MLP-Mixer: An all-MLP architecture for Vision

Presenters: Ronit Joshi, Soumitri Chattopadhyay

Date: 04/15/2024

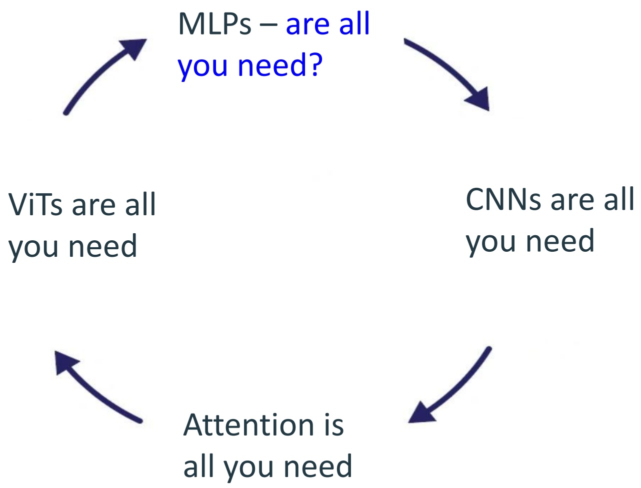
Motivation

- Going beyond ViTs – what's next that can scale up really well with large data/compute?
- Self-attention in ViTs are of quadratic complexity, which limits scalability. Can we do better?
- Is self-attention the only means of global information processing in images, or did we miss something simpler?

The solution lies in… well, the full cycle completion of computer vision: back to MLPs!

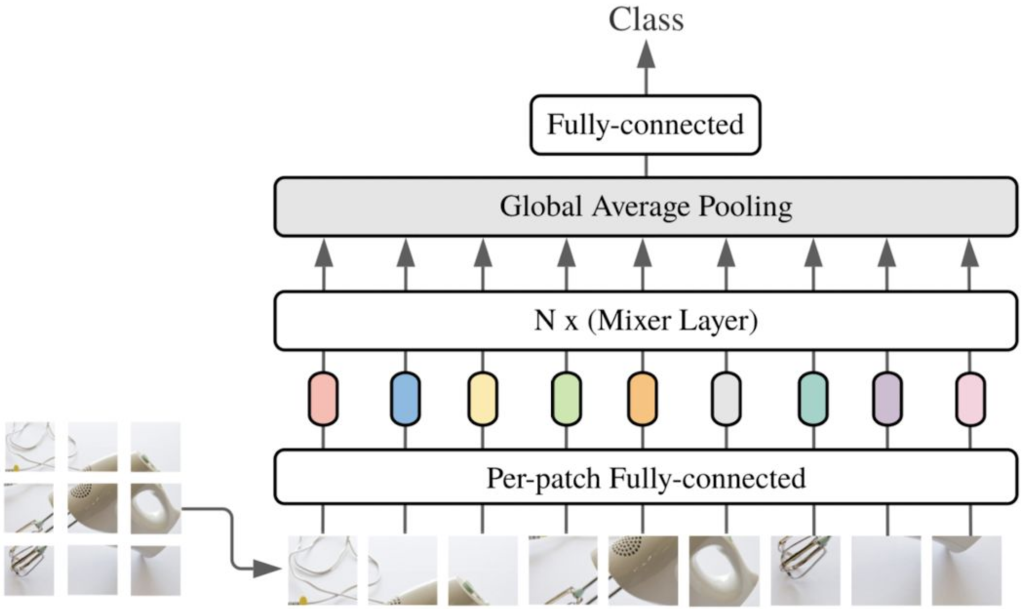
In a nutshell...

- No convolutions, no self-attention. Just **feature mixing**, pure MLP-based!
- No fancy computations/equations; Simple tensor reshapes, multiplications and non-linearity
- Quadratic **self-attention** is replaced by **linear** complexity **token+channel mixing** modules
- Achieves *surprisingly* competitive results against SOTA ViT/CNNs, and shows *great scalability* properties!



MLP-Mixer:
Proposed Architecture

High-level architecture



- Similar processing as ViTs ⟶ BUT, **mixer layers** instead of self-attention
- Standard classification head: Global average pooling layer + Linear classifier

Types of MLP layers

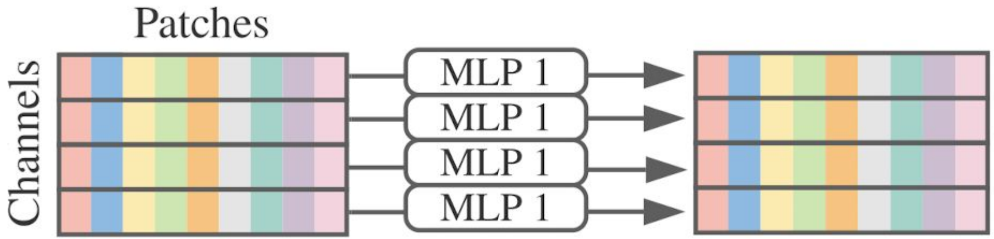
Channel-mixing MLPs

- Each image token has C channels
- Allows communication between different channels
- Operate on tokens independently

Token-mixing MLPs

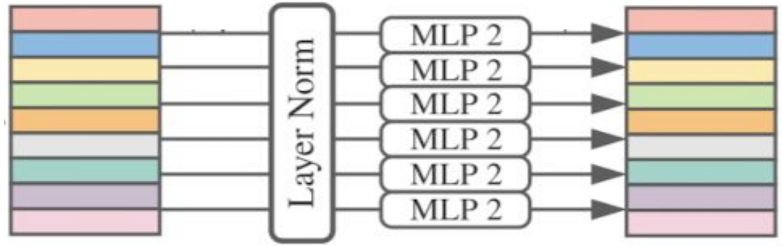
- Allows communication between tokens
- Operates on channels independently

Token-mixing MLP



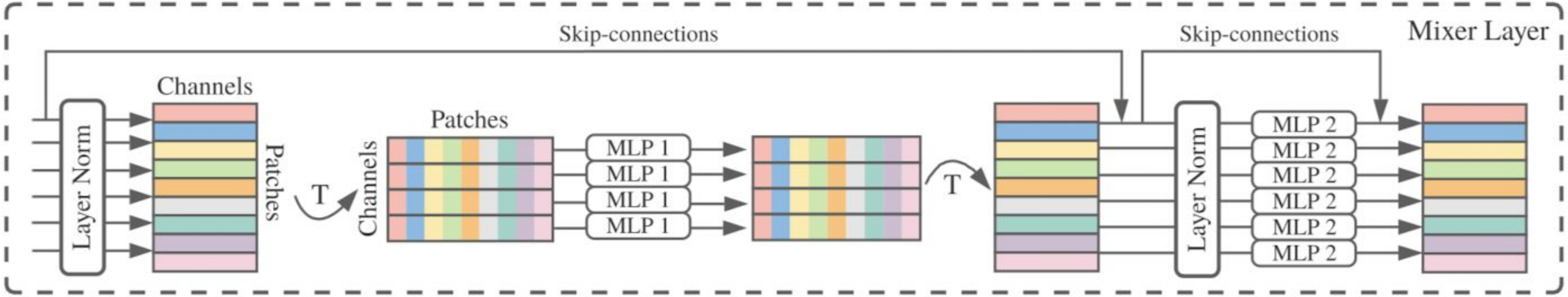
- “Cross location” operation
  - Performed by CNNs (N x N convolutions for N > 1) and larger kernels and transformers
- $\mathbf{X} \in \mathbb{R}^{S \times C}$ is the given input, where S is the number of image patches, and C is the number of channels. This MLP is applied on the **columns** of X (i.e. applied to X transpose)

Channel-mixing MLP



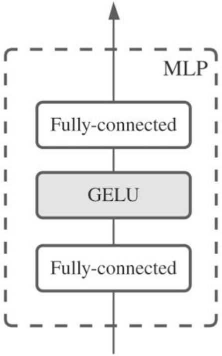
- Applied across all token features independently
- Same MLP layer, shared parameters across all token features
- Aggregates channel information across all tokens

Mixer layer



$$\mathbf{U}_{*,i} = \mathbf{X}_{*,i} + \mathbf{W}_2\,\sigma\big(\mathbf{W}_1\,\mathrm{LayerNorm}(\mathbf{X})_{*,i}\big), \qquad \text{for } i = 1 \ldots C,$$
$$\mathbf{Y}_{j,*} = \mathbf{U}_{j,*} + \mathbf{W}_4\,\sigma\big(\mathbf{W}_3\,\mathrm{LayerNorm}(\mathbf{U})_{j,*}\big), \qquad \text{for } j = 1 \ldots S.$$

σ is an element-wise nonlinearity (GELU)



- The **same channel-mixing MLP** and the **same token-mixing MLP** is applied.
  - Using parameters across channels is not common.
  - Leads to significant memory savings, and doesn't affect performance!
- No position embeddings; token-mixing MLPs are sensitive to ordering

Experiments
&
Results

Experimental Setup

- **Downstream task:** Image classification
- **Datasets:**
  - JFT-300M, ImageNet-21k (pre-training), ImageNet-1k (pre-training + evaluation)
  - CIFAR-10/100, Oxford-IIIT Pets (37 classes), Oxford-Flowers (102 classes), Visual Task Adaptation Benchmark (VTAB)
- **Metrics:**
  - Top-1 accuracy
  - TPUv3-core-days (pre-training time)
  - Throughput (images/sec/core)
- **Model variants:**
  - Scale variants similar to ViTs - Small (S/32), Base (B/32, B/16), Large (L/32, L/16), Huge (H/14)
- **Competitors:**
  - Vanilla ViT with its scale variants
  - ResNet-based BiT

Empirical Results: Transfer Learning

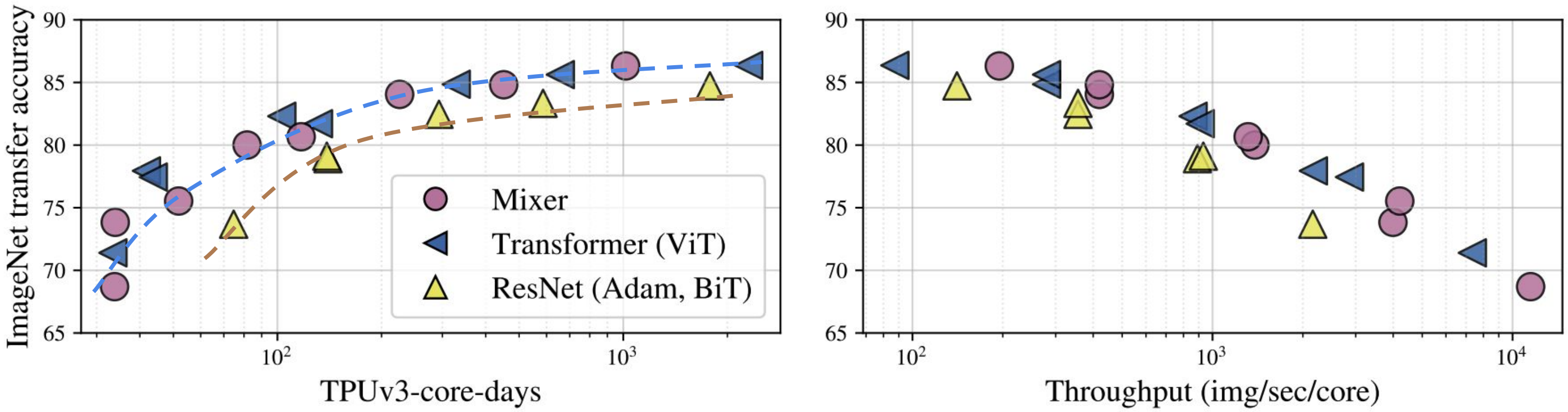
- Mixer achieves top-1 acc. **competitive** to SOTA
- Gap **reduces** with increase in pre-training data (IN-21k → JFT-300M)
- **Throughput** of Mixer is way superior w.r.t. ViTs or CNNs (i.e. BiT)

Mixer yields **superior accuracy vs throughput** tradeoff.

	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 [51]	85.8	—	—	—	120	0.10k
Mixer-L/16	84.15	87.86	93.91	74.95	105	0.41k
ViT-L/16 [14]	85.30	88.62	94.39	72.72	32	0.18k
BiT-R152x4 [22]	85.39	—	94.04	70.64	26	0.94k
Pre-trained on JFT-300M (proprietary)						
NFNet-F4+ [7]	89.2	—	—	—	46	1.86k
Mixer-H/14	87.94	90.18	95.71	75.33	40	1.01k
BiT-R152x4 [22]	87.54	90.54	95.33	76.29	26	9.90k
ViT-H/14 [14]	88.55	90.72	95.97	77.63	15	2.30k
Pre-trained on unlabelled or weakly labelled data (proprietary)						
MPL [34]	90.0	91.12	—	—	—	20.48k
ALIGN [21]	88.64	—	—	79.99	15	14.82k

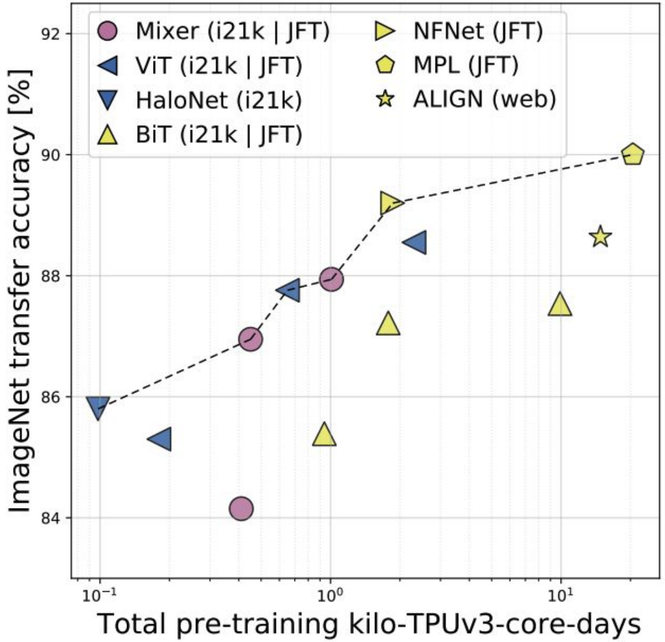
Effect of Model Scaling

- Both ViTs and Mixer scale well w.r.t. compute budget, and lead over CNNs **(Left figure)**
- For given top-1 accuracy, Mixer (and ViTs) have higher throughput w.r.t. CNN
- For given model size, Mixer has higher throughput vs. ViT (albeit lower top-1 acc. score)



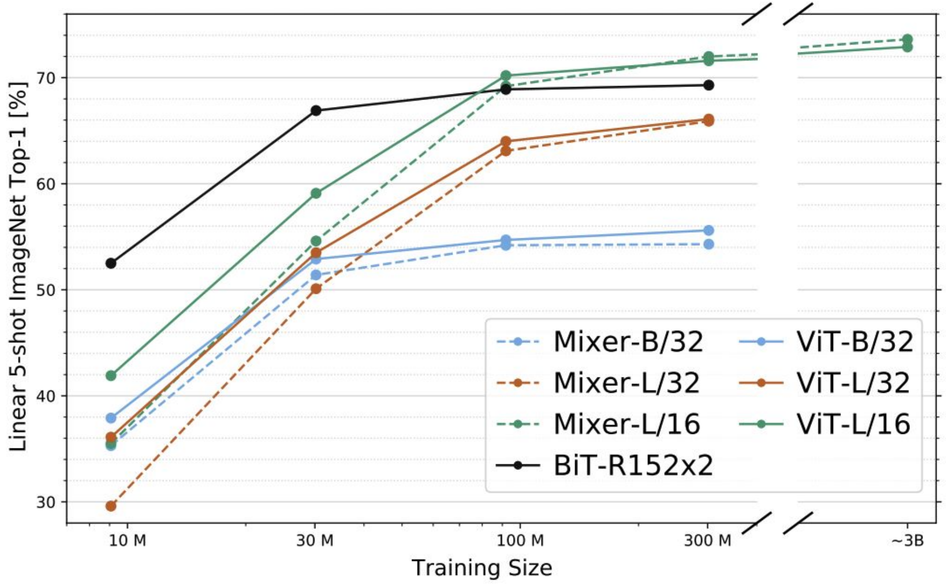
Effect of Compute Scaling

- Points on Pareto frontier (dashed line) depicts there **cannot** be a change in y (or x) *without incurring* a change in x (or y) i.e. indicates **a trade-off**
- Both Mixer and ViT points follow the Pareto frontier, depicting the compute-vs-performance trade-off
- Sort of assurance that with higher compute scaling, Mixers would yield better performances



Effect of Scaling Pre-training Data

- CNN fares better than Mixer/ViT at low data regimes
- CNN quickly saturates with increased training data; ViTs and Mixer scale better
- Smaller variants (Mixer-B, ViT-B) saturate out quicker than larger variants
- Very high scaling of training data ⇒ larger variants (L/16, L/32) of Mixer converges to / outperforms ViTs



Empirical Results: Scaling MLP-Mixers

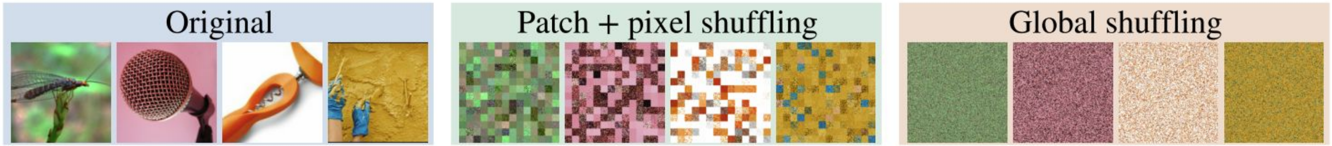
- Summary of scaling evaluations of Mixers w.r.t. CNNs/ViTs – (a) model sizes (**B**ase, **L**arge, **H**uge), (b) pretraining scales (IN-1k, IN-21k, JFT-300M), (c) input resolutions (224, 448)
- Mixers consistently show better throughput, scales better than ViTs (both model size and training data) and achieves competitive performance to SOTA.

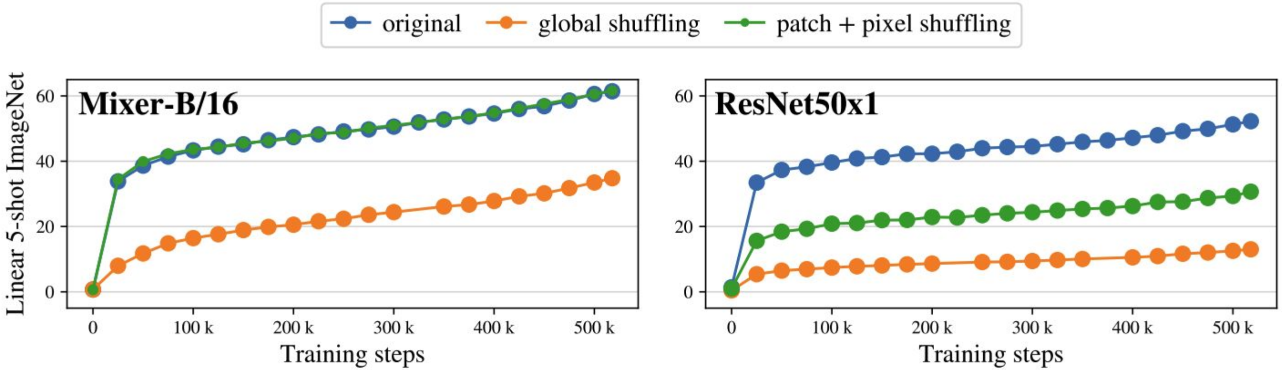
	Image size	Pre-Train Epochs	ImNet top-1	ReaL top-1	Avg. 5 top-1	Throughput (img/sec/core)	TPUv3 core-days
Pre-trained on ImageNet (with extra regularization)							
Mixer-B/16	224	300	76.44	82.36	88.33	1384	0.01k[‡]
ViT-B/16 (☎)	224	300	79.67	84.97	90.79	861	0.02k[‡]
Mixer-L/16	224	300	71.76	77.08	87.25	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[‡]

	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 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

Inductive Biases: Mixer vs. CNNs

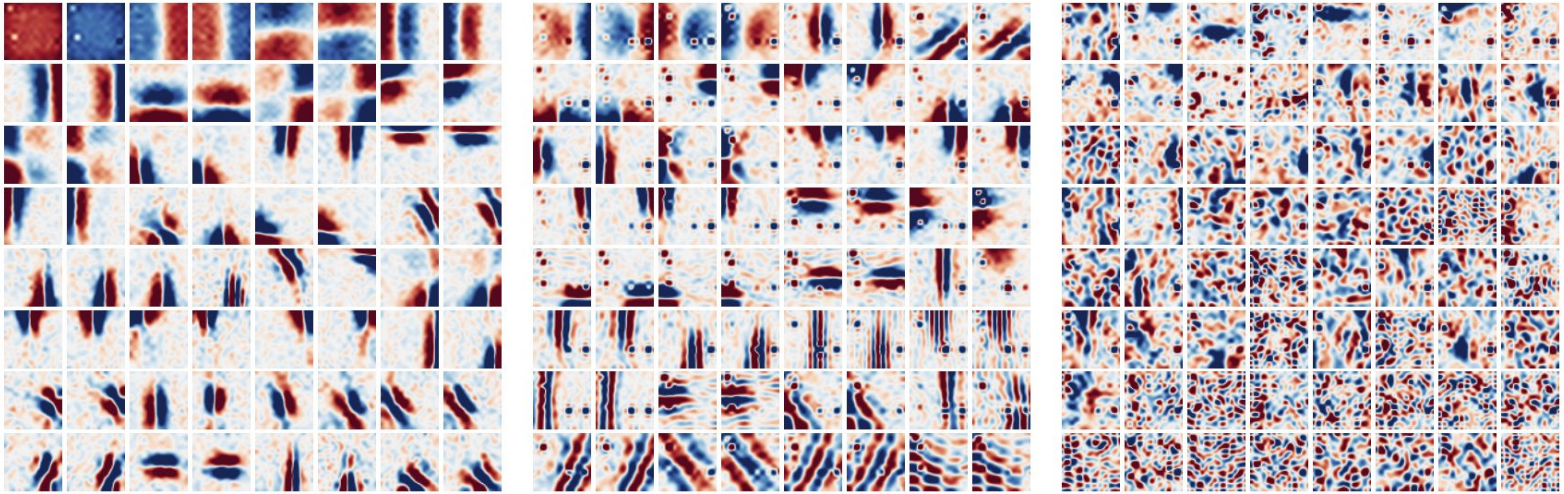
- Mixer is **invariant** to the order of patches *and* pixels within the patches (original = patch+pixel shuffling)
- For global shuffling: Performance drop for Mixer (45%) is **less** compared to CNN (75%)





Feature Visualisations

- While early CNN layers learn local spatial features, **token-mixing** MLPs allow global feature learning.
- Some Mixer-learned features (even early blocks) operate at global level, others at local regions. Deeper layers have **no** identifiable structure.



Summing up...

Key takeaways:

- An all-MLP architecture is a lot simpler than CNN/ViT, but shows very competitive performance to these SOTA models
- Token-mixing MLPs learns global features while being of linear complexity – more efficient vs. quadratic complexity of self-attention in ViTs
- Mixer shows high scalability w.r.t. training data, compute and model capacity – better scaling vs. ViTs
- Shows superior throughput compared to ViTs at a nominal performance expenses, esp. at high capacities

[Question] Does this imply that any network, no matter how simple, with sufficiently high compute + data + capacity, can yield competitive performances on image classification benchmarks?

Summing up...

Thank you!

Questions?