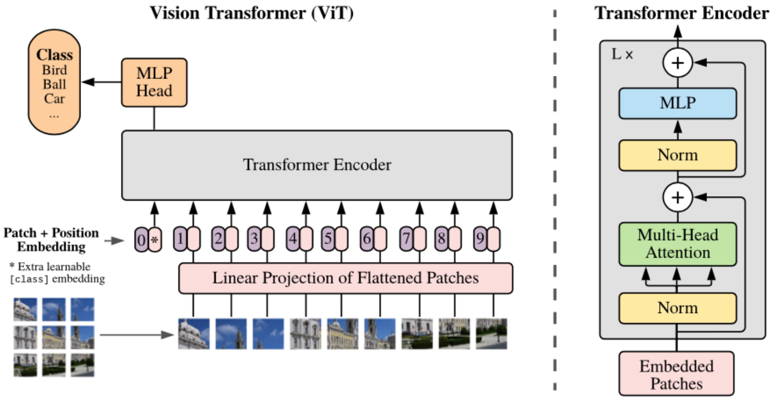
Computer vision and classification deep architectures TRANSFOMERS

Ronan Sicre

Vision Transformers

Transformers use multi-head attention on sequence of patches.



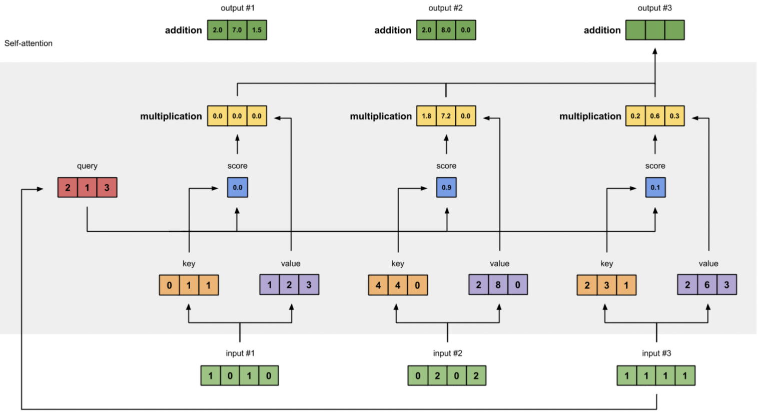
An image is worth 16x16 words: Transformers for image recognition at scale

Vision Transformers

https://towardsdatascience.com/illustrated-self-attention-2d627e33b20a

Vision Transformers

$$Attention(Q, K, V) = Softmax(\frac{QK^T}{\sqrt{d}}V)$$

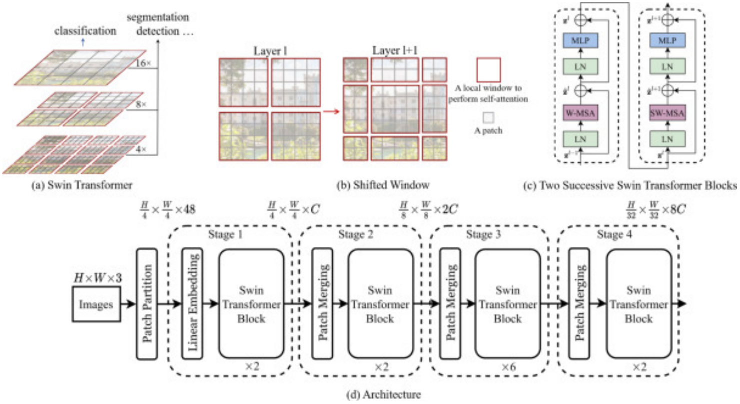


Transformer architectures: Swin Transformers

Shifted windows

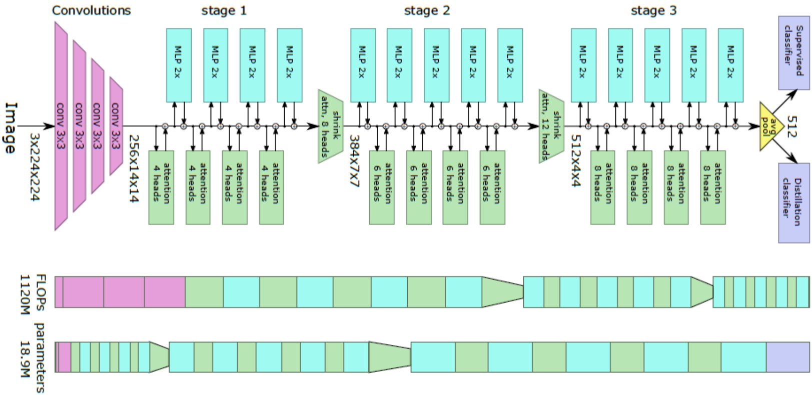
Hierarchical filters (pyramids)

Can cope with multiple down stream tasks (object localization)



Combining CNNs and transformers: LeViT

CNN embedding



Combining CNNs and transformers: Conformer

CNN and transformers in parallel

Orthogonal connections at every block

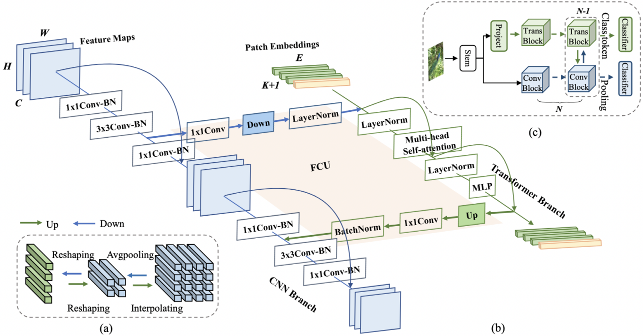
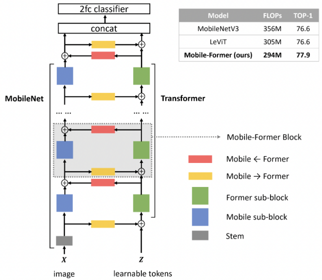


Figure 2: Network architecture of the proposed Conformer. (a) Up-sampling and down-sampling for spatial alignment of feature maps and patch embeddings. (b) Implementation details of the CNN block, the transformer block, and the Feature Coupling Unit (FCU). (c) Thumbnail of Conformer.

Combining CNNs and transformers: Mobileformer

mobileNet v3 + LeViT



Model	FLOPs	TOP-1
MobileNetV3	356M	76.6
LeViT	305M	76.6
Mobile-Former (ours)	**294M**	**77.9**

Figure 1. **Overview of Mobile-Former**, which parallelizes MobileNet [26] on the left side and Transformer [36] on the right side. Different from vision transformer [9] that uses image patches to form tokens, the transformer in Mobile-Former takes *very few learnable tokens* as input that are randomly initialized. *Mobile* (refers to MobileNet) and *Former* (refers to transformer) communicate through a bidirectional bridge, which is modeled by the proposed light-weight cross attention. Best viewed in color.

Self-supervised models

RotNet, Deep Cluster, BYOL, DINO, iBOT

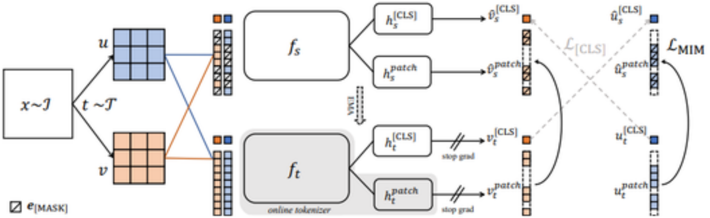


Figure 3: **Overview of iBOT framework, performing masked image modeling with an *online tokenizer*.** Given two views u and v of an image x, each view is passed through a teacher network $h_t \circ f_t$ and a student network $h_s \circ f_s$. iBOT minimizes two losses. The first loss $\mathcal{L}_{[CLS]}$ is self-distillation between cross-view [CLS] tokens. The second loss $\mathcal{L}_{MIM}$ is self-distillation between in-view patch tokens, with some tokens masked and replaced by $e_{[MASK]}$ for the student network. The objective is to reconstruct the masked tokens with the teacher networks' outputs as supervision.