Master in Computer Vision Barcelona

[http://pagines.uab.cat/mcv/]

Xavier Giro-i-Nieto

@DocXavi
xavier.giro@upc.edu

Associate Professor
Universitat Politècnica de Catalunya

Module 6 - Day 8 - Lecture 2

The Transformer in Vision

31th March 2022

Outline

1. Vision Transformer (ViT)
2. Beyond ViT

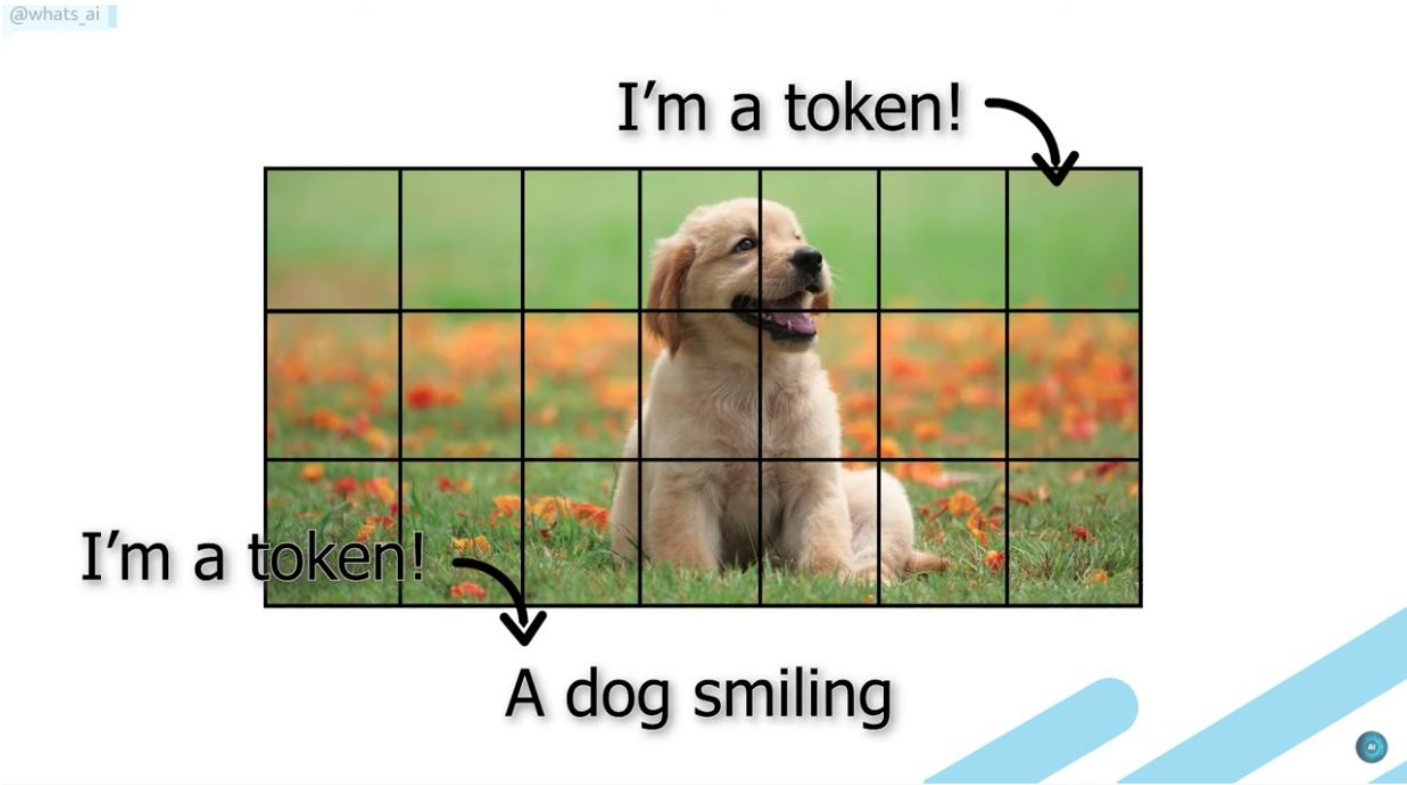
The Transformer for Vision: ViT

#ViT Dosovitskiy, Alexey, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani et al. "An image is worth 16x16 words: Transformers for image recognition at scale." ICLR 2021. [blog] [code] [video by Yannic Kilcher]

Outline

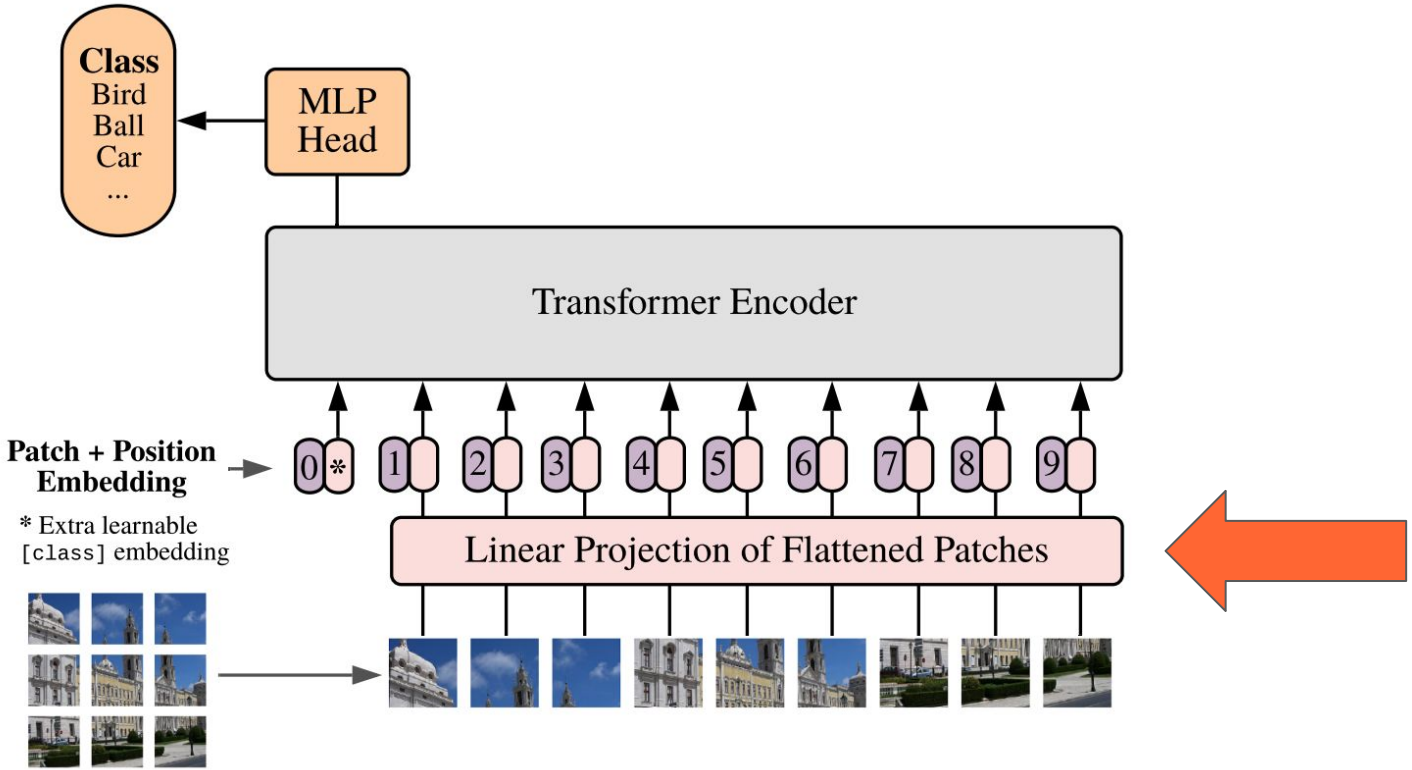
1. Vision Transformer (ViT)
    a. Tokenization
    b. Position embeddings
    c. Class embedding
    d. Receptive field
    e. Performance
2. Beyond ViT

The Transformer for Vision



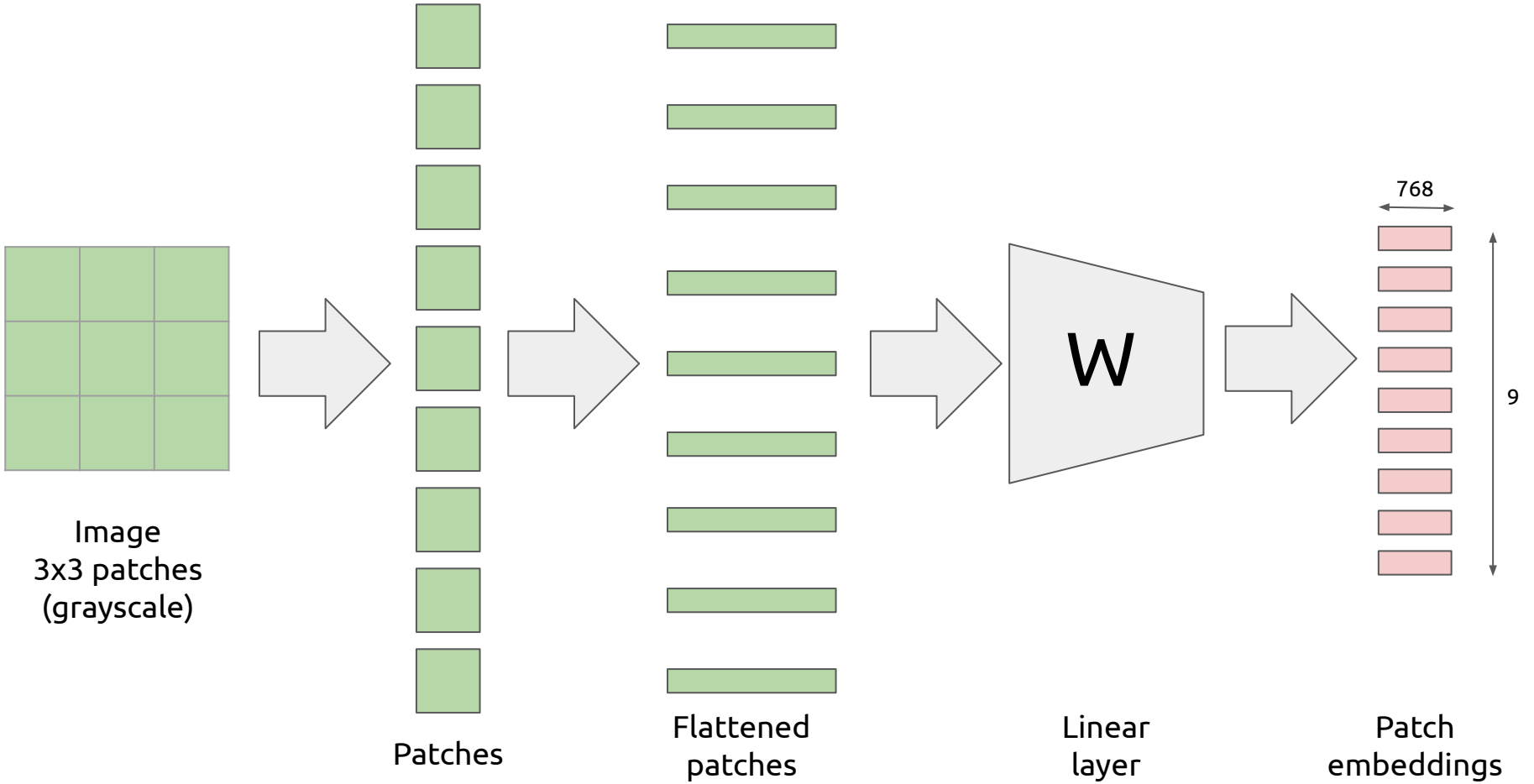
Source: What's AI, Will Transformers Replace CNNs in Computer Vision? + NVIDIA GTC Giveaway (2021)

The Transformer for Vision: ViT



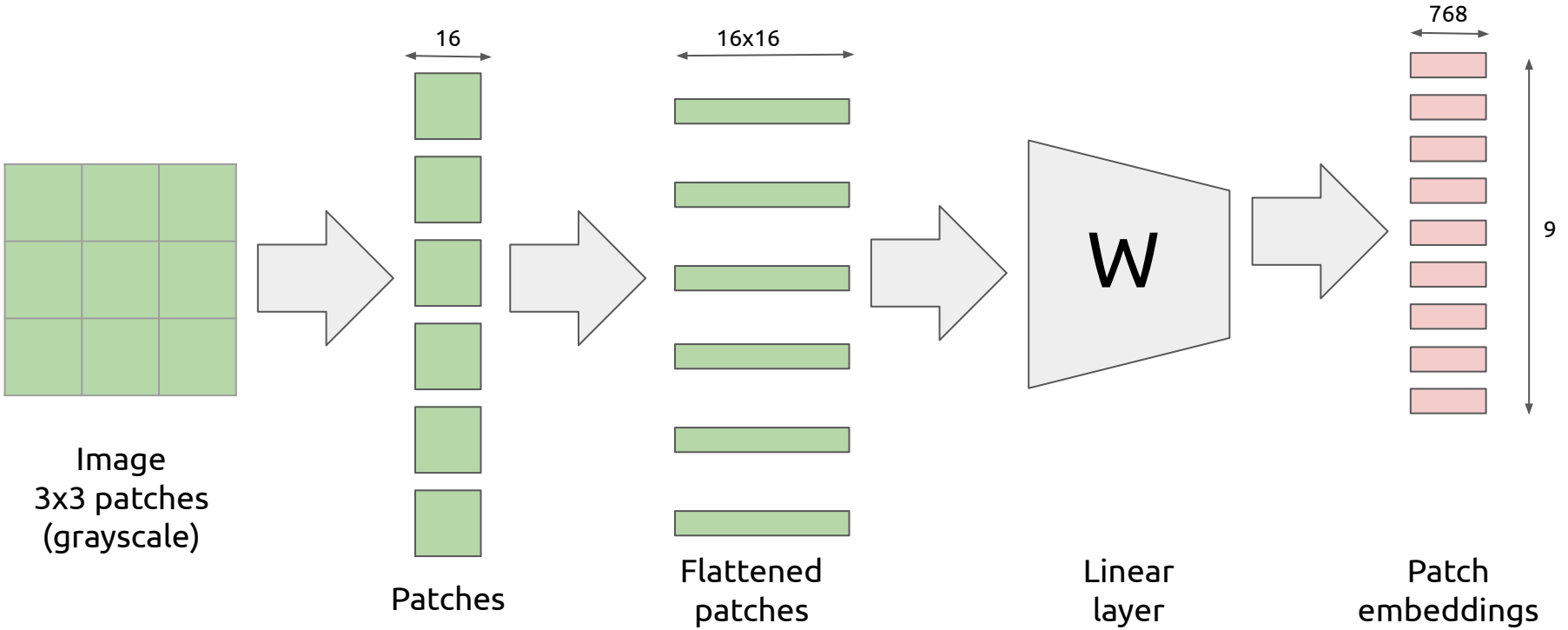
#ViT Dosovitskiy, Alexey, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani et al. "An image is worth 16x16 words: Transformers for image recognition at scale." ICLR 2021. [blog] [code] [video by Yannic Kilcher]

Linear projection of Flattened Patches



The Transformer for Vision

Consider the case of patches of 16x16 pixels and their embedding size of D=768, as in ViT-Base. How could the linear layer be implented with a convolutional layer ?



The Transformer for Vision

...

Consider the case of patches of 16x16 pixels and their embedding size of D=768, as in ViT-Base. How could the linear layer be implented with a convolutional layer ?

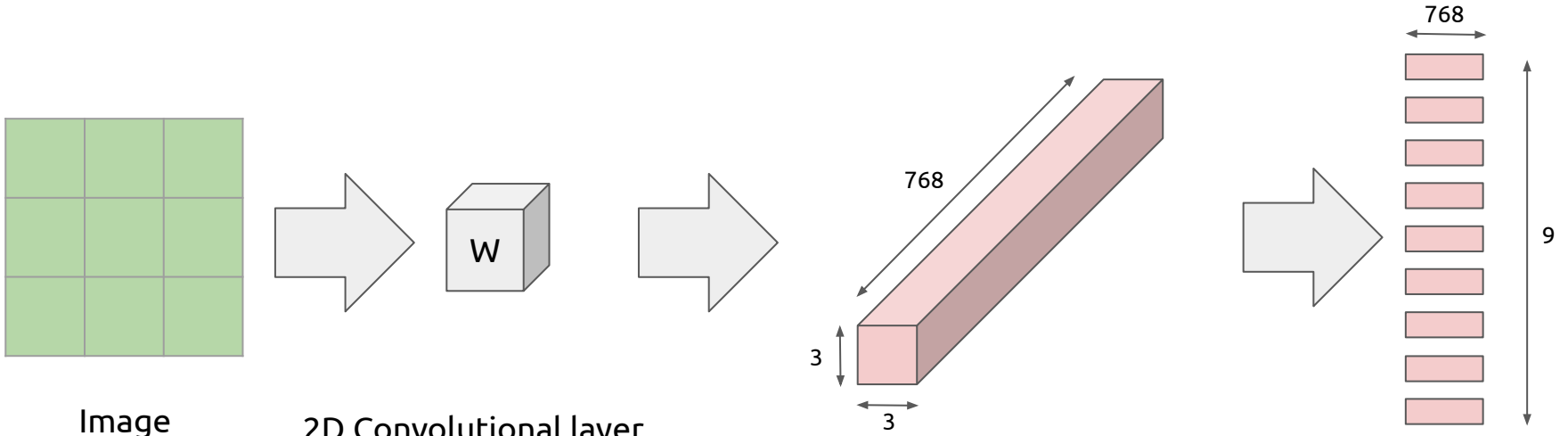


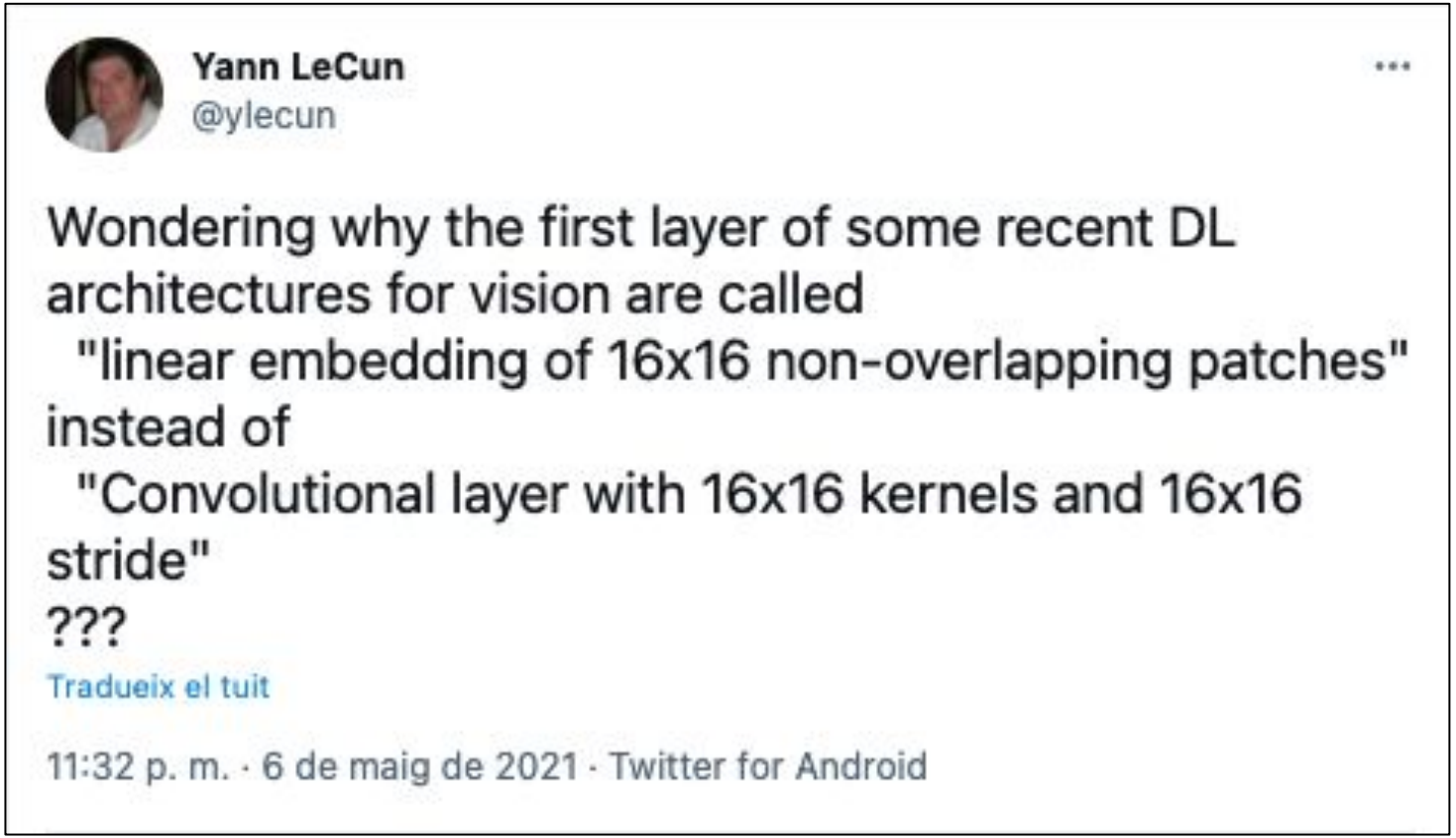
Image
3x3 patches
(grayscale)

2D Convolutional layer

768 filters
Kernel size = 16x16
Stride = 16x16

Patch
embeddings

The Transformer for Vision



Yann LeCun
@ylecun

Wondering why the first layer of some recent DL architectures for vision are called
"linear embedding of 16x16 non-overlapping patches"
instead of
"Convolutional layer with 16x16 kernels and 16x16 stride"
???

Tradueix el tuit

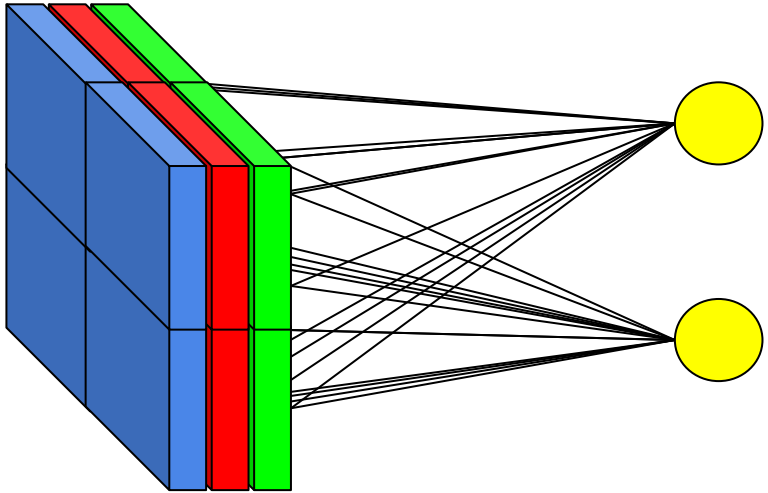
11:32 p. m. · 6 de maig de 2021 · Twitter for Android

The Transformer for Vision

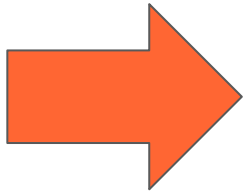
Observation: Fully connected neurons could be implemented as convolutional ones.

3x2x2 tensor
(RGB image of 2x2)

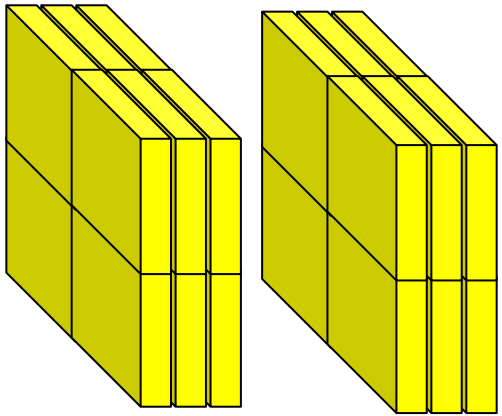
2 fully connected
neurons



(3x2x2 x 2) weights + 1 bias



2 convolutional filters of 3 x 2 x 2
(same size as input tensor)

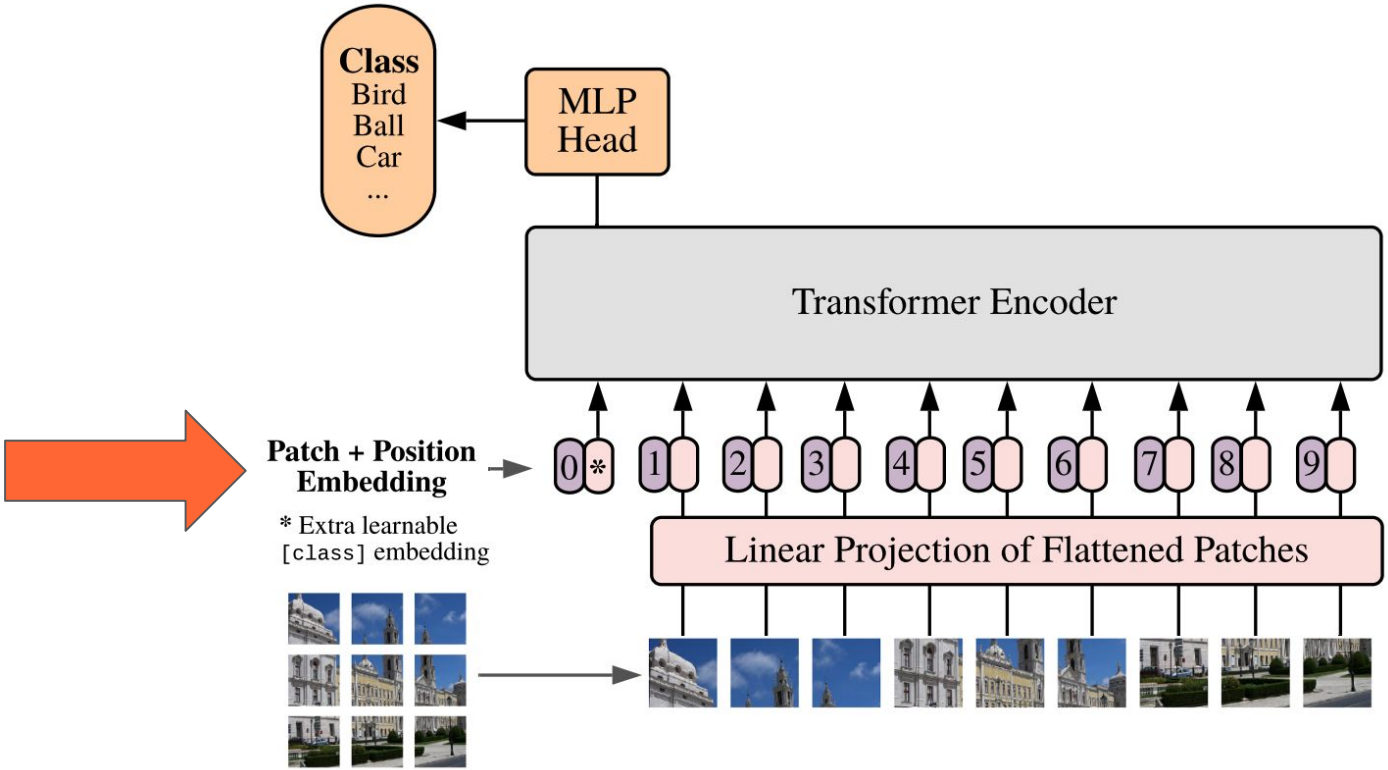


(3x2x2 x 2) weights + 1 bias

Outline

1. Vision Transformer (ViT)
    a. Tokenization
    b. Position embeddings
    c. Class embedding
    d. Receptive field
    e. Performance

Position Embeddings

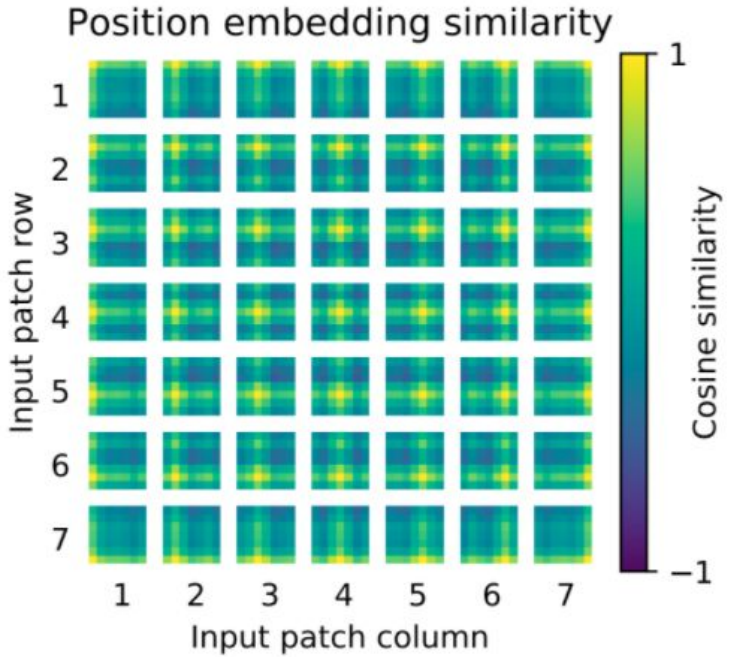


#ViT Dosovitskiy, Alexey, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani et al. "An image is worth 16x16 words: Transformers for image recognition at scale." ICLR 2021. [blog] [code] [video by Yannic Kilcher]

Position embeddings

The model **learns** to encode the relative position between patches.

Each position embedding is most similar to others in the same row and column, indicating that the model has recovered the grid structure of the original images.



#ViT Dosovitskiy, Alexey, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani et al. "An image is worth 16x16 words: Transformers for image recognition at scale." ICLR 2021. [blog] [code] [video by Yannic Kilcher]

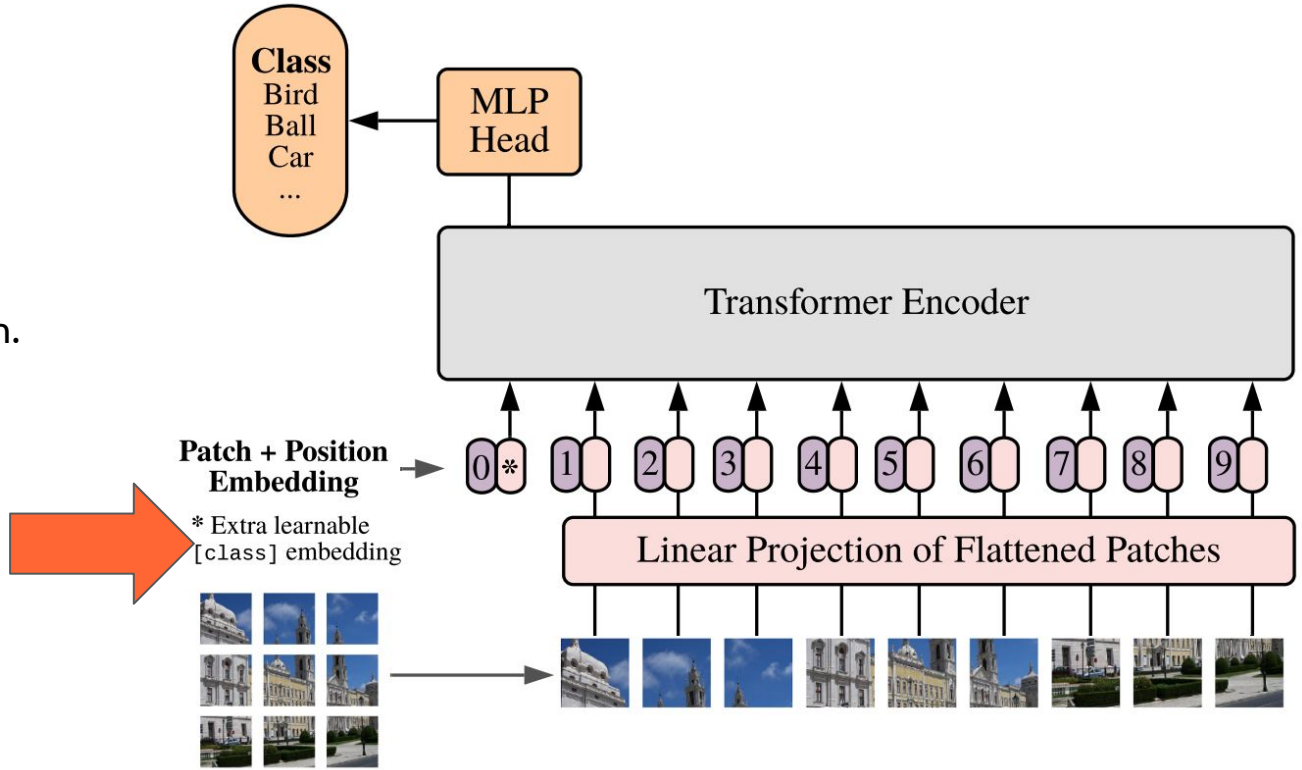
Outline

1. Vision Transformer (ViT)
    a. Tokenization
    b. Position embeddings
    c. Class embedding
    d. Receptive field
    e. Performance

Class embedding

[class] is a special learnable embedding added in front of every input example.

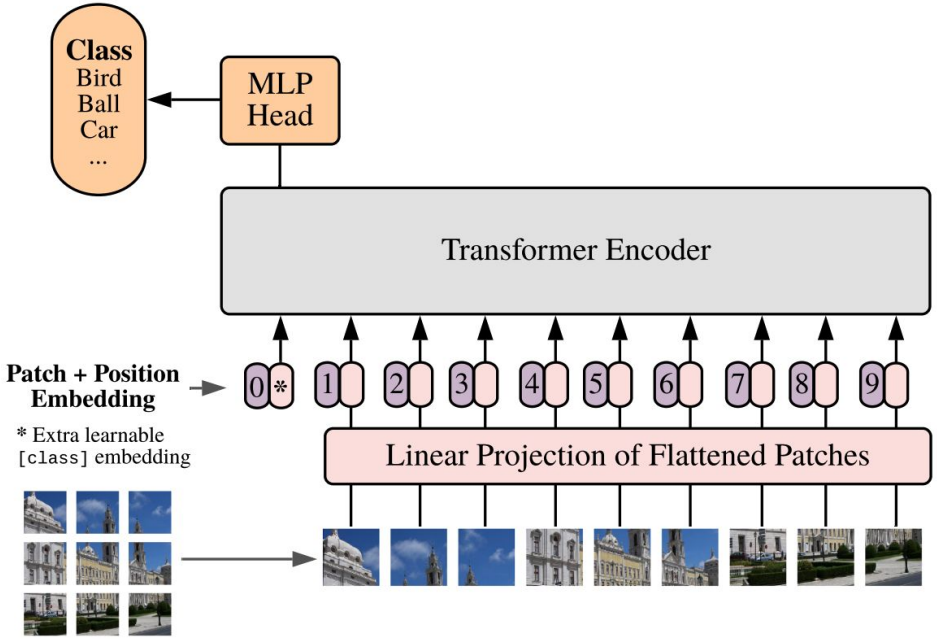
It triggers the class prediction.



#BERT Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. "Bert: Pre-training of deep bidirectional transformers for language understanding." NAACL 2019.

Class embedding

Why does the ViT not have a decoder in its architecture ?

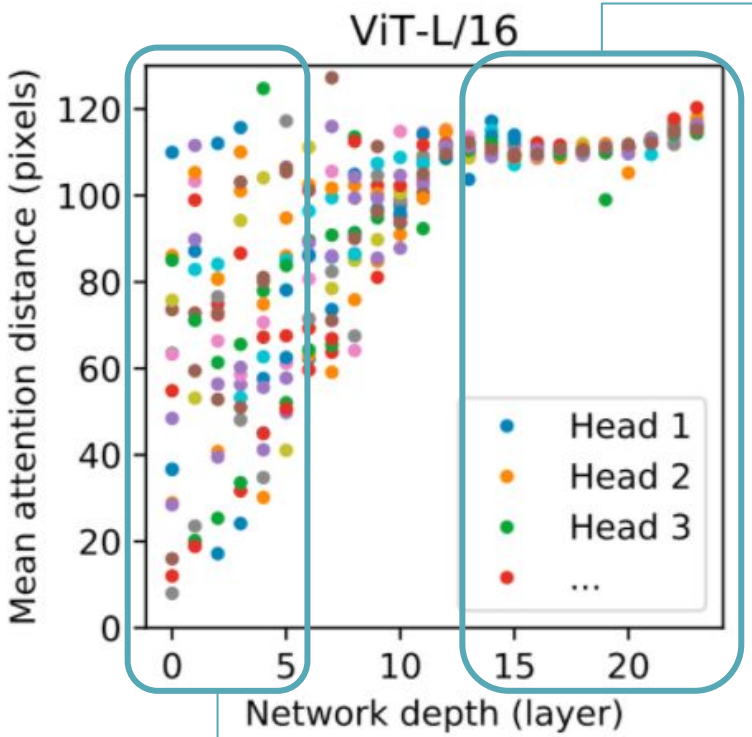


Outline

1. Vision Transformer (ViT)
    a. Tokenization
    b. Position embeddings
    c. Class embedding
    d. Receptive field
    e. Performance

Receptive field

Average spatial distance between one element attending to another for each transformer block:



Deeper layers attend all over the image.

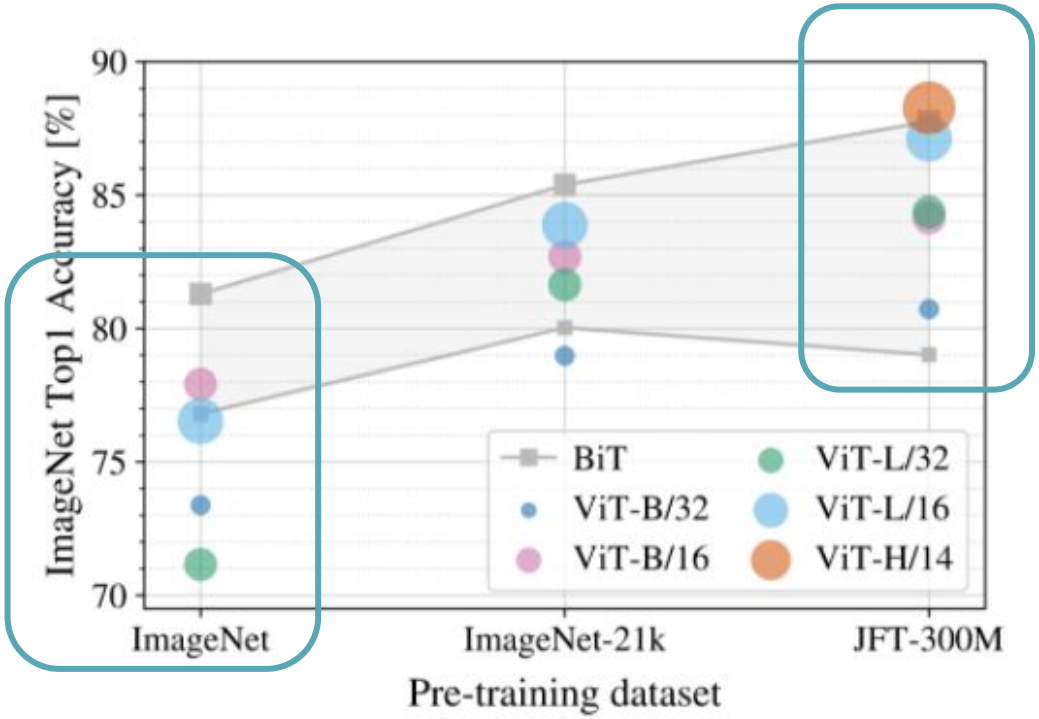
Both short & wide attention ranges in early layers (CNN can only learn short ranges)

Outline

1. Vision Transformer (ViT)
    a. Tokenization
    b. Position embeddings
    c. Class embedding
    d. Receptive field
    e. Performance

Performance: Accuracy

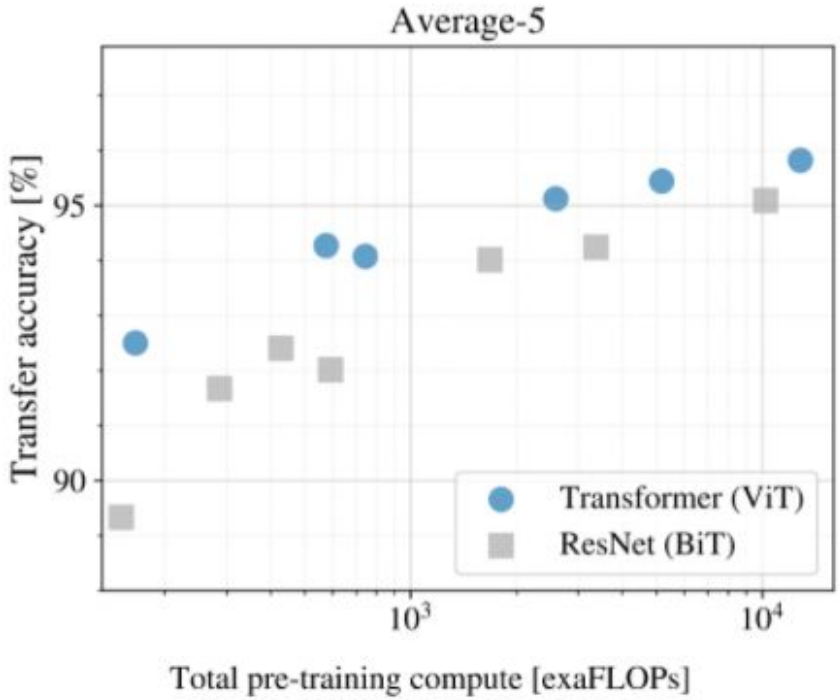
Worse performance than CNN (BiT) with ImageNet data only.



Slight improvement over CNN (BiT) when very large amounts of training data available.

#BiT Kolesnikov, Alexander, Lucas Beyer, Xiaohua Zhai, Joan Puigcerver, Jessica Yung, Sylvain Gelly, and Neil Houlsby. "Big transfer (bit): General visual representation learning." ECCV 2020.
#ViT Dosovitskiy, Alexey, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani et al. "An image is worth 16x16 words: Transformers for image recognition at scale." ICLR 2021. [blog] [code] [video by Yannic Kilcher]

Performance: Computation



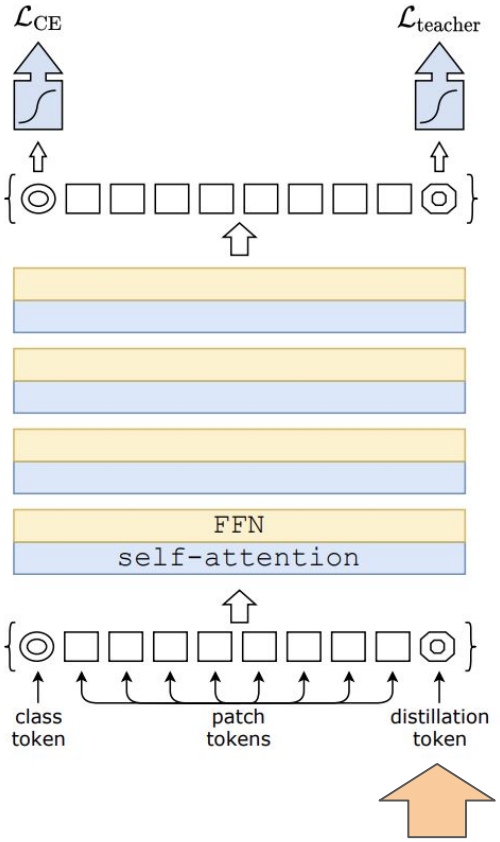
Requires less training computation than comparable CNN (BiT).

#BiT Kolesnikov, Alexander, Lucas Beyer, Xiaohua Zhai, Joan Puigcerver, Jessica Yung, Sylvain Gelly, and Neil Houlsby. "Big transfer (bit): General visual representation learning." ECCV 2020.
#ViT Dosovitskiy, Alexey, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani et al. "An image is worth 16x16 words: Transformers for image recognition at scale." ICLR 2021. [blog] [code] [video by Yannic Kilcher]

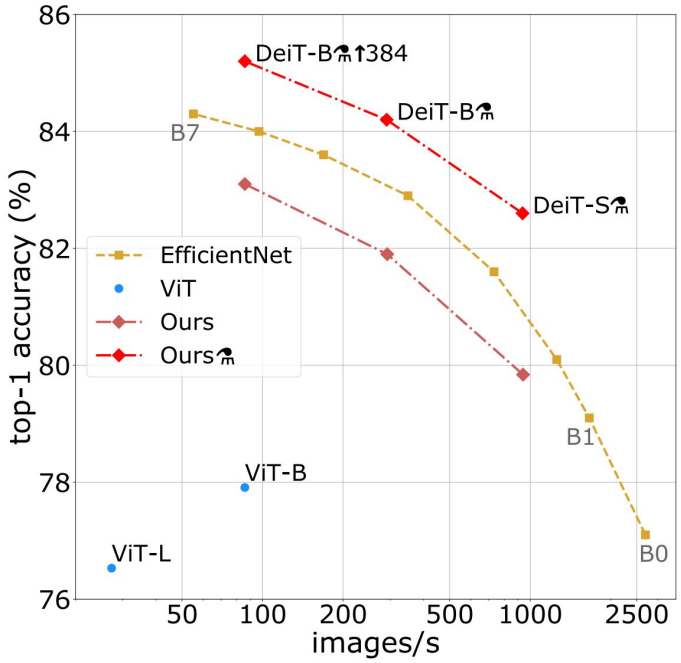
Outline

1. Vision Transformer (ViT)
    a. Tokenization
    b. Position embeddings
    c. Class embedding
    d. Receptive field
    e. Performance
2. **Beyond ViT**
3. Is attention all we need ?

Data-efficient Transformer (DeIT)



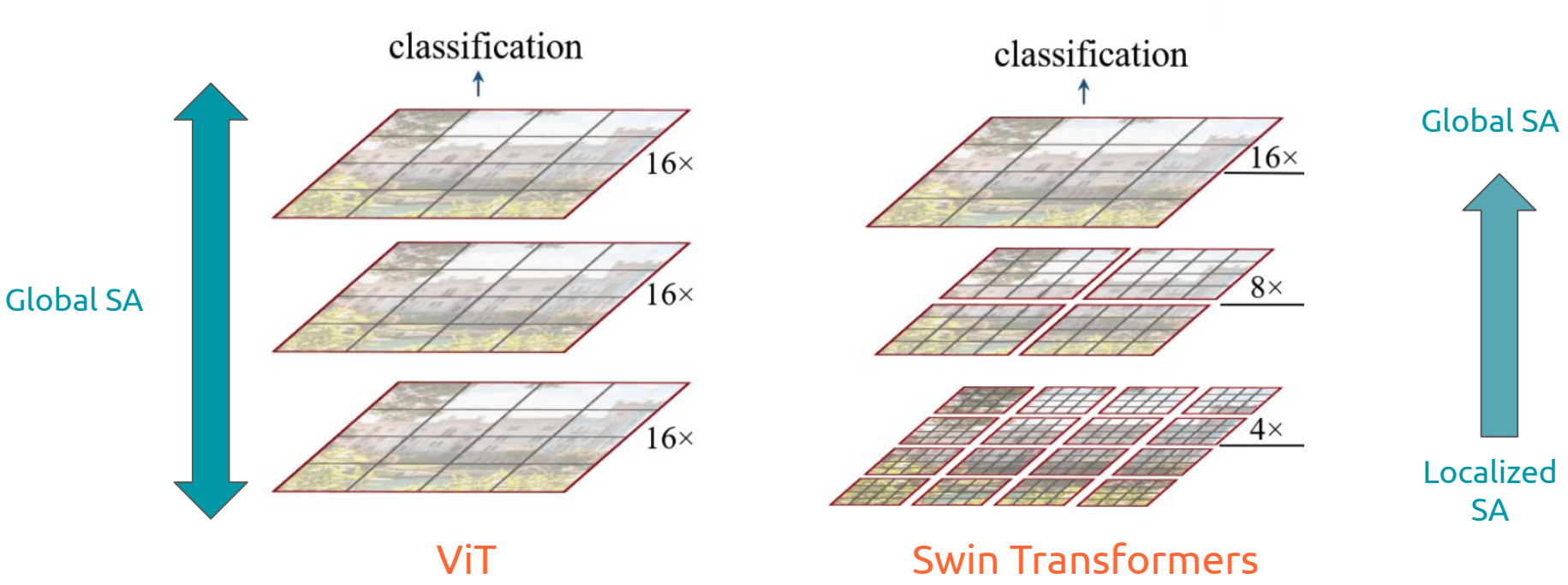
Distillation token that aims at predicting the label estimated by a teacher CNN. This allows introducing the convolutional bias in ViT.



#DeIT Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., & Jégou, H. Training data-efficient image transformers & distillation through attention. ICML 2021.

Shifted WINdow (SWIN) Self-Attention (SA)

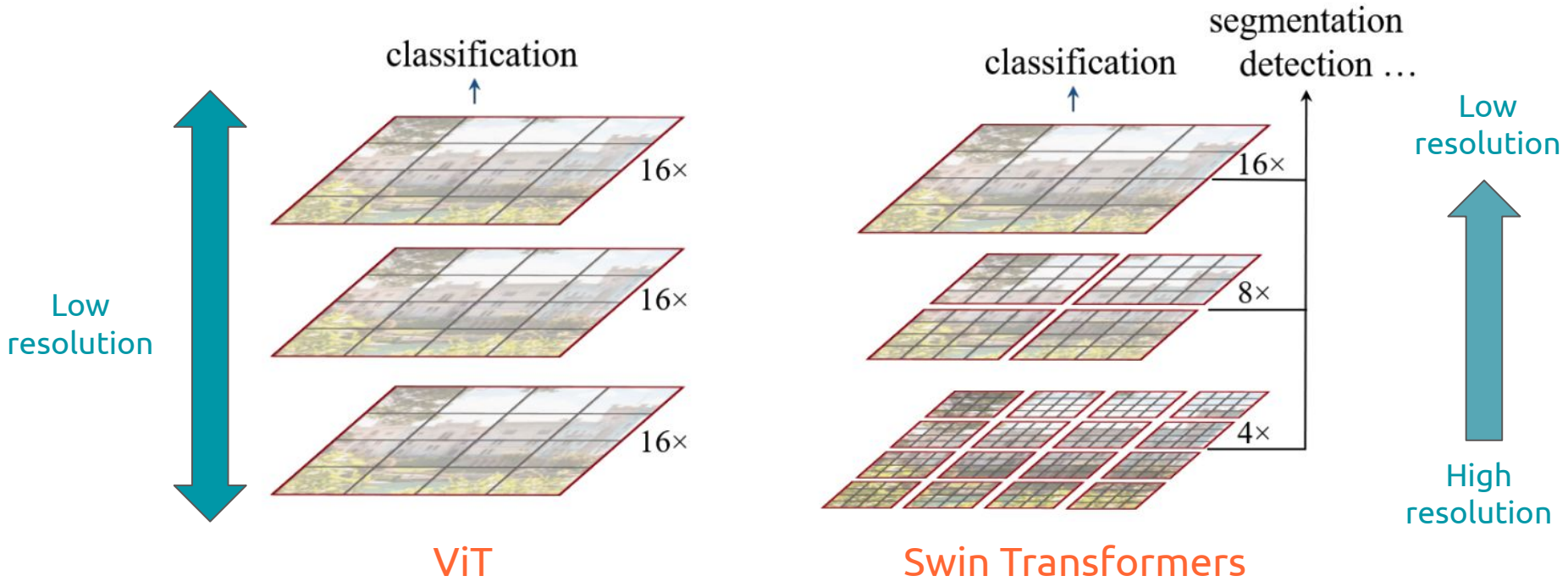
Less computation by self-attenting only in local windows (in grey).



#SWIN Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., ... & Guo, B. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. ICCV 2021 [Best paper award].

Hierarchical ViT Backbone

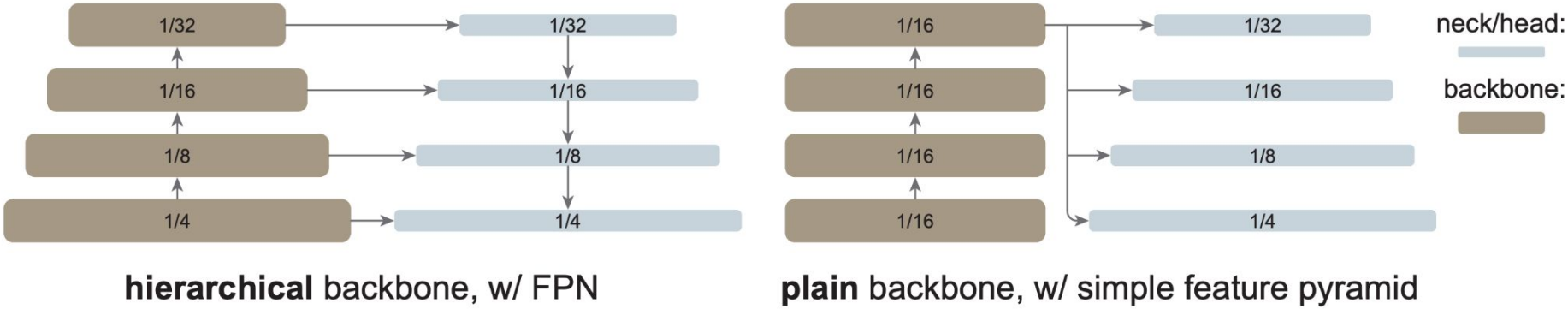
Hierarchical features maps by merging image patches (in red) across layers.



#SWIN Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., ... & Guo, B. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. ICCV 2021 [Best paper award].

Non-Hierarchical ViT Backbone

Multi-scale detection by building a feature pyramid from only the last, large strige (16) fetaure map of the plain backbone.



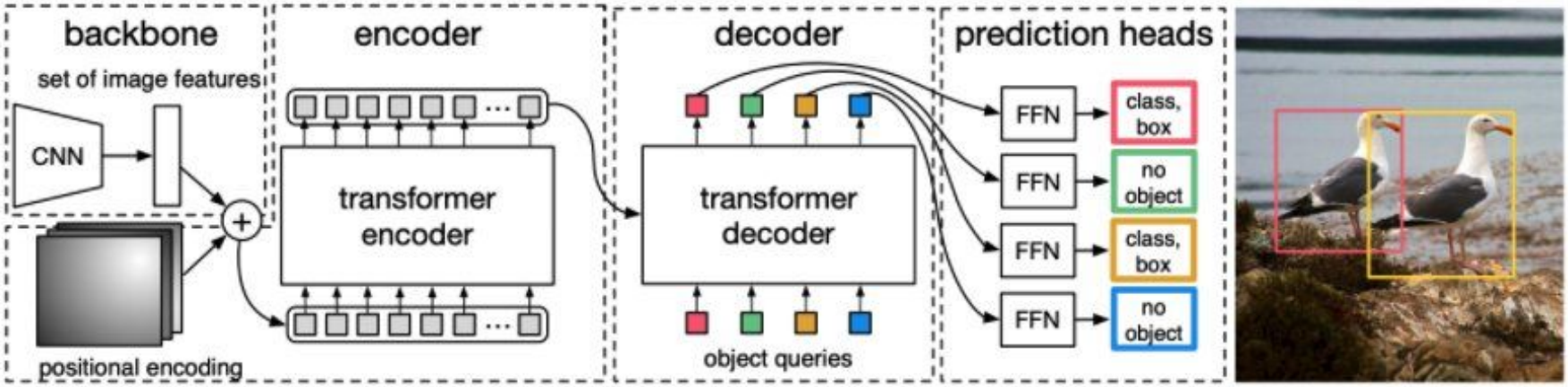
#VitDet Li, Yanghao, Hanzi Mao, Ross Girshick, and Kaiming He. "Exploring Plain Vision Transformer Backbones for Object Detection." arXiv preprint arXiv:2203.16527 (2022).

Outline

1. Vision Transformer (ViT)
    a. Tokenization
    b. Position embeddings
    c. Class embedding
    d. Receptive field
    e. Performance
2. **Beyond ViT**
3. Is attention all we need ?

Object Detection

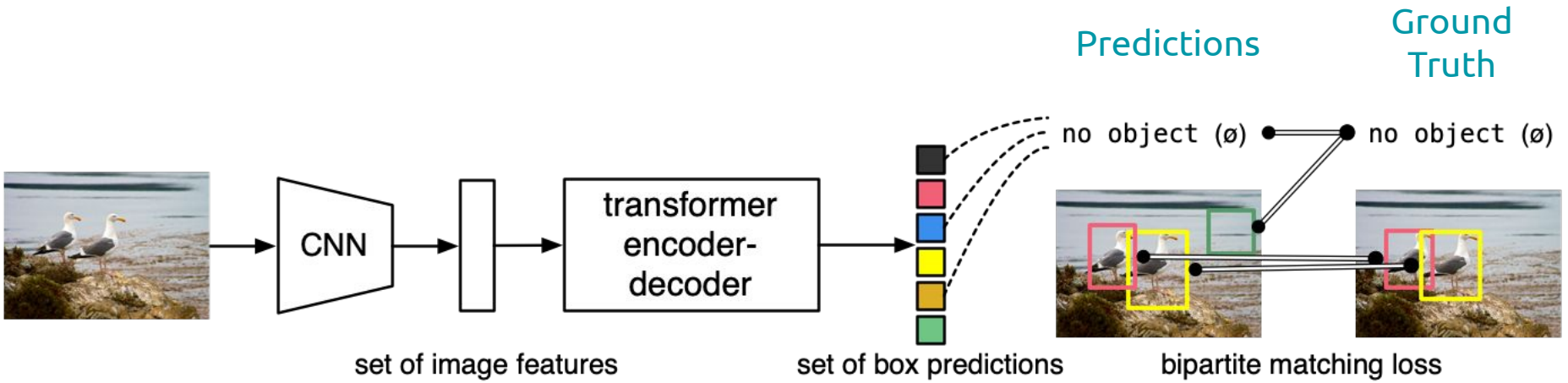
- Object detection formulated as a set prediction problem.
- DETR infers a fixed-size amount of predictions.
- Comparable performance to Faster R-CNN.



#DETR Carion, Nicolas, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. "End-to-End Object Detection with Transformers." ECCV 2020. [code] [colab]

Object Detection

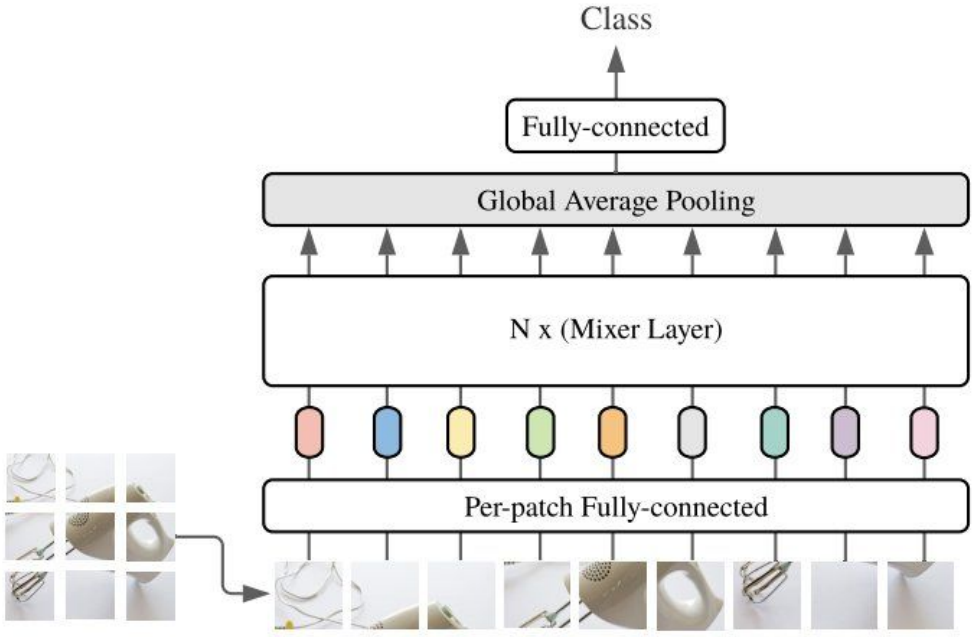
- During training, bipartite matching uniquely assigns predictions with ground truth boxes.
- Prediction with no match should yield a “no object” (∅) class prediction.



#DETR Carion, Nicolas, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. "End-to-End Object Detection with Transformers." ECCV 2020. [code] [colab]

Is attention (or convolutions) all we need ?

"In this paper we show that while convolutions and attention are both sufficient for good performance, neither of them are necessary."

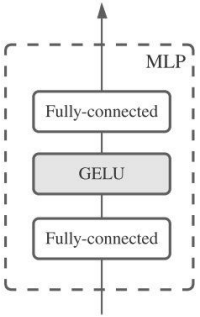


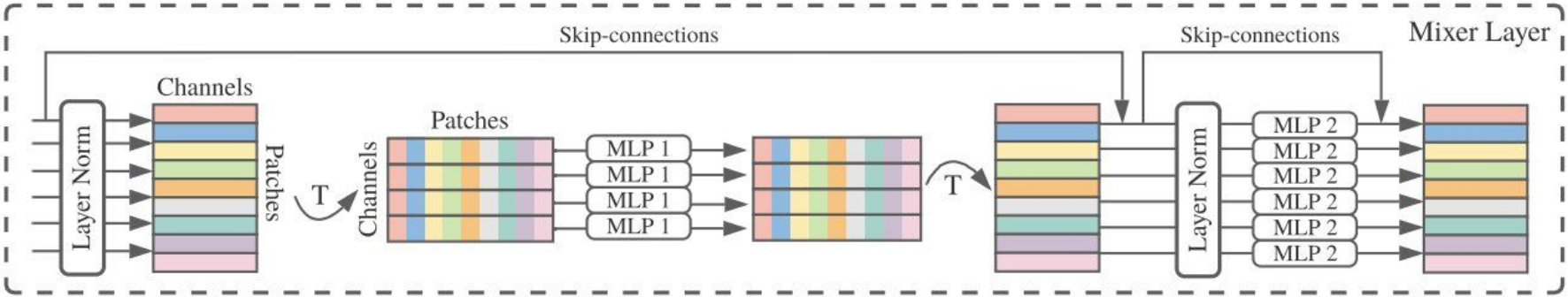
#MLP-Mixer Ilya Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Daniel Keysers, Jakob Uszkoreit, Mario Lucic, Alexey Dosovitskiy, "MLP-Mixer: An all-MLP Architecture for Vision". NeurIPS 2021. [tweet] [video by Yannic Kilcher]

Is attention (or convolutions) all we need ?

Two types of MLP Layers:

- **MLP 1:** Applied independently to image patches (i.e. “mixing” the per-location features”)
- **MLP 2:** applied across patches (i.e. “mixing spatial information”).

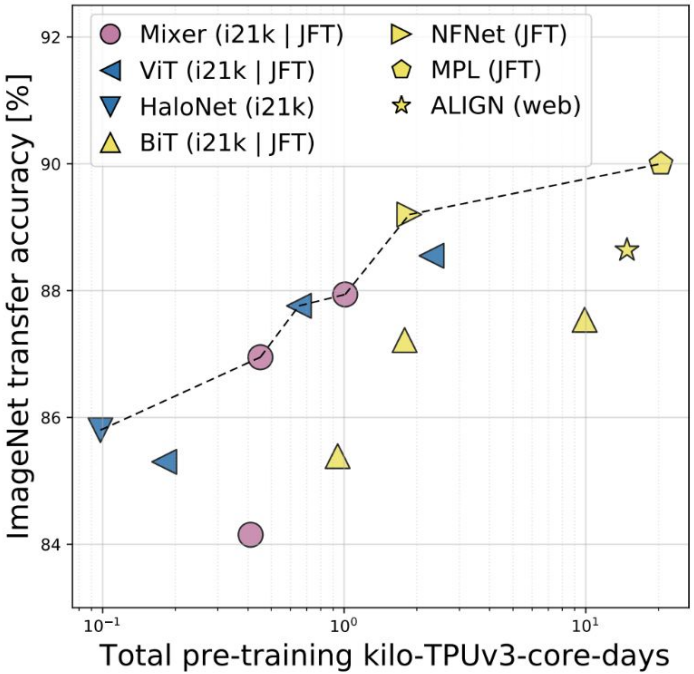




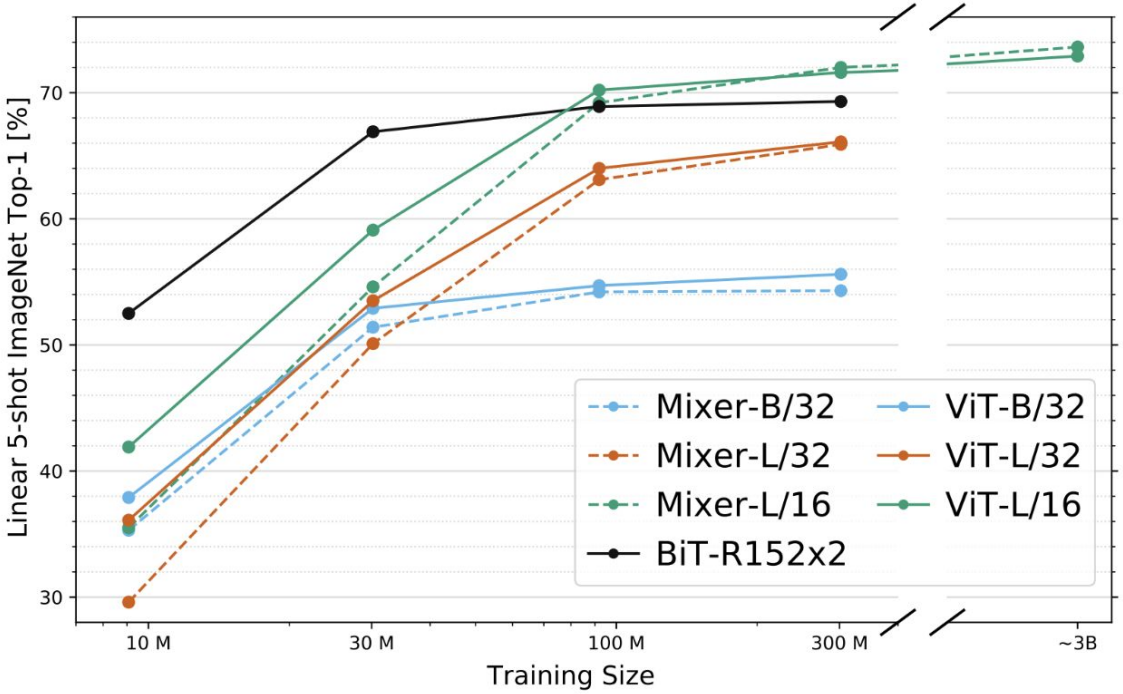
#MLP-Mixer Ilya Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Daniel Keysers, Jakob Uszkoreit, Mario Lucic, Alexey Dosovitskiy, “MLP-Mixer: An all-MLP Architecture for Vision”. NeurIPS 2021. [tweet] [video by Yannic Kilcher] [code]

Is attention (or convolutions) all we need ?

Computation efficiency (train)



Training data efficiency



#MLP-Mixer Ilya Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Daniel Keysers, Jakob Uszkoreit, Mario Lucic, Alexey Dosovitskiy, "MLP-Mixer: An all-MLP Architecture for Vision". NeurIPS 2021. [tweet] [video by Yannic Kilcher]

Is attention (or convolutions) all we need ?

A Alfredo Canziani i 5 més els agrada

Yann LeCun
@ylecun

Well, not *actually* conv free.
1st layer: "Per-patch fully-connected" == "conv layer with 16x16 kernels and 16x16 stride"
other layers: "MLP-Mixer" == "conv layer with 1x1 kernels"

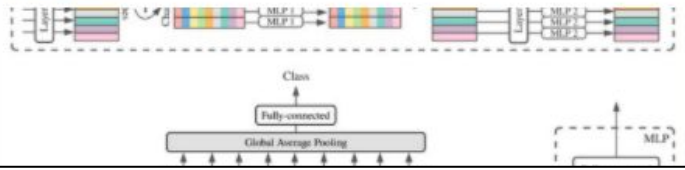
Tradueix el tuit

Neil Houlsby @neilhoulsby · 5 de maig
New paper from Brain Zurich and Berlin!

We try a conv and attention free vision architecture: MLP-Mixer (arxiv.org/abs/2105.01601)

Simple is good, so we went as minimalist as possible (just MLPs!) to see whether modern training methods & data is sufficient...

Mostra el fil



Joan Serrà
@serrjoa

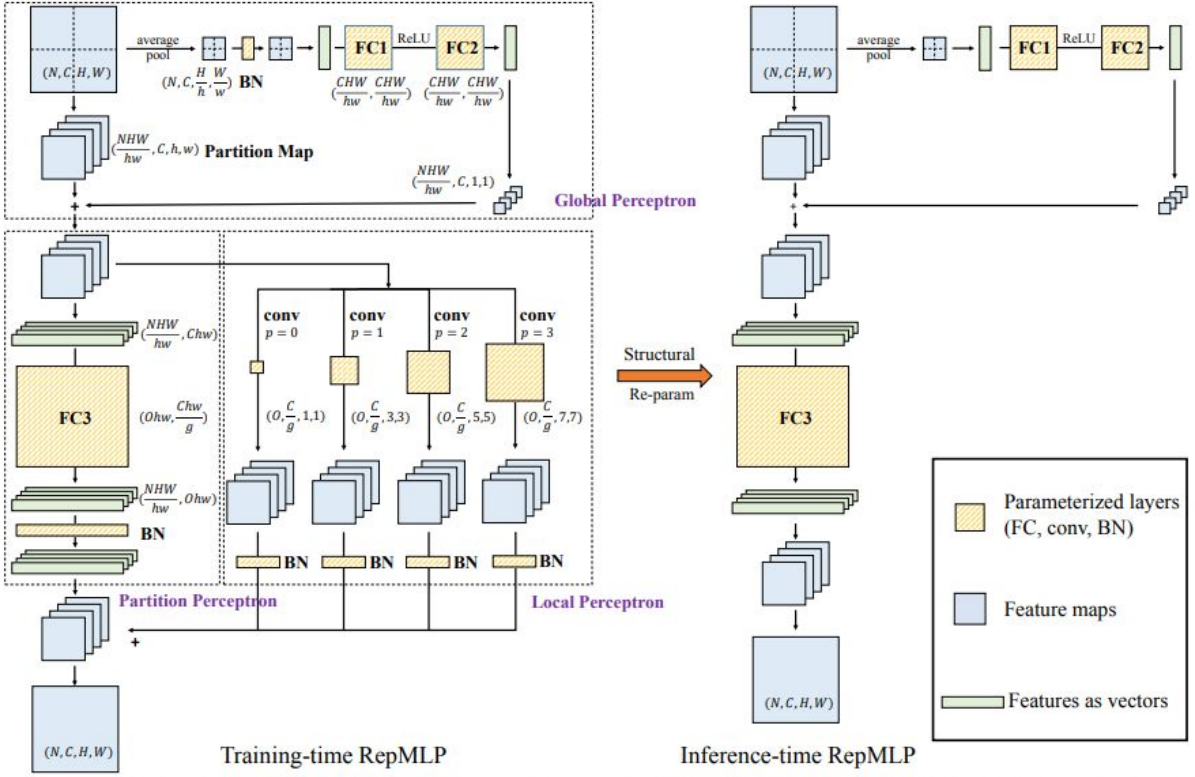
Conv1d is all you need.

Tradueix el tuit

7:52 a. m. · 7 de maig de 2021 · Twitter for iPhone

#MLP-Mixer Ilya Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Daniel Keysers, Jakob Uszkoreit, Mario Lucic, Alexey Dosovitskiy, "MLP-Mixer: An all-MLP Architecture for Vision". NeurIPS 2021. [tweet] [video by Yannic Kilcher]

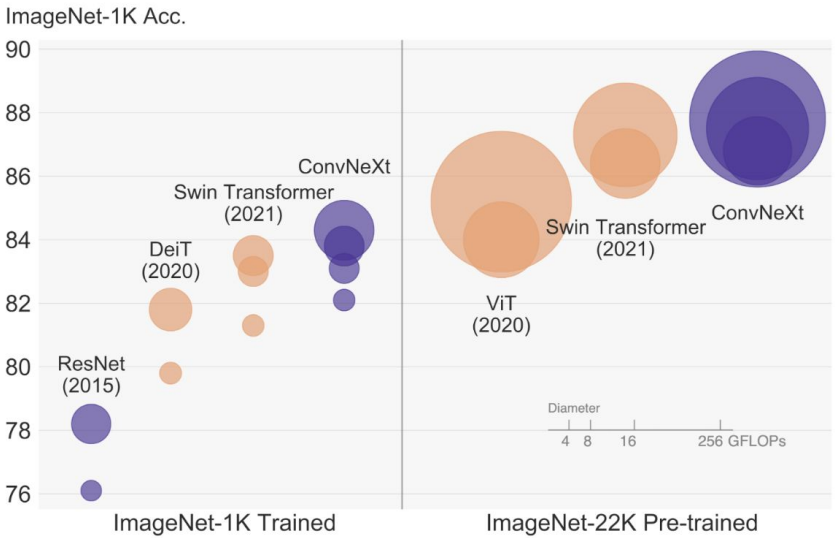
Is attention all we need ?

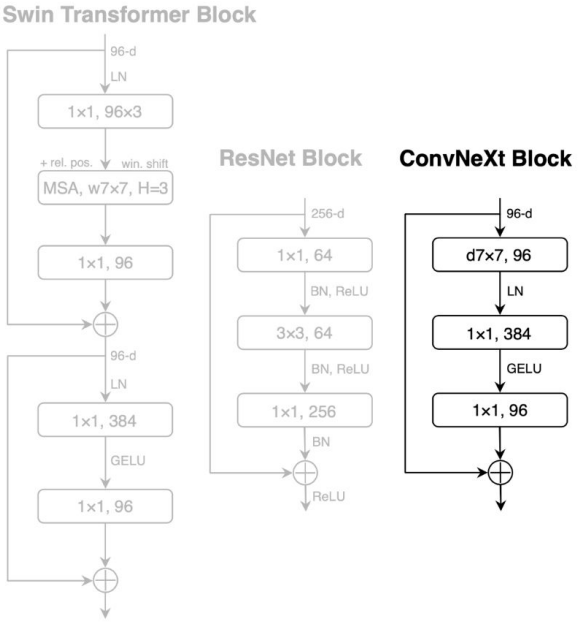


#RepMLP Xiaohan Ding, Xiangyu Zhang, Jungong Han, Guiguang Ding, [“RepMLP: Re-parameterizing Convolutions into Fully-connected Layers for Image Recognition”](#) CVPR 2022. [code] [tweet]

Is attention all we need ?

Gradually “modernize” a standard ResNet towards the design of ViT.



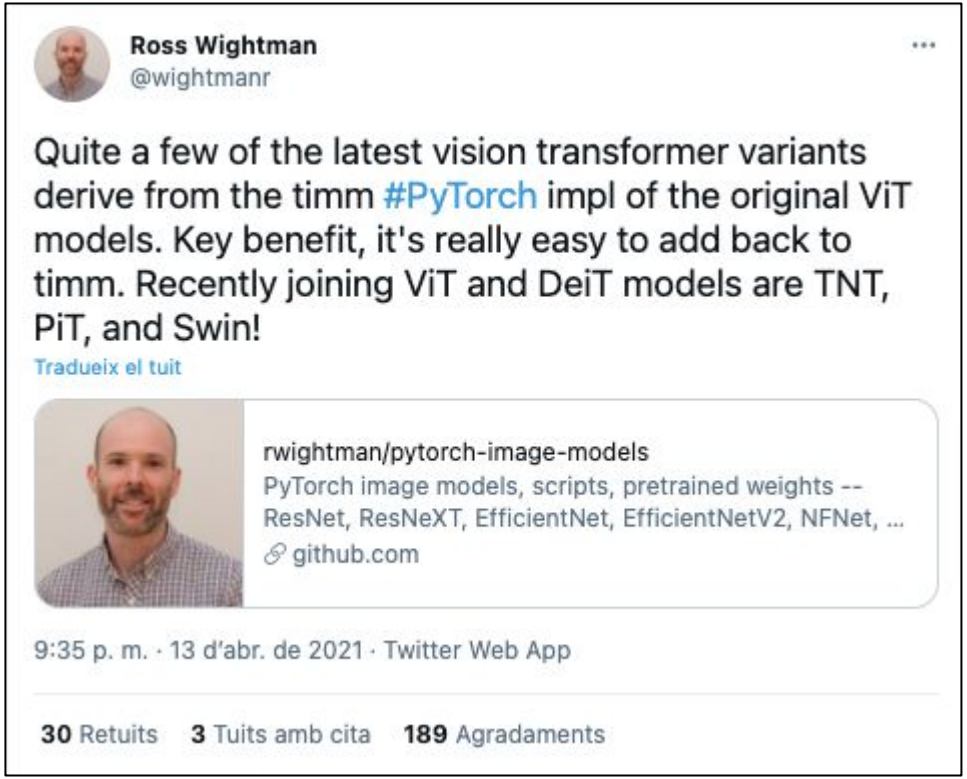


#ConvNeXt Liu, Zhuang, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. "A ConvNet for the 2020s." CVPR 2022. [code]

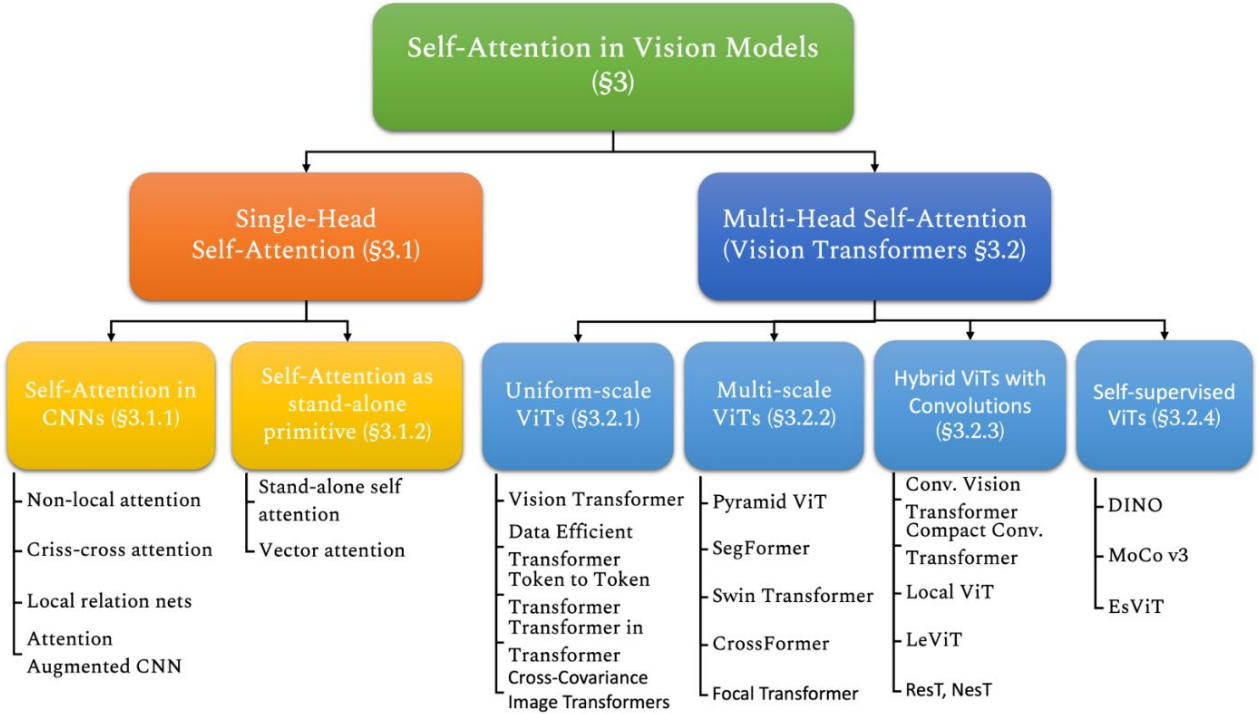
Outline

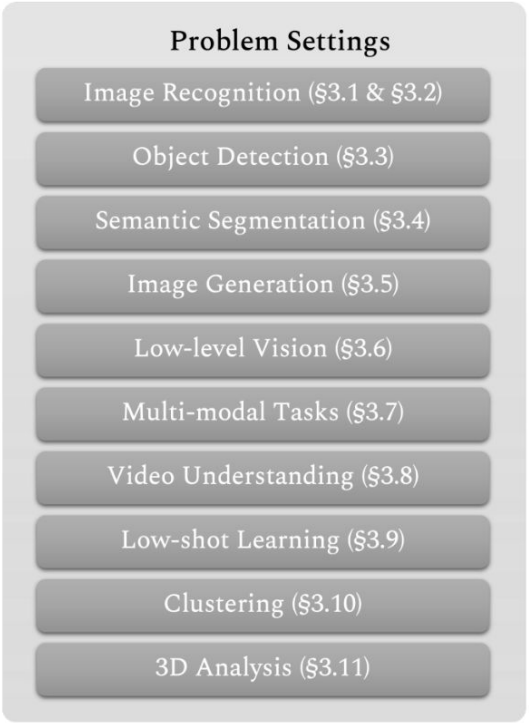
1. Vision Transformer (ViT)
    a. Tokenization
    b. Position embeddings
    c. Class embedding
    d. Receptive field
    e. Performance
2. Beyond ViT

Software


Ross Wightman
@wightmanr
Quite a few of the latest vision transformer variants derive from the timm #PyTorch impl of the original ViT models. Key benefit, it's really easy to add back to timm. Recently joining ViT and DeiT models are TNT, PiT, and Swin!
Tradueix el tuit
rwightman/pytorch-image-models
PyTorch image models, scripts, pretrained weights -- ResNet, ResNeXT, EfficientNet, EfficientNetV2, NFNet, ...
github.com
9:35 p. m. · 13 d'abr. de 2021 · Twitter Web App
30 Retuits 3 Tuits amb cita 189 Agradaments

Learn more

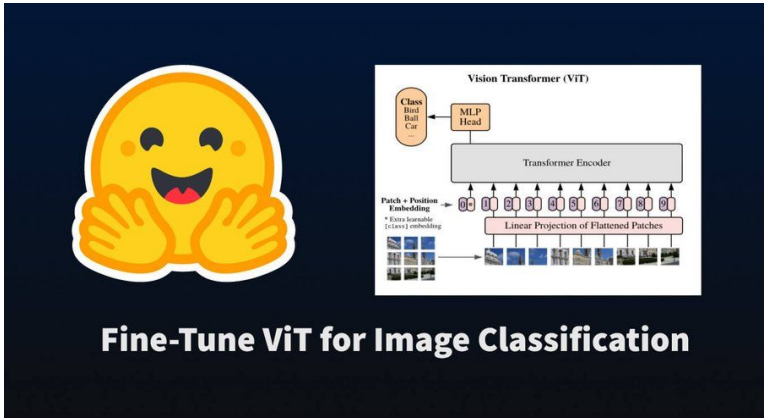




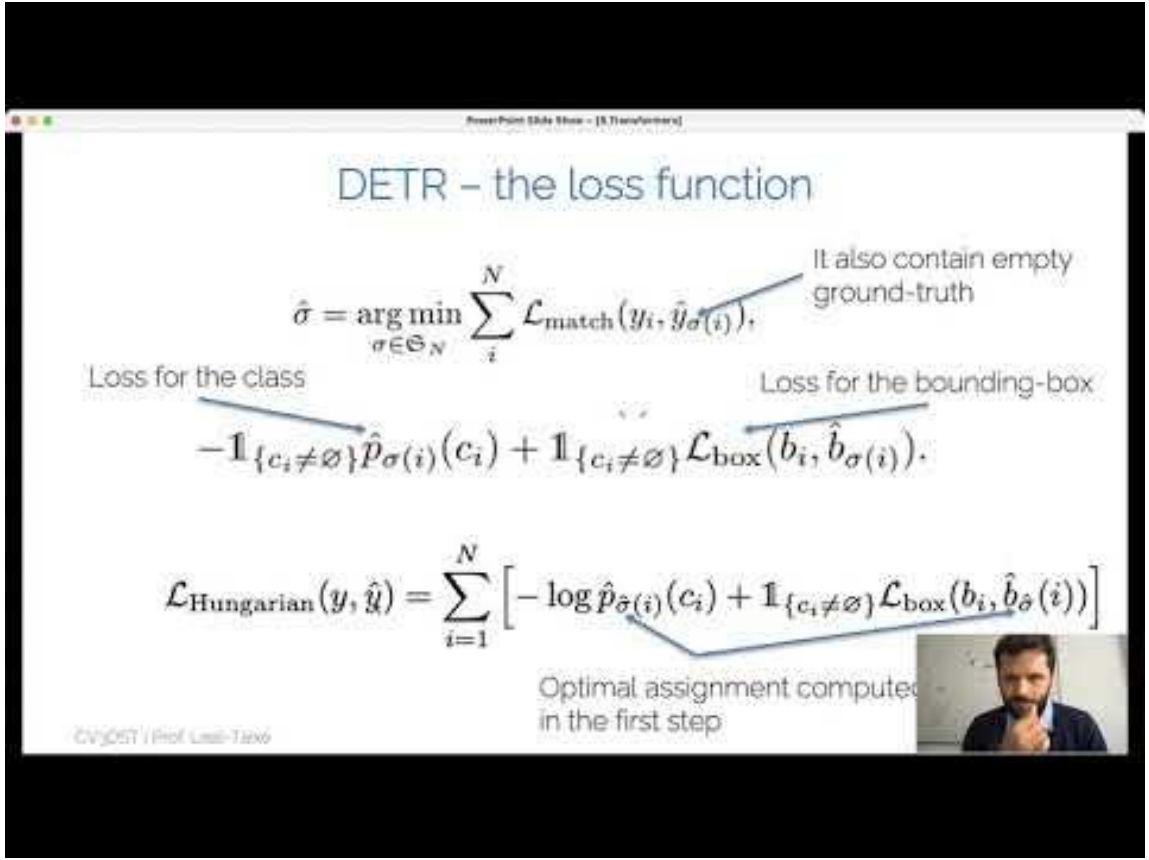
Khan, Salman, Muzammal Naseer, Munawar Hayat, Syed Waqas Zamir, Fahad Shahbaz Khan, and Mubarak Shah. "Transformers in vision: A survey." ACM Computing Surveys (CSUR) (2021).

Learn more

- Paper with code: Twitter thread about ViT (2022)
- IAML Distill Blog: Transformers in Vision (2021)
- Touvron, Hugo, Matthieu Cord, Alaaeldin El-Nouby, Jakob Verbeek, and Hervé Jégou. "Three things everyone should know about Vision Transformers." arXiv preprint arXiv:2203.09795 (2022).
- Tutorial: Fine-Tune ViT for Image Classification with 🤗 Transformers (2022).

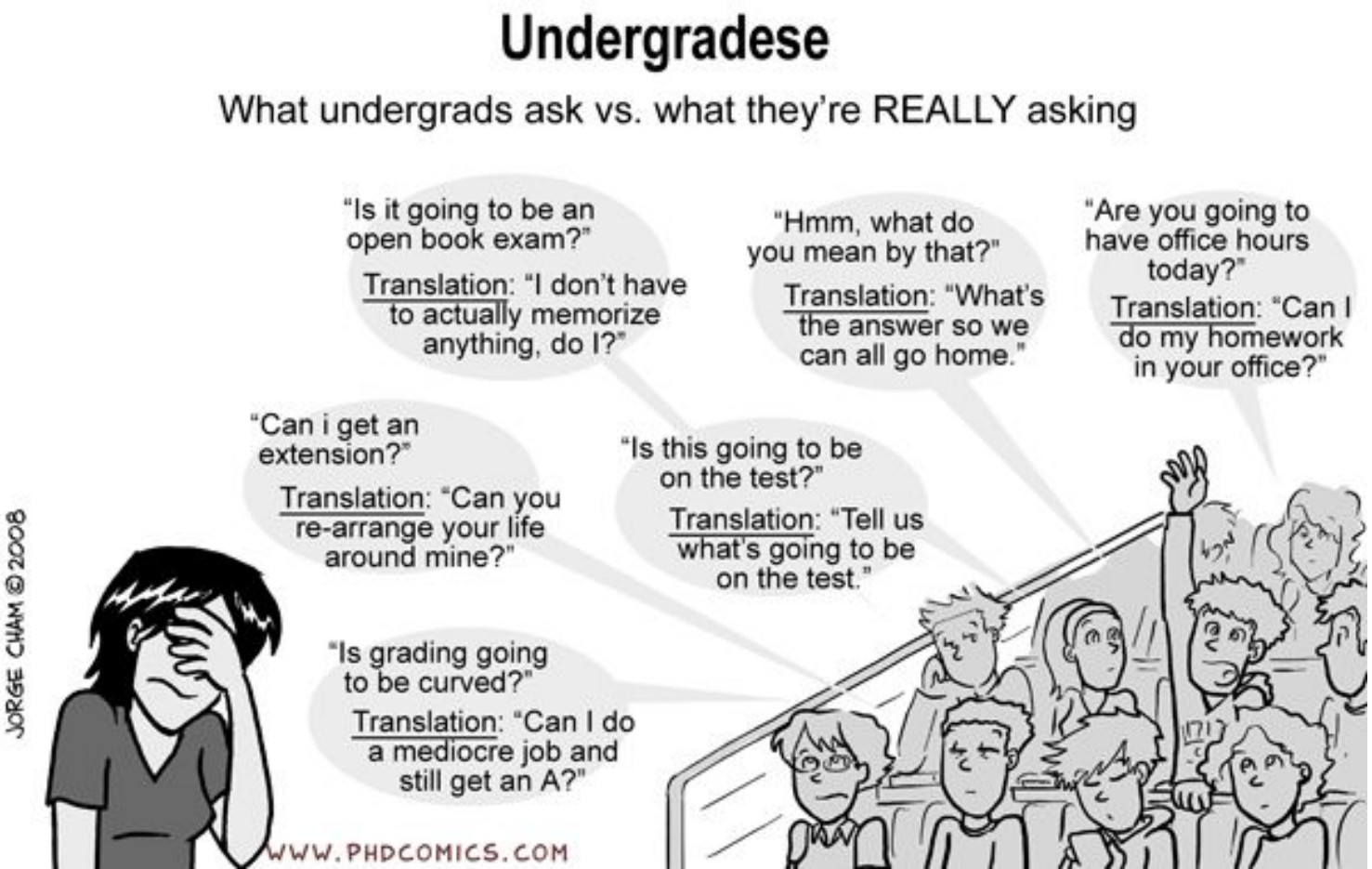


Learn more

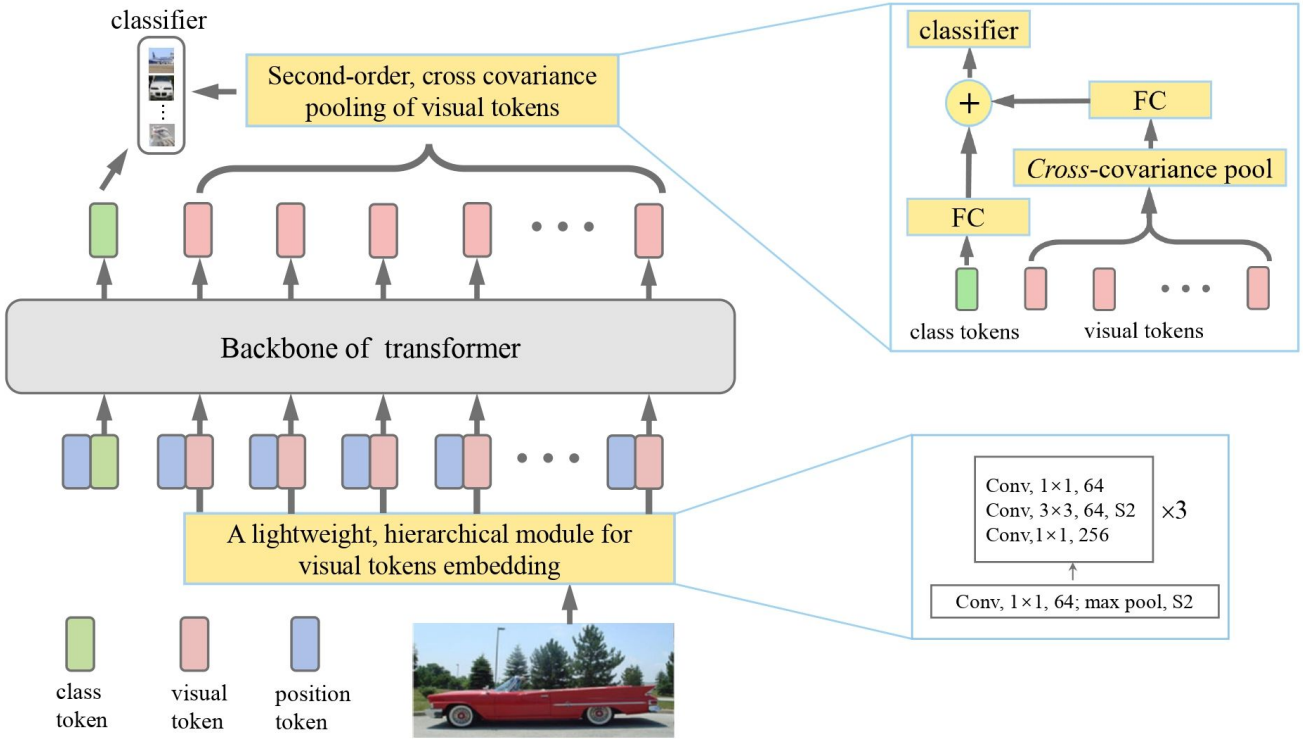


Ismael Elisi, [“Transformers and its use in computer vision”](#) (TUM 2021)

Questions ?


Undergradese
What undergrads ask vs. what they're REALLY asking
"Is it going to be an open book exam?"
Translation: "I don't have to actually memorize anything, do I?"
"Hmm, what do you mean by that?"
Translation: "What's the answer so we can all go home."
"Are you going to have office hours today?"
Translation: "Can I do my homework in your office?"
"Can i get an extension?"
Translation: "Can you re-arrange your life around mine?"
"Is this going to be on the test?"
Translation: "Tell us what's going to be on the test."
"Is grading going to be curved?"
Translation: "Can I do a mediocre job and still get an A?"
JORGE CHAM © 2008
WWW.PHDCOMICS.COM

The Transformer for Vision: So-VIT



#So-VIT Jiangtao Xie, Ruiren Zeng, Qilong Wang, Ziqi Zhou, Peihua Li, "So-ViT: Mind Visual Tokens for Vision Transformer". arXIv 2021. [code] [tweet]

The Transformer for Vision: Lambda Networks

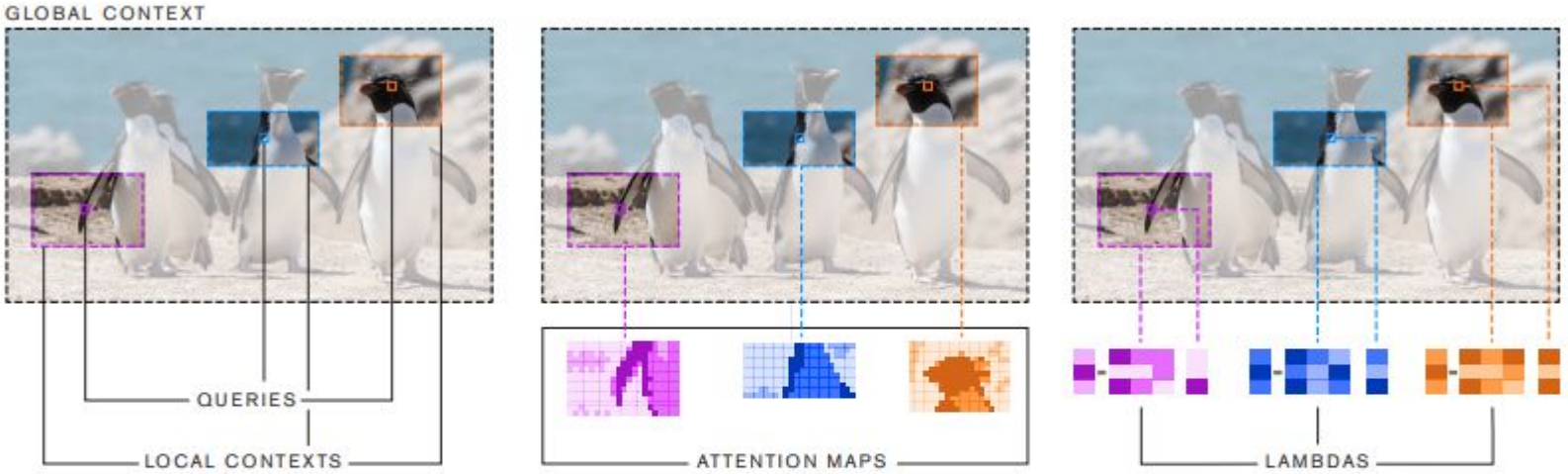


Figure 1: Comparison between attention and lambda layers. (Left) An example of 3 queries and their local contexts within a global context. (Middle) The attention operation associates each query with an attention distribution over its context. (Right) The lambda layer transforms each context into a linear function lambda that is applied to the corresponding query.

Bello, Irwan. "Lambdanetworks: Modeling long-range interactions without attention." ICLR 2021. [code] [video] [review by Yannick Kilcher]

The Transformer for Vision: HaloNet

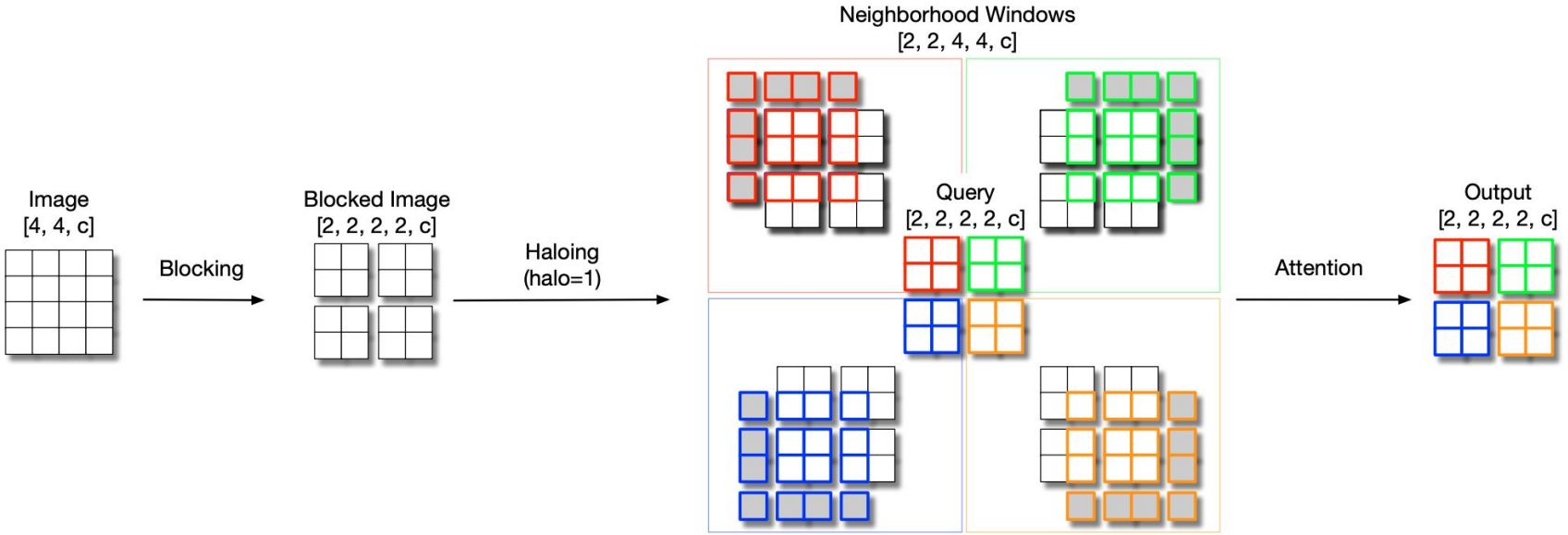


Figure 1. **HaloNet local self-attention architecture: The different stages of blocked local attention for a** $[4, 4, c]$ **image, block size** $b = 2$**, and halo** $h = 1$. The image is first blocked into non-overlapping $[2, 2, c]$ images from which the queries are computed. The subsequent haloing step then extracts a $[4, 4, c]$ memory around each of the blocks which linearly transform to keys and values. The spatial dimensions after attention are the same as the queries.

#HaloNet Vaswani, Ashish, Prajit Ramachandran, Aravind Srinivas, Niki Parmar, Blake Hechtman, and Jonathon Shlens. "Scaling Local Self-Attention For Parameter Efficient Visual Backbones." CVPR 2021.

The Transformer for Vision: Bottleneck

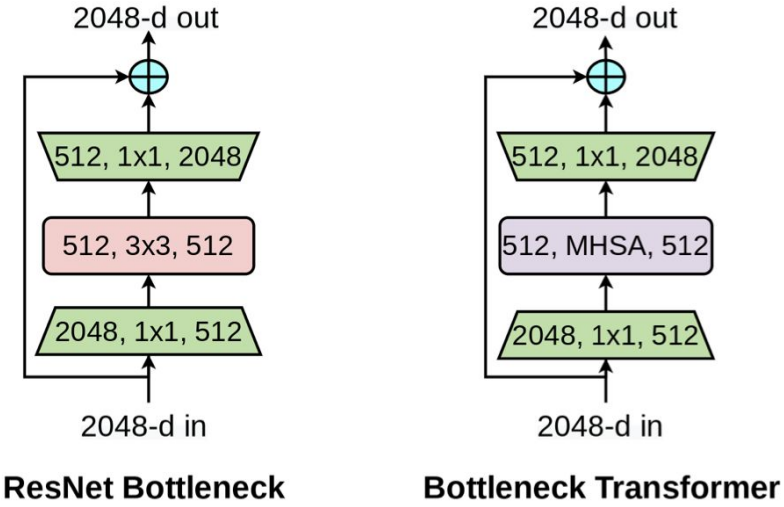
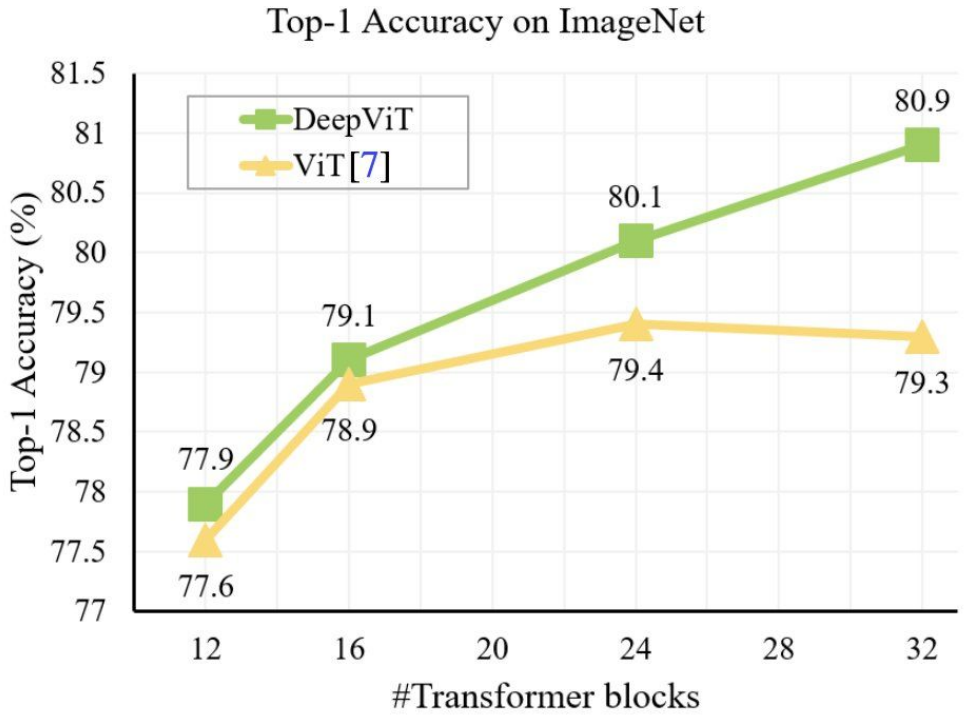


Figure 1: **Left:** A ResNet Bottleneck Block, **Right:** A Bottleneck Transformer (BoT) block. The only difference is the replacement of the spatial 3×3 convolution layer with Multi-Head Self-Attention (MHSA). The structure of the self-attention layer is described in Figure 4.

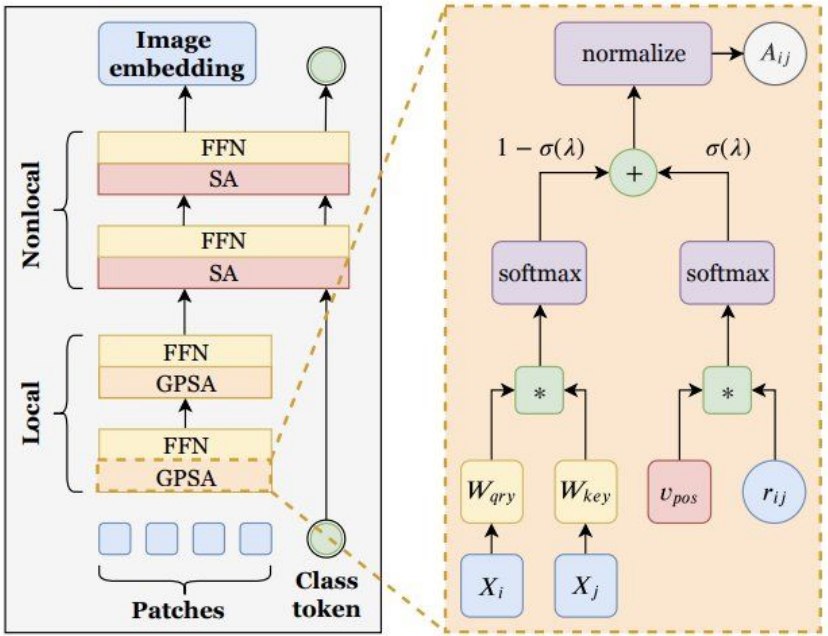
#BotNet Srinivas, Aravind, Tsung-Yi Lin, Niki Parmar, Jonathon Shlens, Pieter Abbeel, and Ashish Vaswani. "Bottleneck transformers for visual recognition." arXiv preprint arXiv:2101.11605 (2021).

The Transformer for Vision: DeepViT



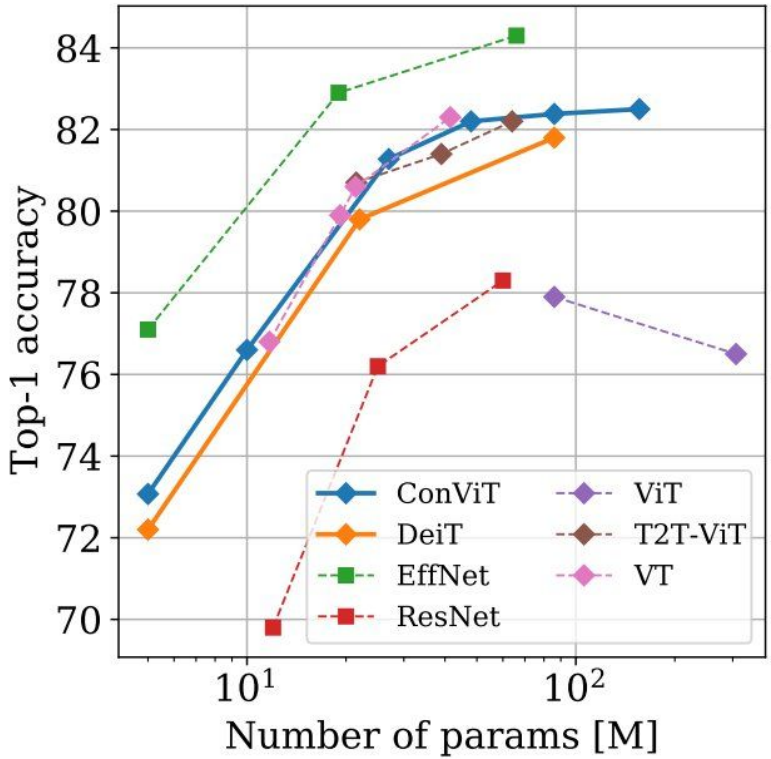
#DeepViT Daquan Zhou, Bingyi Kang, Xiaojie Jin, Linjie Yang, Xiaochen Lian, Qibin Hou, Jiashi Feng, "DeepViT: Towards Deeper Vision Transformer" arXiv 2021. [tweet]

The Transformer for Vision: ConvIT



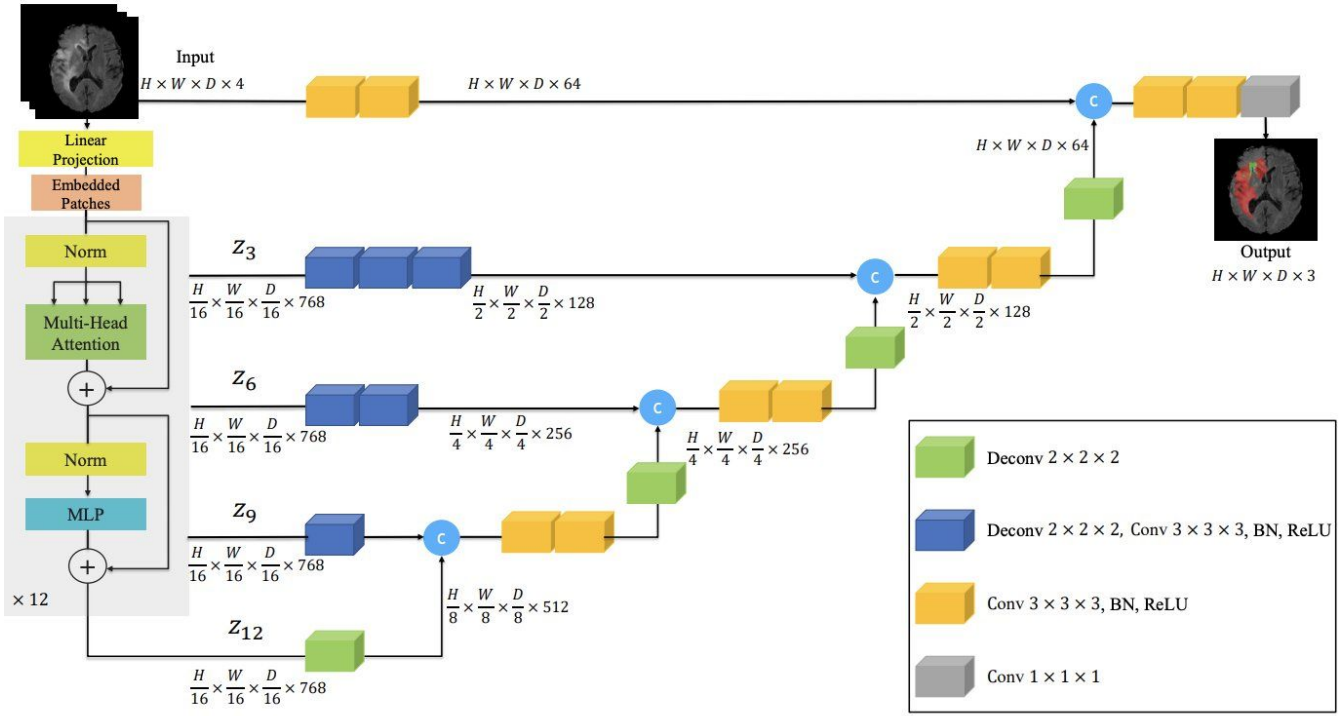
ConViT

Gated Positional Self-Attention



#ConViT Stéphane d'Ascoli, Hugo Touvron, Matthew Leavitt, Ari Morcos, Giulio Biroli, Levent Sagun, "ConViT: Improving Vision Transformers with Soft Convolutional Inductive Biases" arXiv 2021. [tweet]

The Transformer for Vision: Segmentation



#UNETR Ali Hatamizadeh, Dong Yang, Holger Roth, Daguang Xu, [“UNETR: Transformers for 3D Medical Image Segmentation”](#) arXiv 2021

The Transformer for Vision: Segmentation

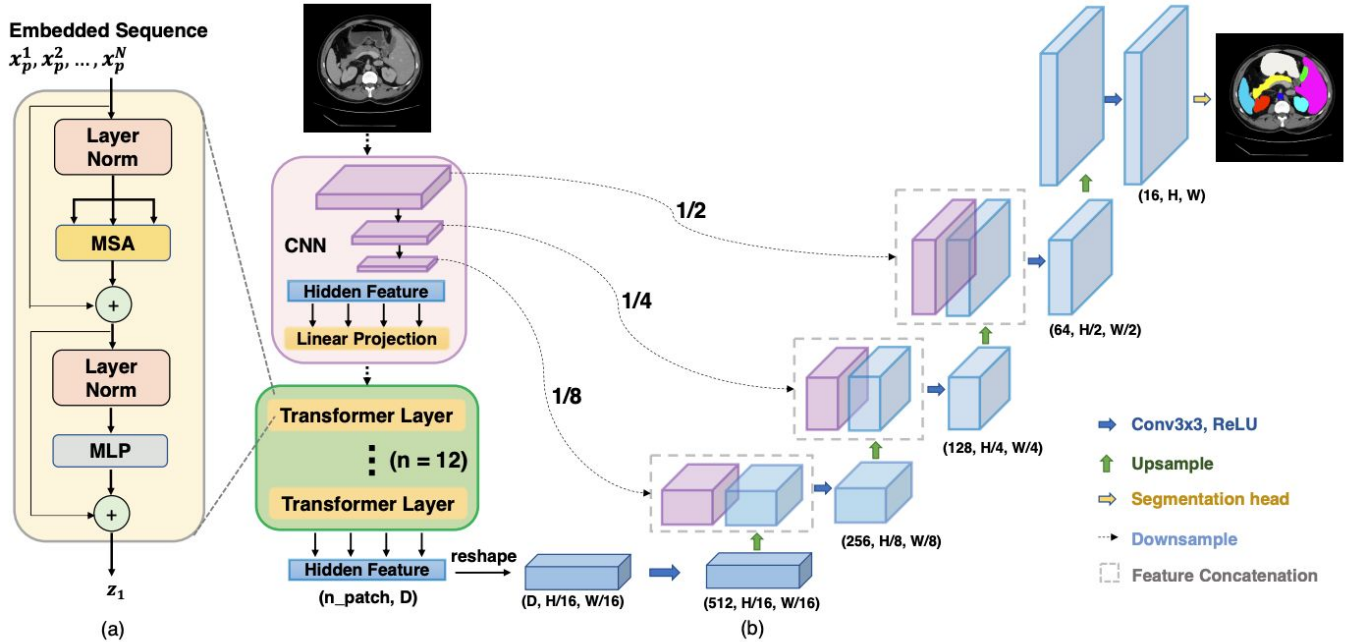
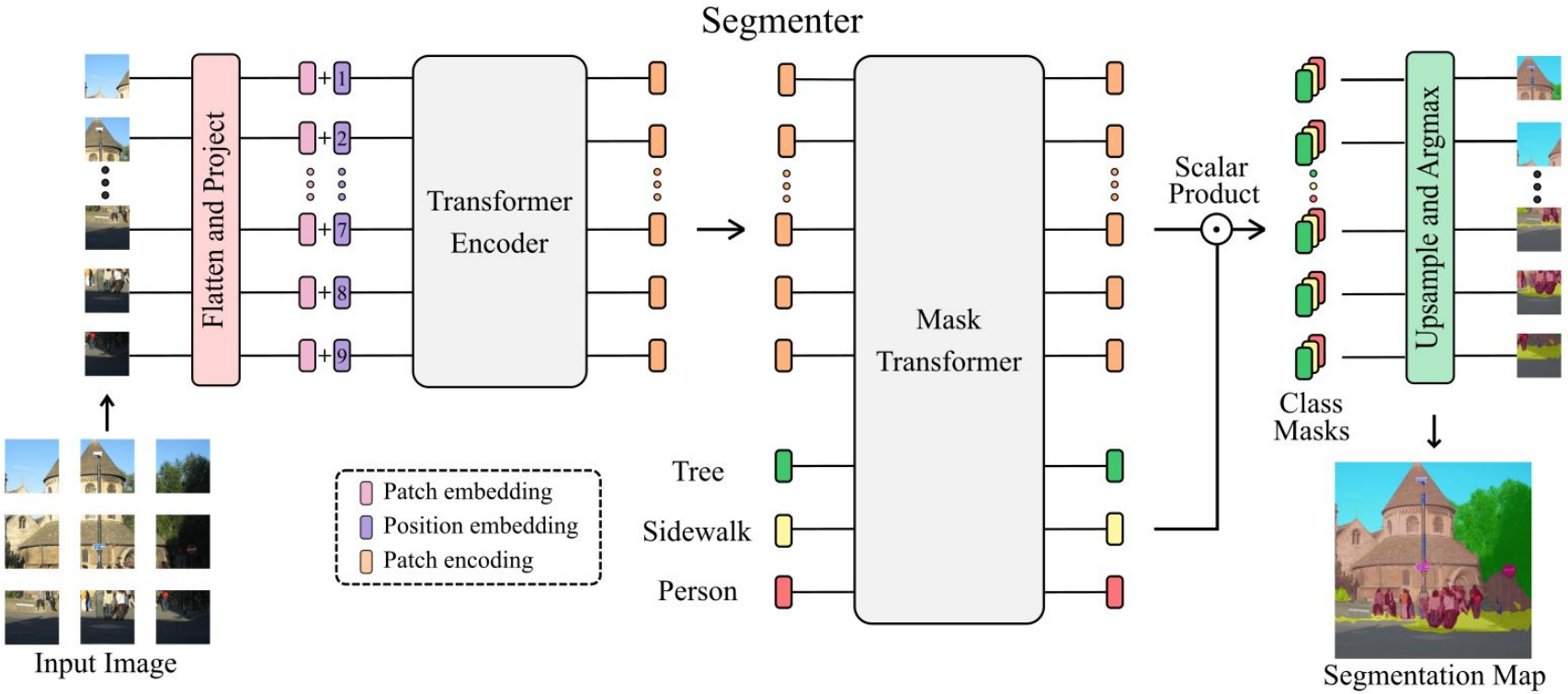


Fig. 1: Overview of the framework. (a) schematic of the Transformer layer; (b) architecture of the proposed TransUNet.

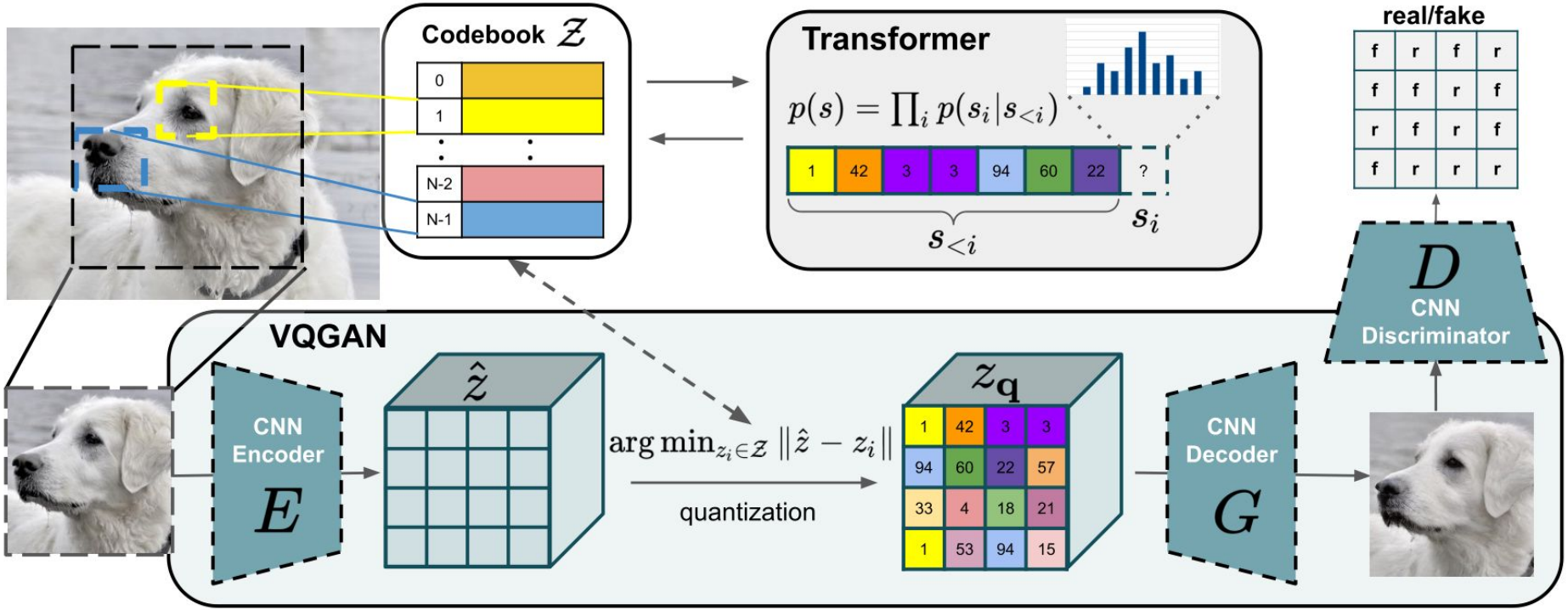
#TransUNet Chen, J., Lu, Y., Yu, Q., Luo, X., Adeli, E., Wang, Y., ... & Zhou, Y. (2021). Transunet: Transformers make strong encoders for medical image segmentation. arXiv preprint arXiv:2102.04306. [code]

The Transformer for Vision: Segmentation



#Segmenter Strudel, R., Garcia, R., Laptev, I., & Schmid, C. Segmenter: Transformer for Semantic Segmentation. ICCV 2021 [code]

The Transformer for Vision: High definition



Patrick Esser*, Robin Rombach*, Björn Ommer, "Taming Transformers for High-Resolution Image Synthesis" CVPR 2021. [code]

The Transformer for Vision: Monocular Depth

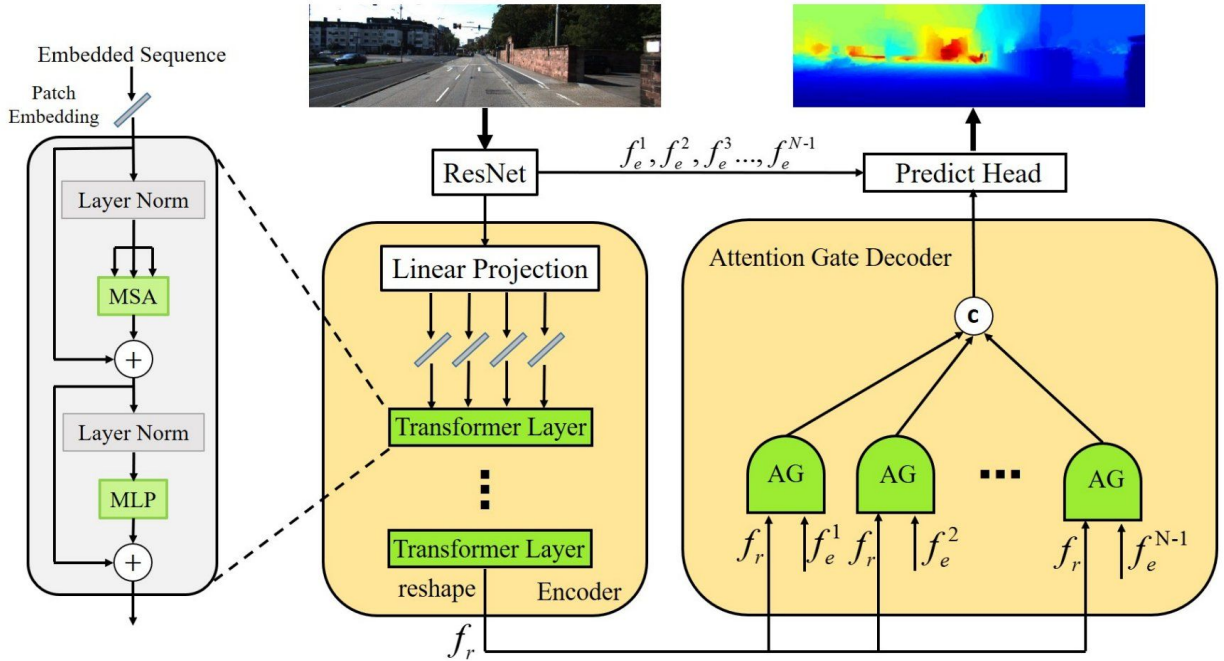
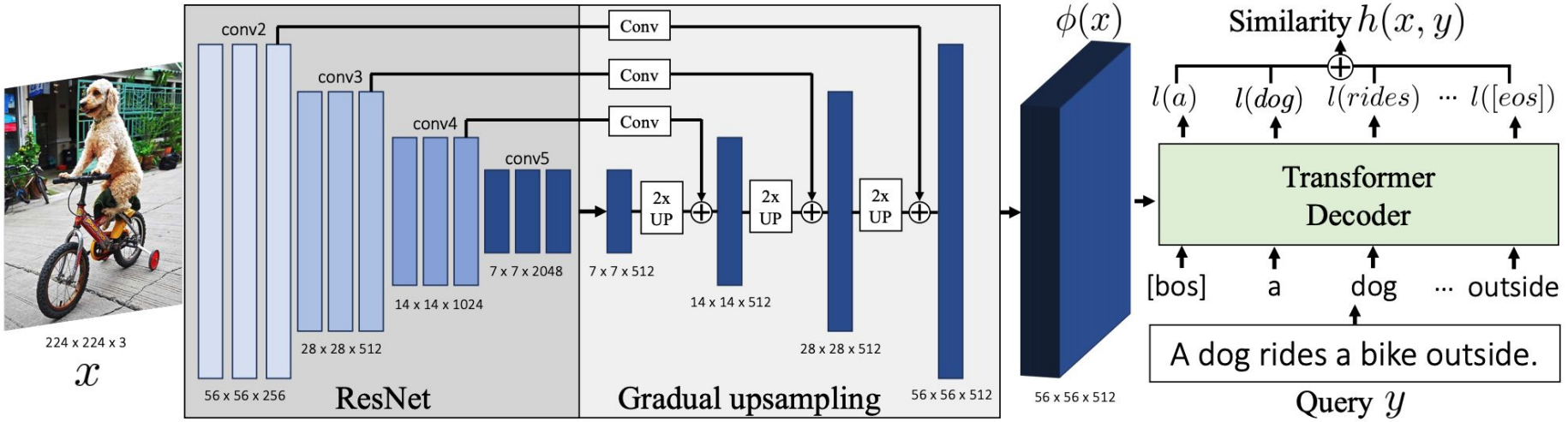


Figure 1: The overview of the proposed TransDepth. The symbols ⓒ and ⊕ denote concatenation and addition operations, respectively. AG is short for attention gate.

#TransDepth Guanglei Yang, Hao Tang, Mingli Ding, Nicu Sebe, Elisa Ricci, [“Transformers Solve the Limited Receptive Field for Monocular Depth Prediction”](). arXiv 2021. [code] [tweet]

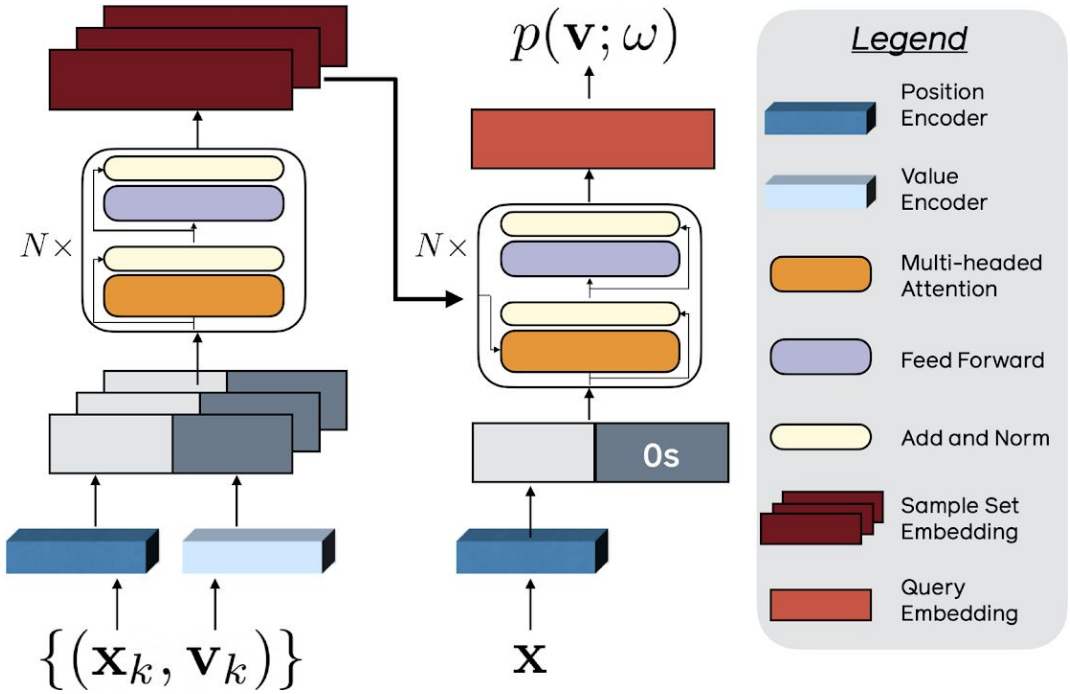
The Transformer for Vision: Retrieval



Miech, A., Alayrac, J. B., Laptev, I., Sivic, J., & Zisserman, A. (2021). Thinking Fast and Slow: Efficient Text-to-Visual Retrieval with Transformers. arXiv preprint arXiv:2103.16553.

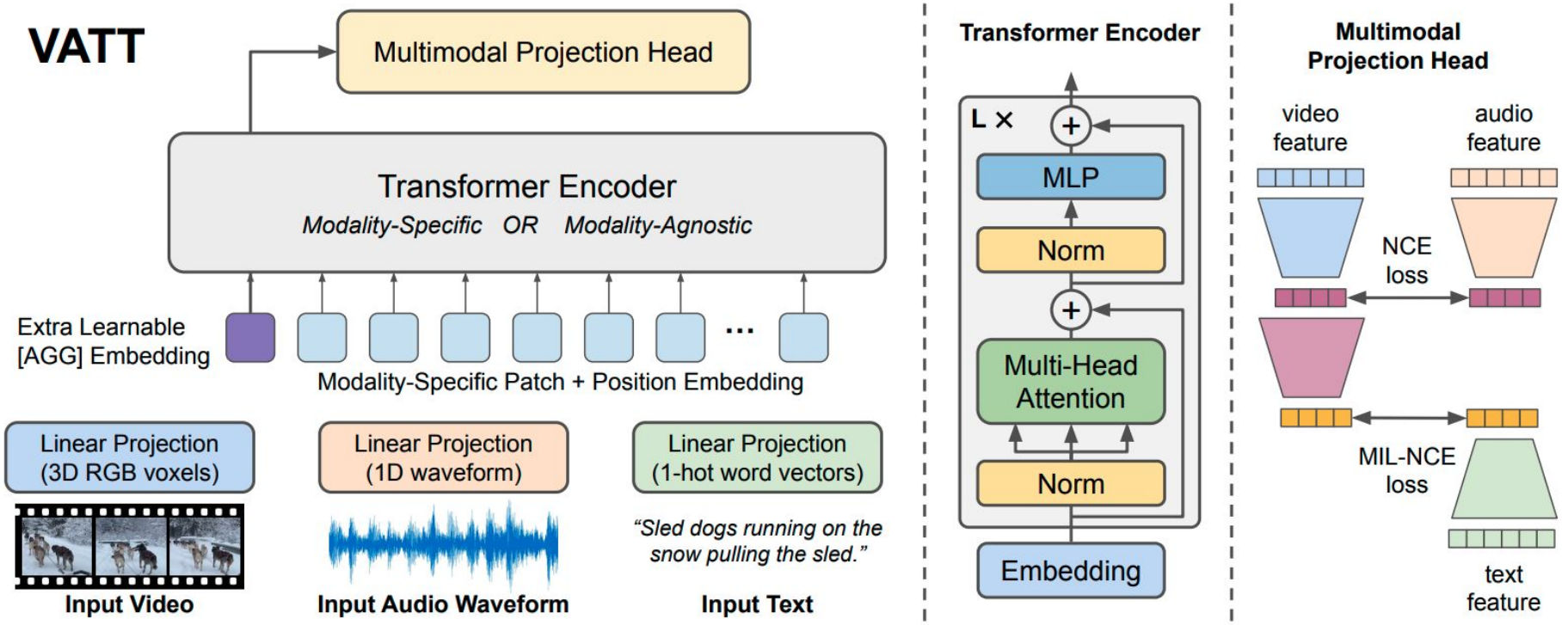
The Transformer for Vision: Generative models

The model takes as input a set of observed pixel locations and values ($\{(\mathbf{x}_k,\mathbf{v}_k)\}$) and can then predict the value distribution for any query position **x**.



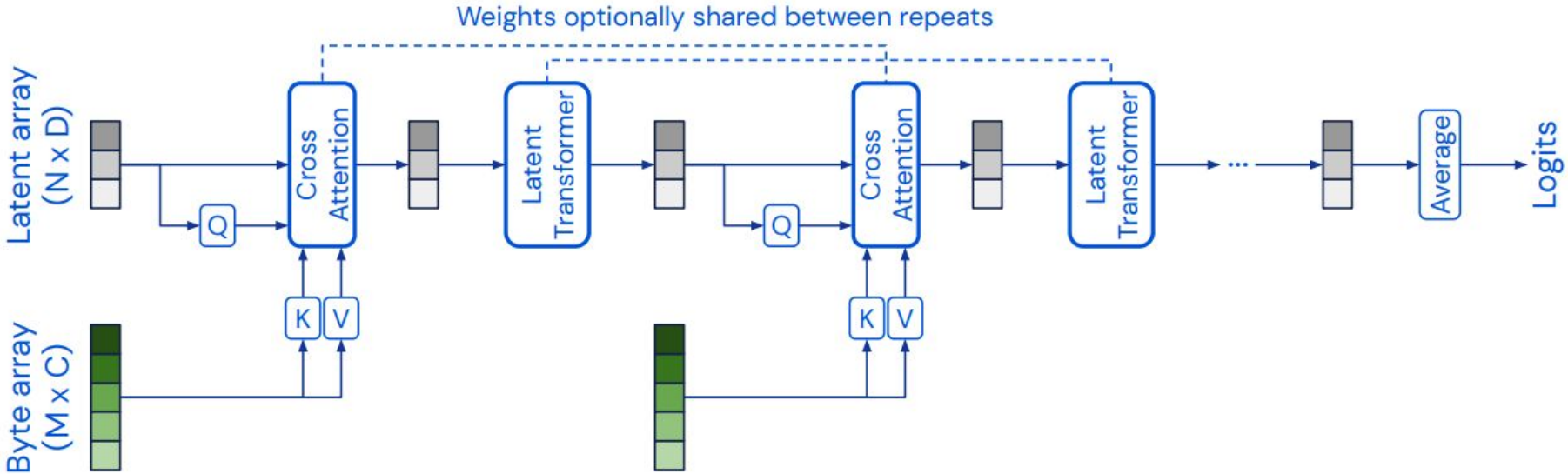
#PixelTransformer Tulsiani, Shubham, and Abhinav Gupta. "PixelTransformer: Sample Conditioned Signal Generation." arXiv preprint arXiv:2103.15813 (2021).

The Transformer for Vision: Multimodal



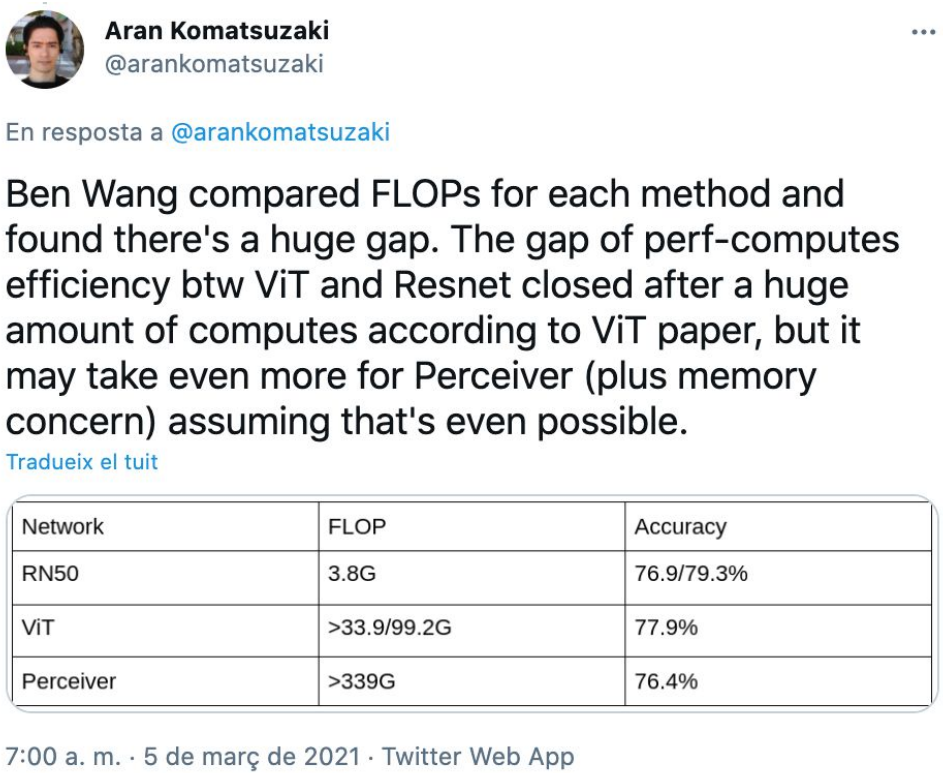
#VATT Hassan Akbari, Linagzhe Yuan, Rui Qian, Wei-Hong Chuang, Shih-Fu Chang, Yin Cui, Boqing Gong, "VATT: Transformers for Multimodal Self-Supervised Learning from Raw Video, Audio and Text" arXiv 2021. [tweet]

The Transformer for Vision: Perceiver



#Perceiver Jaegle, A., Gimeno, F., Brock, A., Zisserman, A., Vinyals, O., & Carreira, J. (2021). Perceiver: General Perception with Iterative Attention. ICML 2021.

The Transformer for Vision: Perceiver



Aran Komatsuzaki
@arankomatsuzaki

En resposta a @arankomatsuzaki

Ben Wang compared FLOPs for each method and found there's a huge gap. The gap of perf-computes efficiency btw ViT and Resnet closed after a huge amount of computes according to ViT paper, but it may take even more for Perceiver (plus memory concern) assuming that's even possible.

Tradueix el tuit

Network	FLOP	Accuracy
RN50	3.8G	76.9/79.3%
ViT	>33.9/99.2G	77.9%
Perceiver	>339G	76.4%

7:00 a. m. · 5 de març de 2021 · Twitter Web App

#Perceiver Jaegle, A., Gimeno, F., Brock, A., Zisserman, A., Vinyals, O., & Carreira, J. (2021). Perceiver: General Perception with Iterative Attention. arXiv preprint arXiv:2103.03206.

The Transformer for Vision: VisFormer

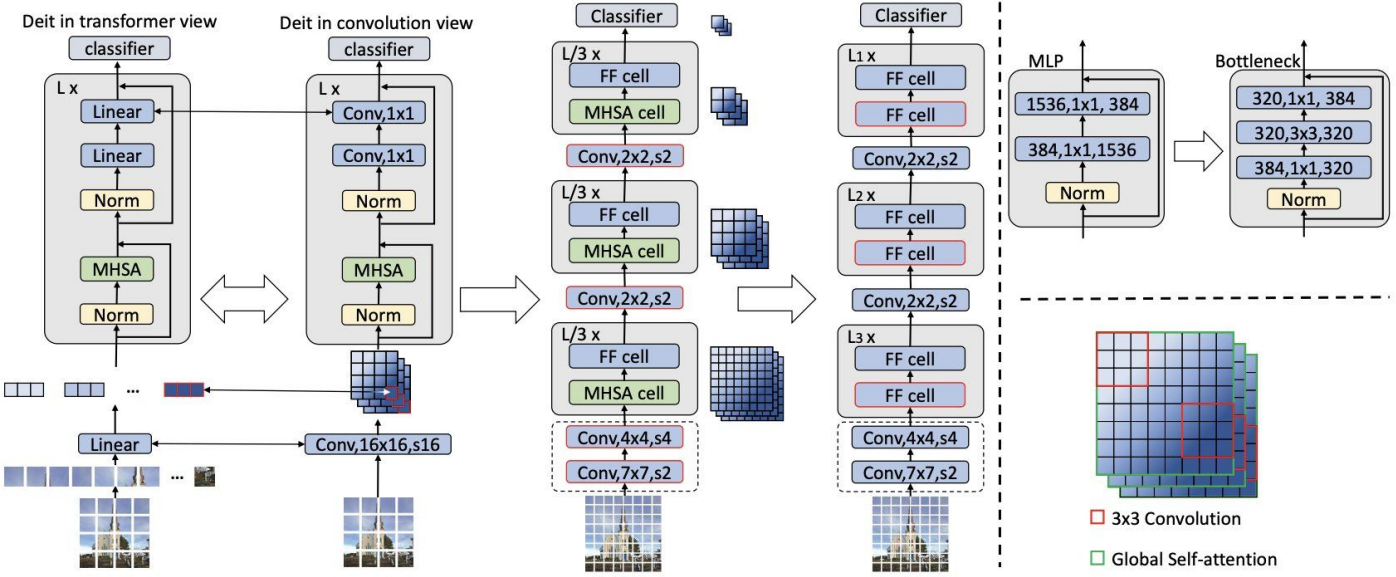
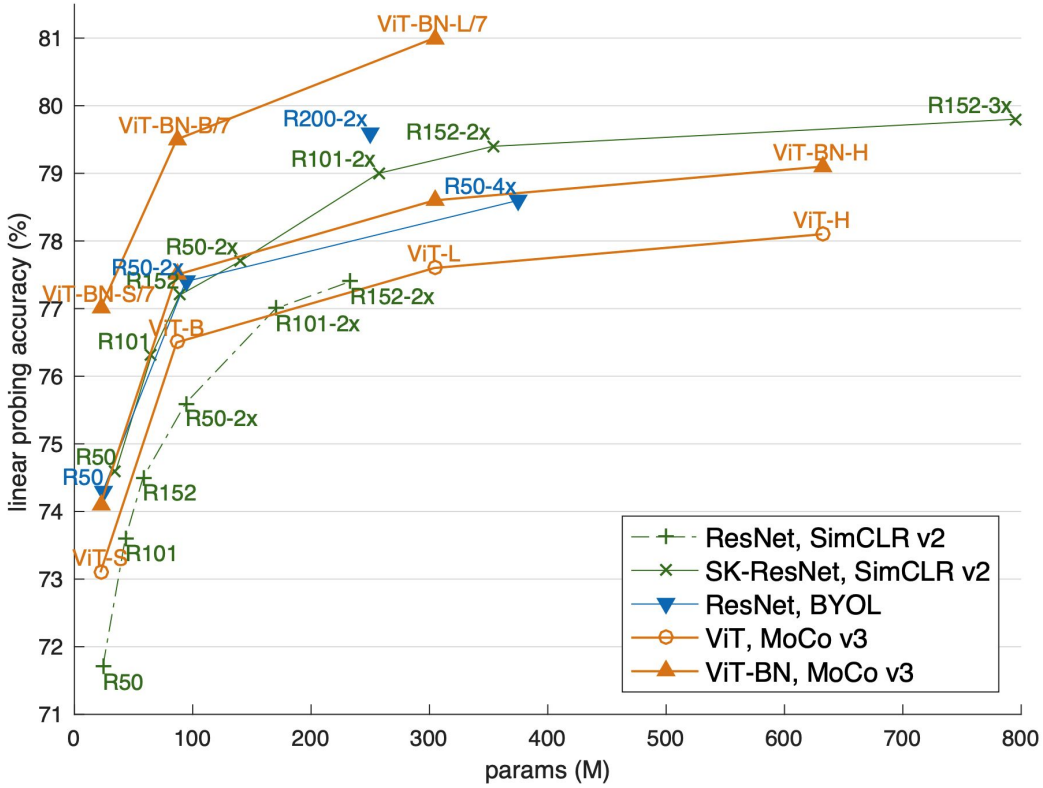


Figure 1. The transition process that starts with DeiT and ends with ResNet-50. To save space, we only show three important steps, that is, (i) replacing the patch flattening module with step-wise patch embedding (elaborated in Section 3.3.2), (ii) introducing the stage-wise design (in Section 3.3.3), and (iii) replacing the self-attention module with convolution (Section 3.3.7). The upper-right area shows a relatively minor modifications, inserting 3×3 convolution (Section 3.3.5). The lower-right area compares the receptive fields of a 3×3 convolution and self-attention. This figure is best viewed in color.

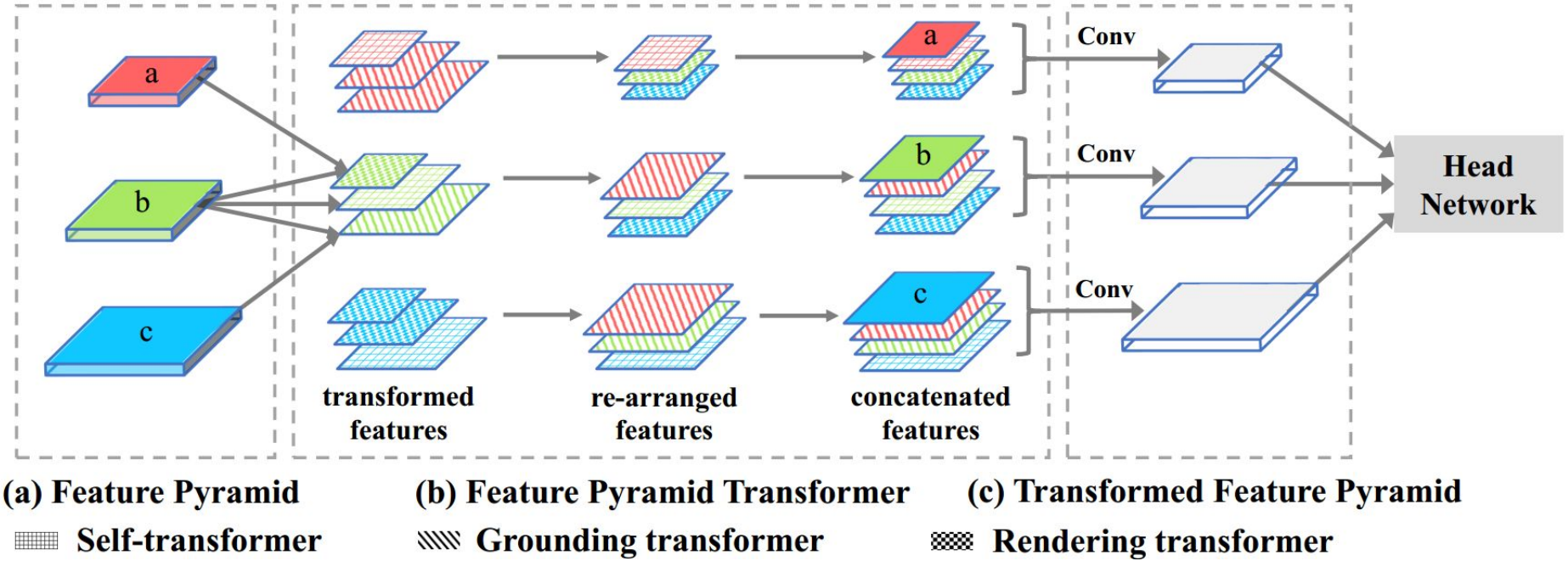
#VisFormer Zhengsu Chen, Lingxi Xie, Jianwei Niu, Xuefeng Liu, Longhui Wei, Qi Tian, "Visformer: The Vision-friendly Transformer" arXiv 2021. [tweet]

The Transformer for Vision: Self-supervised



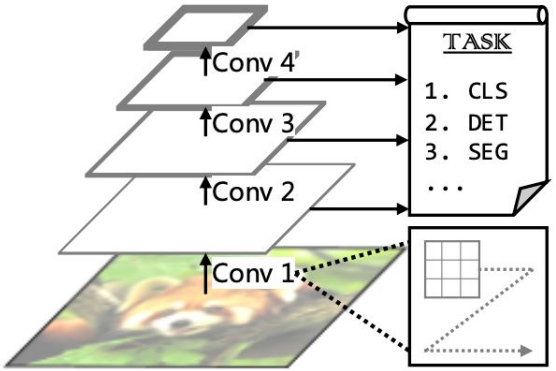
#MoCo-v3 Xinlei Chen, Saining Xie, Kaiming He, "An Empirical Study of Training Self-Supervised Visual Transformers" arXiv 2021.

The Transformer for Vision: Feature Pyramid

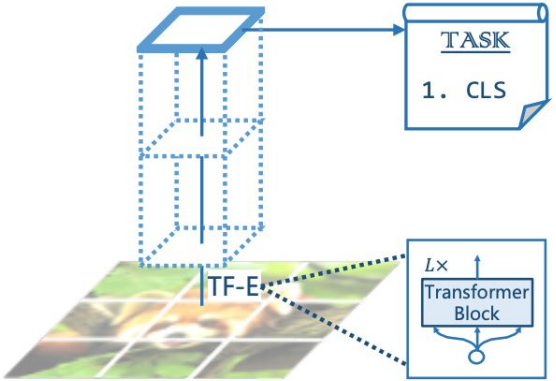


#FPT Zhang, D., Zhang, H., Tang, J., Wang, M., Hua, X., & Sun, Q. Feature pyramid transformer. ECCV 2020.

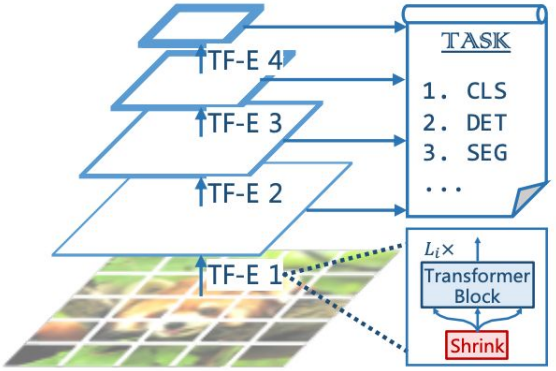
The Transformer for Vision: PVT



(a) CNNs: VGG [41], ResNet [15], *etc.*



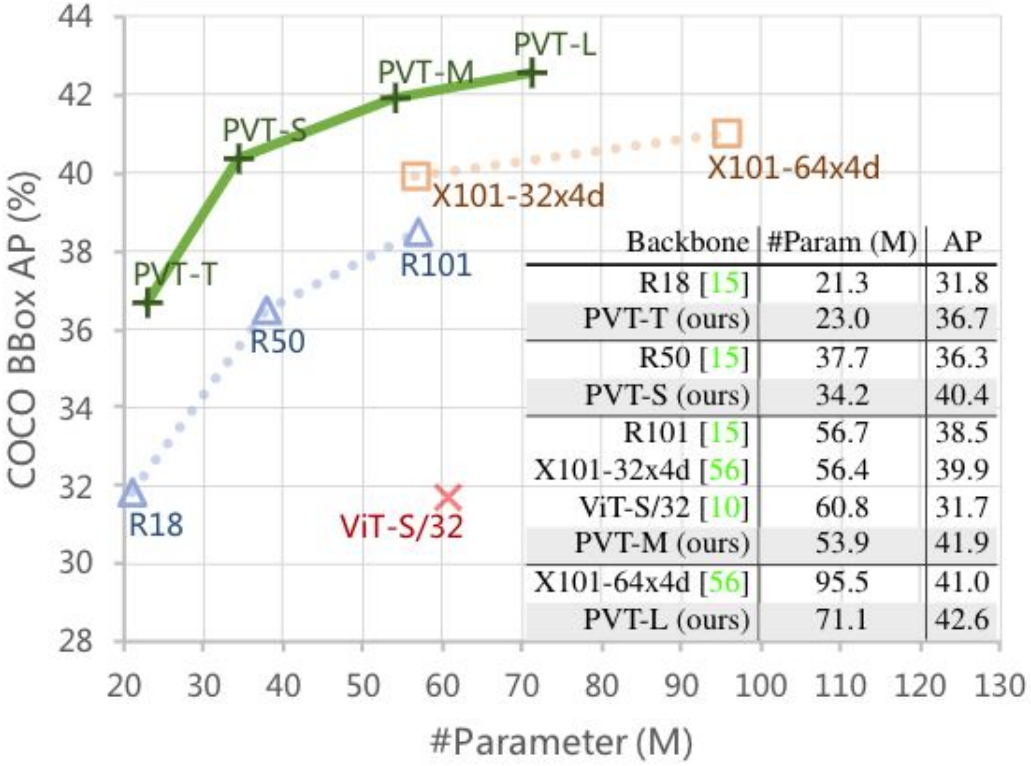
(b) Vision Transformer [10]



(c) Pyramid Vision Transformer (ours)

#PVT Wang, W., Xie, E., Li, X., Fan, D. P., Song, K., Liang, D., ... & Shao, L. Pyramid Vision Transformer: A Versatile Backbone for Dense Prediction without Convolutions. ICCV 2021. [code]

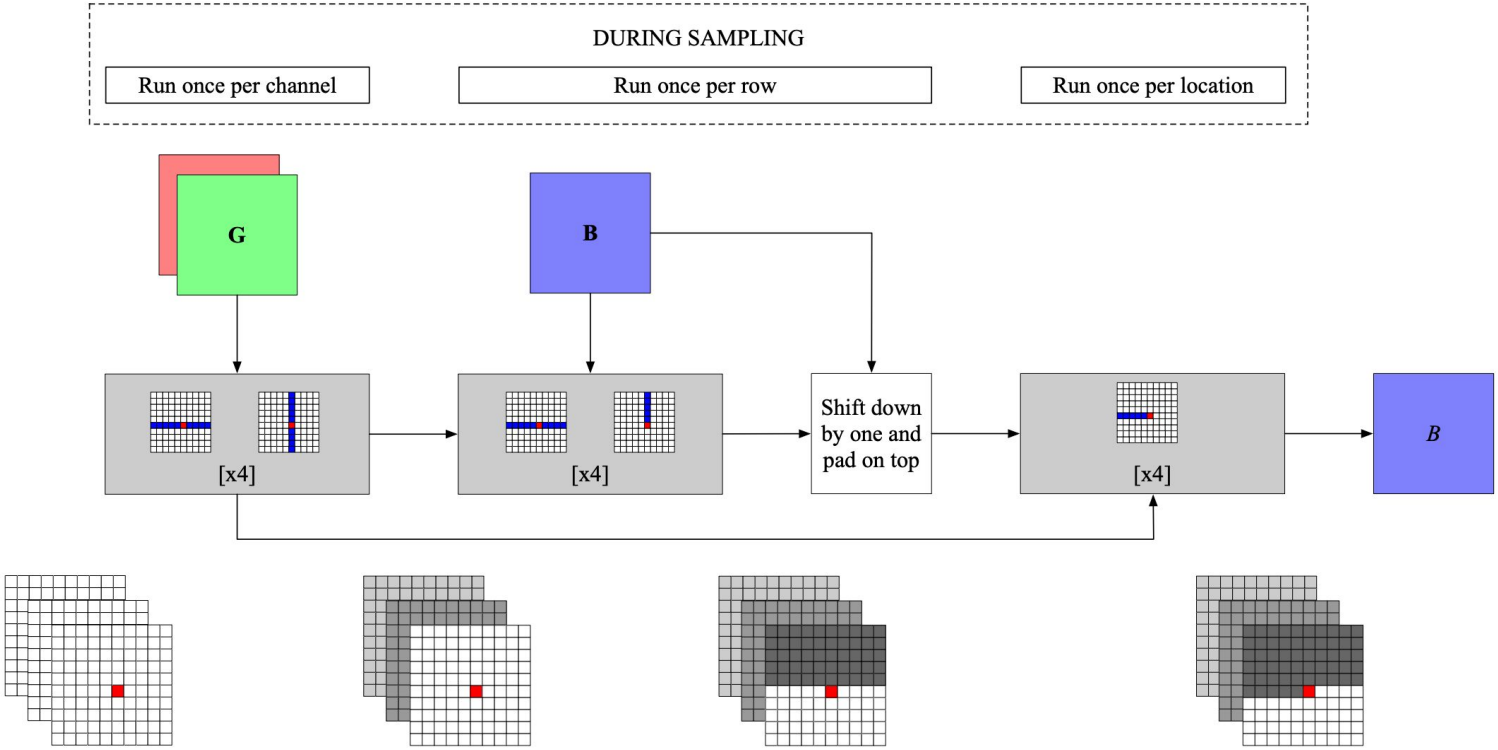
The Transformer for Vision: PVT



Backbone	#Param (M)	AP
R18 [15]	21.3	31.8
PVT-T (ours)	23.0	36.7
R50 [15]	37.7	36.3
PVT-S (ours)	34.2	40.4
R101 [15]	56.7	38.5
X101-32x4d [56]	56.4	39.9
ViT-S/32 [10]	60.8	31.7
PVT-M (ours)	53.9	41.9
X101-64x4d [56]	95.5	41.0
PVT-L (ours)	71.1	42.6

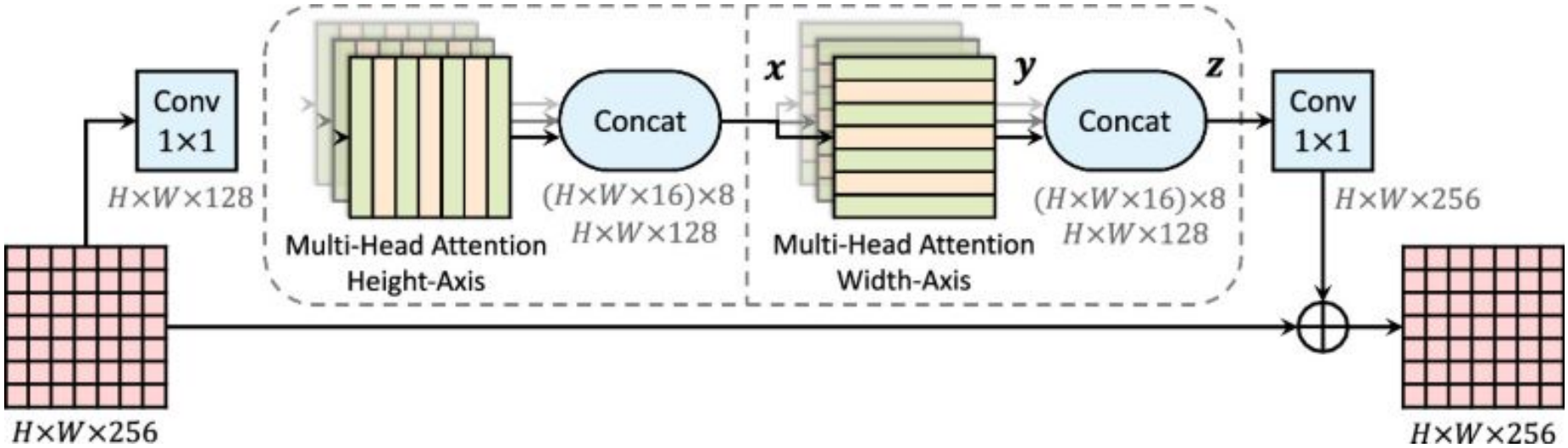
#PVT Wang, W., Xie, E., Li, X., Fan, D. P., Song, K., Liang, D., ... & Shao, L. Pyramid Vision Transformer: A Versatile Backbone for Dense Prediction without Convolutions. ICCV 2021. [code]

Axial Attention



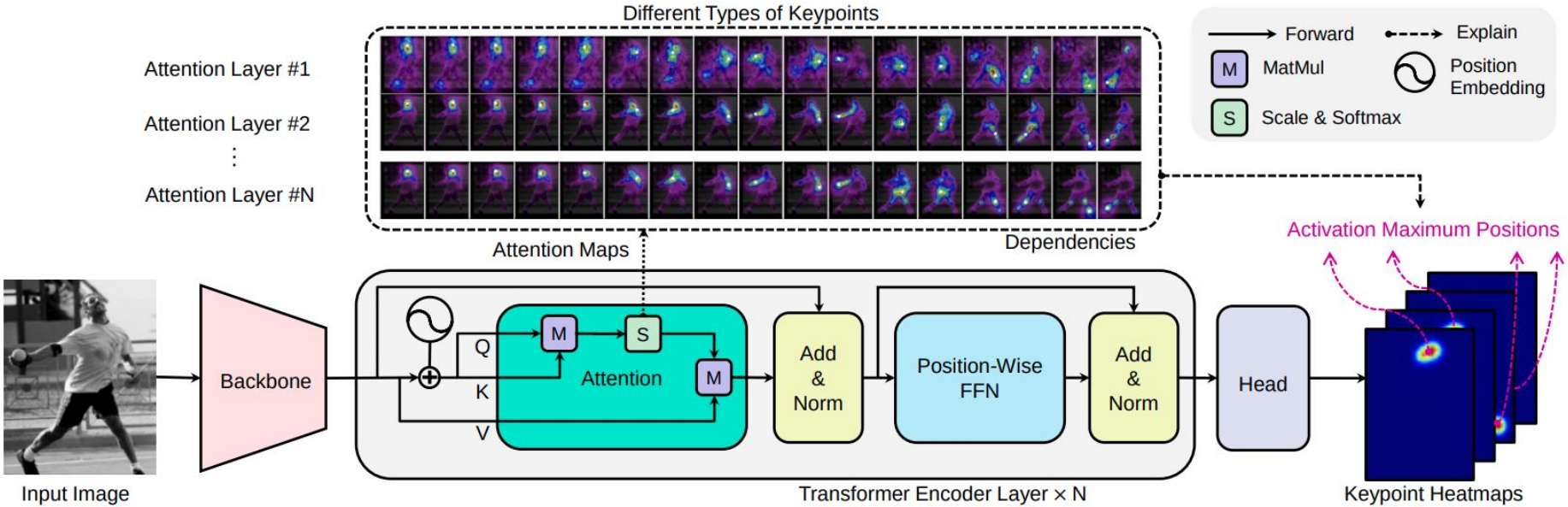
Ho, J., Kalchbrenner, N., Weissenborn, D., & Salimans, T. Axial attention in multidimensional transformers. arXiv 2019.

The Transformer for Vision: Segmentation



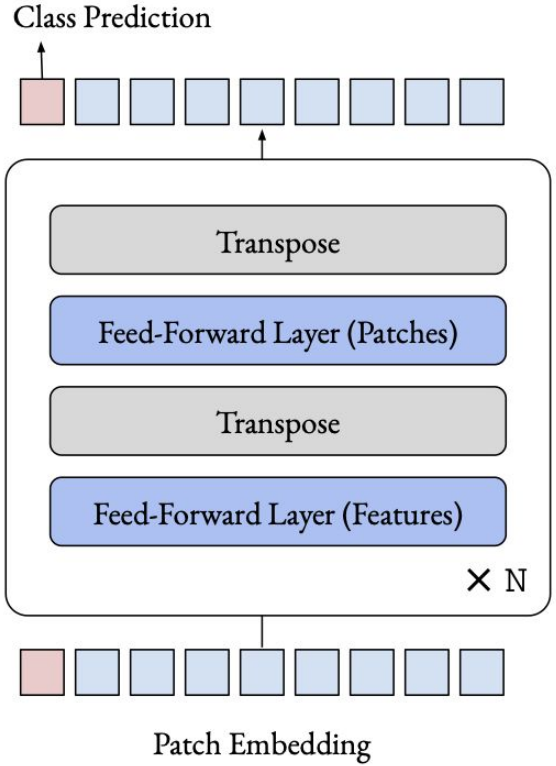
Wang, H., Zhu, Y., Green, B., Adam, H., Yuille, A., & Chen, L. C. Axial-deeplab: Stand-alone axial-attention for panoptic segmentation. ECCV 2020. [video by Yannic Kilcher]

Pose estimation



Yang, Sen, Zhibin Quan, Mu Nie, and Wankou Yang. "Transpose: Keypoint localization via transformer." ICCV 2021. [code][talk]

Is attention all we need ?



Luke Melas-Kyriazi, "Do You Even Need Attention? A Stack of Feed-Forward Layers Does Surprisingly Well on ImageNet" arXiv 2021. [code]