MLP-Mixer: An all-MLP Architecture for Vision

Presented by Hector Maeso Garcia

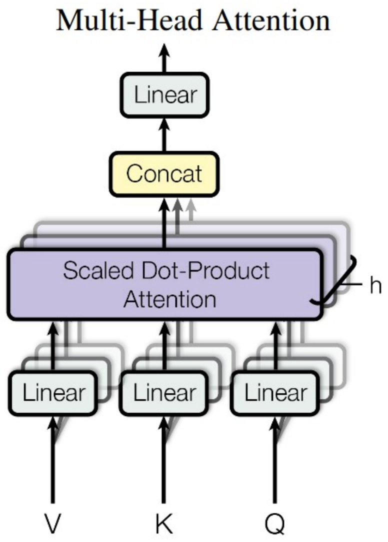
Wednesday, 8th May

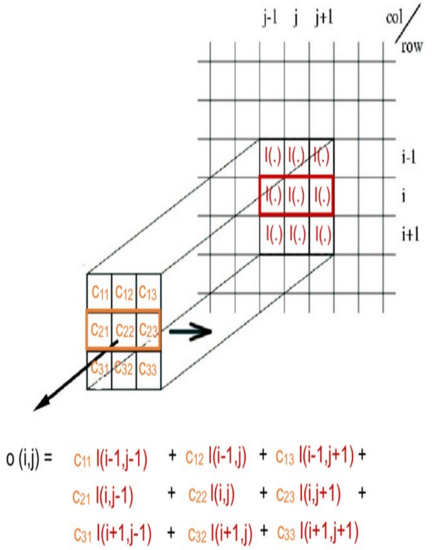
Introduction

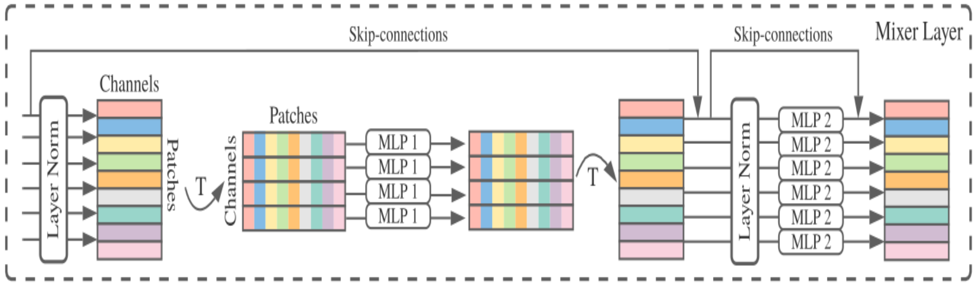
- Released by Google Brain in 2021

- Authored by: Ilya Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, Mario Lucic, Alexey Dosovitskiy

- Propose an architecture for computer vision based in **Multi-Layer Perceptrons**







Schedule

1. Motivation
2. Convolution and Attention
3. MLP Mixer Architecture
4. Experiments and results
5. Conclusion
6. Personal opinion

Motivation

"...while convolutions and attention are both sufficient for good performance, neither of them are necessary."

"We hope that these results spark further research beyond the realms of well established CNNs and Transformers."

Review on Convolution

- Convolutions are linear, local, shift invariant transformations.

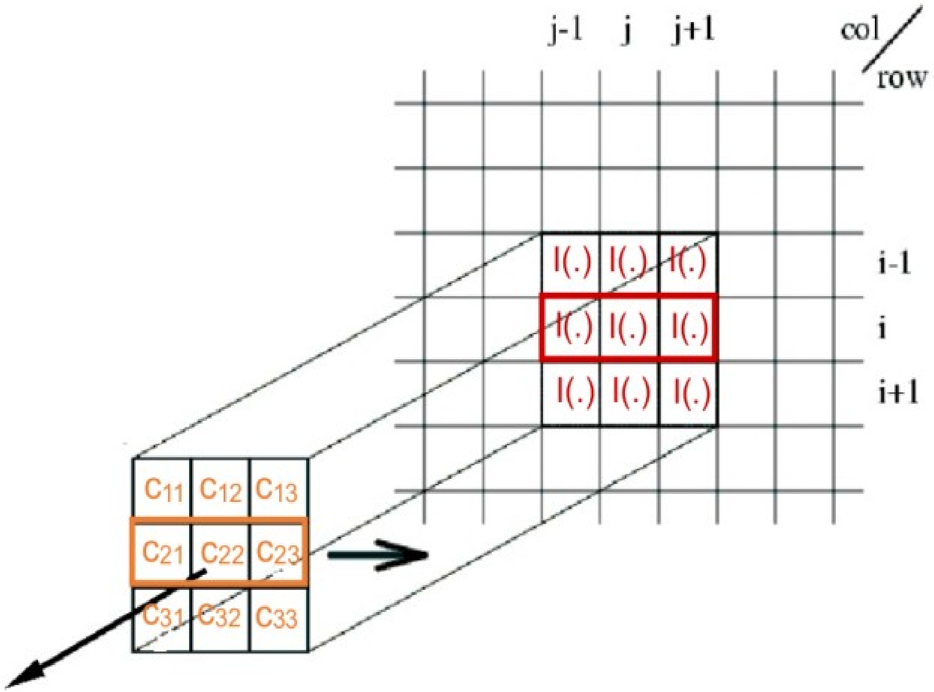
- Convolution with **channels**: Kernel shape is

$$k \times k \times C_{\text{in}} \times C_{\text{out}}$$

- **Separable** convolutions: C different $k \times k$

Kernels, with

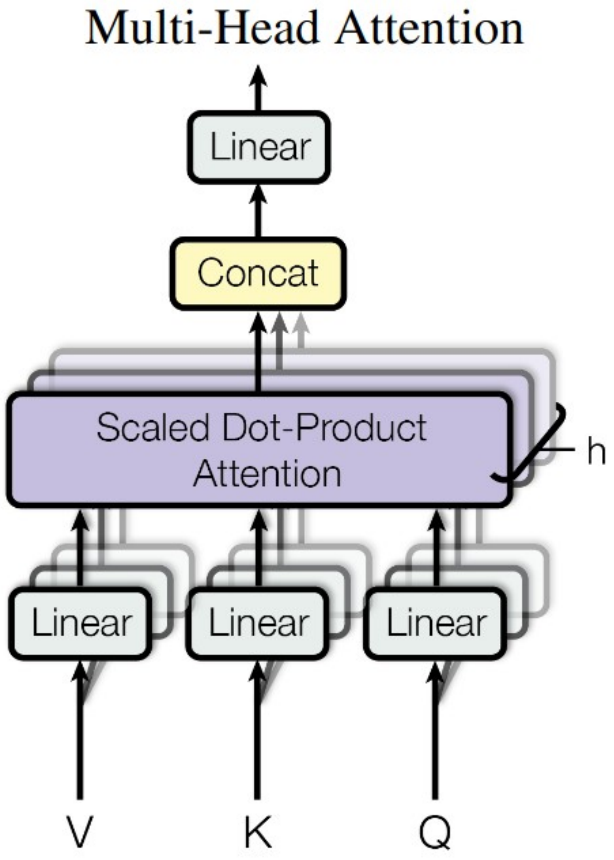
$$C_{\text{in}} = C_{\text{out}} := C$$



$$o(i,j) = c_{11}\, I(i-1,j-1) + c_{12}\, I(i-1,j) + c_{13}\, I(i-1,j+1) + c_{21}\, I(i,j-1) + c_{22}\, I(i,j) + c_{23}\, I(i,j+1) + c_{31}\, I(i+1,j-1) + c_{32}\, I(i+1,j) + c_{33}\, I(i+1,j+1)$$

Review on Self-Attention

- Attention Is All You Need, Vaswani et. al.

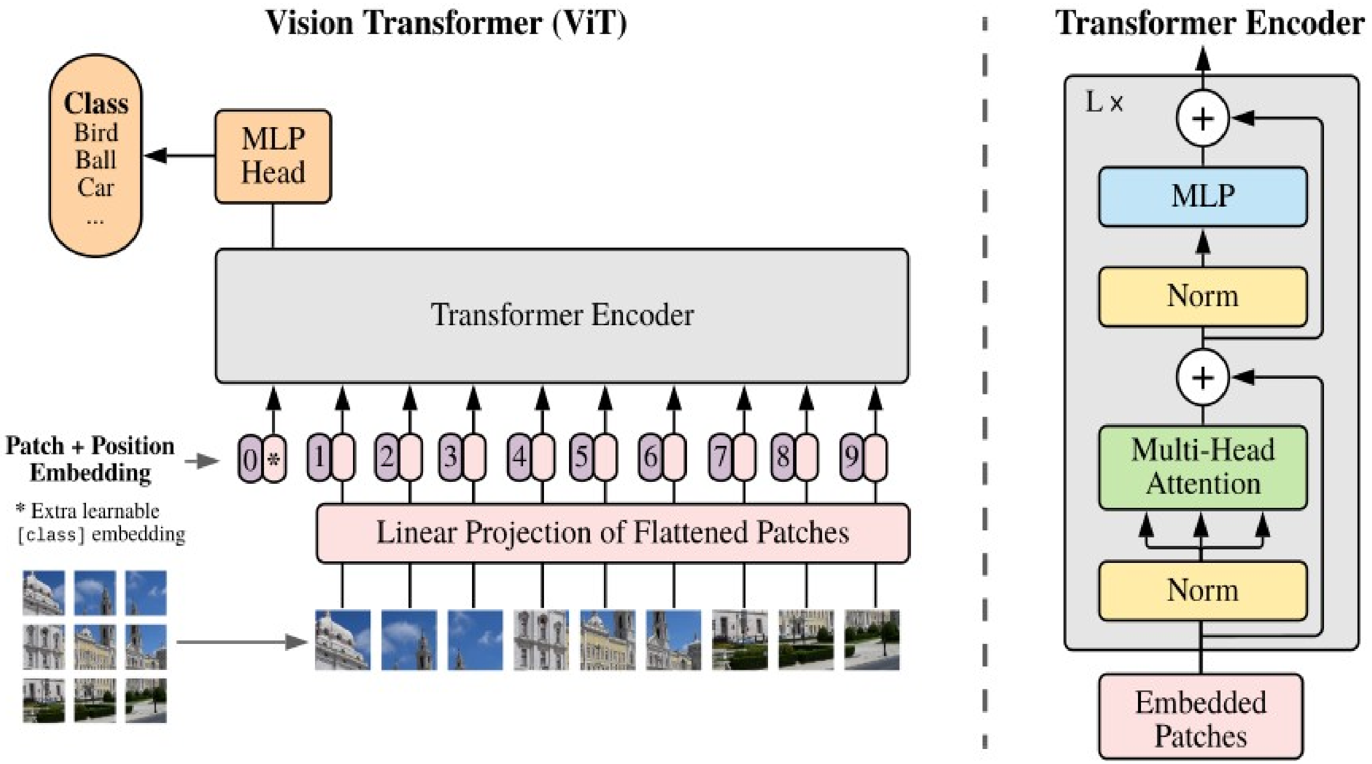


The Vision Transformer

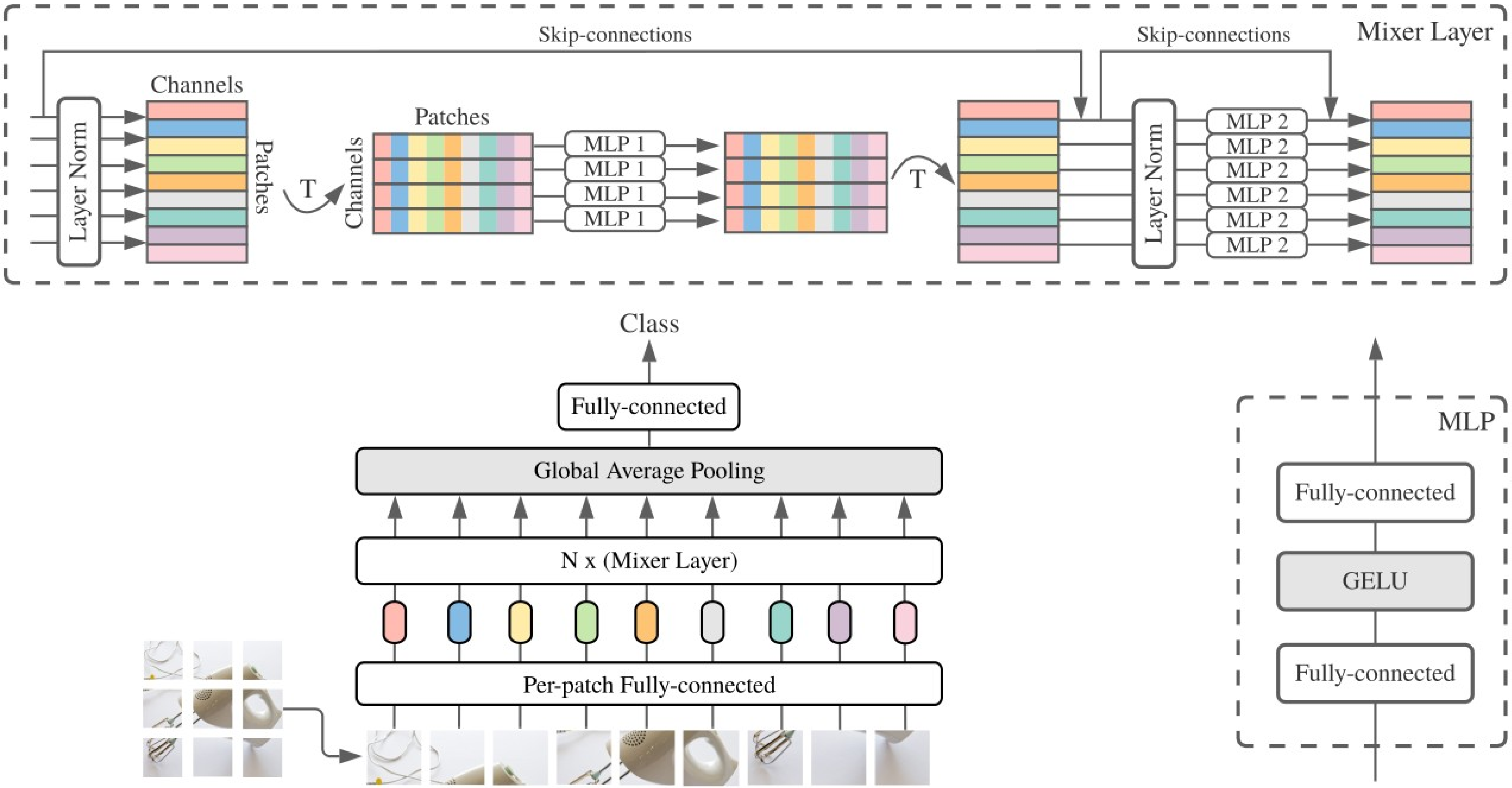
- An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale, [Dositovsky et al](#).

- Proposed in 2021 by Google Brain research.

- Divide input image into patches and map them into tokens. Shared **linear projection** + **positional embeddings**.

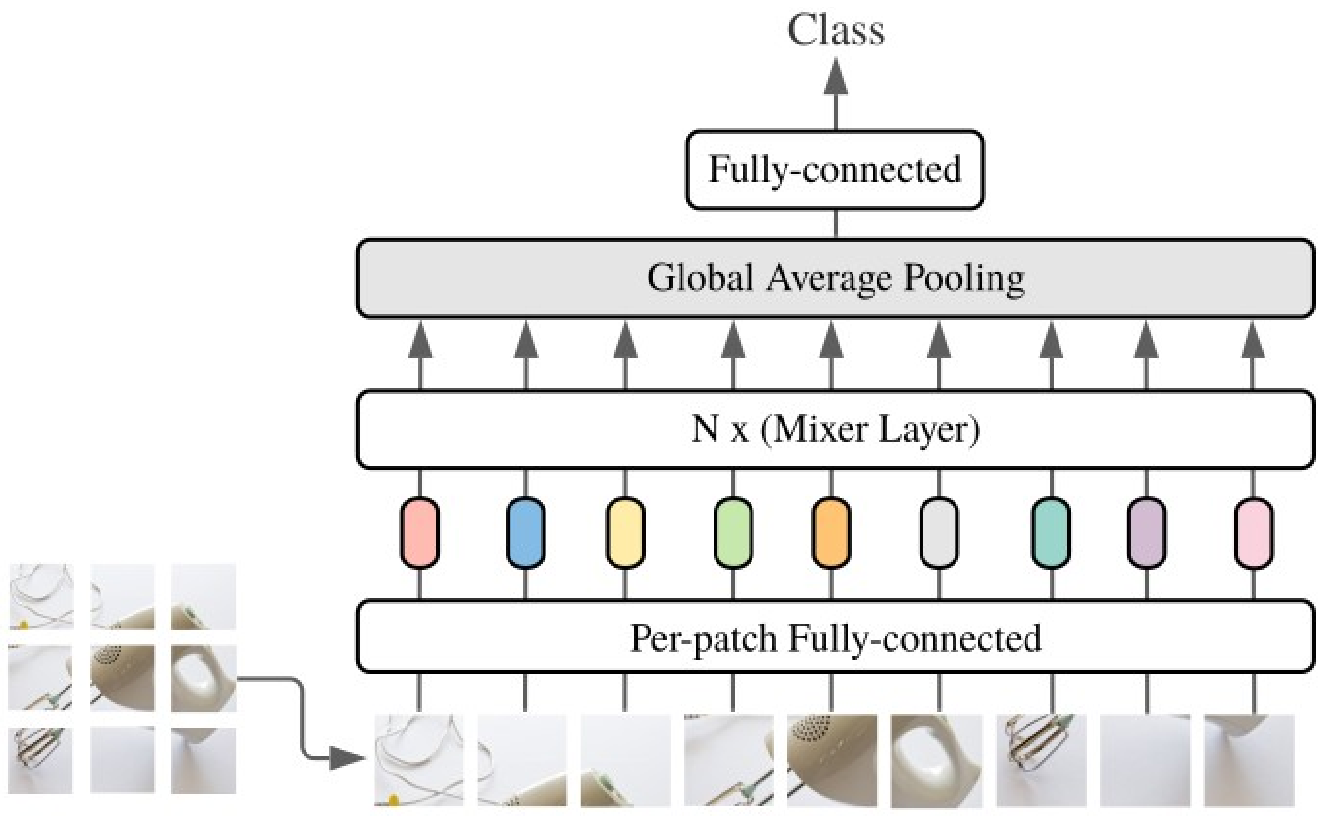


MLP-Mixer Model

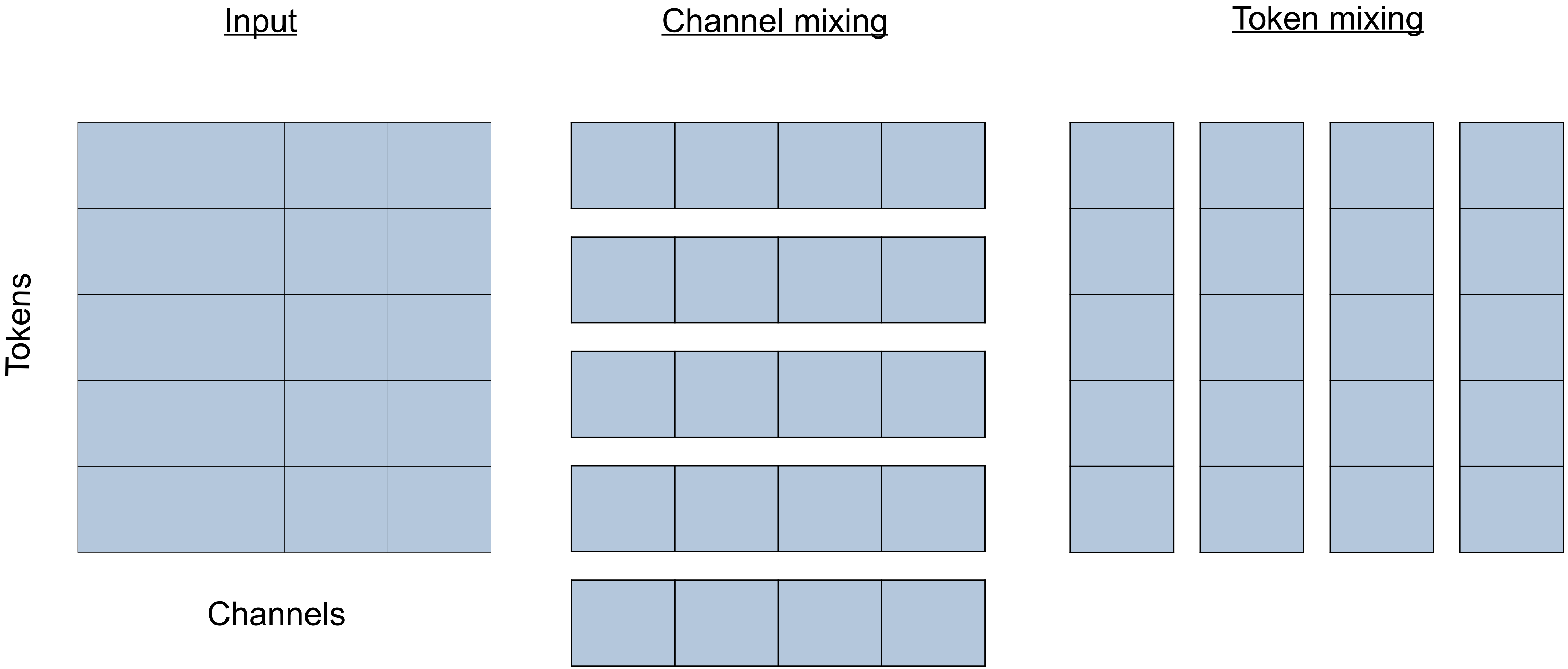


MLP-Mixer Model

- Divide input image into S patches of size P x P.
- Linear projection of each patch into a token with C channels.
- No positional embeddings.
- Input and output shape S x C, with S the number of patches.



Mixer Layer



Mixer Layer

$$\mathbf{U}_{*,i} = \mathbf{X}_{*,i} + \mathbf{W}_2\, \sigma\big(\mathbf{W}_1\, \text{LayerNorm}(\mathbf{X})_{*,i}\big)$$
$$\text{for } i = 1 \ldots C$$

$$\mathbf{Y}_{j,*} = \mathbf{U}_{j,*} + \mathbf{W}_4\, \sigma\big(\mathbf{W}_3\, \text{LayerNorm}(\mathbf{U})_{j,*}\big)$$
$$\text{for } j = 1 \ldots S.$$

Mixer Layer

<u>Channel mixing (MLP 2)</u>:

communication between different **channels**

operate on each **patch** independently

shared across all rows (**patches**)

mix features at a given spatial location

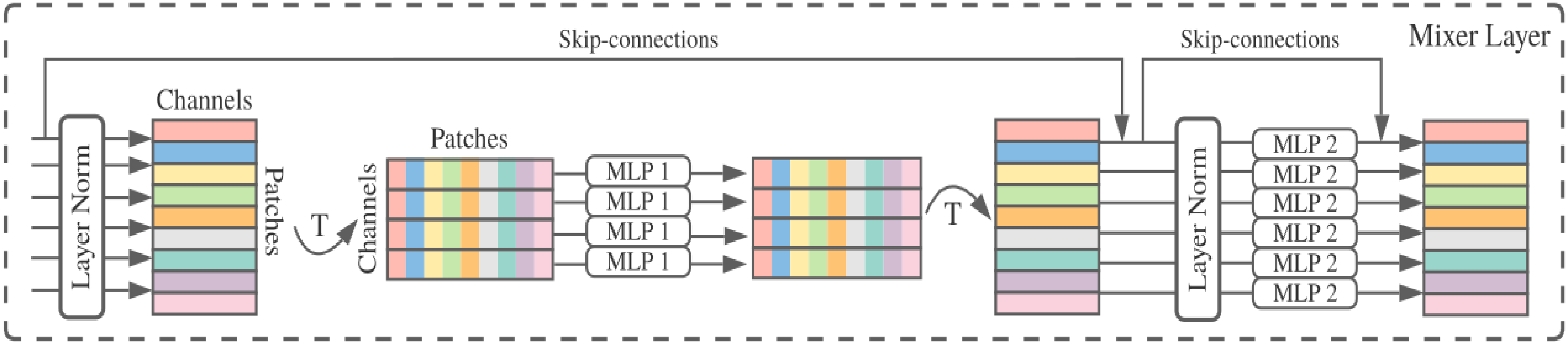
<u>Token mixing (MLP 1):</u>

communication between different **patches/tokens**

operate on each **channel** independently

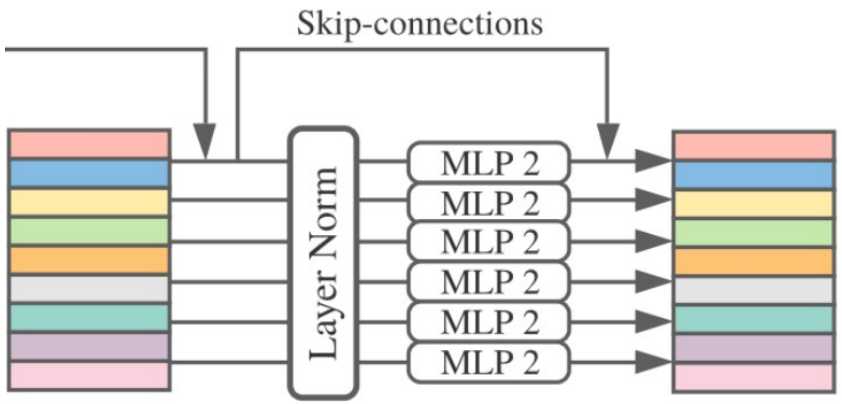
shared across all columns (**channels**)

mix features between different spatial locations



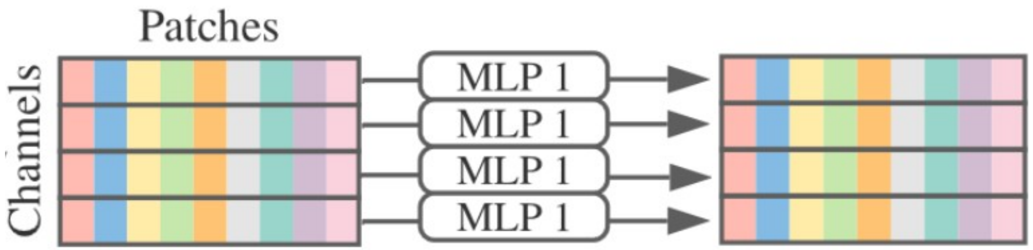
Mixer Layer is a special case of Convolutional block

<u>Channel mixing (MLP 2):</u>



It is a 1x1 convolution

<u>Token mixing (MLP 1):</u>



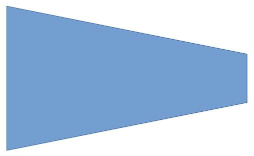
Separable convolutions with **parameter sharing** and full receptive field

.

Comparison to CNNs and Vision Transformers

	Mix features locally	Mix features across different locations
MLP-Mixer	Channel mixing	Token mixing
CNNs	NxN convolutions	NxN convolutions, N>1 Pooling Dilated convolutions Receptive field
Attention-based models	Keys, queries and values	Attention scores, Output

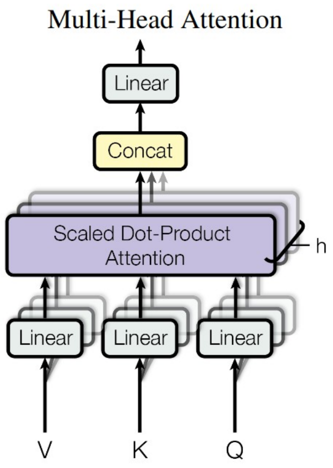
Comparison to CNNs and Vision Transformers

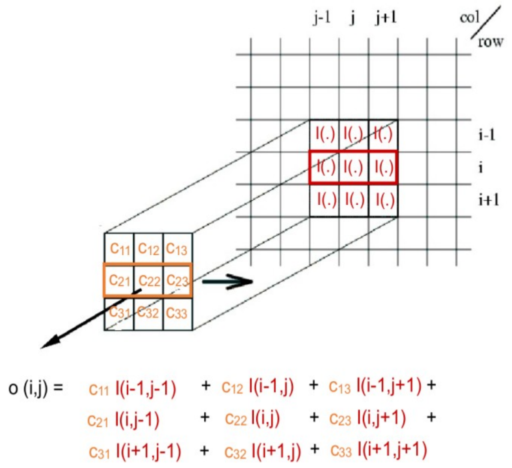
	Model Size as a function of input size	Input vs output
MLP-Mixer	Linear	
CNNs	-	
Attention-based models	Quadratic	

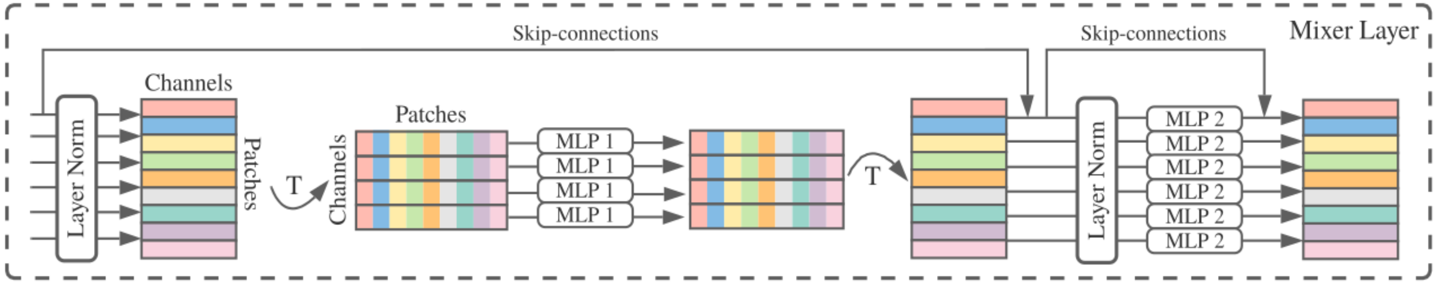
Experiments

Metrics

- Accuracy on downstream tasks (classification)
- Total pre-training cost
- Test time throughput







Downstream tasks

Dataset	# Images	# Classes
ILSVRC2012	1.3M	1k
CIFAR10/100	50k	10/100
Oxford-IIIT Pets	3.7k	36
Oxford Flowers-102	2k	102
VTAB-1k	19 x 1k	-

Oxford-IIIT Pets

Oxford Flowers-102

Pre-training

Dataset	# Images	# Classes
ImageNet	1M	1000
ImageNet-21k	14M	21k
JFT-300M	300M	18k

Data Augmentation + Regularization

-RandAugment

-Mixup

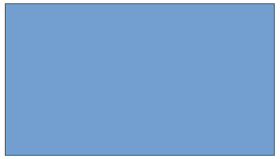
-Dropout

-Stochastic depth

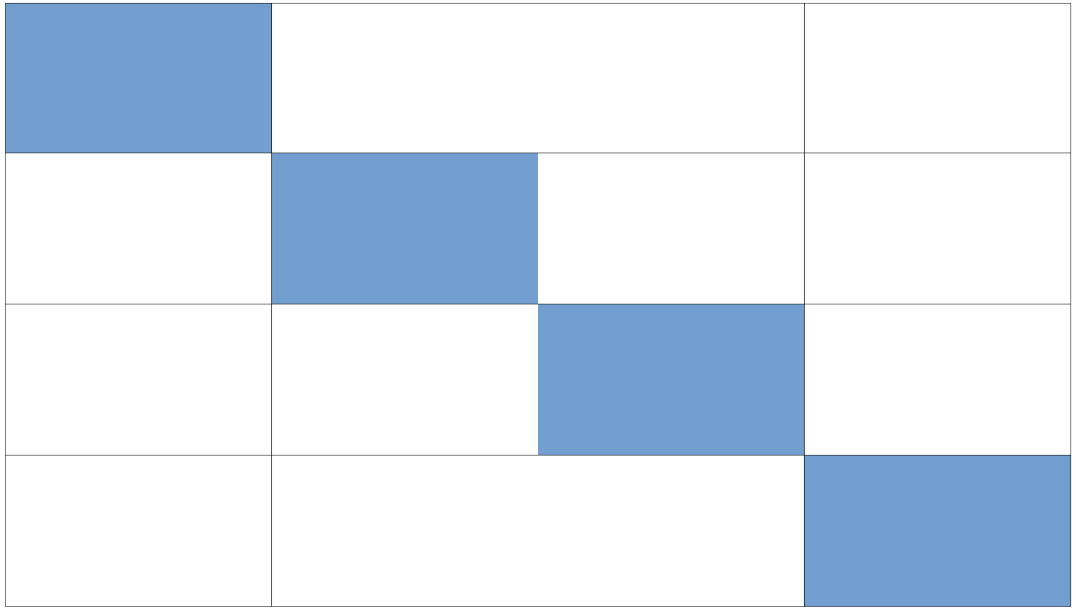
Fine-tuning

1. Fine tune at higher ressolution than pre-training.

2. Keep same patch size P, larger number of patches S



$$\mathbf{W}_1 \in \mathbb{R}^{D_S \times S}$$



$$\mathbf{W}_1' \in \mathbb{R}^{(K^2 \cdot D_S) \times (K^2 \cdot S)}$$

Models

Mixer architectures

B – Base

L – Large

H – Huge

Convolutional architectures

Big Transfer (BiT)

NFNets

MPL

ALIGN

Attention-based architectures

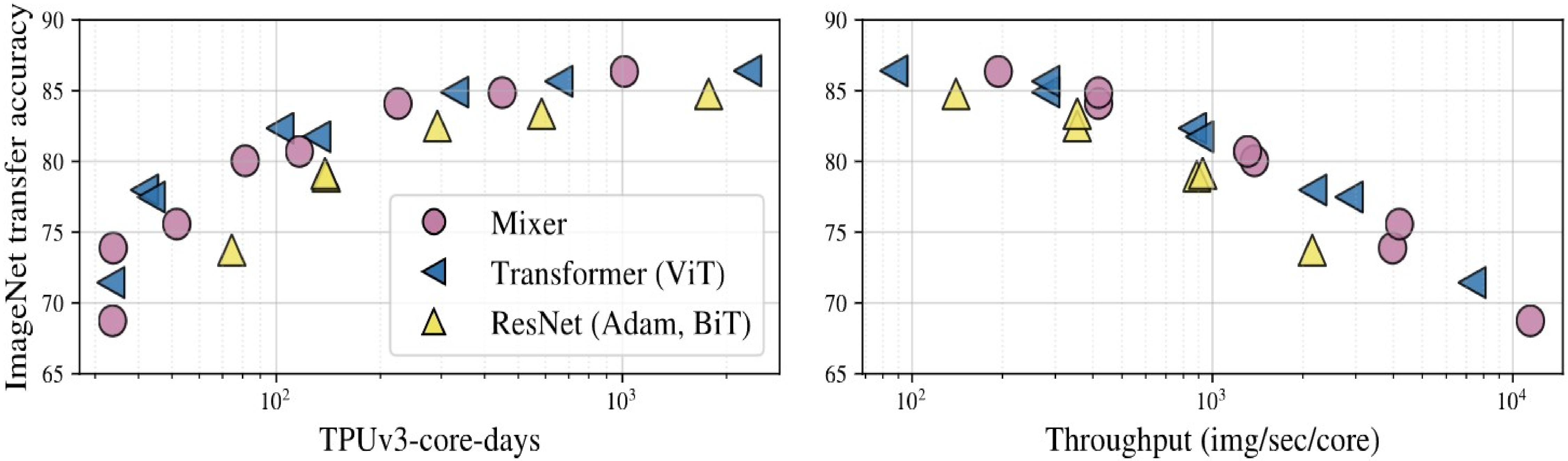
Vision Transformer (ViT)

HaloNets

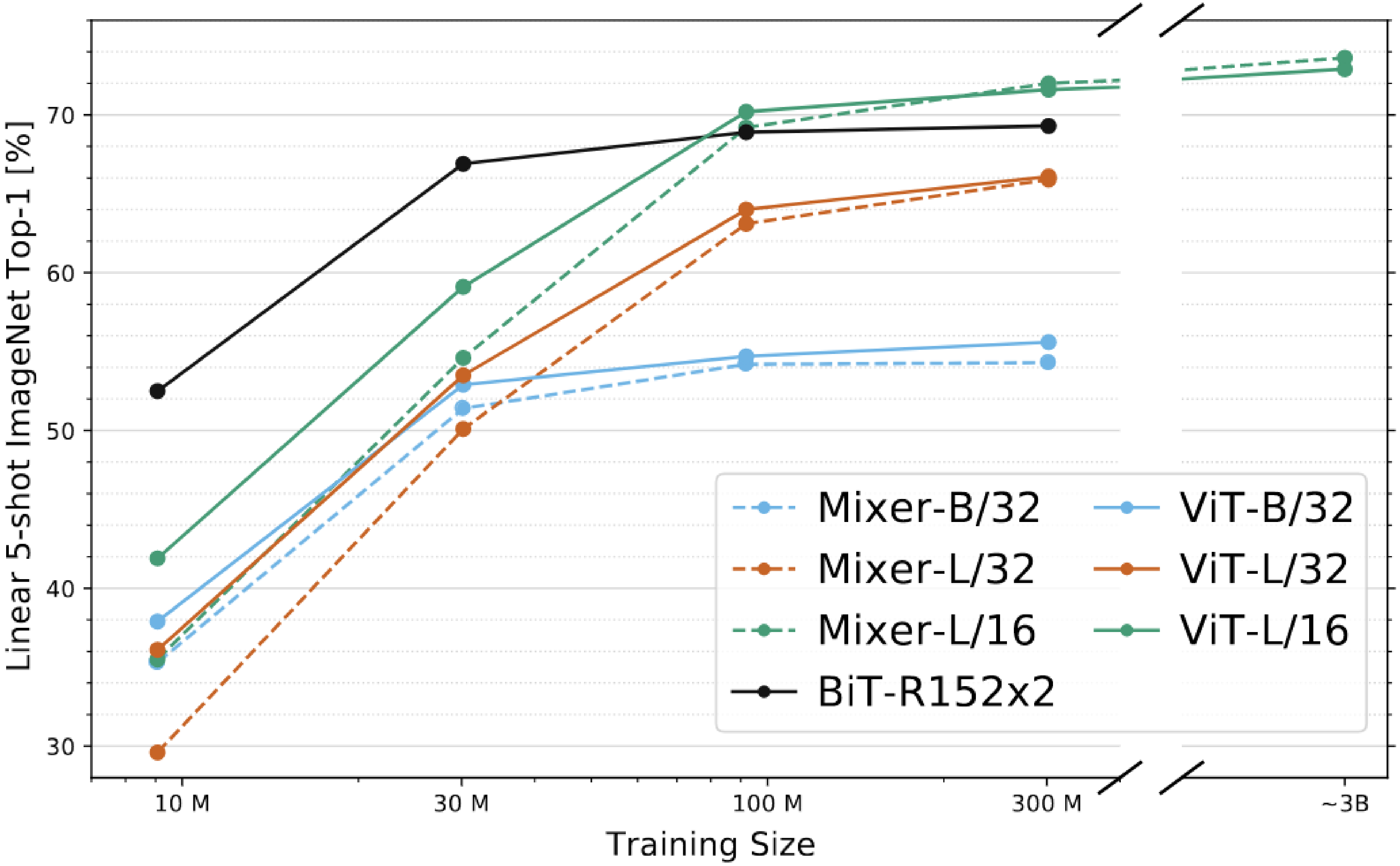
Results

	ImNet top-1	ReaL top-1	Avg 5 top-1	VTAB-1k 19 tasks	Throughput img/sec/core	TPUv3 core-days
Pre-trained on ImageNet-21k (public)						
HaloNet	85.8	—	—	—	120	0.10k
Mixer-L/16	84.15	87.86	93.91	74.95	105	0.41k
ViT-L/16	85.30	88.62	94.39	72.72	32	0.18k
BiT-R152x4	85.39	—	94.04	70.64	26	0.94k
Pre-trained on JFT-300M (proprietary)						
NFNet-F4+	89.2	—	—	—	46	1.86k
Mixer-H/14	87.94	90.18	95.71	75.33	40	1.01k
BiT-R152x4	87.54	90.54	95.33	76.29	26	9.90k
ViT-H/14	88.55	90.72	95.97	77.63	15	2.30k

Role of model size



Role of pre-training dataset size



Conclusions and related work

Mixer model is competitive with state of the art models in terms of the tradeoff between accuracy and computational costs.

Both for computer vision and other realms, it is worth exploring architectures beyond CNNs and attention-based networks.

Related work:

Pay Attention to MLPs, Liu et al.

TSMixer: Lightweight MLP-Mixer Model for Multivariate Time Series Forecasting, Chen et al.

Mixer is more than just a model, Ji et al.

Multi-Scale MLP-Mixer for image classification, Zhang et al.

pNLP-Mixer: an Efficient all-MLP Architecture for Language, Fusco et al.

Personal opinion

Challenge existing neural network architectures.

Model very well described.

Extensive comparison with CNNs and Attention-based networks.

Discuss things that didn't work.

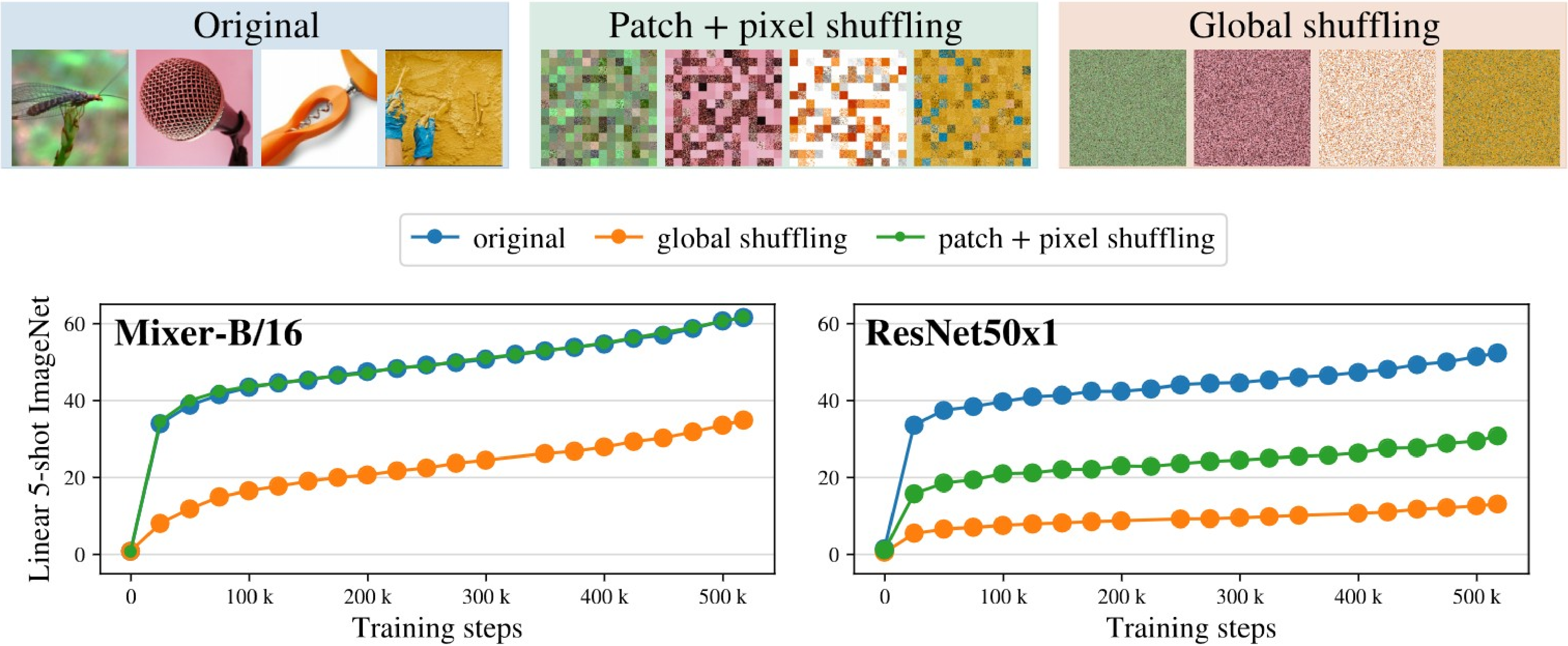
Experiments not very clear, for instance regularization.

Propose to move away from convolutions and self-attention but publish Vision Transformer one week earlier.

Parameter sharing for token mixer is only backed by empirical results.

Additional content

Invariance to input permutations



Related work

Skip connections, batch normalization.

Depth-wise convolutions.

Share parameters in depth-wise convolutions for NLP.

Augment CNNs with non-local operations.

Convert image to sequence of patches and embed them.

Fully connected network, data augmentation, pre-training with autoencoder.

Fully connected network with custom optimization and regularization.

Tables

	Image size	Pre-Train Epochs	ImNet top-1	ReaL top-1	Avg. 5 top-1	Throughput (img/sec/core)	TPUv3 core-days
Pre-trained on ImageNet (with extra regularization)							
● Mixer-B/16	224	300	76.44	82.36	88.33	1384	0.01k(‡)
● ViT-B/16 (☎)	224	300	79.67	84.97	90.79	861	0.02k(‡)
● Mixer-L/16	224	300	71.76	77.08	87.25	419	0.04k(‡)
● ViT-L/16 (☎)	224	300	76.11	80.93	89.66	280	0.05k(‡)
Pre-trained on ImageNet-21k (with extra regularization)							
● Mixer-B/16	224	300	80.64	85.80	92.50	1384	0.15k(‡)
● ViT-B/16 (☎)	224	300	84.59	88.93	94.16	861	0.18k(‡)
● Mixer-L/16	224	300	82.89	87.54	93.63	419	0.41k(‡)
● ViT-L/16 (☎)	224	300	84.46	88.35	94.49	280	0.55k(‡)
● Mixer-L/16	448	300	83.91	87.75	93.86	105	0.41k(‡)
Pre-trained on JFT-300M							
● Mixer-S/32	224	5	68.70	75.83	87.13	11489	0.01k
● Mixer-B/32	224	7	75.53	81.94	90.99	4208	0.05k
● Mixer-S/16	224	5	73.83	80.60	89.50	3994	0.03k
● BiT-R50x1	224	7	73.69	81.92	—	2159	0.08k
● Mixer-B/16	224	7	80.00	85.56	92.60	1384	0.08k
● Mixer-L/32	224	7	80.67	85.62	93.24	1314	0.12k
● BiT-R152x1	224	7	79.12	86.12	—	932	0.14k
● BiT-R50x2	224	7	78.92	86.06	—	890	0.14k
● BiT-R152x2	224	14	83.34	88.90	—	356	0.58k
● Mixer-L/16	224	7	84.05	88.14	94.51	419	0.23k
● Mixer-L/16	224	14	84.82	88.48	94.77	419	0.45k
● ViT-L/16	224	14	85.63	89.16	95.21	280	0.65k
● Mixer-H/14	224	14	86.32	89.14	95.49	194	1.01k
● BiT-R200x3	224	14	84.73	89.58	—	141	1.78k
● Mixer-L/16	448	14	86.78	89.72	95.13	105	0.45k
● ViT-H/14	224	14	86.65	89.56	95.57	87	2.30k
● ViT-L/16 [14]	512	14	87.76	90.54	95.63	32	0.65k

Tables

Specification	S/32	S/16	B/32	B/16	L/32	L/16	H/14
Number of layers	8	8	12	12	24	24	32
Patch resolution $P \times P$	32×32	16×16	32×32	16×16	32×32	16×16	14×14
Hidden size C	512	512	768	768	1024	1024	1280
Sequence length S	49	196	49	196	49	196	256
MLP dimension D_C	2048	2048	3072	3072	4096	4096	5120
MLP dimension D_S	256	256	384	384	512	512	640
Parameters (M)	19	18	60	59	206	207	431