For each block, yellow and green indicate low- and high-frequency

high-frequency max-pooling operation and convolution
low-frequency mixer is implemented by a vanilla self-attention in ViTs.

Technically, given the input feature map $\boldsymbol{X} \in \mathbb{R}^{N \times C}$, it is factorized $\boldsymbol{X}$ into $\boldsymbol{X}_h \in \mathbb{R}^{N \times C_h}$ and $\boldsymbol{X}_l \in \mathbb{R}^{N \times C_l}$ along the channel dimension, where $C_h + C_l = C$. Then, $\boldsymbol{X}_h$ and $\boldsymbol{X}_l$ are assigned to high-frequency mixer and low-frequency mixer respectively.

Inception mixer.

MLCF

High-frequency mixer :

$$\boldsymbol{X}_{h1} \in \mathbb{R}^{N \times \frac{C_h}{2}} \qquad \boldsymbol{X}_{h2} \in \mathbb{R}^{N \times \frac{C_h}{2}}$$

$$\begin{aligned} \boldsymbol{Y}_{h1} &= \mathrm{FC}\left(\mathrm{MaxPool}\left(\boldsymbol{X}_{h1}\right)\right) \\ \boldsymbol{Y}_{h2} &= \mathrm{DwConv}\left(\mathrm{FC}\left(\boldsymbol{X}_{h2}\right)\right) \end{aligned}$$

Low-frequency mixer

$$\boldsymbol{Y}_l = \mathrm{Upsample}\left(\mathrm{MSA}\left(\mathrm{AvePooling}\left(\boldsymbol{X}_l\right)\right)\right)$$

the kernel size and stride for the pooling and upsample layers are set to 2 only **at the first two stages.**(reduces the computational overhead)

Low-high-frequency mixers

$$\boldsymbol{Y_c} = \mathrm{Concat}\left(\boldsymbol{Y}_l, \boldsymbol{Y}_{h1}, \boldsymbol{Y}_{h2}\right) \qquad \boldsymbol{Y} = \mathrm{FC}\left(\boldsymbol{Y_c} + \mathrm{DwConv}\left(\boldsymbol{Y_c}\right)\right)$$

$$\begin{aligned} \boldsymbol{Y} &= \boldsymbol{X} + \mathrm{ITM}\left(\mathrm{LN}\left(\boldsymbol{X}\right)\right) \\ \boldsymbol{H} &= \boldsymbol{Y} + \mathrm{FFN}\left(\mathrm{LN}\left(\boldsymbol{Y}\right)\right) \end{aligned}$$

Frequency ramp structure

$$\frac{C_h}{C} + \frac{C_l}{C} = 1$$

define a channel ratio to better balance the high-frequency and low frequency components,

Frequency ramp structure : Configurations of iFormers

Like humans, by capturing the details in high frequency components, lower layers can capture visual elementary features, and also gradually gather local information to achieve a global understanding of the input.

Stage	Layer	iFormer-S	iFormer-B	iFormer-L
1	Patch Embedding	3×3, stride 2, 48 3×3, stride 2, 96	3×3, stride 2, 48 3×3, stride 2, 96	3×3, stride 2, 48 3×3, stride 2, 96
	iFormer Block	$\begin{bmatrix} C_h/h = 2/3 \\ C_l/h = 1/3 \\ \text{pool stride 2} \end{bmatrix} \times 3$	$\begin{bmatrix} C_h/h = 2/3 \\ C_l/h = 1/3 \\ \text{pool stride 2} \end{bmatrix} \times 4$	$\begin{bmatrix} C_h/h = 2/3 \\ C_l/h = 1/3 \\ \text{pool stride 2} \end{bmatrix} \times 4$
2	Patch Embedding	2×2, stride 2, 192	2×2, stride 2, 192	2×2, stride 2, 192
	iFormer Block	$\begin{bmatrix} C_h/h = 1/2 \\ C_l/h = 1/2 \\ \text{pool stride 2} \end{bmatrix} \times 3$	$\begin{bmatrix} C_h/h = 1/2 \\ C_l/h = 1/2 \\ \text{pool stride 2} \end{bmatrix} \times 6$	$\begin{bmatrix} C_h/h = 1/2 \\ C_l/h = 1/2 \\ \text{pool stride 2} \end{bmatrix} \times 6$
3	Patch Embedding	2×2, stride 2, 320	2×2, stride 2, 384	2×2, stride 2, 448
	iFormer Block	$\begin{bmatrix} C_h/h = 3/10 \to 1/10 \\ C_l/h = 7/10 \to 9/10 \\ \text{pool stride 1} \end{bmatrix} \times 9$	$\begin{bmatrix} C_h/h = 4/12 \to 2/12 \\ C_l/h = 8/12 \to 10/12 \\ \text{pool stride 1} \end{bmatrix} \times 14$	$\begin{bmatrix} C_h/h = 4/14 \to 2/14 \\ C_l/h = 10/14 \to 12/14 \\ \text{pool stride 1} \end{bmatrix} \times 18$
4	Patch Embedding	2×2, stride 2, 384	2×2, stride 2, 512	2×2, stride 2, 640
	iFormer Block	$\begin{bmatrix} C_h/h = 1/12 \\ C_l/h = 11/12 \\ \text{pool stride 1} \end{bmatrix} \times 3$	$\begin{bmatrix} C_h/h = 1/16 \\ C_l/h = 15/16 \\ \text{pool stride 1} \end{bmatrix} \times 6$	$\begin{bmatrix} C_h/h = 1/20 \\ C_l/h = 19/20 \\ \text{pool stride 1} \end{bmatrix} \times 8$
#Param. (M)		20	48	87
FLOPs (G)		4.8	9.4	14.0

$$\frac{C_h}{C} + \frac{C_l}{C} = 1$$

Comparison of different types of models on ImageNet-1K

MLCF

Model Size	Arch.	Method	#Param. (M)	FLOPs (G)	Input Size Train	Input Size Test	ImageNet Top-1	ImageNet Top-5
small model size (~20M)	CNN	RSB-ResNet-50 [47, 62]	26	4.1	224	224	80.4	-
		ConvNeXt-T [30]	28	4.5	224	224	82.1	-
	ViT	Deit-S [29]	22	4.6	224	224	79.8	95.0
		PVT-S [6]	25	3.8	224	224	79.8	-
		T2T-14 [38]	22	5.2	224	224	80.7	-
		Swin-T [5]	29	4.5	224	224	81.3	95.5
		Focal-T [63]	29	4.9	224	224	82.2	95.9
		CSwin-T [64]	23	4.3	224	224	82.7	-
	Hybrid	CvT-13 [25]	20	4.5	224	224	81.6	-
		CoAtNet-0 [24]	25	4.2	224	224	81.6	-
		Container [65]	22	8.1	224	224	82.7	-
		ViTAE-S [23]	24	5.6	224	224	82.0	95.9
		ViTAEv2-S [66]	19	5.7	224	224	82.6	96.2
		UniFormer-S [22]	22	3.6	224	224	82.9	-
		iFormer-S	**20**	**4.8**	**224**	**224**	**83.4**	**96.6**
large model size (~100M)	CNN	RegNetY-16GF [29, 67]	84	16.0	224	224	82.9	-
		ConvNeXt-B [30]	89	15.4	224	224	83.8	-
	ViT	DeiT-B [29]	86	17.5	224	224	81.8	95.6
		Swin-B [5]	88	15.4	224	224	83.3	96.5
		Focal-B [63]	90	16.0	224	224	83.8	96.5
		CSwin-B [64]	78	15.0	224	224	84.2	-
	Hybrid	BoTNet-T7 [68]	79	19.3	256	256	84.2	-
		CoAtNet-3 [24]	168	34.7	224	224	84.5	-
		ViTAEv2-B [66]	90	24.3	224	224	84.6	96.9
		iFormer-L	**87**	**14.0**	**224**	**224**	**84.8**	**97.0**

The same data augmentations and regularization methods in DeiT for fair comparison.

iFormer surpasses
both the SoTA ViTs and hybrid ViTs.

Fine-tuning Results with larger resolution (384 x 384)

Method	#Param. (M)	FLOPs (G)	Input Size Train	Input Size Test	ImageNet Top-1
EfficientNet-B5 [72]	30	9.9	456	456	83.6
EfficientNetV2-S [73]	22	8.5	384	384	83.9
CSwin-T↑384 [64]	23	14.0	224	384	84.3
CvT-13↑384 [25]	20	16.3	224	384	83.0
CoAtNet-0↑384 [24]	20	13.4	224	384	83.9
ViTAEv2-S↑384 [66]	19	17.8	224	384	83.8
iFormer-S↑384	**20**	**16.1**	**224**	**384**	**84.6**
EfficientNet-B7 [72]	66	39.2	600	600	84.3
EfficientNetV2-M [73]	54	25.0	480	480	85.1
ViTAEv2-48M ↑384 [66]	49	41.1	224	384	84.7
CSwin-S↑384 [64]	35	22.0	224	384	85.0
CoAtNet-1↑384 [24]	42	27.4	224	384	85.1
iFormer-B↑384	**48**	**30.5**	**224**	**384**	**85.7**
EfficientNetV2-L [73]	121	53	480	480	85.7
Swin-B↑384 [5]	88	47.0	224	384	84.2
CSwin-B↑384 [64]	78	47.0	224	384	85.4
ViTAEv2-B↑384 [66]	90	74.4	224	384	85.3
CoAtNet-2↑384 [24]	75	49.8	224	384	85.7
iFormer-L↑384	**87**	**45.3**	**224**	**384**	**85.8**

iFormer consistently outperforms the counterparts by a significant margin across different computation settings. These results clearly demonstrate the advantages of iFormer on image classifications.

Results on Object detection and Instance segmentation

MLCF

Method	#Param. (M)	FLOPs (G)	Mask R-CNN 1 × AP^b	AP^b_{50}	AP^b_{70}	AP^m	AP^m_{50}	AP^m_{75}
ResNet50 [47]	44	260	38.0	58.6	41.4	34.4	55.1	36.7
PVT-S [6]	44	245	40.4	62.9	43.8	37.8	60.1	40.3
TwinsP-S [75]	44	245	42.9	65.8	47.1	40.0	62.7	42.9
Twins-S [75]	44	228	43.4	66.0	47.3	40.3	63.2	43.4
Swin-T [5]	48	264	42.2	64.6	46.2	39.1	61.6	42.0
ViL-S [76]	45	218	44.9	67.1	49.3	41.0	64.2	44.1
Focal-T [63]	49	291	44.8	67.7	49.2	41.0	64.7	44.2
UniFormer-S_{h14} [22]	41	269	45.6	68.1	49.7	41.6	64.8	45.0
iFormer-S	**40**	**263**	**46.2**	**68.5**	**50.6**	**41.9**	**65.3**	**45.0**
ResNet101 [47]	63	336	40.4	61.1	44.2	36.4	57.7	38.8
X101-32	63	340	41.9	62.5	45.9	37.5	59.4	40.2
PVT-M [6]	64	302	42.0	64.4	45.6	39.0	61.6	42.1
TwinsP-B [75]	64	302	44.6	66.7	48.9	40.9	63.8	44.2
Twins-B [75]	76	340	45.2	67.6	49.3	41.5	64.5	44.8
Swin-S [5]	69	354	44.8	66.6	48.9	40.9	63.4	44.2
Focal-S [63]	71	401	47.4	69.8	51.9	42.8	66.6	46.1
CSWin-S [64]	54	342	47.9	70.1	52.6	43.2	67.1	46.2
UniFormer-B [22]	69	399	47.4	69.7	52.1	43.1	66.0	46.5
iFormer-B	**67**	**351**	**48.3**	**70.3**	**53.2**	**43.4**	**67.2**	**46.7**

iFormer as the backbone in Mask R-CNN.

The FLOPs are measured at resolution 800x1280

Semantic Segmentation

MLCF

Method	#Param. (M)	FLOPs (G)	mIoU (%)
ResNet50 [47]	29	183	36.7
PVT-S [6]	28	161	39.8
TwinsP-S [75]	28	162	44.3
Twins-S [75]	28	144	43.2
Swin-T [5]	32	182	41.5
UniFormer-S_{h32} [22]	25	199	46.2
UniFormer-S [22]	25	247	46.6
UniFormer-B [22]	54	471	48.0
iFormer-S	**24**	**181**	**48.6**

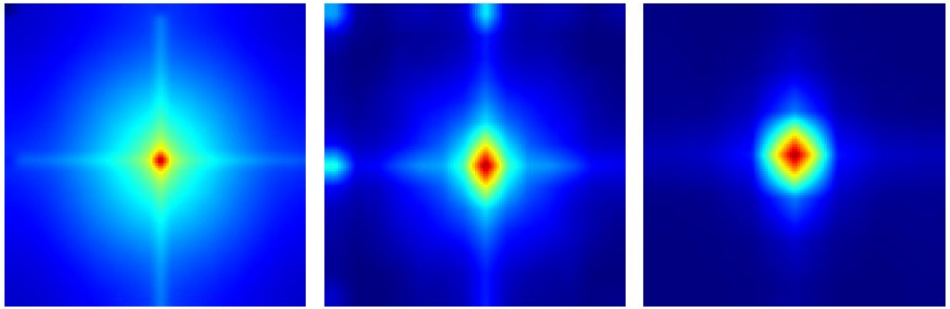
Semantic segmentation with semantic FPN on ADE20K. FLOPs are measured at resolution 512x2048.

evaluate the generality of iFormer through benchmark on semantic segmentation, i.e., ADE20K.

The dataset contains 20K training images and 2K validation images. We adopt iFormer pretrained on ImageNet as the backbone of the Semantic FPN framework. Following PVT and UniFormer, we use AdamW with an initial learning rate of 2x10^-4 with cosine learning rate schedule to train 80k iterations. All experiments are implemented on mmsegmentation codebase

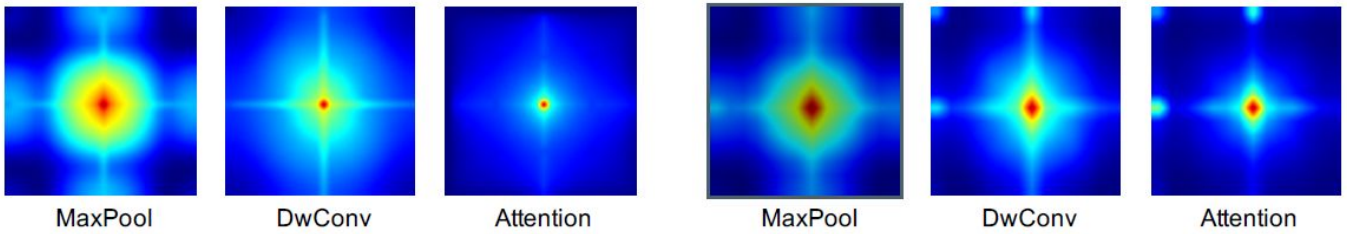
Fourier spectrum of 6-th, 12-th and 18-th layers for the iFormer-S.

MLCF



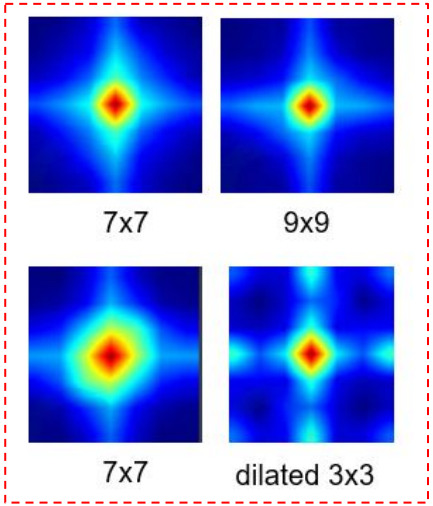
(a) 6-*th* layer (b) 12-*th* layer (c) 18-*th* layer

Fourier spectrum of iFormer-S for the MaxPool, DwConv and Attention branches in the Inception mixer. We can observe that attention mixer tends to reduce high frequencies, while MaxPool and DwConv enhance them.



(a) 4-*th* layer (b) 8-*th* layer

Figure 4: **(a) (b) Fourier spectrum of iFormer-S for the MaxPool, DwConv and Attention branches in the Inception mixer.** We can observe that attention mixer tends to reduce high-frequencies, while MaxPool and DwConv enhance them.



Effect of kernel type on frequency

Ablation study

Table 7: Ablation study of down-up structure for Self-attention and kernel size of convolution.

	Down- and Up	#Param. (M)	FLOPs (G)	Top-1(%)
Former-S	✗	20	7.0	83.6
	✓	20	4.8	83.4
	Kernel Size			Top-1(%)
Former-S	5×5			83.2
	3×3			83.4

adopting the down- and up-sample structure, iFormer-S gets a similar accuracy (83.4%) but has much less FLOPs (4.8 G).

the self-attention has learned that is learnt by large-kernel DWConv. Besides, the smaller kernel is more conducive and effective for capturing high-frequency.

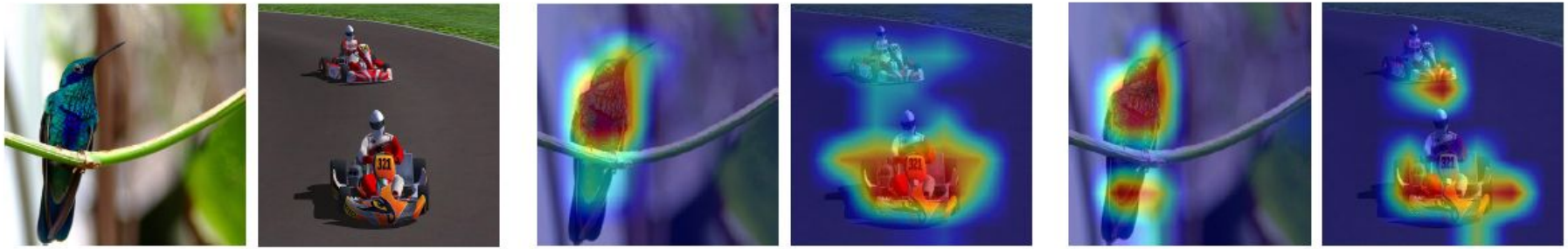
Table 5: Ablation study of Inception mixer and frequency ramp structure on ImageNet-1K. All the models are trained for 100 epochs.

	Attention	MaxPool	DwConv	#Param. (M)	FLOPs (G)	Top-1(%)
Mixer	✓	✗	✗	21	5.2	80.8
	✓	✓	✗	20	4.9	81.0
	✓	✓	✓	20	4.8	81.2
Structure	$C_l/C \downarrow, C_h/C \uparrow$			19	4.7	80.5
	$C_l/C = C_h/C$			19	4.7	80.7
	$C_l/C \uparrow, C_h/C \downarrow$			20	4.8	81.2

Attention	MaxPool	DwConv	Top-1(%)
✓	✓	X	81.2
✓	X	✓	81.4
✓	✓	✓	81.5

Inception token mixer. The Inception mixer is proposed to augment the perception capability of ViTs in the frequency spectrum. To evaluate the effects of the components in the Inception mixer, we increasingly remove each branch from the full model

Grad-CAM of Swin-T and iFormer-S trained on ImageNet.



(a) Input (b) Swin-T (c) iFormer-S

Figure 5: **Grad-CAM [80] activation maps of Swin-T [5] and iFormer-S trained on ImageNet.**

Input Swin-T iFormer-S Input Swin-T iFormer-S

Conclusions

MLCF

iFormer adopts a **channel splitting mechanism** to simply and efficiently couple **convolution, maxpooling and self-attention**, giving more concentrations on high frequencies and expanding the perception capability of the Transformer in the frequency spectrum

iFormer **outperforms** representative vision Transformers on **image classification, object detection and semantic segmentation**, demonstrating the great potential of our iFormer to serve as a general-purpose backbone for computer vision.

we further design a **frequency ramp structure**, enabling effective trade-off between **high-frequency and low-frequency** components across all layers

Limitation.
It is not trained on large scale datasets, e.g., ImageNet-21K
A straightforward solution would be to use neural architecture search.

Openreview (NeurIPS 2022 Oral)

https://openreview.net/forum?id=qf12cWVSksq

MLCF

This paper proposes a novel **multi-branch style architecture** for vision tasks, motivated by a **frequency perspective** of deep network behaviors.

Official Blind Review 1 (Rating: 7: Accept)
Q: The **design of a frequency ramp structure** that uses a channel ratio to balance the high- and low-frequency seems straightforward. I wonder other **designs for frequency information** balancing and how they perform. (from linear scaling)
A: cosine scaling / NAS
We then consider using a **regularization method** to help improve the ability of attention to learn high-frequency information.

Official Blind Review #2 (Rating: 8: Strong Accept)
Q: the **multi-branched network** seems to have better generalization and optimization properties compared with the single-branched counter parts **Lack of discussion with previous multi-branched network structure.**
A: this work **aims to disclose the problem of vanilla ViTs**, while ResNeXt/Inception aims to improve the efficiency of CNNs.

Official Blind Review #3 (8: Strong Accept)
Q: How does the **feature fusion module** compares to direct concatenation in performance?
A: Our feature fusion module for iFormer-S achieves **83.4%**, while the result of direct concatenation is **83.0%**

Official Blind Review #4 (Rating: 7: Accept - increase ratings)
Q: the **fair settings** instead of only on an intermediate setting.
A: this study will provide **valuable insights for the community** to design efficient and effective Transformer architectures.
Q: we can apply **large kernel DWConv** to achieve better performance than the naive MaxPool operation.
A: the self-attention has learned the information that is learnt by large-kernel DWConv. Besides, **the smaller kernel is more conducive and effective** for capturing high-frequency information.

MLCF

Thanks

Any Questions?

You can send mail to
Susang Kim(healess1@gmail.com)