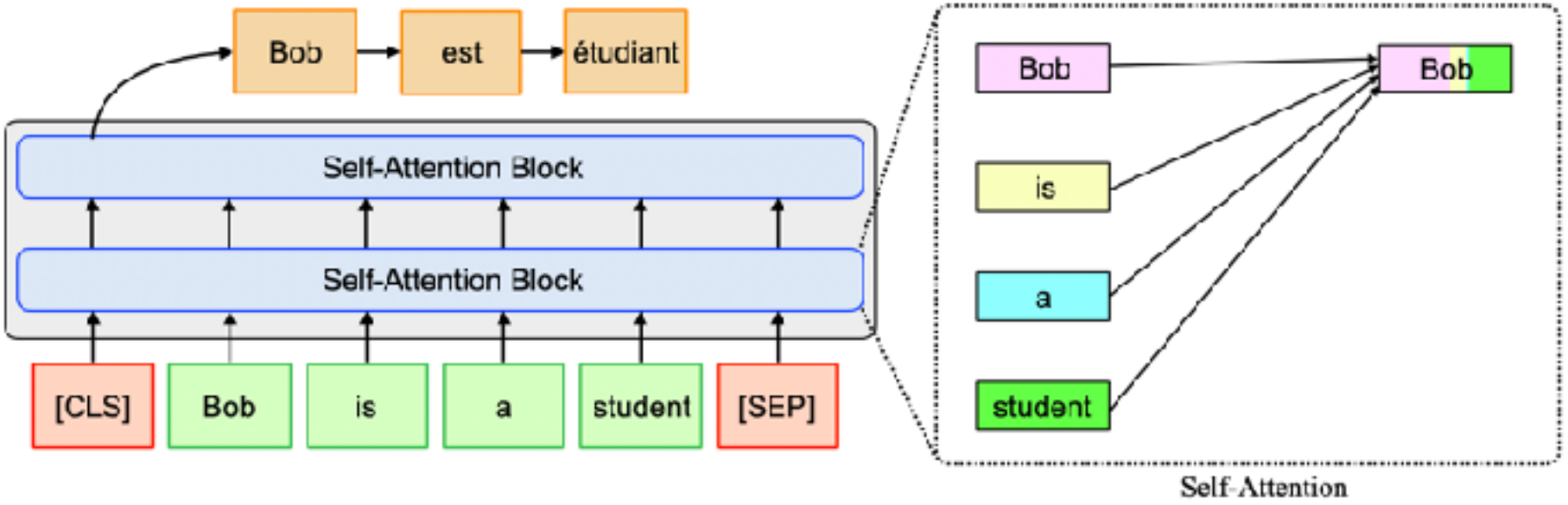
An Image Is Worth 16x16 Words: Transformers for Image Recognition at Scale

ICLR 2020

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov,
Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner,
Mostafa Dehghani, Matthias Minderer, Georg Heigold,
Sylvain Gelly, Jakob Uszkoreit, Neil Houlsby

Recap of Self-Attention

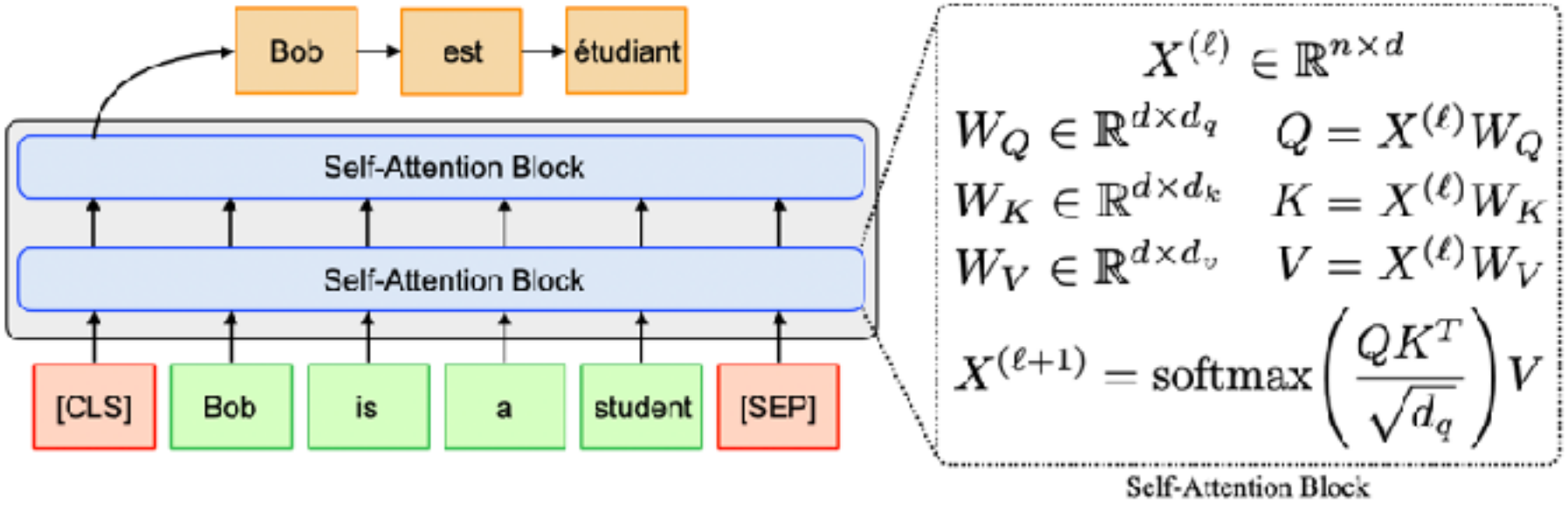
- Self-attention enables capturing long-range dependencies among words.



Vaswani et al., ”Attention is All You Need”, NIPS 2017

Recap of Self-Attention

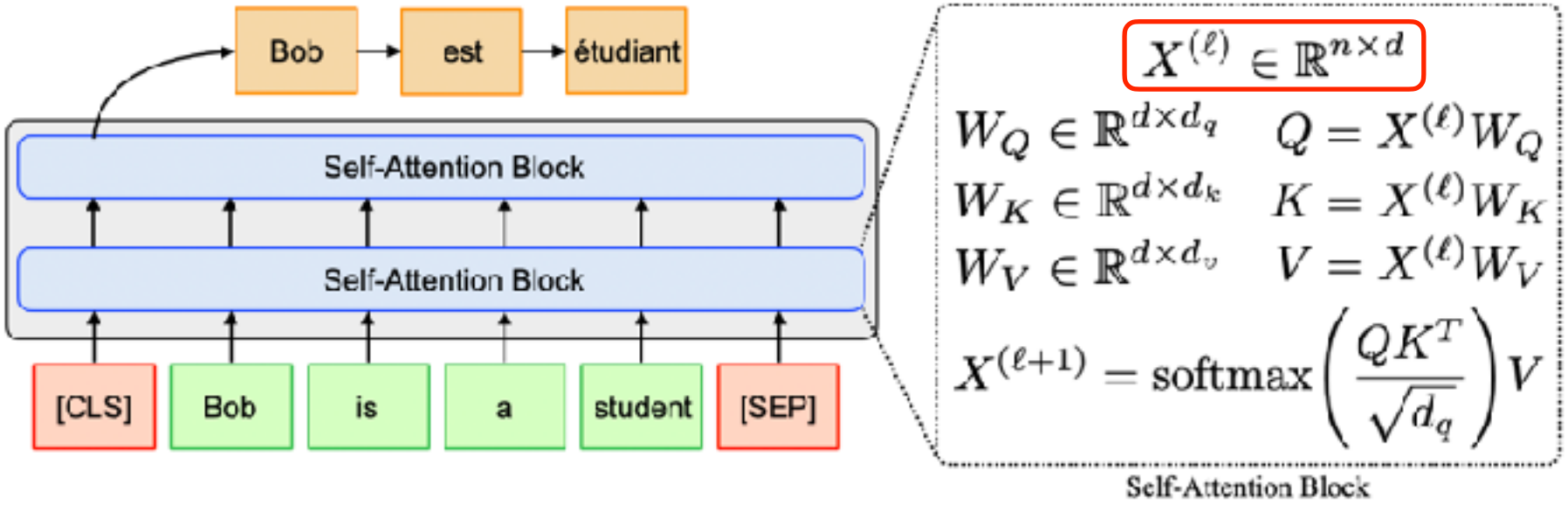
- Self-attention enables capturing long-range dependencies among words.



Vaswani et al., ”Attention is All You Need”, NIPS 2017

Recap of Self-Attention

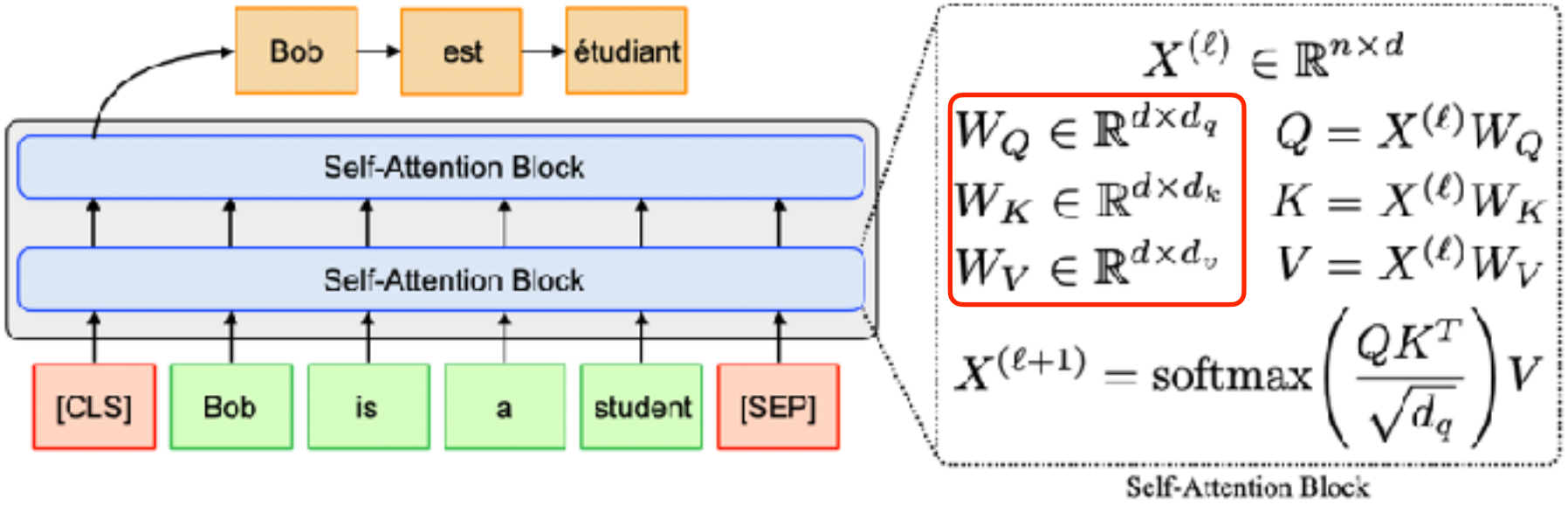
- Self-attention enables capturing long-range dependencies among words.



Vaswani et al., ”Attention is All You Need”, NIPS 2017

Recap of Self-Attention

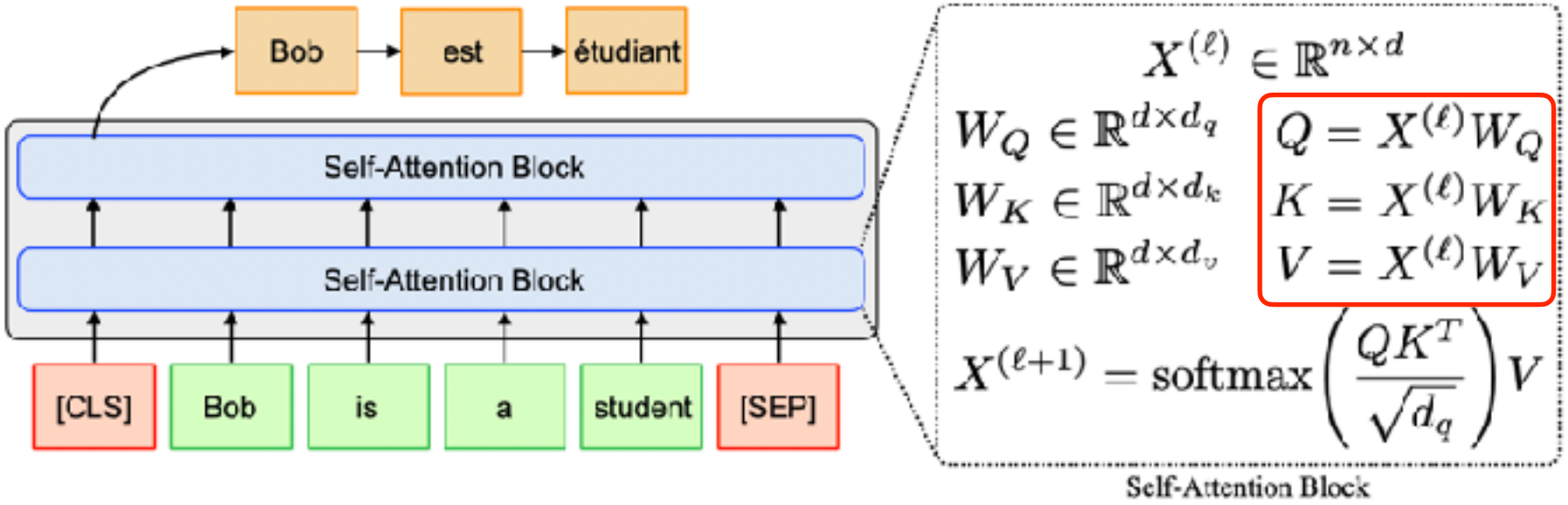
- Self-attention enables capturing long-range dependencies among words.



Vaswani et al., ”Attention is All You Need”, NIPS 2017

Recap of Self-Attention

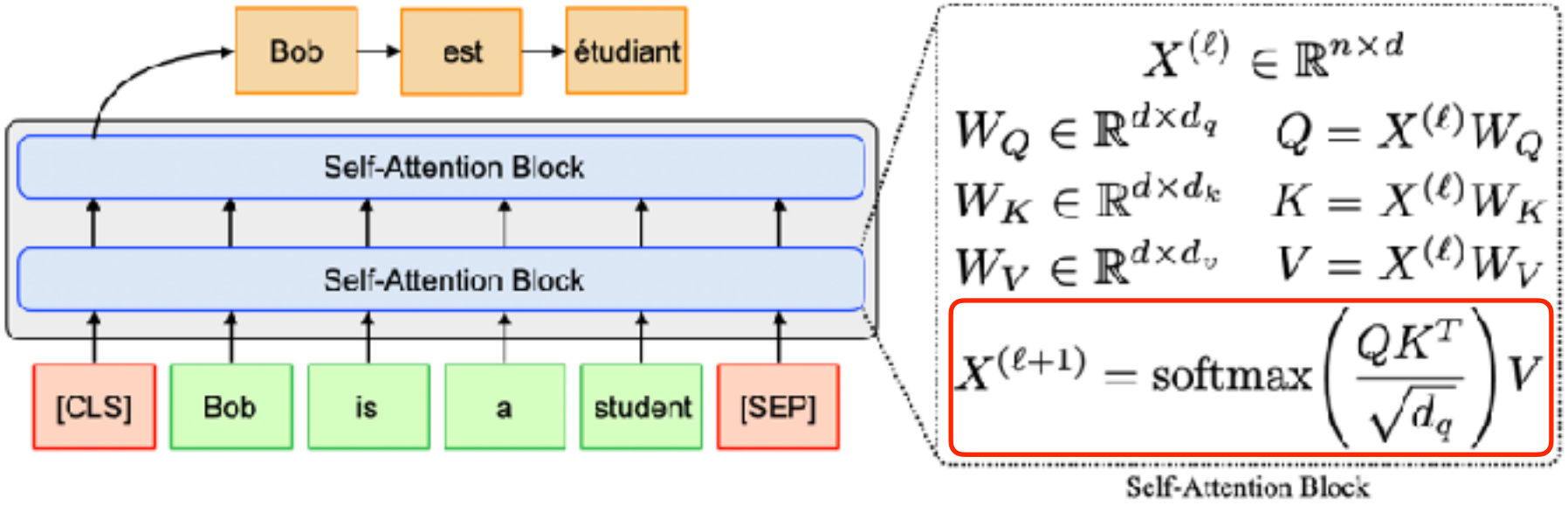
- Self-attention enables capturing long-range dependencies among words.



Vaswani et al., ”Attention is All You Need”, NIPS 2017

Recap of Self-Attention

- Self-attention enables capturing long-range dependencies among words.

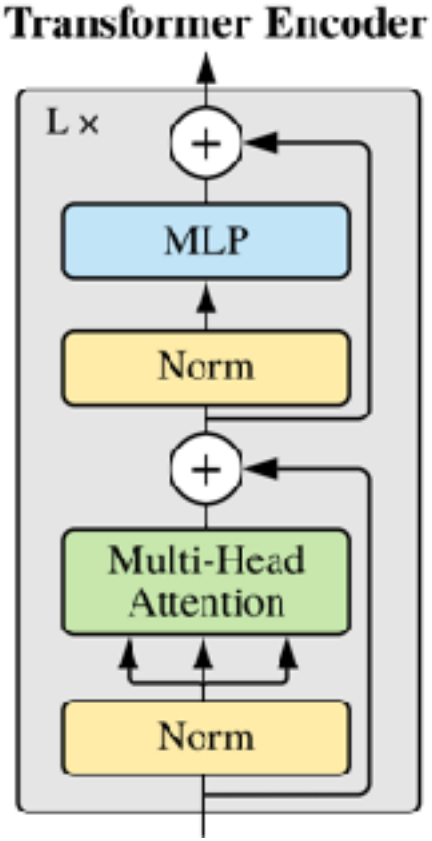


Vaswani et al., ”Attention is All You Need”, NIPS 2017

Transformers for Visual Data

- How do we apply transformers on visual data (e.g., images or videos)?

???



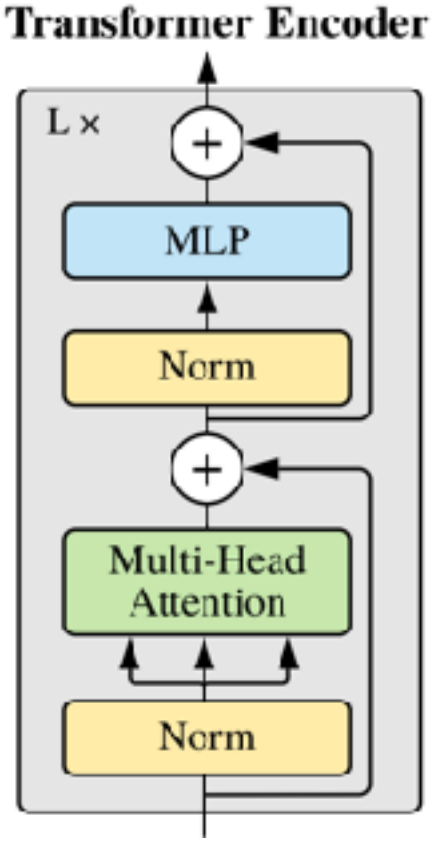
Why Transformers in CV?

1. In the recent years, transformers have revolutionized the field of natural language processing (NLP).
2. Convolutional networks are designed to capture short-range local connections in visual data while transformers can capture long-range dependencies.
3. Convolutional networks can be difficult to scale to larger model sizes, which are needed for modern big data regimes.
4. Transformers are much better suited for multimodal learning.

Challenges

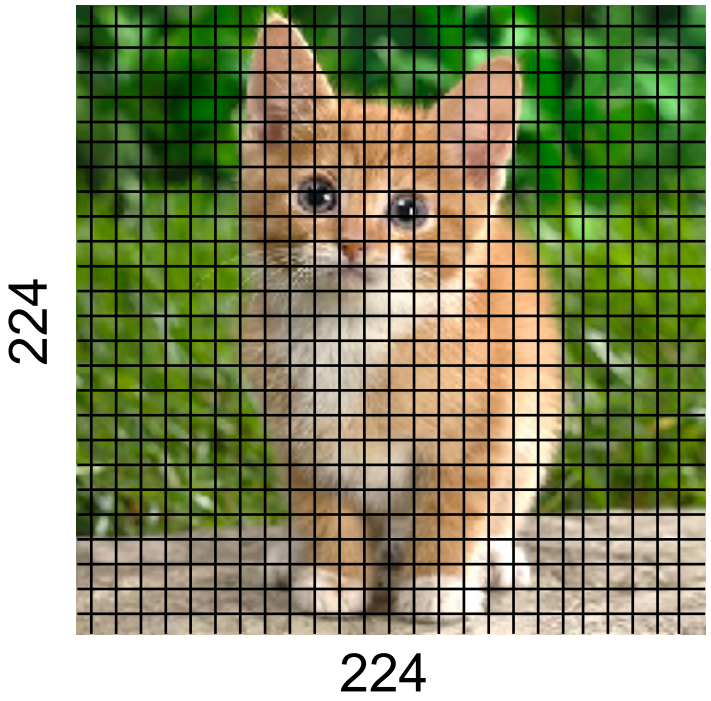
- How do we apply transformers on visual data (e.g., images or videos)?

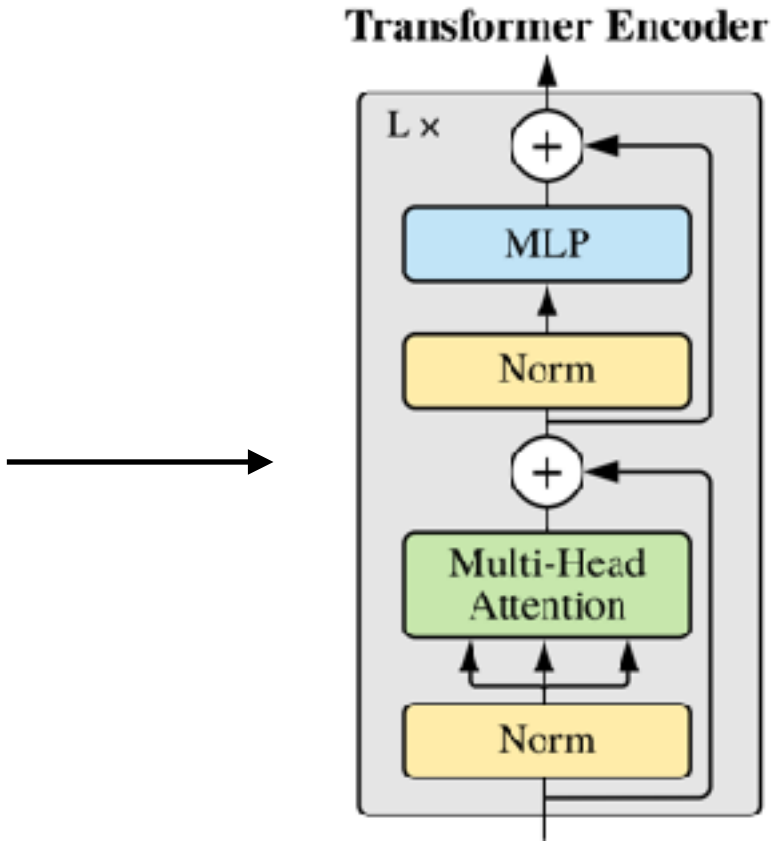
???



Challenges

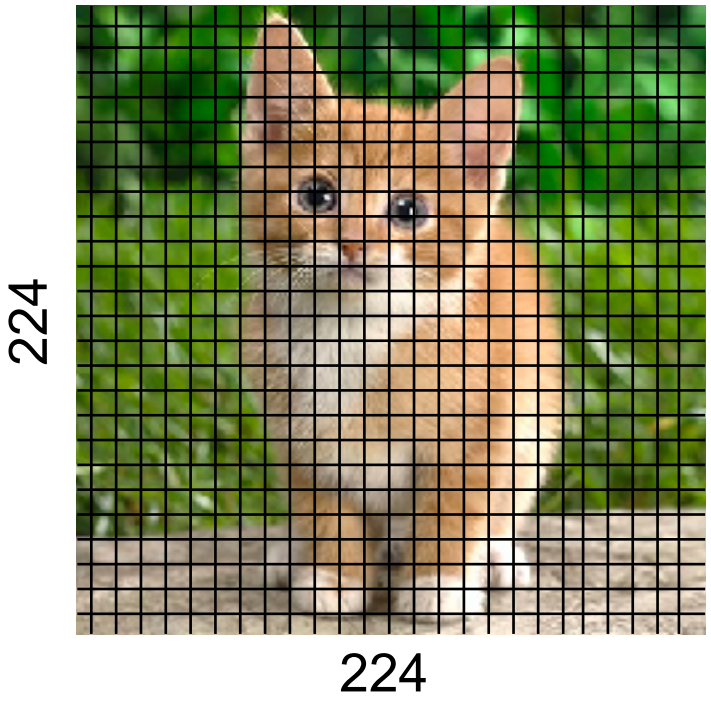
- How do we apply transformers on visual data (e.g., images or videos)?

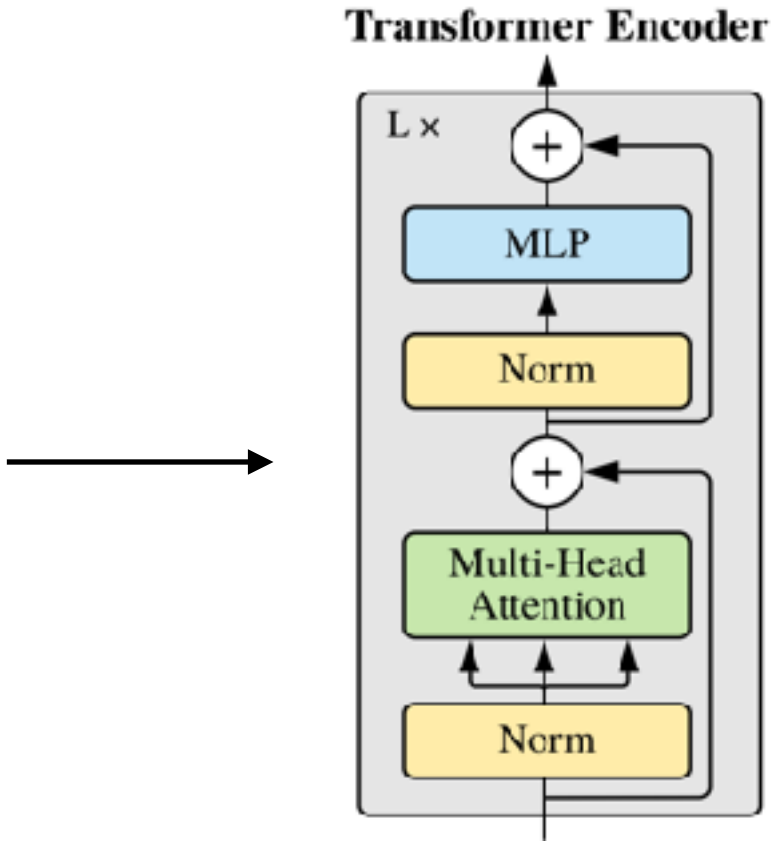




Challenges

- How do we apply transformers on visual data (e.g., images or videos)?

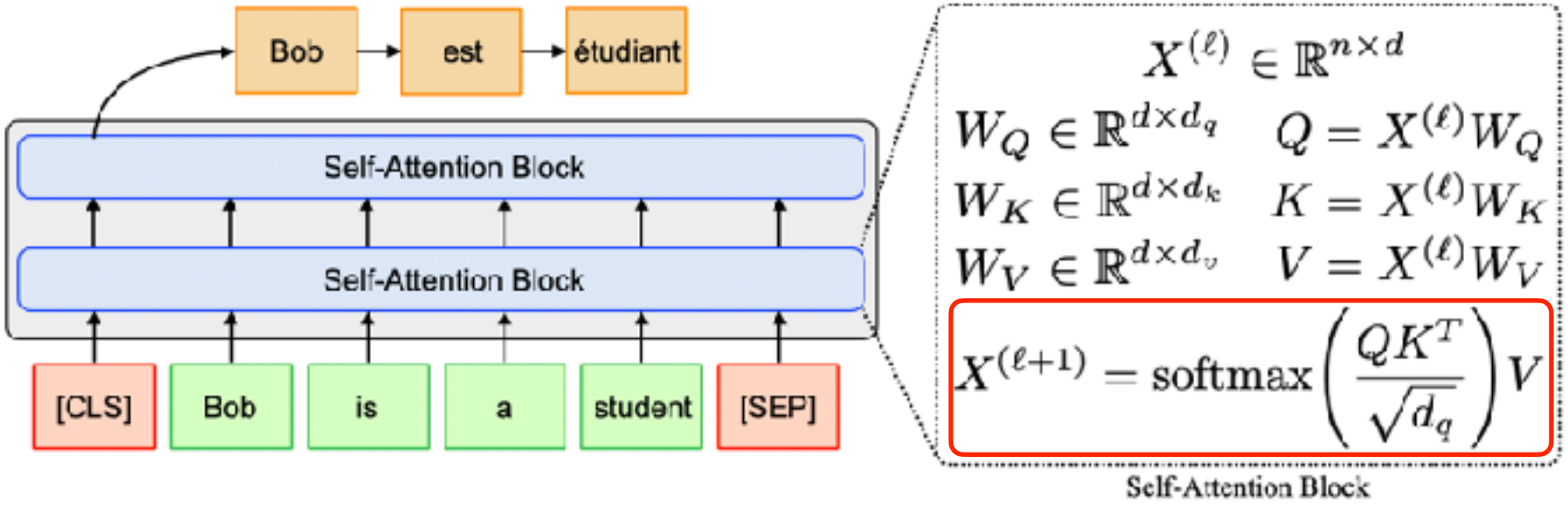




We can then feed the resulting 224^2 = 50,176 image tokens into a standard transformer model.

Recap of Self-Attention

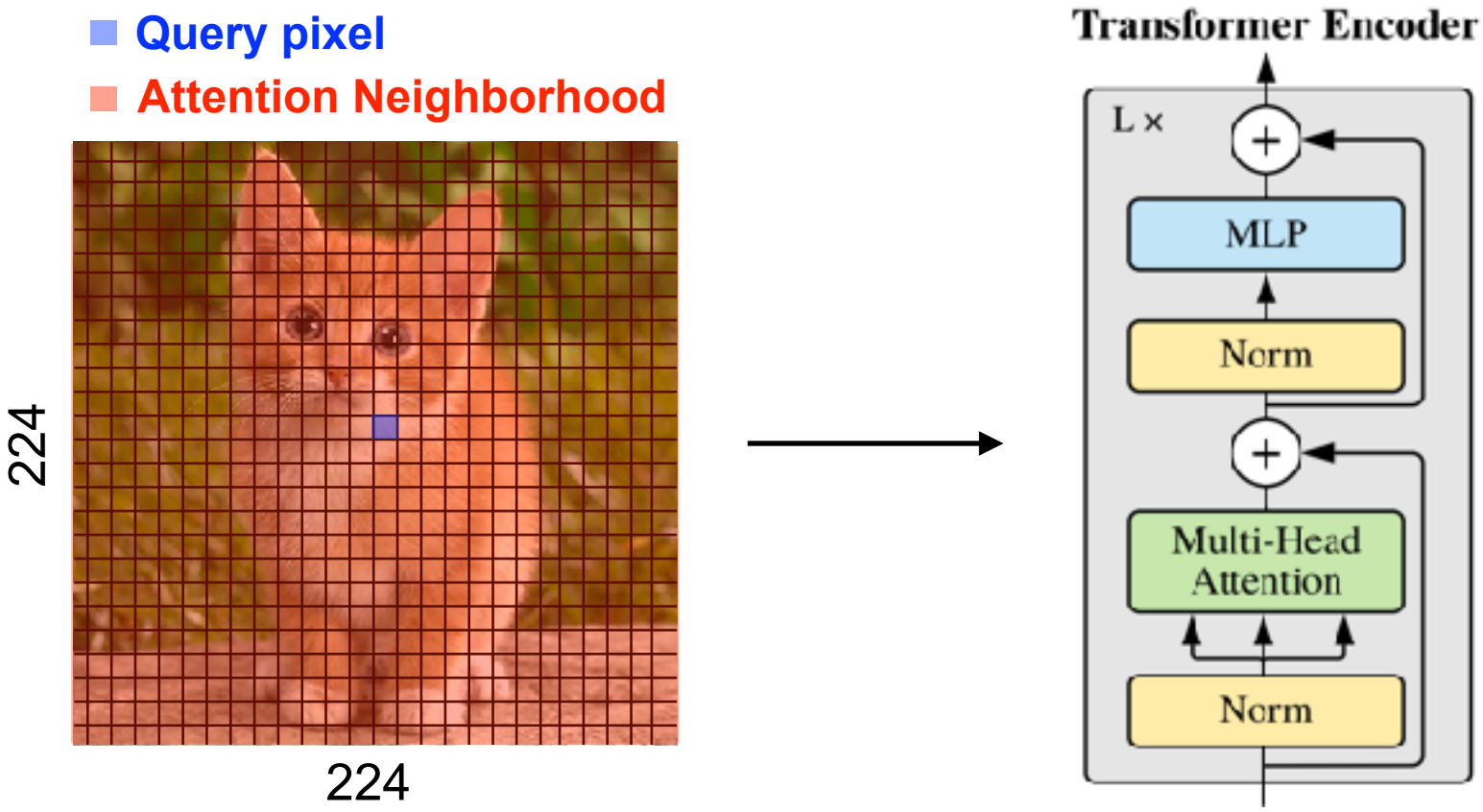
- Self-attention enables capturing long-range dependencies among words.



Vaswani et al., ”Attention is All You Need”, NIPS 2017

Challenges

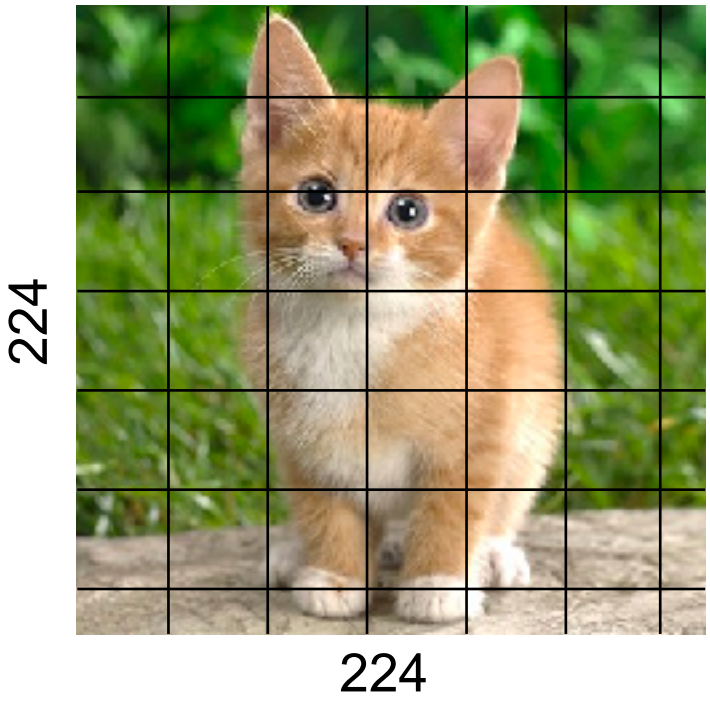
- How do we apply transformers on visual data (e.g., images or videos)?

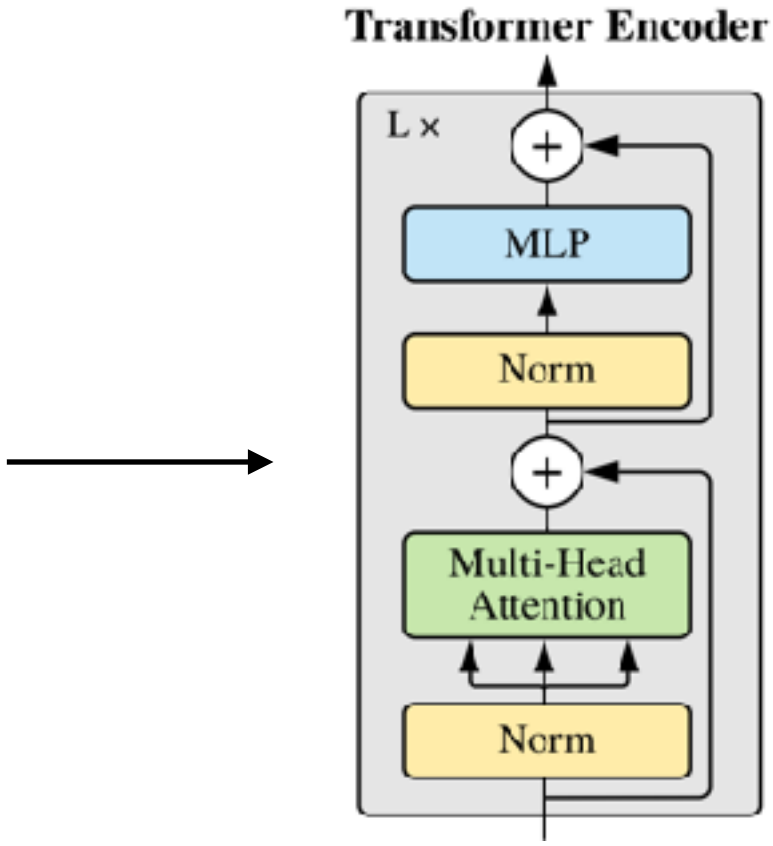


The self-attention cost scales quadratically with the number of tokens (e.g., 50,176^2 = 2.5B).

Challenges

- How do we apply transformers on visual data (e.g., images or videos)?

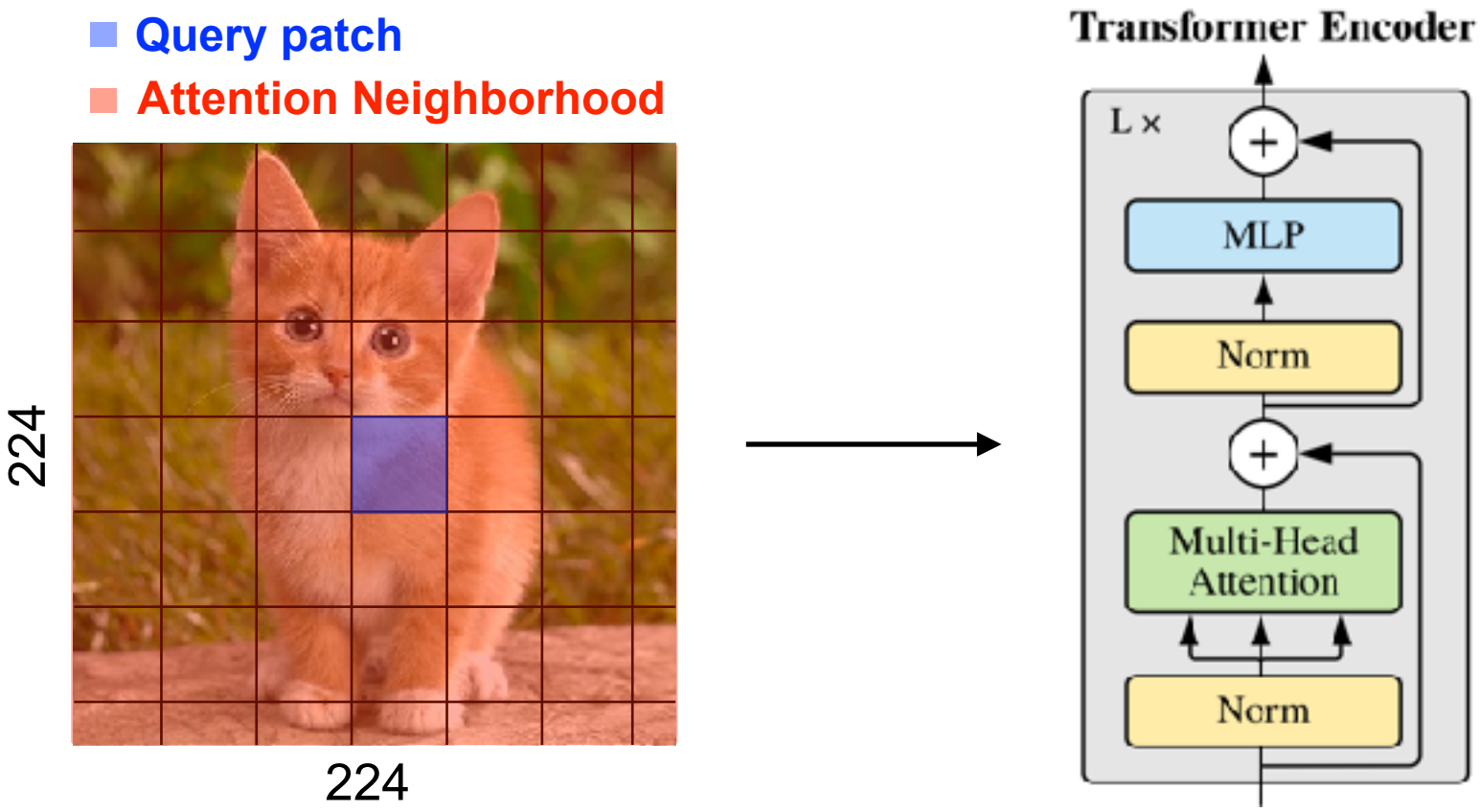




Divide an image into a set of non-overlapping 16x16 patches

Challenges

- How do we apply transformers on visual data (e.g., images or videos)?



This results in 196 patches for a 224x224 image, which enables the application of standard global self-attention.

Image Classification

- The goal is to identify the category of a given image.

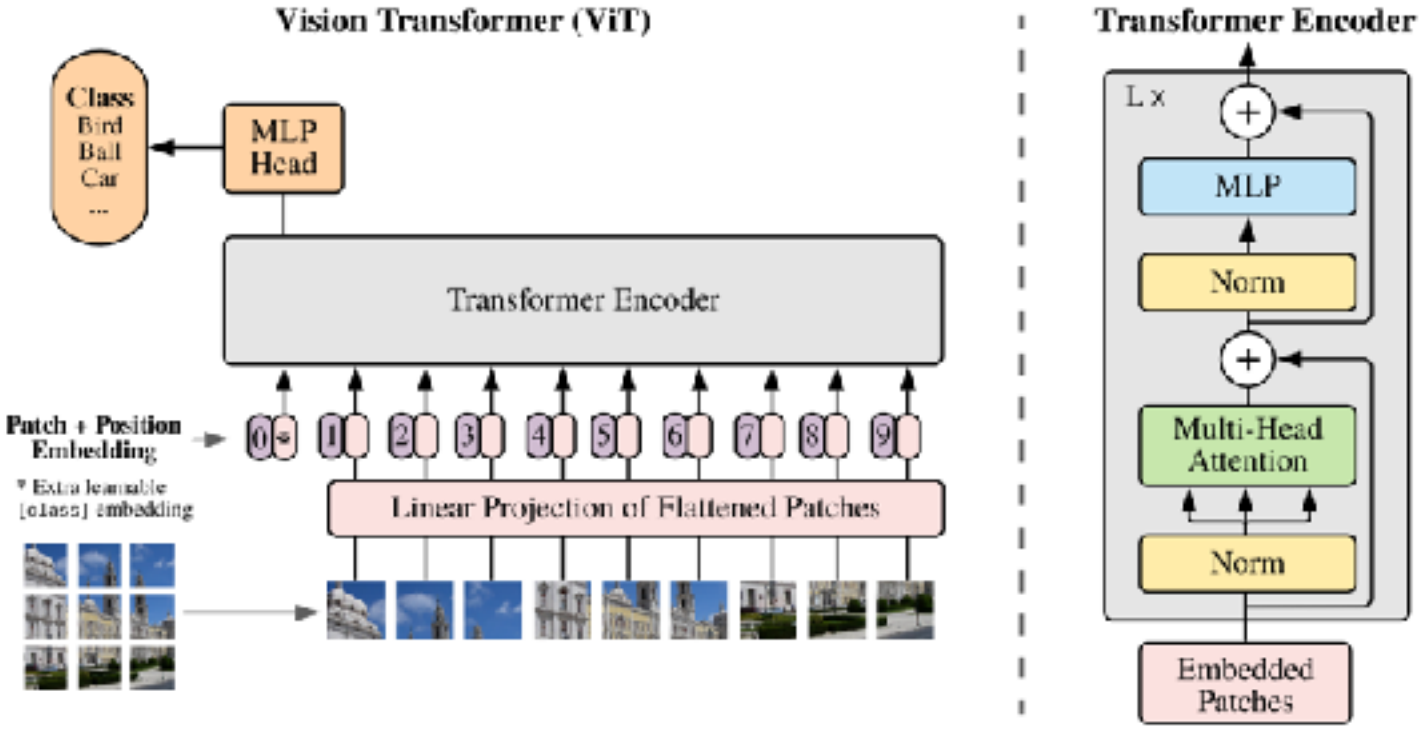
Input:

Classification

Output:

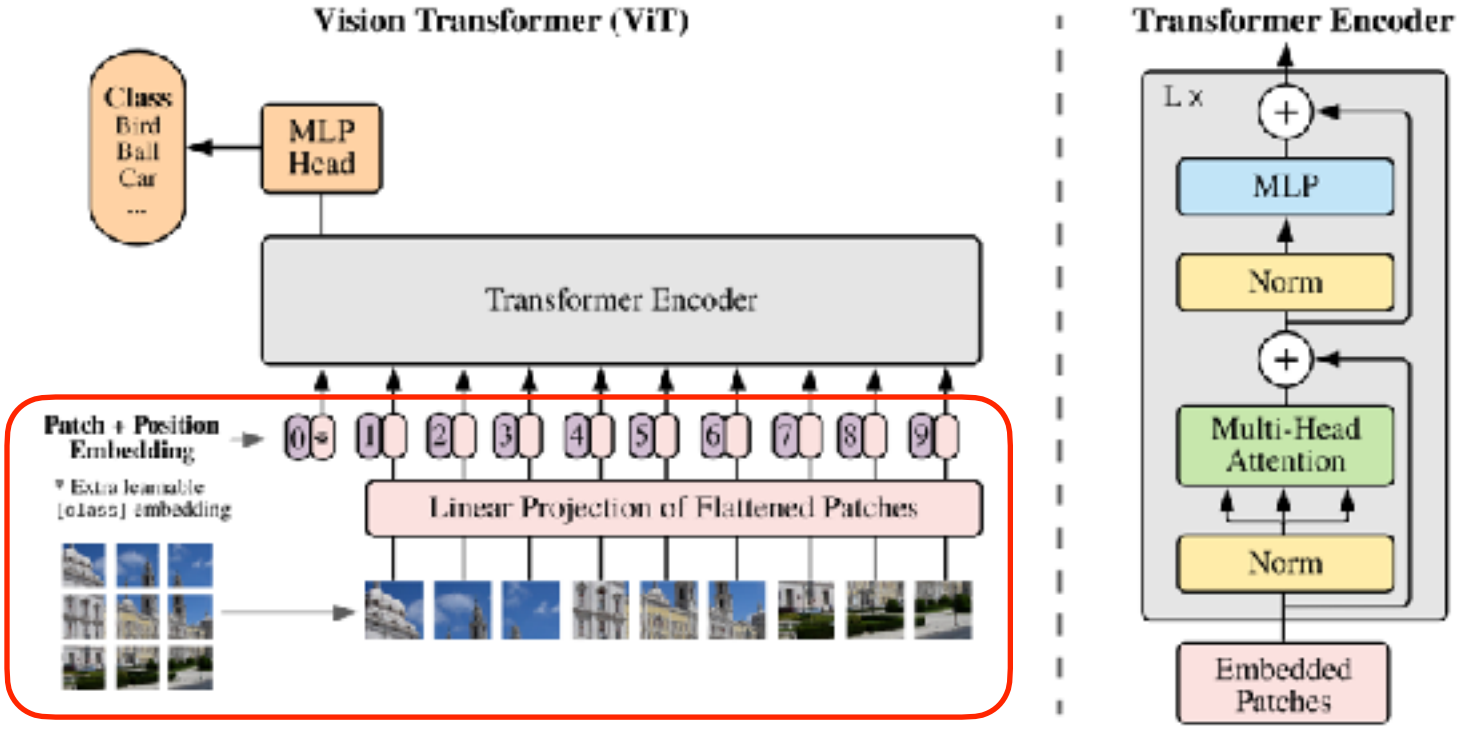
Vision Transformer (ViT)

- The authors split an image into fixed-size patches, linearly embed each of them, and add position embeddings.
- The resulting sequence of vectors is then fed into a standard Transformer encoder.



Vision Transformer (ViT)

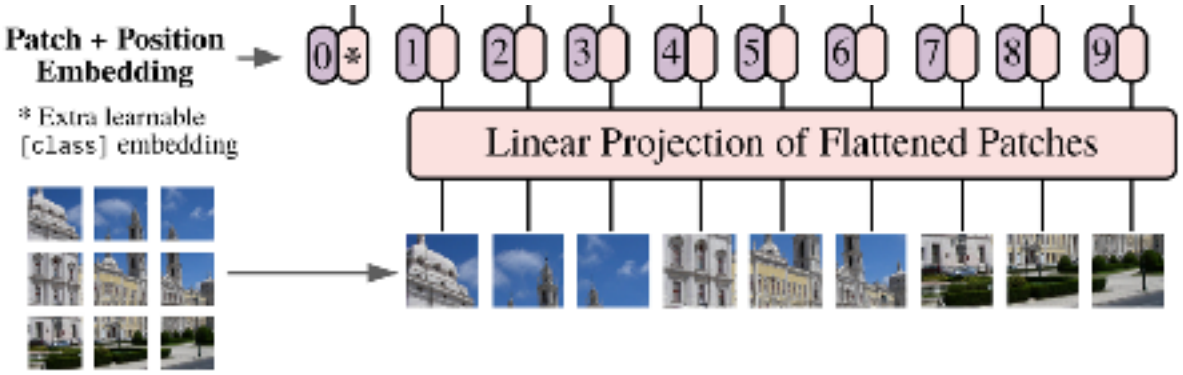
- The authors split an image into fixed-size patches, linearly embed each of them, and add position embeddings.
- The resulting sequence of vectors is then fed into a standard Transformer encoder.



Linear Projection of Flattened Patches

- The authors reshape the image $x \in \mathbb{R}^{H \times W \times C}$ into a sequence of flattened 2D patches $x_p \in \mathbb{R}^{N \times (P^2 \cdot C)}$.
- The patches are then linearly projected using a trainable projection layer E.

$$\mathbf{z}_0 = [\mathbf{x}_{\text{class}}; \mathbf{x}_p^1\mathbf{E}; \mathbf{x}_p^2\mathbf{E}; \cdots; \mathbf{x}_p^N\mathbf{E}] + \mathbf{E}_{pos},$$



Linear Projection of Flattened Patches

- The authors reshape the image $x \in \mathbb{R}^{H \times W \times C}$ into a sequence of flattened 2D patches $x_p \in \mathbb{R}^{N \times (P^2 \cdot C)}$.
- The patches are then linearly projected using a trainable projection layer E.

$$\mathbf{z}_0 = [\mathbf{x}_{\text{class}}; \mathbf{x}_p^1\mathbf{E}; \mathbf{x}_p^2\mathbf{E}; \cdots; \mathbf{x}_p^N\mathbf{E}] + \mathbf{E}_{pos},$$

A sequence of flattened patch vectors.

Linear Projection of Flattened Patches

- The authors reshape the image $x \in \mathbb{R}^{H \times W \times C}$ into a sequence of flattened 2D patches $x_p \in \mathbb{R}^{N \times (P^2 \cdot C)}$.
- The patches are then linearly projected using a trainable projection layer E.

$$\mathbf{z}_0 = [\mathbf{x}_{\text{class}}; \mathbf{x}_p^1\mathbf{E}; \mathbf{x}_p^2\mathbf{E}; \cdots; \mathbf{x}_p^N\mathbf{E}] + \mathbf{E}_{pos},$$

A trainable linear projection (i.e., a fully connected layer).

$$\mathbf{E} \in \mathbb{R}^{(P^2 \cdot C) \times D}$$

Linear Projection of Flattened Patches

- The authors reshape the image $x \in \mathbb{R}^{H \times W \times C}$ into a sequence of flattened 2D patches $x_p \in \mathbb{R}^{N \times (P^2 \cdot C)}$.
- The patches are then linearly projected using a trainable projection layer E.

$$\mathbf{z}_0 = [\mathbf{x}_{\text{class}}; \mathbf{x}_p^1\mathbf{E}; \mathbf{x}_p^2\mathbf{E}; \cdots; \mathbf{x}_p^N\mathbf{E}] + \mathbf{E}_{pos},$$

D dimensional linear embeddings of the patches.

Linear Projection of Flattened Patches

- The authors reshape the image $x \in \mathbb{R}^{H \times W \times C}$ into a sequence of flattened 2D patches $x_p \in \mathbb{R}^{N \times (P^2 \cdot C)}$.
- The patches are then linearly projected using a trainable projection layer E.

$$\mathbf{z}_0 = [\mathbf{x}_{\text{class}}; \mathbf{x}_p^1\mathbf{E}; \mathbf{x}_p^2\mathbf{E}; \cdots; \mathbf{x}_p^N\mathbf{E}] + \mathbf{E}_{pos},$$

A special learnable embedding, which will serve as the image representation used for classification.

Linear Projection of Flattened Patches

- The authors reshape the image $x \in \mathbb{R}^{H \times W \times C}$ into a sequence of flattened 2D patches $x_p \in \mathbb{R}^{N \times (P^2 \cdot C)}$.
- The patches are then linearly projected using a trainable projection layer E.

$$\mathbf{z}_0 = [\mathbf{x}_{\text{class}}; \mathbf{x}_p^1\mathbf{E}; \mathbf{x}_p^2\mathbf{E}; \cdots; \mathbf{x}_p^N\mathbf{E}] + \mathbf{E}_{pos},$$

Learnable 1D position embeddings.

$$\mathbf{E}_{pos} \in \mathbb{R}^{(N+1) \times D}$$

Linear Projection of Flattened Patches

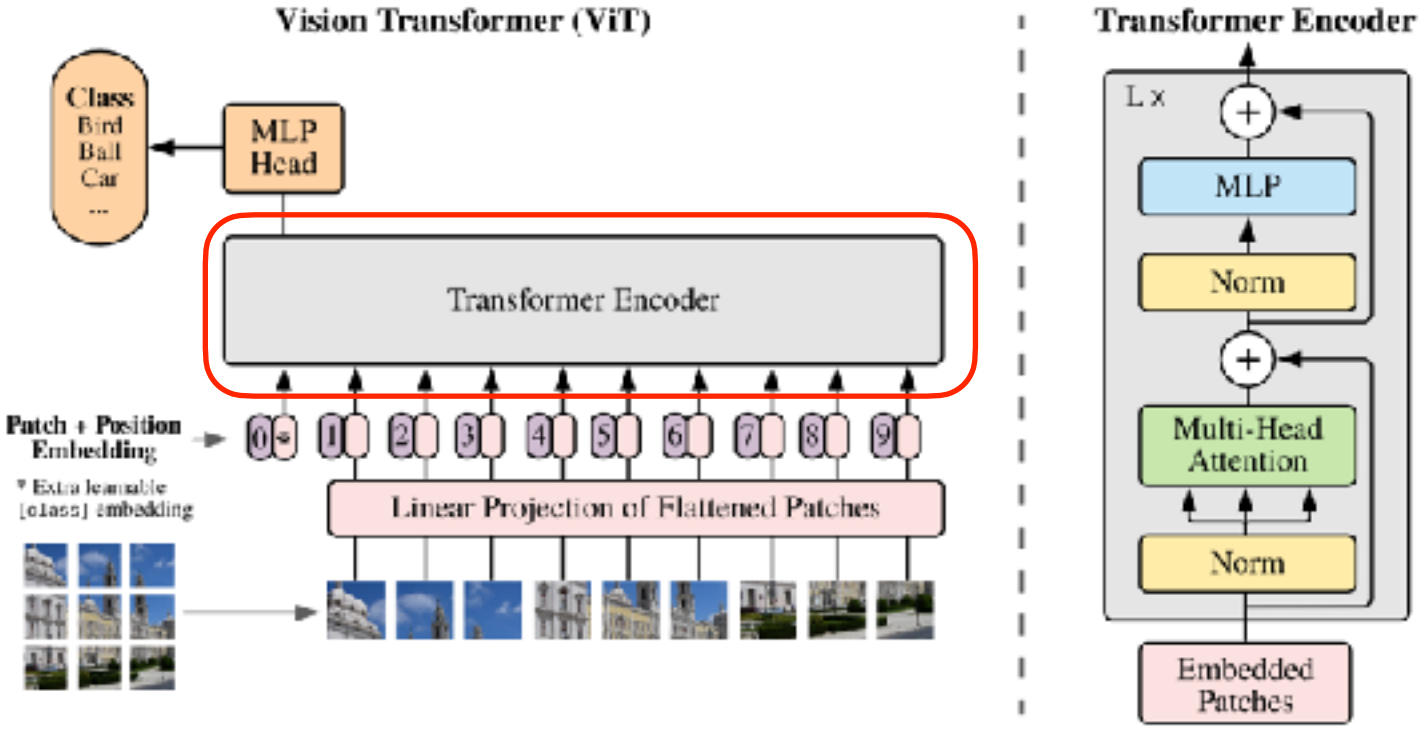
- The authors reshape the image $x \in \mathbb{R}^{H \times W \times C}$ into a sequence of flattened 2D patches $x_p \in \mathbb{R}^{N \times (P^2 \cdot C)}$.
- The patches are then linearly projected using a trainable projection layer E.

$$\mathbf{z}_0 = [\mathbf{x}_{\text{class}}; \mathbf{x}_p^1\mathbf{E}; \mathbf{x}_p^2\mathbf{E}; \cdots; \mathbf{x}_p^N\mathbf{E}] + \mathbf{E}_{pos},$$

A sequence of vectors used as input to the Transformer at layer 0.

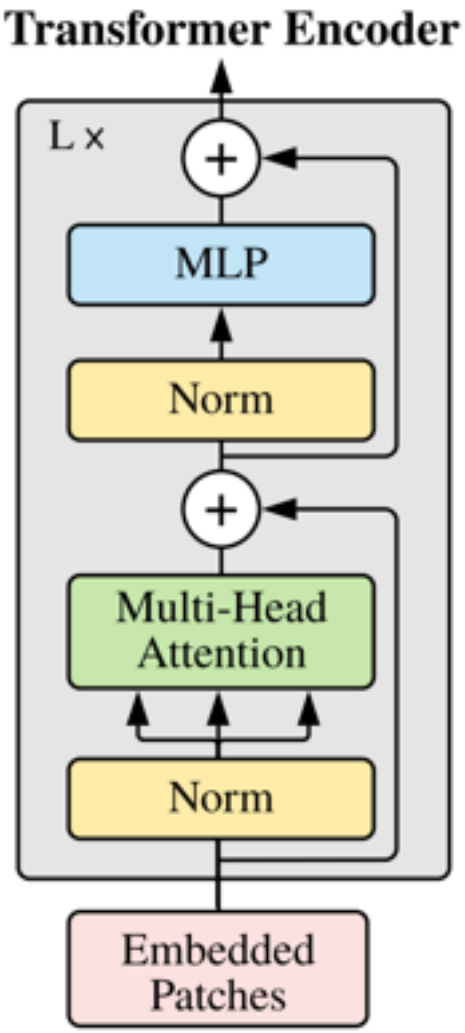
Vision Transformer (ViT)

- The authors split an image into fixed-size patches, linearly embed each of them, and add position embeddings.
- The resulting sequence of vectors is then fed into a standard Transformer encoder.



Transformer Encoder

- The Transformer encoder consists of alternating layers of self-attention (MSA) and MLP blocks.
- Layernorm (LN) is applied before every block, and residual connections after every block.
- The MLP contains two layers with a GELU non-linearity.



Vision Transformer (ViT)

- The authors split an image into fixed-size patches, linearly embed each of them, and add position embeddings.
- The resulting sequence of vectors is then fed into a standard Transformer encoder.

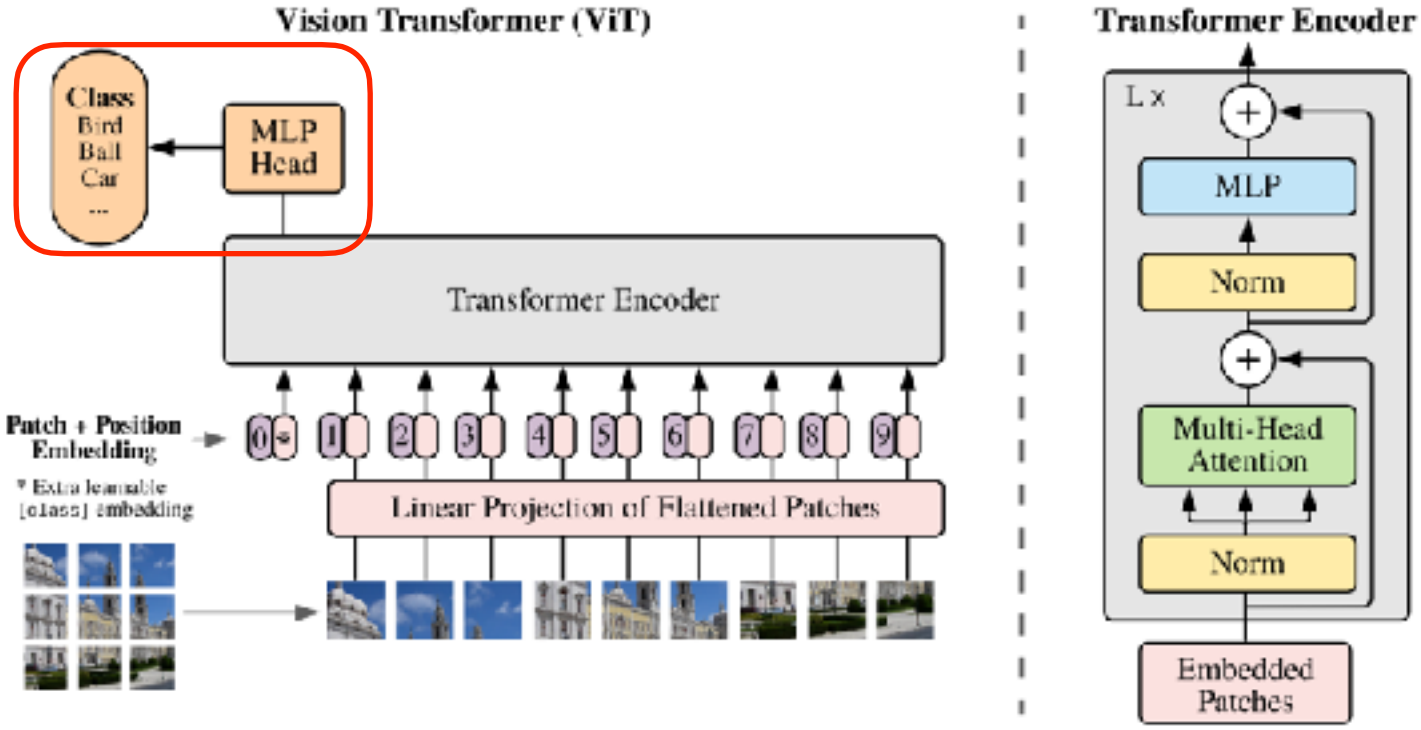


Image Classification

- The final image embedding is obtained from the final block for the classification token.
- On top of this representation, the authors append a 1-hidden-layer MLP, which is used to predict the final image categories.

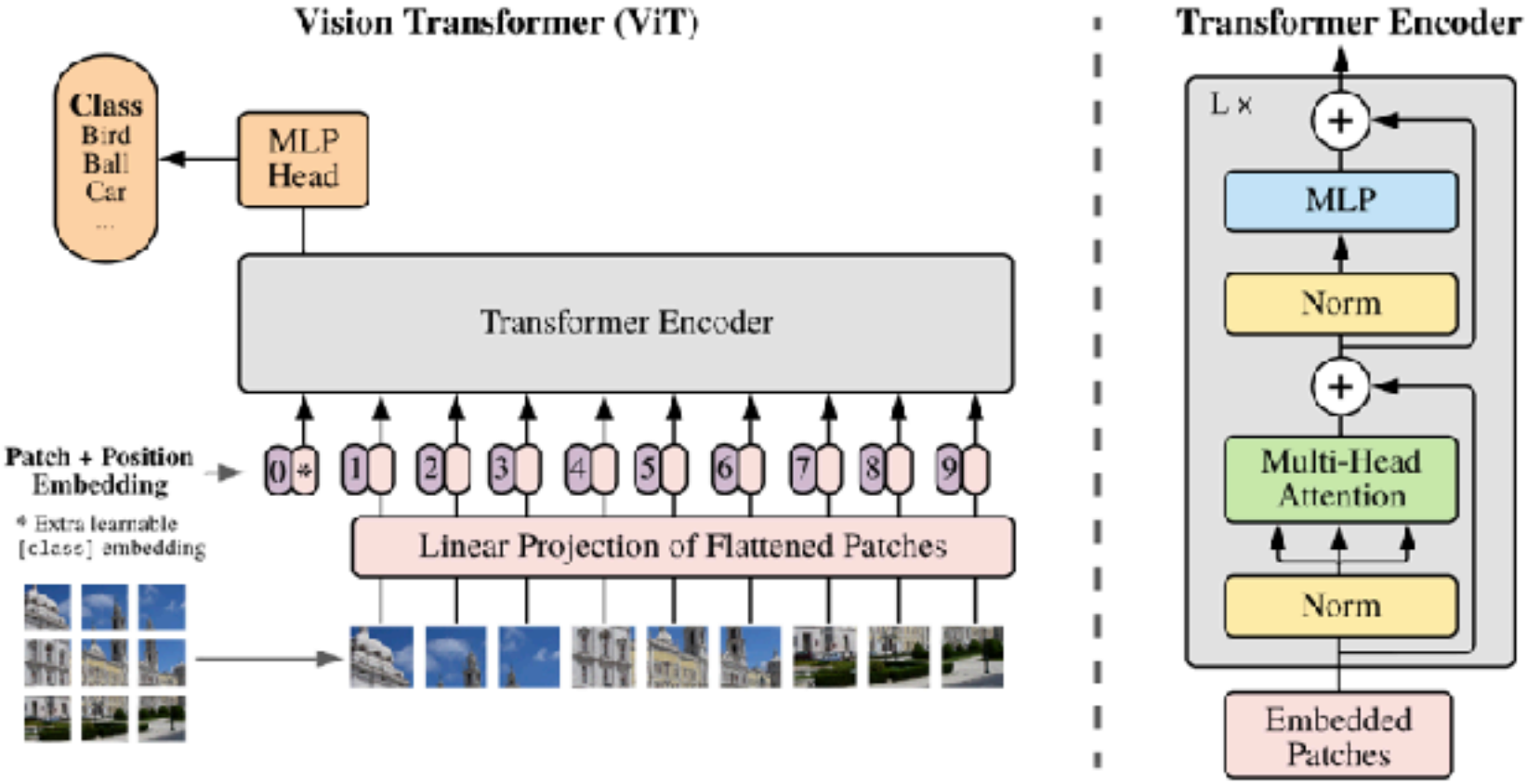
$$\mathbf{z}_0 = [\mathbf{x}_{\text{class}}; \mathbf{x}_p^1\mathbf{E}; \mathbf{x}_p^2\mathbf{E}; \cdots; \mathbf{x}_p^N\mathbf{E}] + \mathbf{E}_{pos},$$

A special learnable embedding, which will serve as the image representation used for classification.

a) a special CLS token appended to the input sequence

b) final classification stage

Vision Transformer (ViT)



ViT Architectures

- The ViT architecture variants are based on those used by BERT for language modeling.
- Notation: ViT-L/16 means the “Large” variant with 16×16 input patch size.

Model	Layers	Hidden size D	MLP size	Heads	Params
ViT-Base	12	768	3072	12	86M
ViT-Large	24	1024	4096	16	307M
ViT-Huge	32	1280	5120	16	632M

ViT Architectures

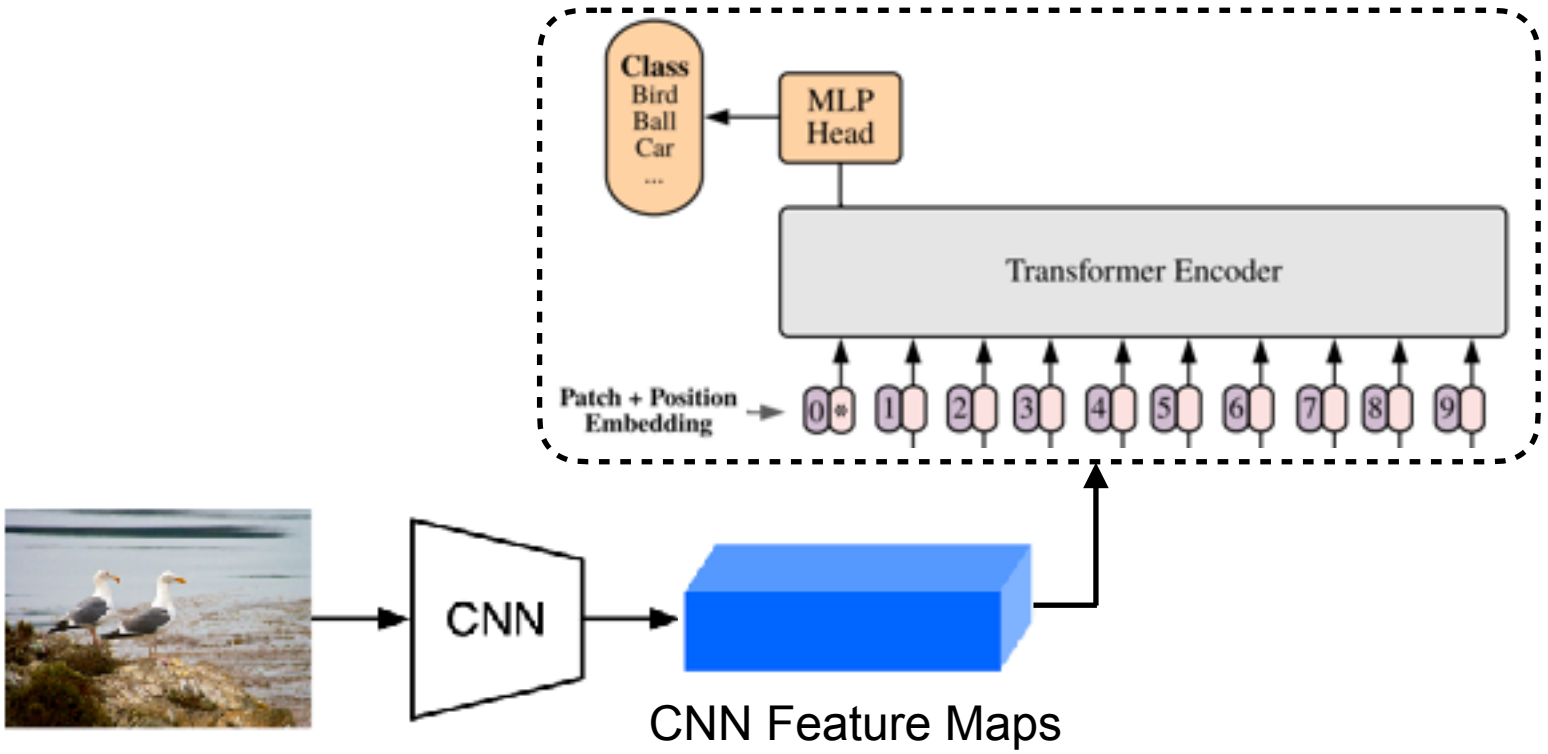
- The ViT architecture variants are based on those used by BERT for language modeling.
- Notation: ViT-L/16 means the "Large" variant with 16×16 input patch size.

Model	Layers	Hidden size D	MLP size	Heads	Params
ViT-Base	12	768	3072	12	86M
ViT-Large	24	1024	4096	16	307M
ViT-Huge	32	1280	5120	16	632M

Network	#param.
ResNet-18 [21]	12M
ResNet-50 [21]	25M
ResNet-101 [21]	45M
ResNet-152 [21]	60M

Hybrid CNN-ViT Architectures

- As an alternative to raw image patches, the input sequence can be formed from feature maps of a CNN.
- The patch embedding projection E is applied to patches extracted from a CNN feature map.



Experiments

- The authors evaluate the representation learned by (1) ResNet, (2) ViT, and (3) hybrid architectures.

Experiments

- The authors evaluate the representation learned by (1) ResNet, (2) ViT, and (3) hybrid architectures.
- Datasets used for pre-training:
    1. ImageNet: 1K classes and 1.3M images.
    2. ImageNet-21k: 21K classes and 14M images.
    3. JFT: 18K classes and 303M images.

Experiments

- The authors evaluate the representation learned by (1) ResNet, (2) ViT, and (3) hybrid architectures.
- Datasets used for pre-training:
    1. ImageNet: 1K classes and 1.3M images.
    2. ImageNet-21k: 21K classes and 14M images.
    3. JFT: 18K classes and 303M images.
- The performance is evaluated using a standard accuracy metric.

Transfer Learning Results

- A model is (1) pre-trained on a large-scale dataset, and then (2) fine-tuned on a smaller image classification dataset.
- The authors report mean and standard deviation of the accuracies, averaged over three fine-tuning runs.

	Ours-JFT (ViT-H/14)	Ours-JFT (ViT-L/16)	Ours-I21k (ViT-L/16)	BiT-L (ResNet152x4)	Noisy Student (EfficientNet-L2)
ImageNet	**88.55** ± 0.04	87.76 ± 0.03	85.30 ± 0.02	87.54 ± 0.02	88.4/88.5*
ImageNet ReaL	**90.72** ± 0.05	90.54 ± 0.03	88.62 ± 0.05	90.54	90.55
CIFAR-10	**99.50** ± 0.06	99.42 ± 0.03	99.15 ± 0.03	99.37 ± 0.06	–
CIFAR-100	**94.55** ± 0.04	93.90 ± 0.05	93.25 ± 0.05	93.51 ± 0.08	–
Oxford-IIIT Pets	**97.56** ± 0.03	97.32 ± 0.11	94.67 ± 0.15	96.62 ± 0.23	–
Oxford Flowers-102	99.68 ± 0.02	**99.74** ± 0.00	99.61 ± 0.02	99.63 ± 0.03	–
VTAB (19 tasks)	**77.63** ± 0.23	76.28 ± 0.46	72.72 ± 0.21	76.29 ± 1.70	–
TPUv3-core-days	2.5k	0.68k	0.23k	9.9k	12.3k

Transfer Learning Results

- A model is (1) pre-trained on a large-scale dataset, and then (2) fine-tuned on a smaller image classification dataset.
- The authors report mean and standard deviation of the accuracies, averaged over three fine-tuning runs.

	Ours-JFT (ViT-H/14)	Ours-JFT (ViT-L/16)	Ours-I21k (ViT-L/16)	BiT-L (ResNet152x4)	Noisy Student (EfficientNet-L2)
ImageNet	**88.55** ± 0.04	87.76 ± 0.03	85.30 ± 0.02	87.54 ± 0.02	88.4/88.5*
ImageNet ReaL	**90.72** ± 0.05	90.54 ± 0.03	88.62 ± 0.05	90.54	90.55
CIFAR-10	**99.50** ± 0.06	99.42 ± 0.03	99.15 ± 0.03	99.37 ± 0.06	–
CIFAR-100	**94.55** ± 0.04	93.90 ± 0.05	93.25 ± 0.05	93.51 ± 0.08	–
Oxford-IIIT Pets	**97.56** ± 0.03	97.32 ± 0.11	94.67 ± 0.15	96.62 ± 0.23	–
Oxford Flowers-102	99.68 ± 0.02	**99.74** ± 0.00	99.61 ± 0.02	99.63 ± 0.03	–
VTAB (19 tasks)	**77.63** ± 0.23	76.28 ± 0.46	72.72 ± 0.21	76.29 ± 1.70	–
TPUv3-core-days	2.5k	0.68k	0.23k	9.9k	12.3k

Downstream evaluation datasets/tasks

Transfer Learning Results

- A model is (1) pre-trained on a large-scale dataset, and then (2) fine-tuned on a smaller image classification dataset.
- The authors report mean and standard deviation of the accuracies, averaged over three fine-tuning runs.

	Ours-JFT (ViT-H/14)	Ours-JFT (ViT-L/16)	Ours-I21k (ViT-L/16)	BiT-L (ResNet152x4)	Noisy Student (EfficientNet-L2)
ImageNet	**88.55** ± 0.04	87.76 ± 0.03	85.30 ± 0.02	87.54 ± 0.02	88.4/88.5*
ImageNet ReaL	**90.72** ± 0.05	90.54 ± 0.03	88.62 ± 0.05	90.54	90.55
CIFAR-10	**99.50** ± 0.06	99.42 ± 0.03	99.15 ± 0.03	99.37 ± 0.06	–
CIFAR-100	**94.55** ± 0.04	93.90 ± 0.05	93.25 ± 0.05	93.51 ± 0.08	–
Oxford-IIIT Pets	**97.56** ± 0.03	97.32 ± 0.11	94.67 ± 0.15	96.62 ± 0.23	–
Oxford Flowers-102	99.68 ± 0.02	**99.74** ± 0.00	99.61 ± 0.02	99.63 ± 0.03	–
VTAB (19 tasks)	**77.63** ± 0.23	76.28 ± 0.46	72.72 ± 0.21	76.29 ± 1.70	–
TPUv3-core-days	2.5k	0.68k	0.23k	9.9k	12.3k

Pretraining cost in terms of TPU days

Transfer Learning Results

- A model is (1) pre-trained on a large-scale dataset, and then (2) fine-tuned on a smaller image classification dataset.
- The authors report mean and standard deviation of the accuracies, averaged over three fine-tuning runs.

	Ours-JFT (ViT-H/14)	Ours-JFT (ViT-L/16)	Ours-I21k (ViT-L/16)	BiT-L (ResNet152x4)	Noisy Student (EfficientNet-L2)
ImageNet	**88.55** ± 0.04	87.76 ± 0.03	85.30 ± 0.02	87.54 ± 0.02	88.4/88.5*
ImageNet ReaL	**90.72** ± 0.05	90.54 ± 0.03	88.62 ± 0.05	90.54	90.55
CIFAR-10	**99.50** ± 0.06	99.42 ± 0.03	99.15 ± 0.03	99.37 ± 0.06	–
CIFAR-100	**94.55** ± 0.04	93.90 ± 0.05	93.25 ± 0.05	93.51 ± 0.08	–
Oxford-IIIT Pets	**97.56** ± 0.03	97.32 ± 0.11	94.67 ± 0.15	96.62 ± 0.23	–
Oxford Flowers-102	99.68 ± 0.02	**99.74** ± 0.00	99.61 ± 0.02	99.63 ± 0.03	–
VTAB (19 tasks)	**77.63** ± 0.23	76.28 ± 0.46	72.72 ± 0.21	76.29 ± 1.70	–
TPUv3-core-days	2.5k	0.68k	0.23k	9.9k	12.3k

Two ViT variants pre-trained on the JFT dataset.

Transfer Learning Results

- A model is (1) pre-trained on a large-scale dataset, and then (2) fine-tuned on a smaller image classification dataset.
- The authors report mean and standard deviation of the accuracies, averaged over three fine-tuning runs.

	Ours-JFT (ViT-H/14)	Ours-JFT (ViT-L/16)	Ours-I21k (ViT-L/16)	BiT-L (ResNet152x4)	Noisy Student (EfficientNet-L2)
ImageNet	**88.55** ± 0.04	87.76 ± 0.03	85.30 ± 0.02	87.54 ± 0.02	88.4/88.5*
ImageNet ReaL	**90.72** ± 0.05	90.54 ± 0.03	88.62 ± 0.05	90.54	90.55
CIFAR-10	**99.50** ± 0.06	99.42 ± 0.03	99.15 ± 0.03	99.37 ± 0.06	–
CIFAR-100	**94.55** ± 0.04	93.90 ± 0.05	93.25 ± 0.05	93.51 ± 0.08	–
Oxford-IIIT Pets	**97.56** ± 0.03	97.32 ± 0.11	94.67 ± 0.15	96.62 ± 0.23	–
Oxford Flowers-102	99.68 ± 0.02	**99.74** ± 0.00	99.61 ± 0.02	99.63 ± 0.03	–
VTAB (19 tasks)	**77.63** ± 0.23	76.28 ± 0.46	72.72 ± 0.21	76.29 ± 1.70	–
TPUv3-core-days	2.5k	0.68k	0.23k	9.9k	12.3k

ViT pre-trained on Imagenet-21K.

Transfer Learning Results

- A model is (1) pre-trained on a large-scale dataset, and then (2) fine-tuned on a smaller image classification dataset.
- The authors report mean and standard deviation of the accuracies, averaged over three fine-tuning runs.

	Ours-JFT (ViT-H/14)	Ours-JFT (ViT-L/16)	Ours-I21k (ViT-L/16)	BiT-L (ResNet152x4)	Noisy Student (EfficientNet-L2)
ImageNet	**88.55** ± 0.04	87.76 ± 0.03	85.30 ± 0.02	87.54 ± 0.02	88.4/88.5*
ImageNet ReaL	**90.72** ± 0.05	90.54 ± 0.03	88.62 ± 0.05	90.54	90.55
CIFAR-10	**99.50** ± 0.06	99.42 ± 0.03	99.15 ± 0.03	99.37 ± 0.06	–
CIFAR-100	**94.55** ± 0.04	93.90 ± 0.05	93.25 ± 0.05	93.51 ± 0.08	–
Oxford-IIIT Pets	**97.56** ± 0.03	97.32 ± 0.11	94.67 ± 0.15	96.62 ± 0.23	–
Oxford Flowers-102	99.68 ± 0.02	**99.74** ± 0.00	99.61 ± 0.02	99.63 ± 0.03	–
VTAB (19 tasks)	**77.63** ± 0.23	76.28 ± 0.46	72.72 ± 0.21	76.29 ± 1.70	–
TPUv3-core-days	2.5k	0.68k	0.23k	9.9k	12.3k

ResNet-based state-of-the-art approach pretrained on JFT.

Transfer Learning Results

- A model is (1) pre-trained on a large-scale dataset, and then (2) fine-tuned on a smaller image classification dataset.
- The authors report mean and standard deviation of the accuracies, averaged over three fine-tuning runs.

	Ours-JFT (ViT-H/14)	Ours-JFT (ViT-L/16)	Ours-I21k (ViT-L/16)	BiT-L (ResNet152x4)	Noisy Student (EfficientNet-L2)
ImageNet	**88.55** ± 0.04	87.76 ± 0.03	85.30 ± 0.02	87.54 ± 0.02	88.4/88.5*
ImageNet ReaL	**90.72** ± 0.05	90.54 ± 0.03	88.62 ± 0.05	90.54	90.55
CIFAR-10	**99.50** ± 0.06	99.42 ± 0.03	99.15 ± 0.03	99.37 ± 0.06	–
CIFAR-100	**94.55** ± 0.04	93.90 ± 0.05	93.25 ± 0.05	93.51 ± 0.08	–
Oxford-IIIT Pets	**97.56** ± 0.03	97.32 ± 0.11	94.67 ± 0.15	96.62 ± 0.23	–
Oxford Flowers-102	99.68 ± 0.02	**99.74** ± 0.00	99.61 ± 0.02	99.63 ± 0.03	–
VTAB (19 tasks)	**77.63** ± 0.23	76.28 ± 0.46	72.72 ± 0.21	76.29 ± 1.70	–
TPUv3-core-days	2.5k	0.68k	0.23k	9.9k	12.3k

EfficientNet-based state-of-the-art semi-supervised approach pretrained on JFT (without using JFT labels).

Transfer Learning Results

- A model is (1) pre-trained on a large-scale dataset, and then (2) fine-tuned on a smaller image classification dataset.
- The authors report mean and standard deviation of the accuracies, averaged over three fine-tuning runs.

	Ours-JFT (ViT-H/14)	Ours-JFT (ViT-L/16)	Ours-I21k (ViT-L/16)	BiT-L (ResNet152x4)	Noisy Student (EfficientNet-L2)
ImageNet	**88.55** ± 0.04	87.76 ± 0.03	85.30 ± 0.02	87.54 ± 0.02	88.4/88.5*
ImageNet ReaL	**90.72** ± 0.05	90.54 ± 0.03	88.62 ± 0.05	90.54	90.55
CIFAR-10	**99.50** ± 0.06	99.42 ± 0.03	99.15 ± 0.03	99.37 ± 0.06	–
CIFAR-100	**94.55** ± 0.04	93.90 ± 0.05	93.25 ± 0.05	93.51 ± 0.08	–
Oxford-IIIT Pets	**97.56** ± 0.03	97.32 ± 0.11	94.67 ± 0.15	96.62 ± 0.23	–
Oxford Flowers-102	99.68 ± 0.02	**99.74** ± 0.00	99.61 ± 0.02	99.63 ± 0.03	–
VTAB (19 tasks)	**77.63** ± 0.23	76.28 ± 0.46	72.72 ± 0.21	76.29 ± 1.70	–
TPUv3-core-days	2.5k	0.68k	0.23k	9.9k	12.3k

Transfer Learning Results

- A model is (1) pre-trained on a large-scale dataset, and then (2) fine-tuned on a smaller image classification dataset.
- The authors report mean and standard deviation of the accuracies, averaged over three fine-tuning runs.

	Ours-JFT (ViT-H/14)	Ours-JFT (ViT-L/16)	Ours-I21k (ViT-L/16)	BiT-L (ResNet152x4)	Noisy Student (EfficientNet-L2)
ImageNet	**88.55** ± 0.04	87.76 ± 0.03	85.30 ± 0.02	87.54 ± 0.02	88.4/88.5*
ImageNet ReaL	**90.72** ± 0.05	90.54 ± 0.03	88.62 ± 0.05	90.54	90.55
CIFAR-10	**99.50** ± 0.06	99.42 ± 0.03	99.15 ± 0.03	99.37 ± 0.06	–
CIFAR-100	**94.55** ± 0.04	93.90 ± 0.05	93.25 ± 0.05	93.51 ± 0.08	–
Oxford-IIIT Pets	**97.56** ± 0.03	97.32 ± 0.11	94.67 ± 0.15	96.62 ± 0.23	–
Oxford Flowers-102	99.68 ± 0.02	**99.74** ± 0.00	99.61 ± 0.02	99.63 ± 0.03	–
VTAB (19 tasks)	**77.63** ± 0.23	76.28 ± 0.46	72.72 ± 0.21	76.29 ± 1.70	–
TPUv3-core-days	2.5k	0.68k	0.23k	9.9k	12.3k

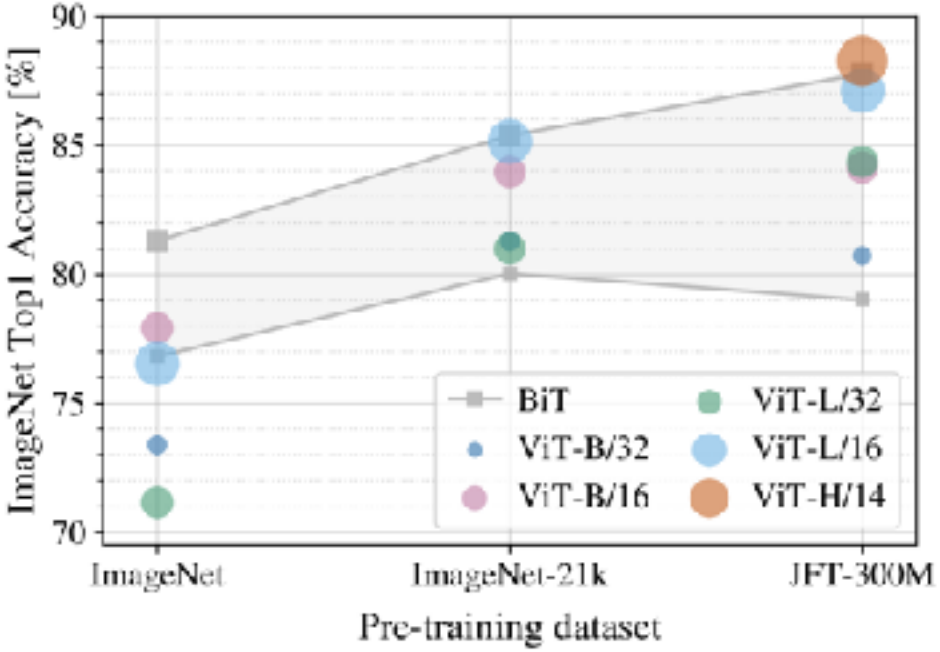
Transfer Learning Results

- A model is (1) pre-trained on a large-scale dataset, and then (2) fine-tuned on a smaller image classification dataset.
- The authors report mean and standard deviation of the accuracies, averaged over three fine-tuning runs.

	Ours-JFT (ViT-H/14)	Ours-JFT (ViT-L/16)	Ours-I21k (ViT-L/16)	BiT-L (ResNet152x4)	Noisy Student (EfficientNet-L2)
ImageNet	**88.55** ± 0.04	87.76 ± 0.03	85.30 ± 0.02	87.54 ± 0.02	88.4/88.5*
ImageNet ReaL	**90.72** ± 0.05	90.54 ± 0.03	88.62 ± 0.05	90.54	90.55
CIFAR-10	**99.50** ± 0.06	99.42 ± 0.03	99.15 ± 0.03	99.37 ± 0.06	–
CIFAR-100	**94.55** ± 0.04	93.90 ± 0.05	93.25 ± 0.05	93.51 ± 0.08	–
Oxford-IIIT Pets	**97.56** ± 0.03	97.32 ± 0.11	94.67 ± 0.15	96.62 ± 0.23	–
Oxford Flowers-102	99.68 ± 0.02	**99.74** ± 0.00	99.61 ± 0.02	99.63 ± 0.03	–
VTAB (19 tasks)	**77.63** ± 0.23	76.28 ± 0.46	72.72 ± 0.21	76.29 ± 1.70	–
TPUv3-core-days	2.5k	0.68k	0.23k	9.9k	12.3k

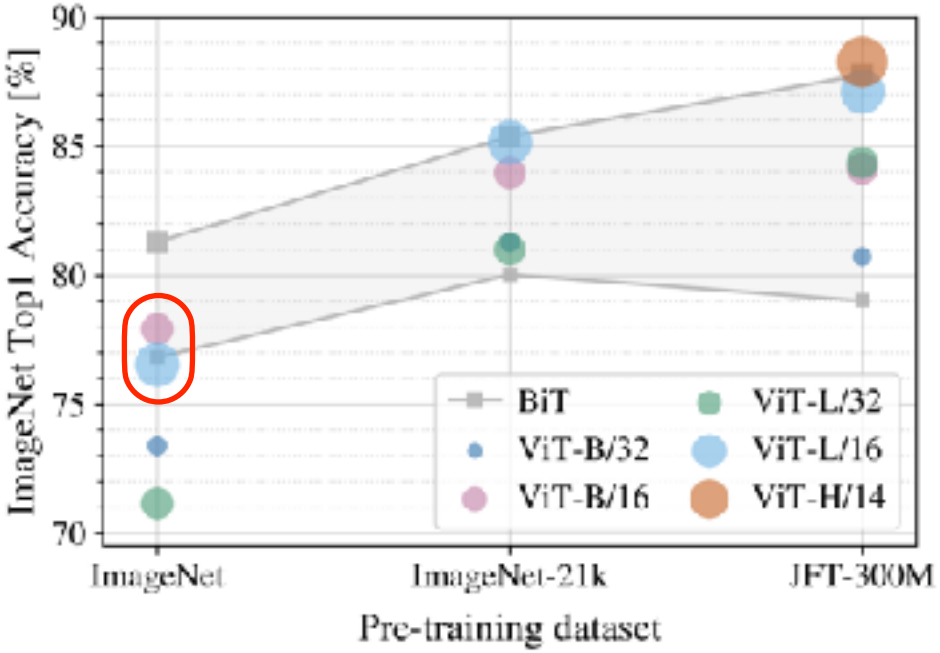
Pre-training Data Requirements

- The ViT models are pre-trained on datasets of increasing size: ImageNet, ImageNet-21k, and JFT300M.
- ImageNet accuracy is reported after finetuning on ImageNet.



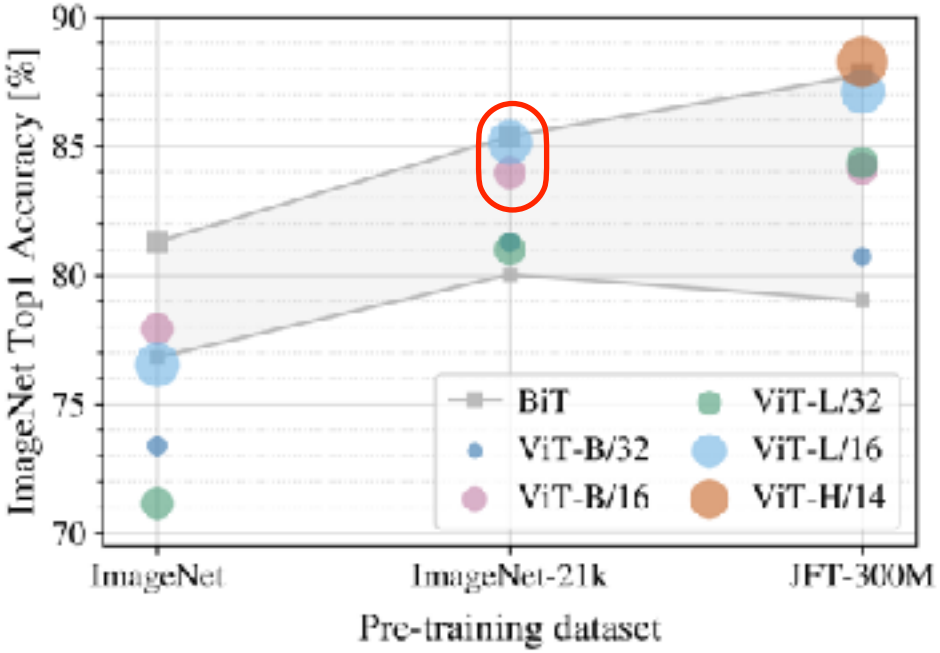
Pre-training Data Requirements

- The ViT models are pre-trained on datasets of increasing size: ImageNet, ImageNet-21k, and JFT300M.
- ImageNet accuracy is reported after finetuning on ImageNet.



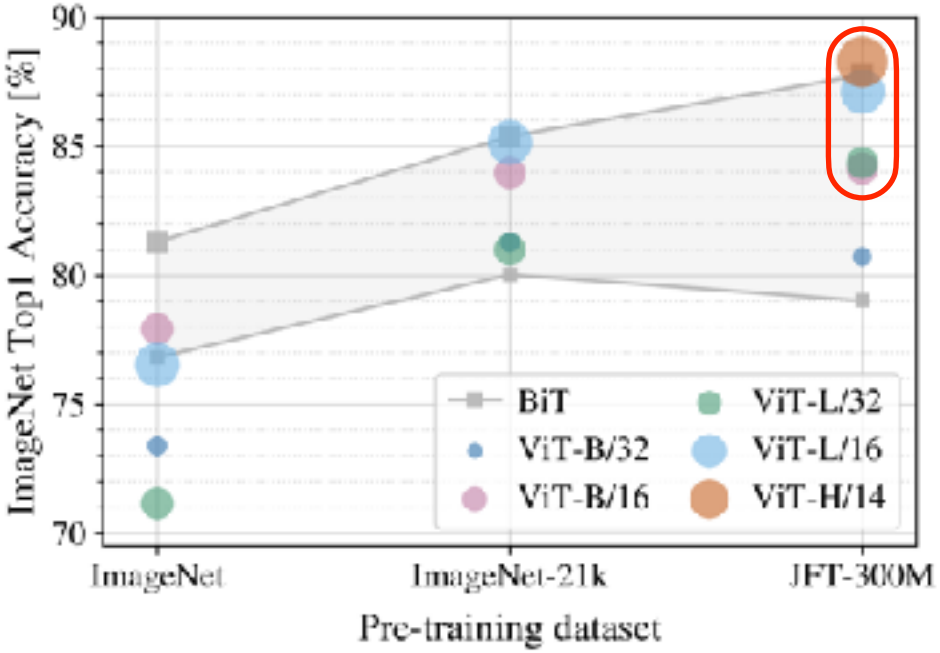
Pre-training Data Requirements

- The ViT models are pre-trained on datasets of increasing size: ImageNet, ImageNet-21k, and JFT300M.
- ImageNet accuracy is reported after finetuning on ImageNet.



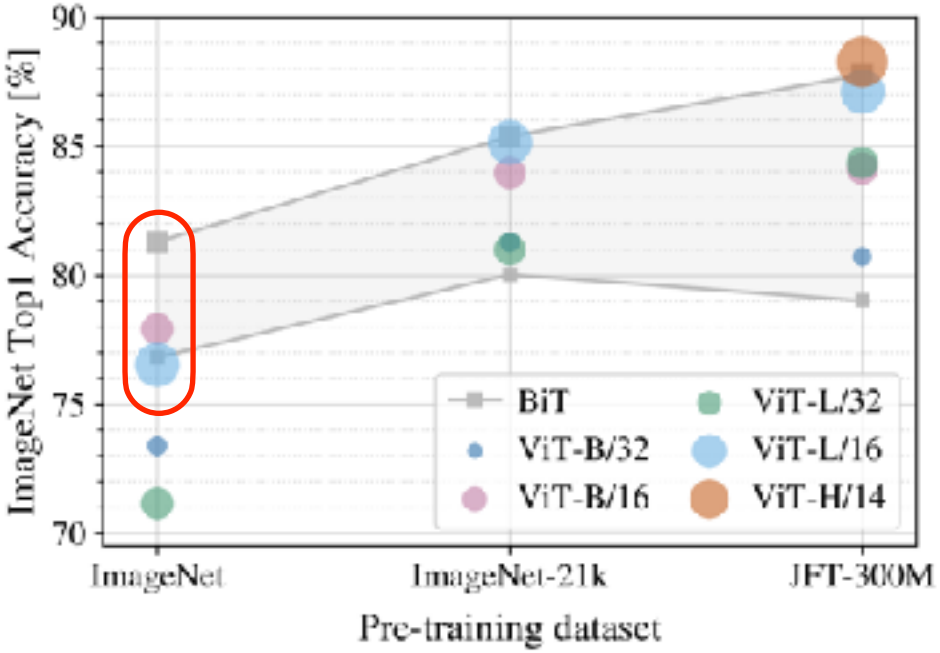
Pre-training Data Requirements

- The ViT models are pre-trained on datasets of increasing size: ImageNet, ImageNet-21k, and JFT300M.
- ImageNet accuracy is reported after finetuning on ImageNet.



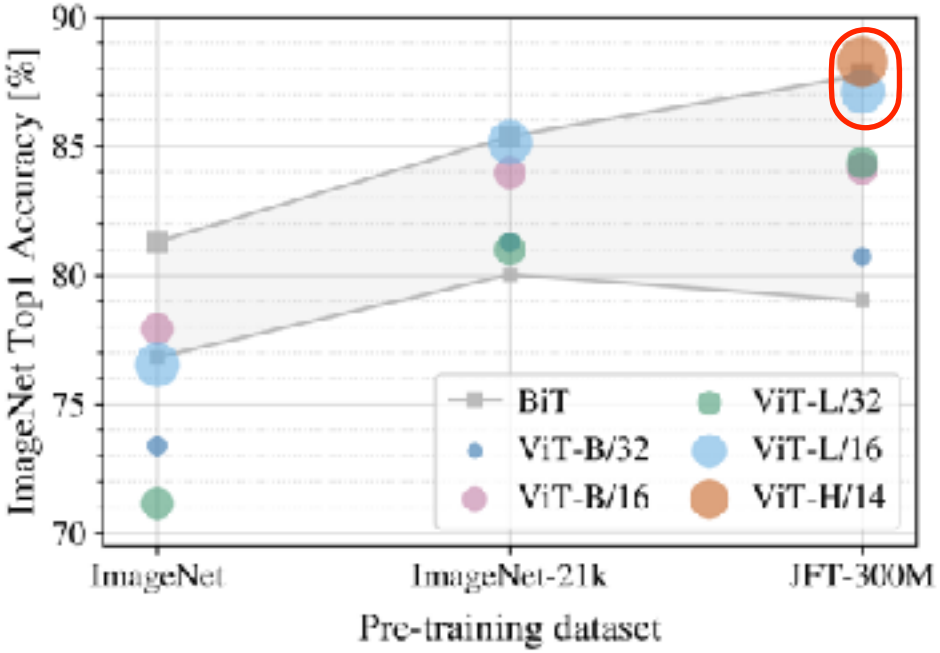
Pre-training Data Requirements

- The ViT models are pre-trained on datasets of increasing size: ImageNet, ImageNet-21k, and JFT300M.
- ImageNet accuracy is reported after finetuning on ImageNet.



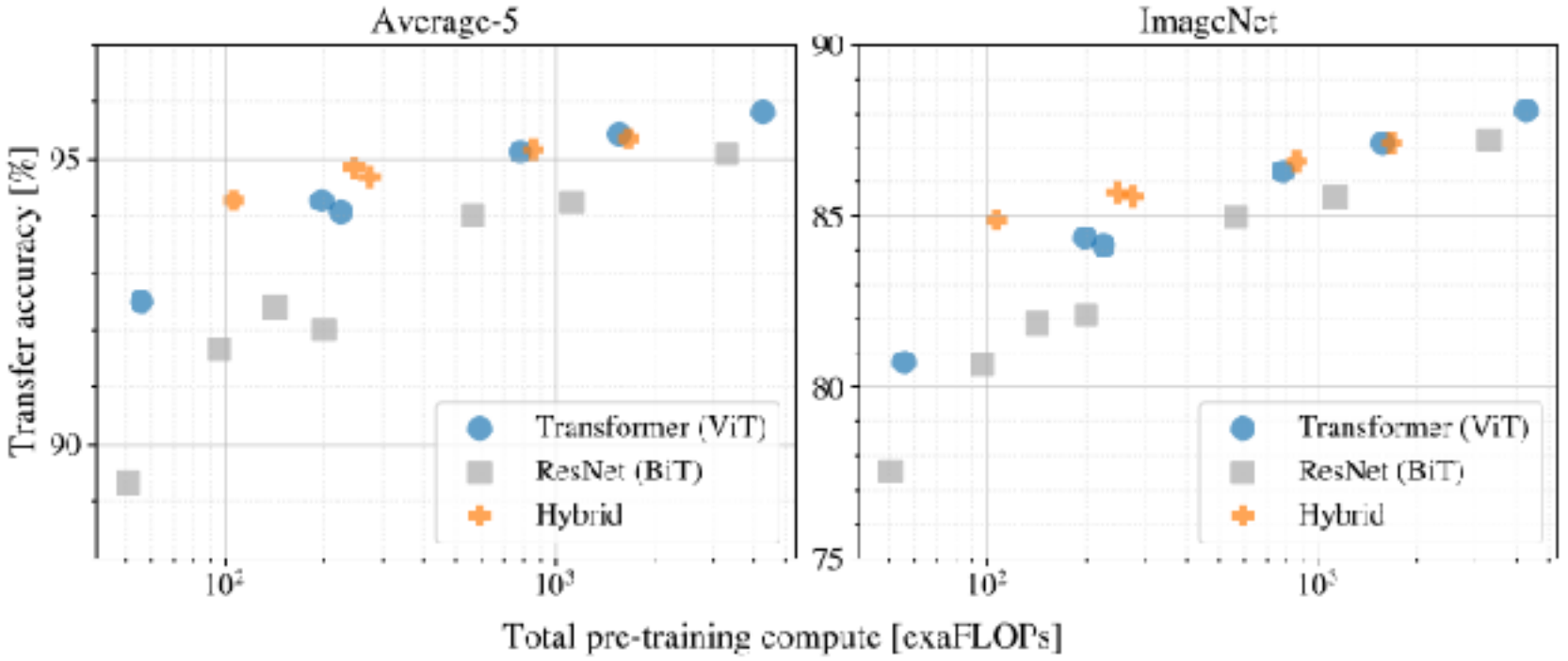
Pre-training Data Requirements

- The ViT models are pre-trained on datasets of increasing size: ImageNet, ImageNet-21k, and JFT300M.
- ImageNet accuracy is reported after finetuning on ImageNet.



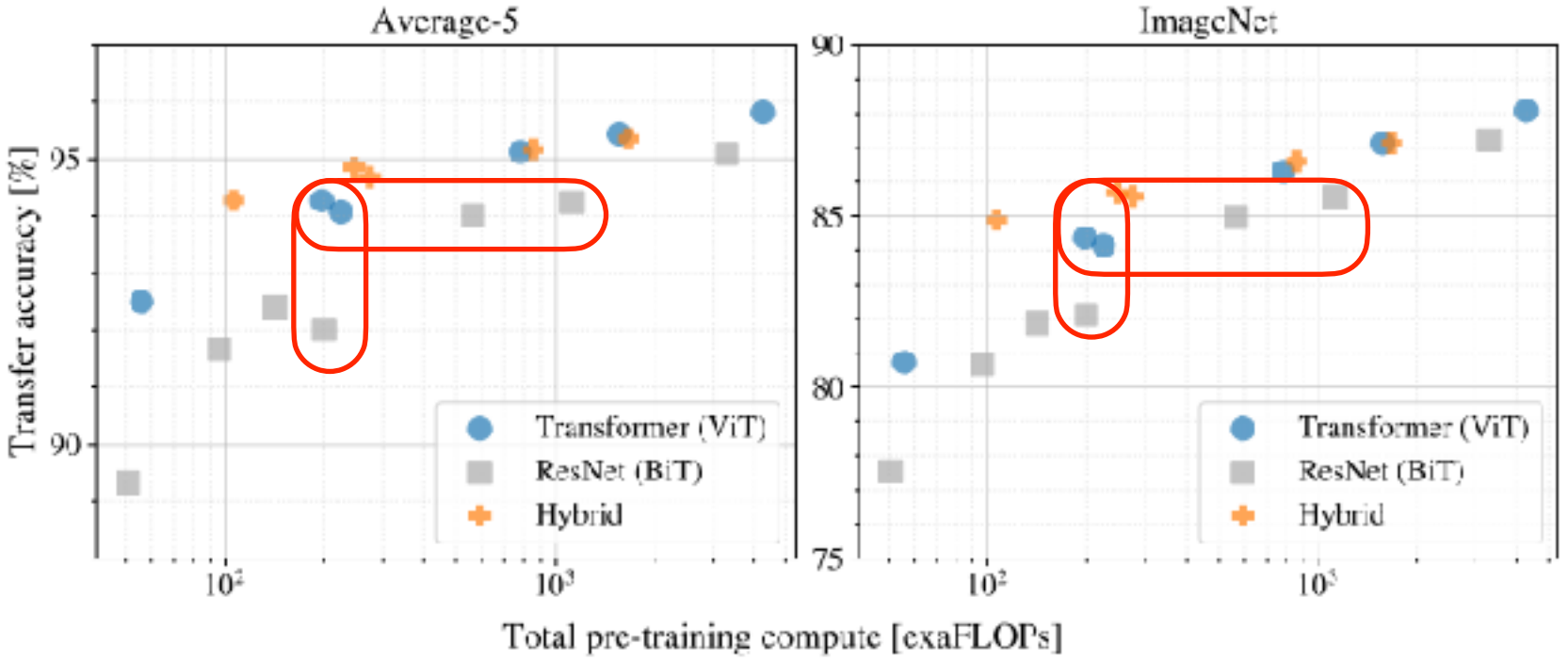
Scaling Study

- Performance versus pre-training compute for different architectures: Vision Transformers, ResNets, and hybrids.



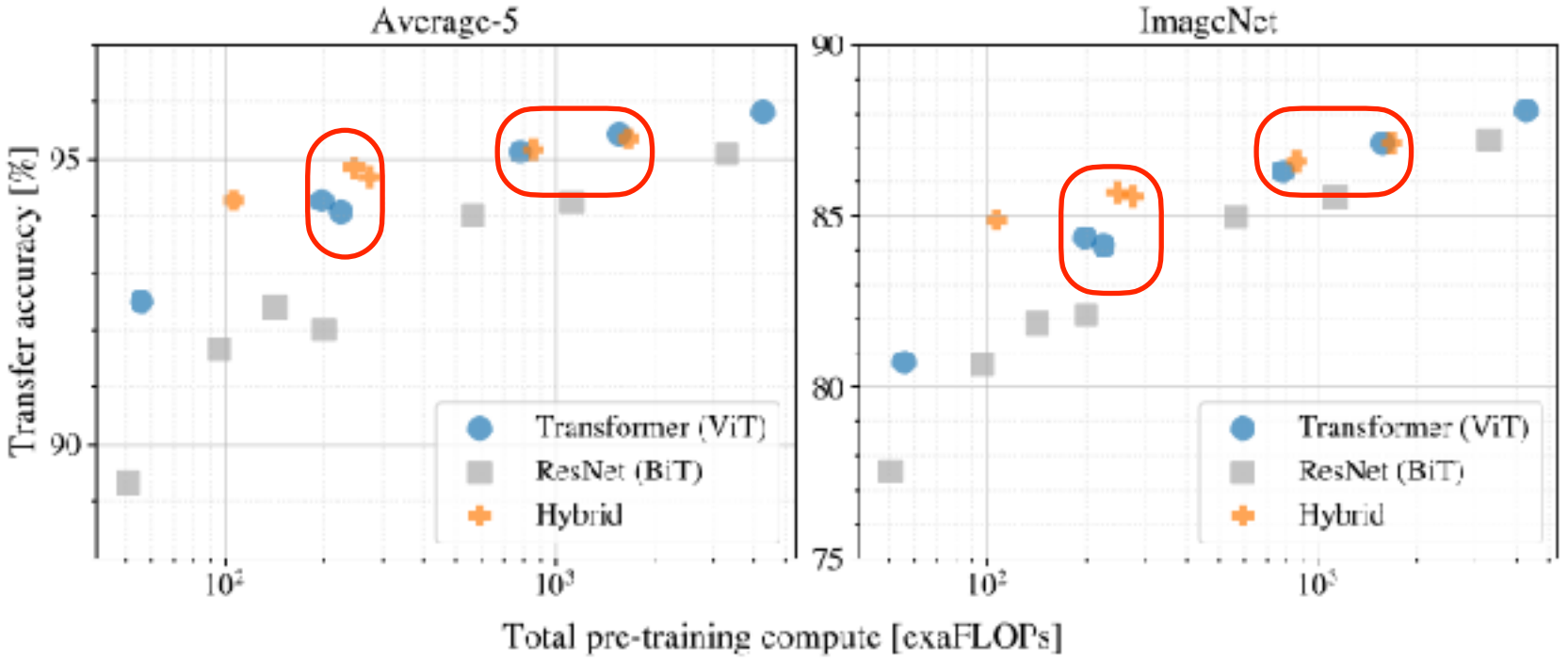
Scaling Study

- Performance versus pre-training compute for different architectures: Vision Transformers, ResNets, and hybrids.



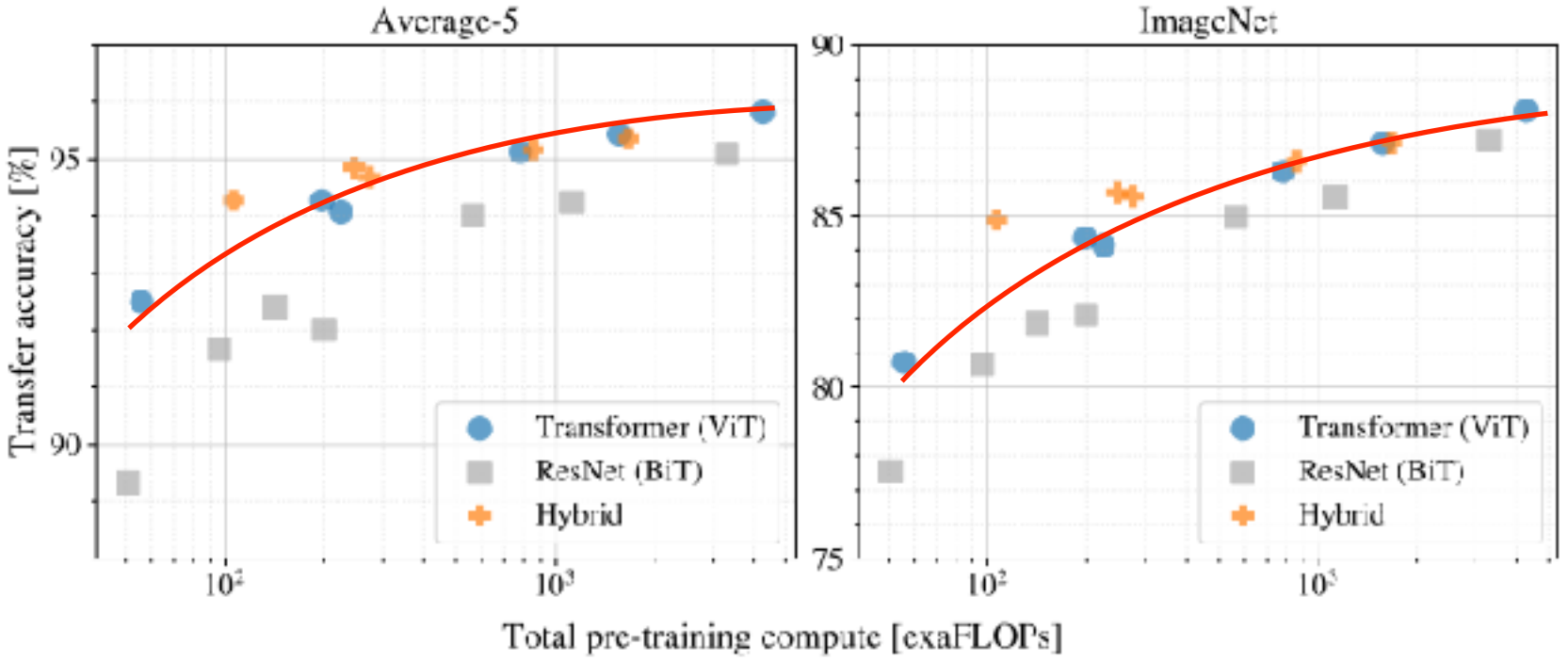
Scaling Study

- Performance versus pre-training compute for different architectures: Vision Transformers, ResNets, and hybrids.



Scaling Study

- Performance versus pre-training compute for different architectures: Vision Transformers, ResNets, and hybrids.



Visualized Attention

- The authors visualize the attention from the output token to the input space.

Summary

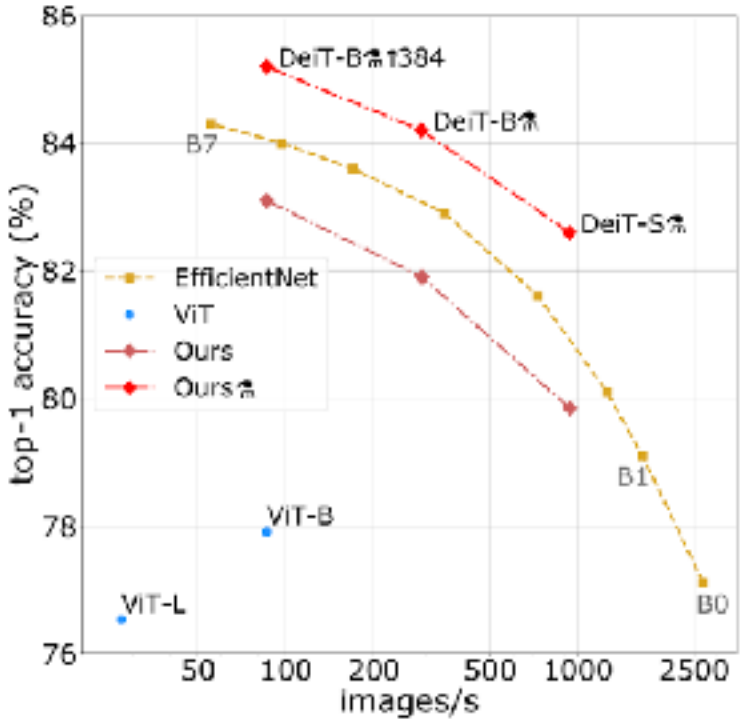
- The availability of large-scale datasets makes ViTs appealing for modern big-data regimes where *all* can be learned from the data.

- In such cases, strong inductive biases of older models (e.g., ResNets) become unnecessary and may be harmful as they limit the model's expressivity.

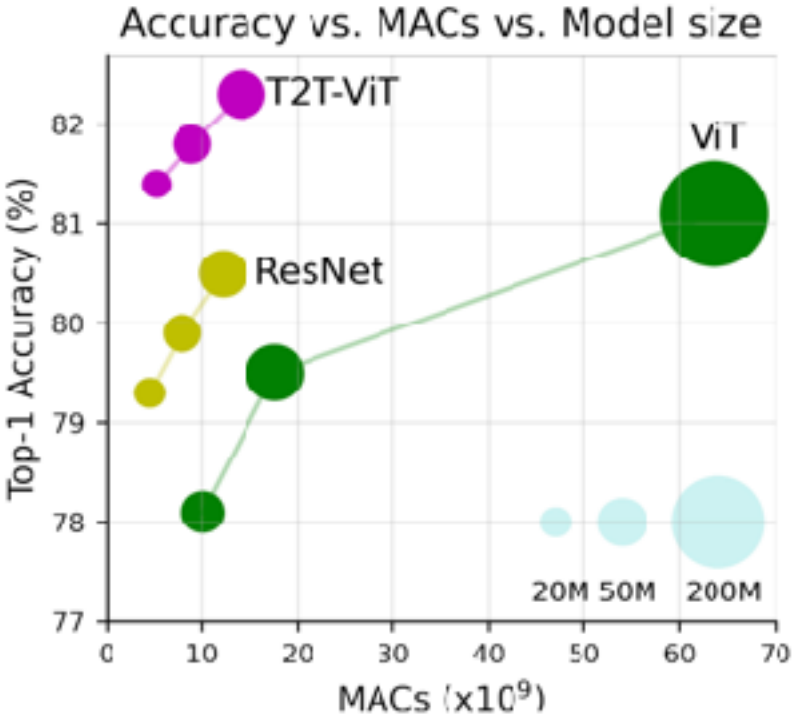
- Despite limited technical novelty, the proposed method's simplicity, effectiveness, and generality makes it a valuable contribution to the research community.

Weaknesses to Address Next Week

- Data efficiency, i.e., training ViTs on ImageNet alone.



Touvron et al., "Training data-efficient image transformers & distillation through attention", ICML 2021



Yuan et al., "Tokens-to-Token ViT: Training Vision Transformers from Scratch on ImageNet", ICCV 2021

Paper Selection Assignment

- The reading list is posted here.

- Select the following:
  1. Seven 30min or 45min papers for standard paper presentations (marked **red** and **purple** in the schedule). Any combo of the papers suffice (e.g., five 30min & two 45min papers, all 30min papers, etc.)
  2. Three 20min papers for paper battles (marked **green** in the schedule).

- Make sure that the papers that you selected will **NOT** be presented by me.

- Rank the papers in each of these lists in descending order of preference (from highest to lowest) and upload them to Canvas **by Wednesday, Jan 17th, 11:59 PM** (please include paper IDs in your lists!!).

- I will then update the website with the paper assignments.