# An Image is Worth 16x16 Words

## Transformers for Image Recognition at Scale

Dosovitskiy et, al. 2021, ICLR.

Presented by:
Joshua Owoyemi, (PhD)
@ MLT __init__ 2021.07.18

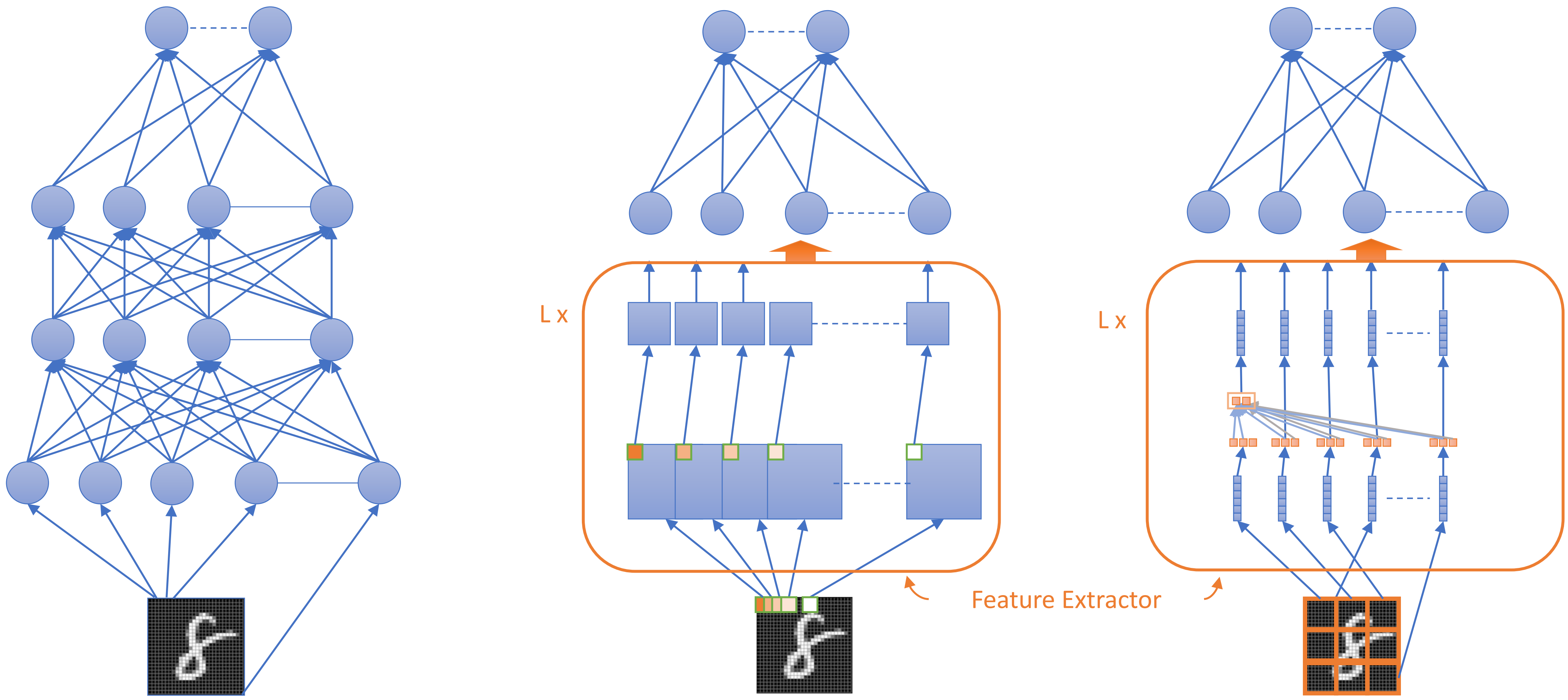
# Outline

- Introduction
- From MLP to Transformers
- Recap on Self-Attention
- Transformers for image recognition
- Experiments and Results

# Introduction

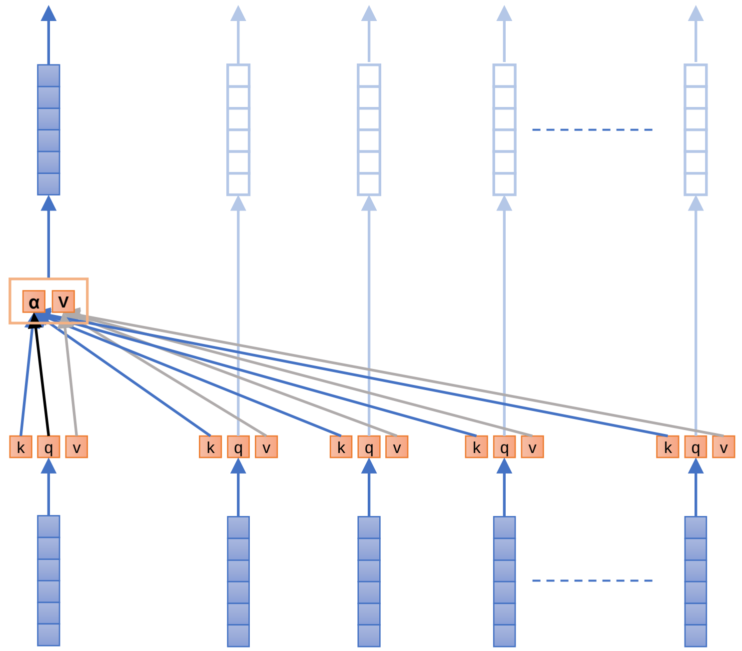
- Claims of the paper
  - CNNs are not necessary
  - Attention-based network can result in substantially fewer computation
- Why this paper is important
  - Challenges the inductive bias of CNNs
  - Opens up alternate perspectives to solving vision-related tasks
- The main idea of the paper
  - Apply Transformers (NLP-centered network) to vision related task

# MLPs to Transformers



# Recap on Self-Attention

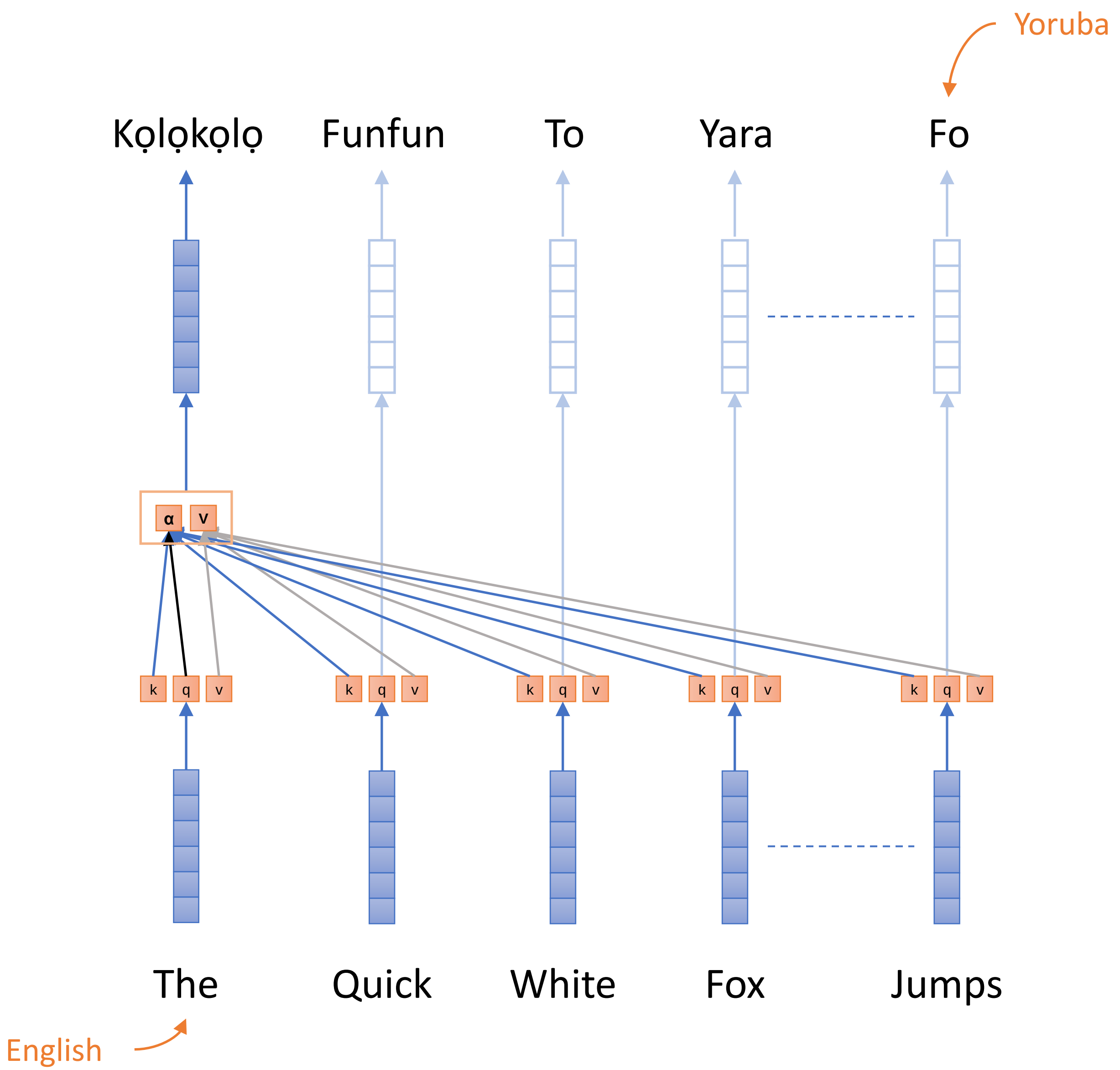
- Attention mechanism helps to build and quantify interdependence*
- Self-attention builds interdependence within input elements
- Has become de-facto standard in language processing tasks




*Attention Mechanism, Gabriel Loye

# Recap on Self-Attention

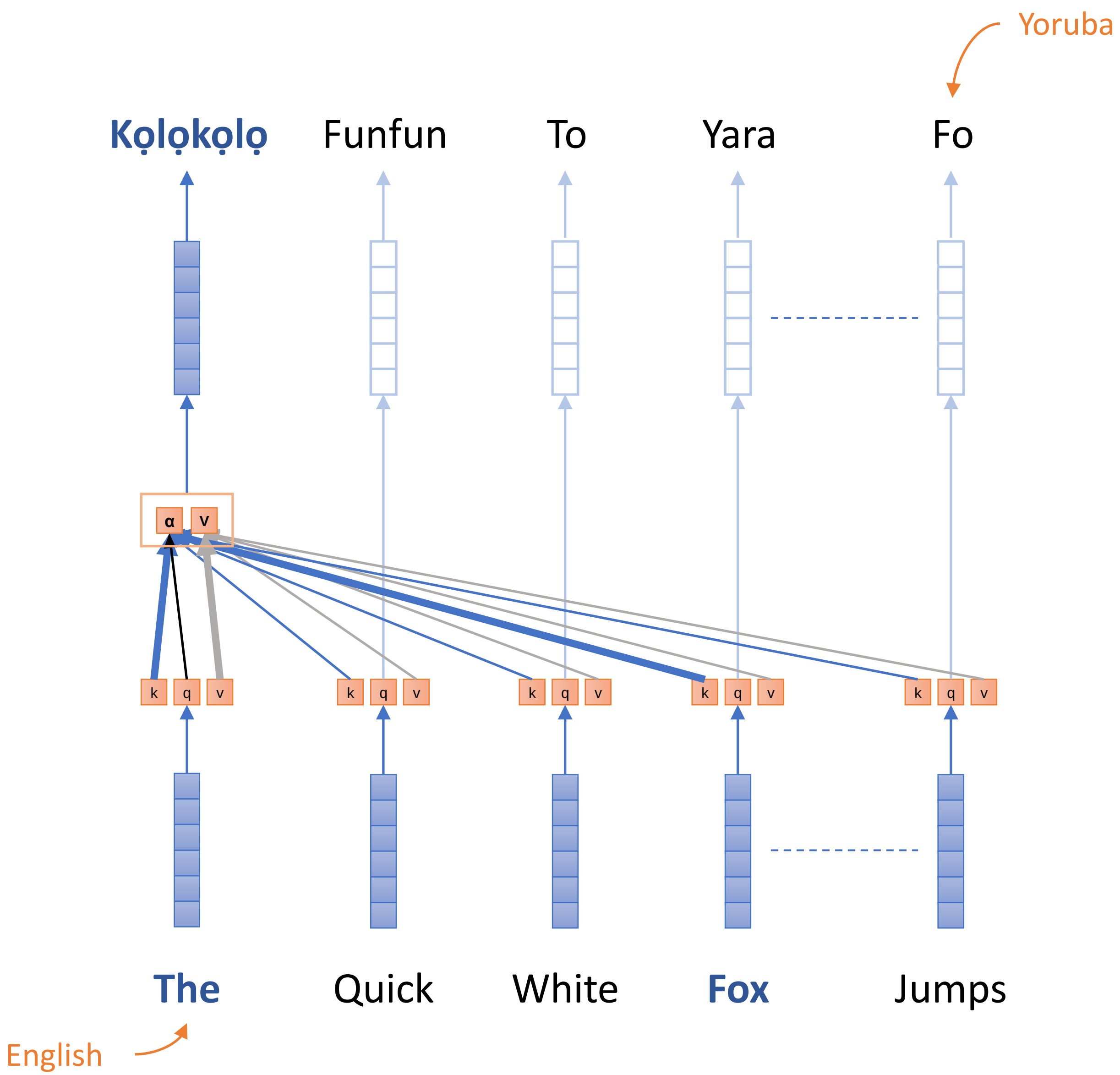
- Attention mechanism helps to build and quantify interdependence*
- Self-attention builds interdependence within input elements
- Has become de-facto standard in language processing tasks




*[Attention Mechanism, Gabriel Loye](#)

# Recap on Self-Attention

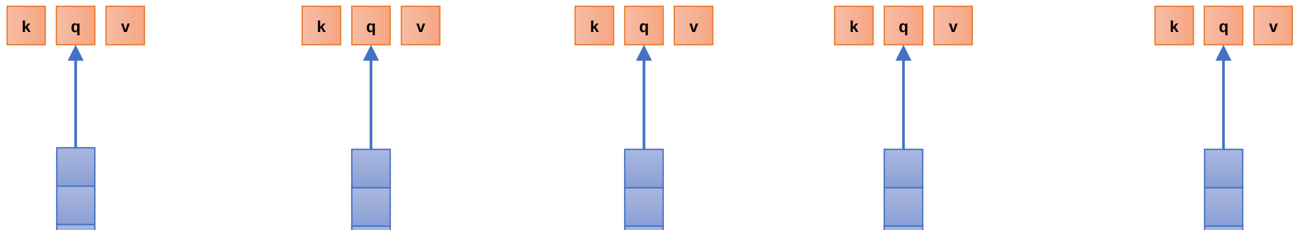
- Attention mechanism helps to build and quantify interdependence*
- Self-attention builds interdependence within input elements
- Has become de-facto standard in language processing tasks




*[Attention Mechanism, Gabriel Loye]

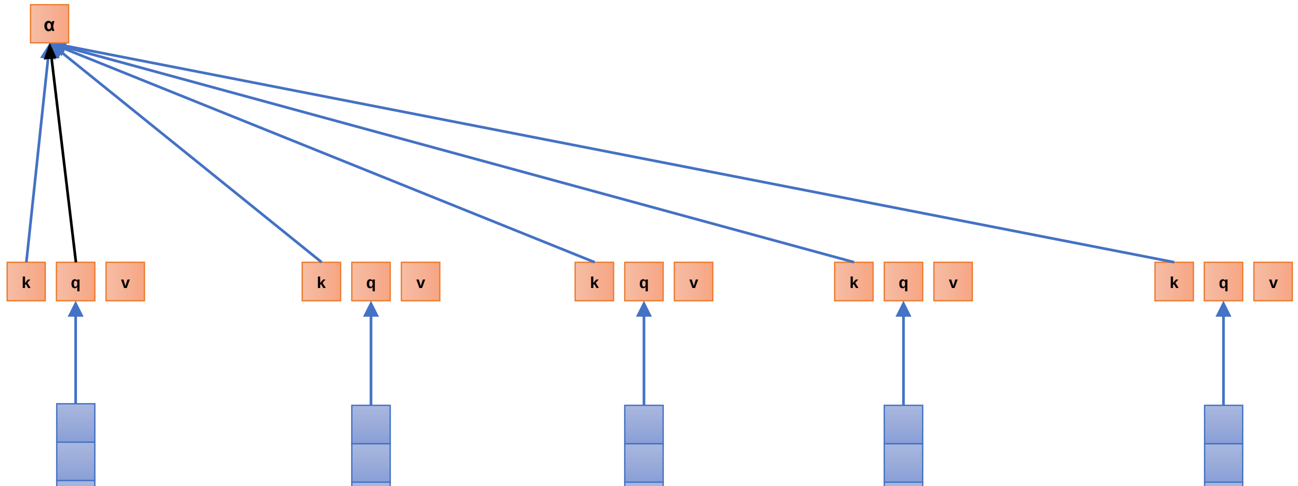
# Recap on Self-Attention

- Map input vectors to
  - $Query$: $q_i = W_Q X_i$,
  - $Key$: $k_i = W_K X_i$,
  - $Value$: $v_i = W_V X_i$,



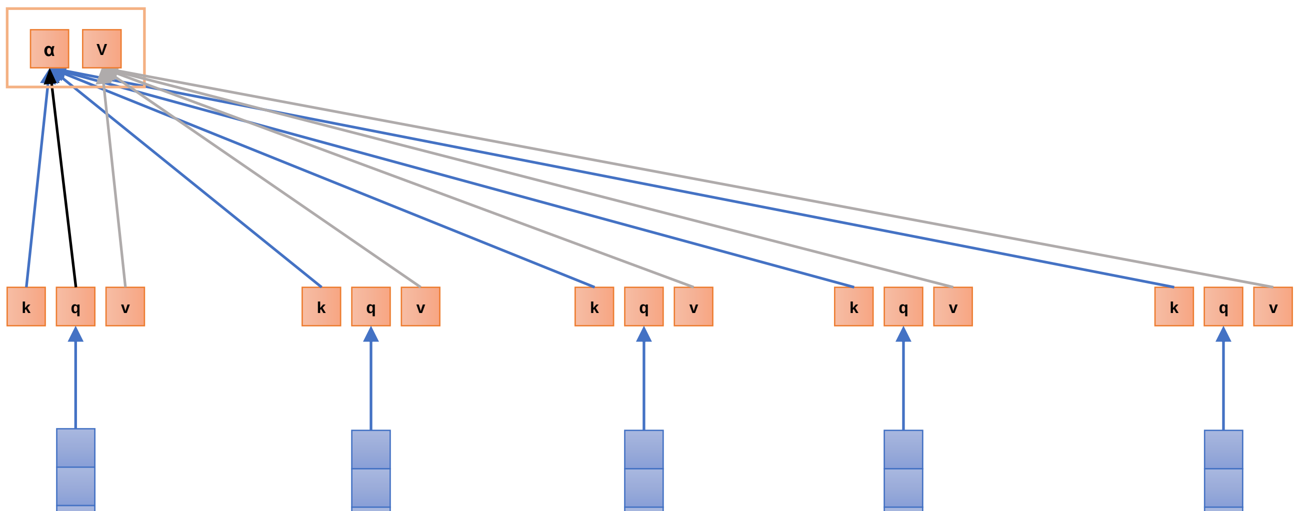
# Recap on Self-Attention

- Map input vectors to
  - $Query\colon\ q_i = W_Q X_i,$
  - $Key\colon\ k_i = W_K X_i,$
  - $Value\colon\ v_i = W_V X_i,$
- Compute weights vector: $A = softmax(\boldsymbol{q}\boldsymbol{k}^T) \epsilon \mathbb{R}^{N\times N}$



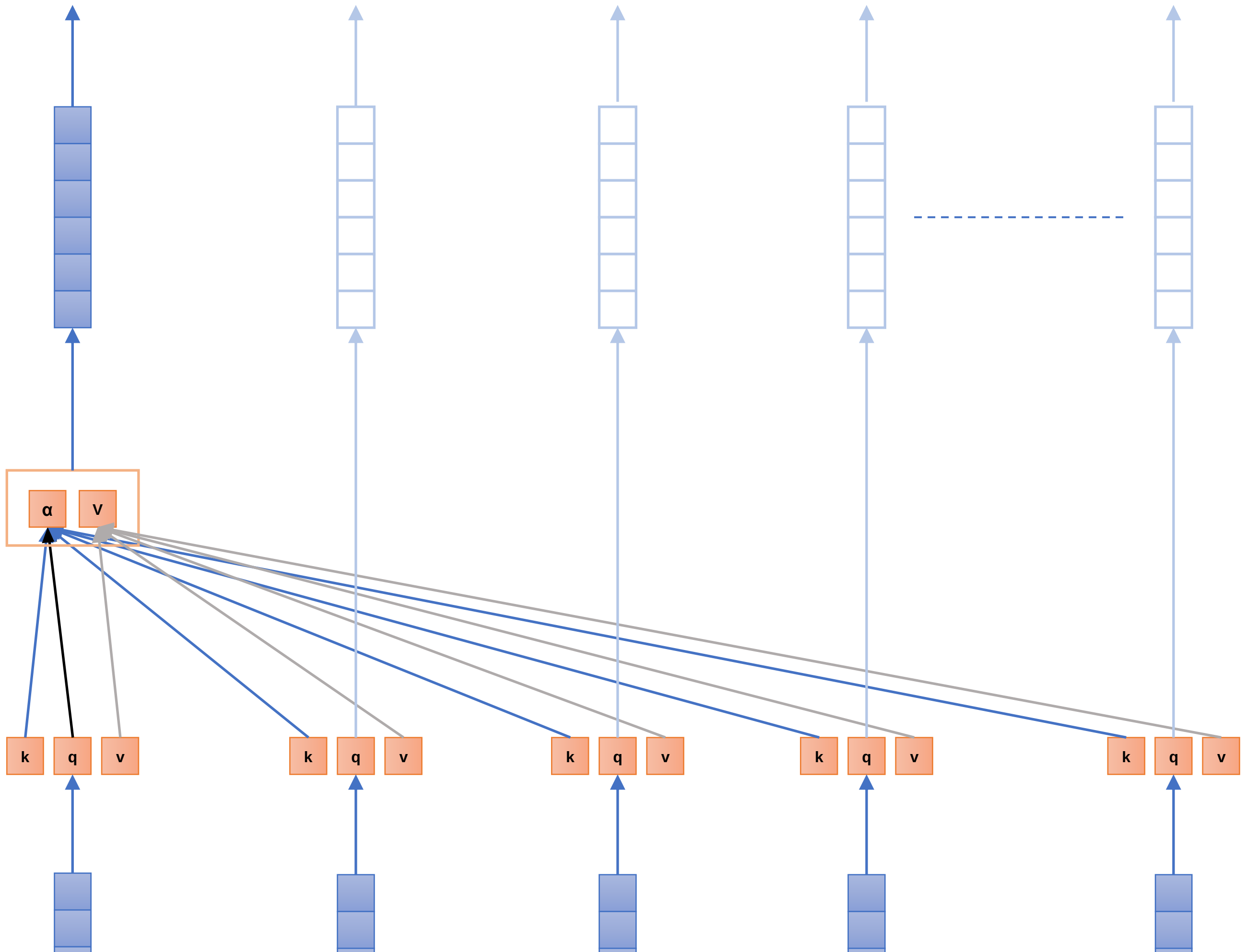
# Recap on Self-Attention

- Map input vectors to
  - $Query$: $q_i = W_Q X_i$,
  - $Key$: $k_i = W_K X_i$,
  - $Value$: $v_i = W_V X_i$,
- Compute weights vector: $A = softmax(\boldsymbol{q}\boldsymbol{k}^T) \epsilon \mathbb{R}^{N \times N}$
- Obtain context vectors: $C_j = A\boldsymbol{v}$

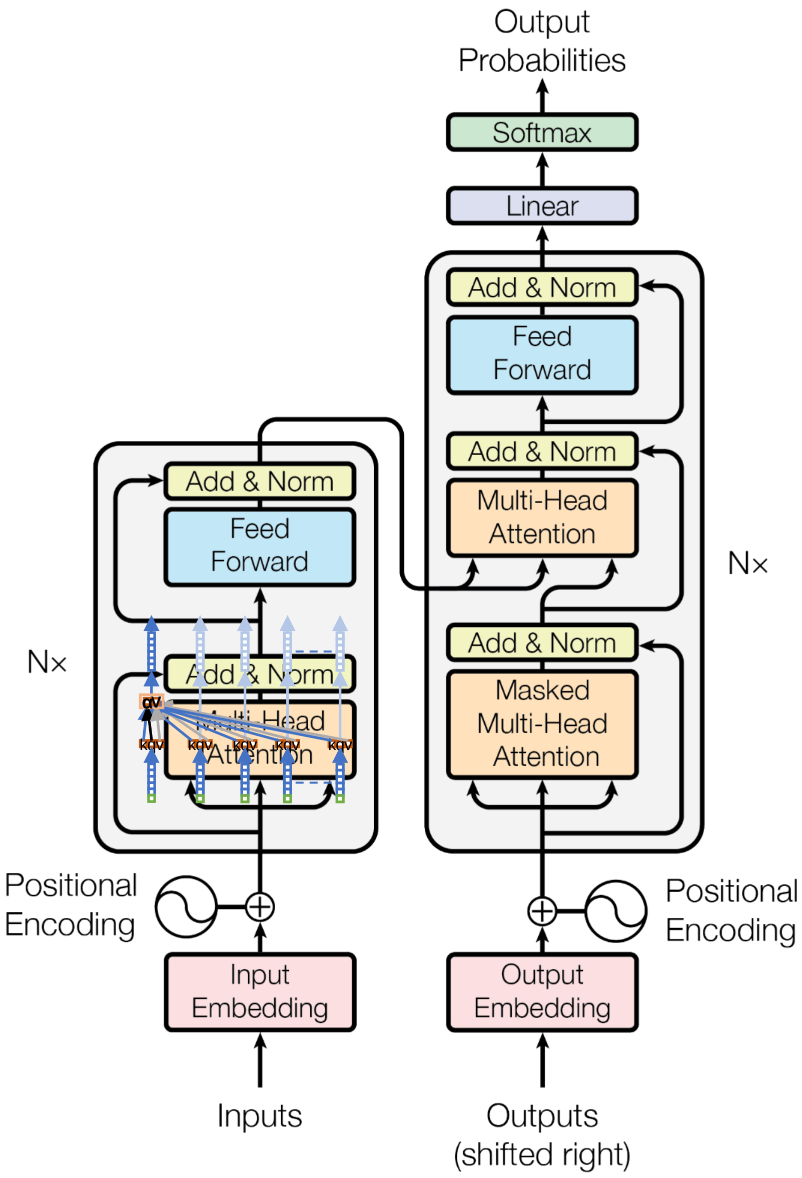


# Recap on Self-Attention

- Self-attention layer: $C_j = f(X, X, X)$
  - $Inputs$: $X = [x_1, x_2, \dots x_m]$
  - $Parameters$: $W_K, W_Q, W_V$



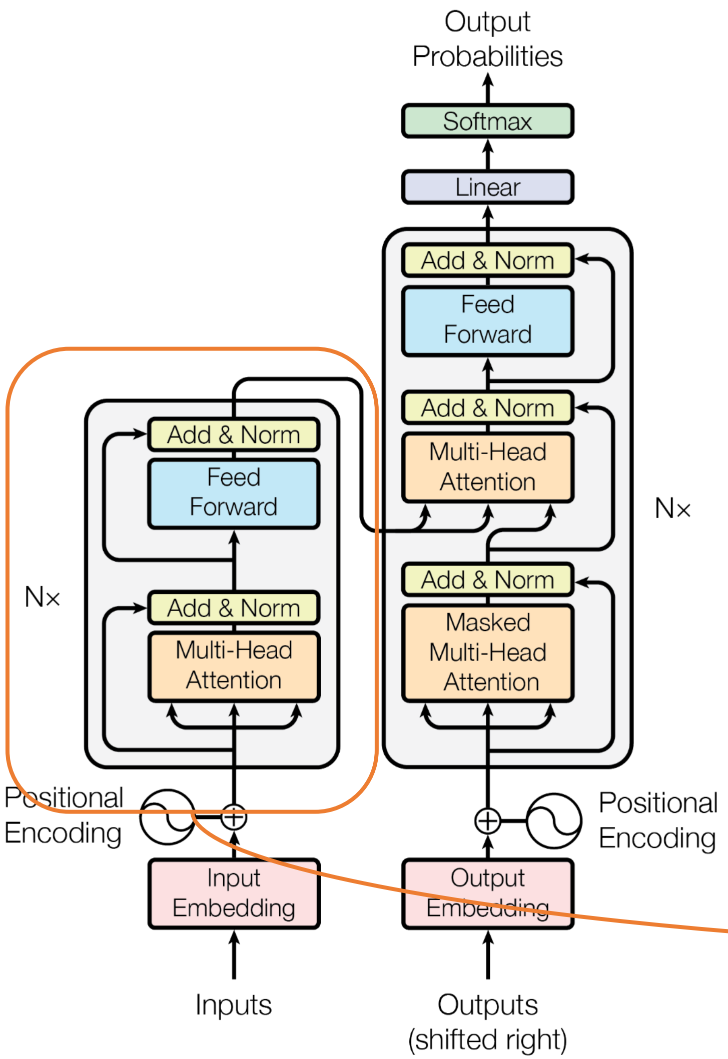
# The Transformer Model



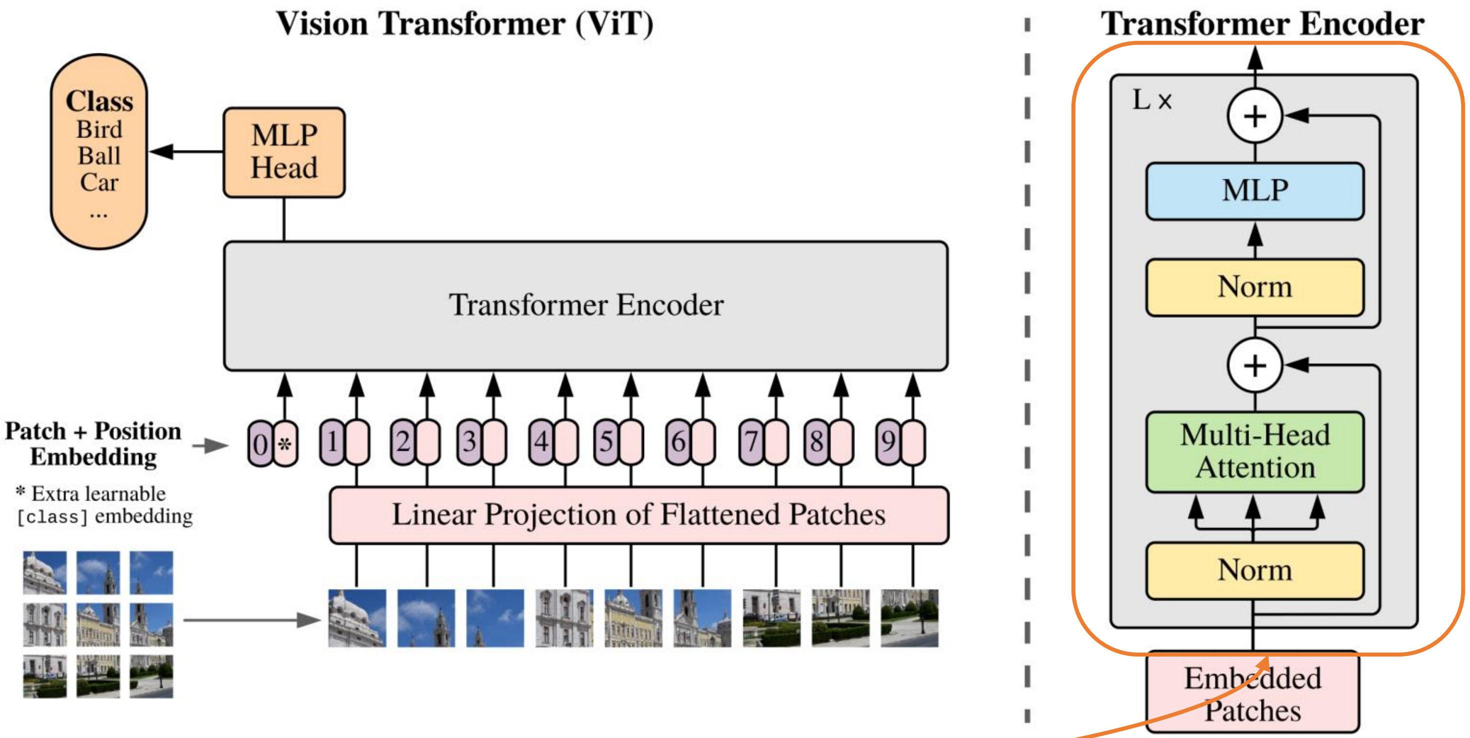

*Attention Is All You Need, Vaswani et al. 2017

# The Transformer Model for Vision

*

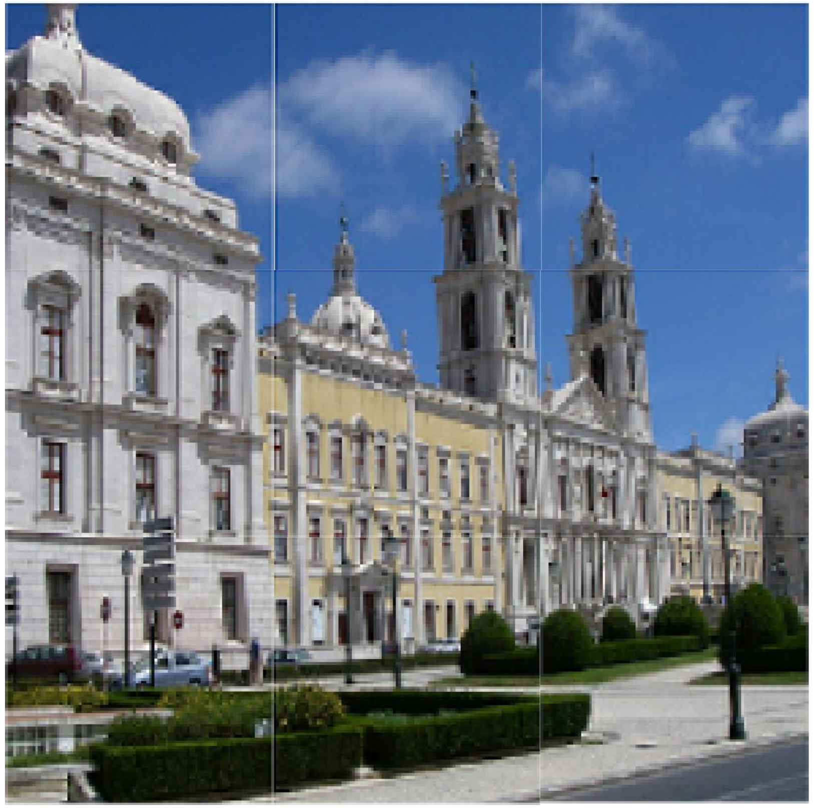



+




*[Attention Is All You Need, Vaswani et al. 2017](), +[An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale, Dosovitskiy et al. 2021]()

# Transformers for Image Recognition



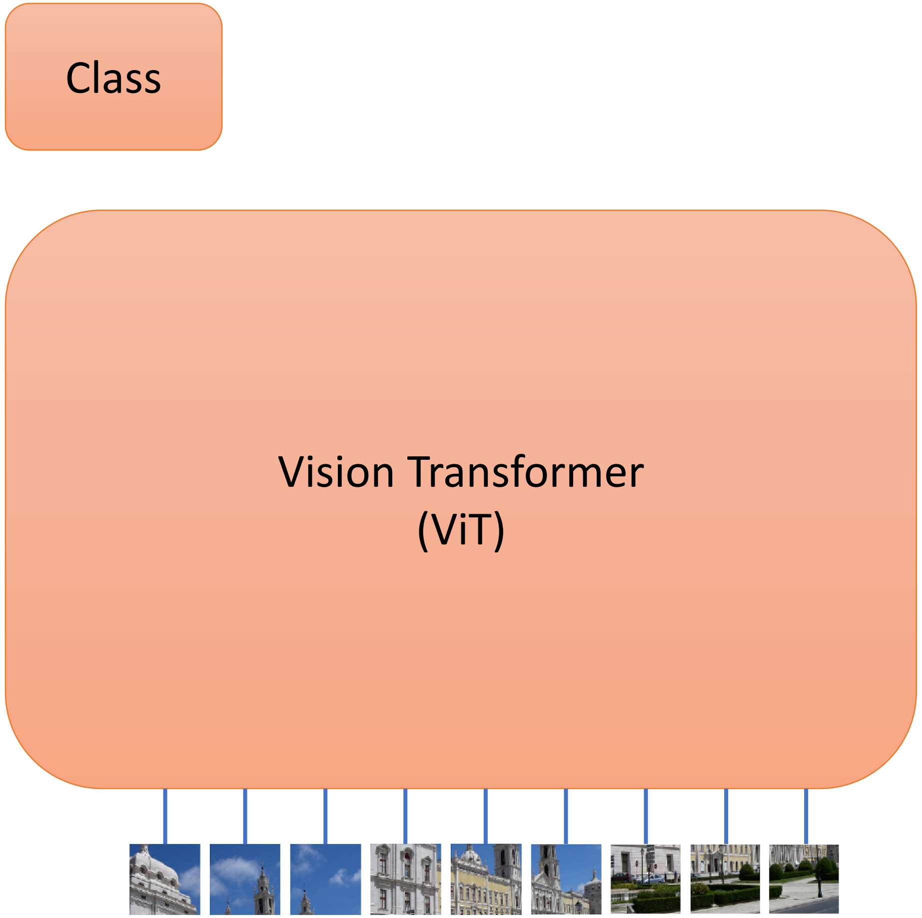
Vision Transformer (ViT)

Class

Needs a sequence of token embeddings

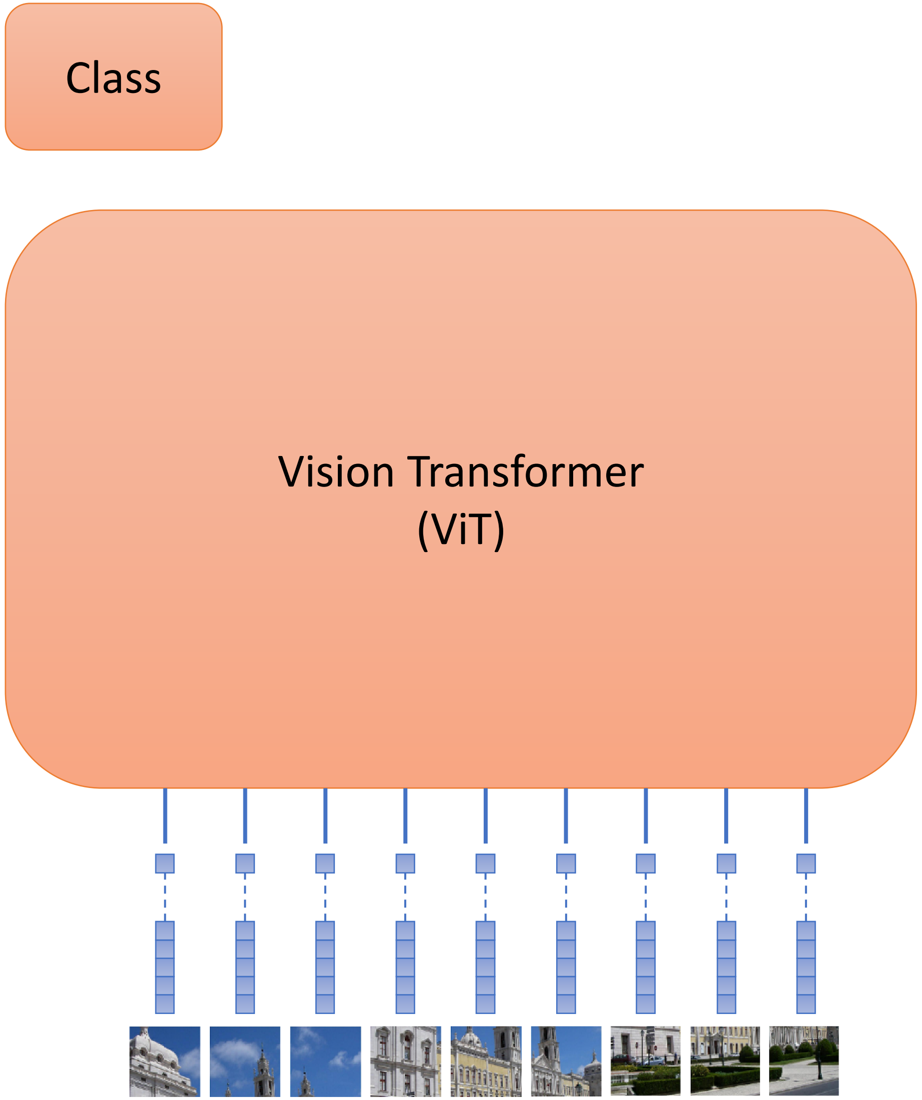
# Transformers for Image Recognition

- Split image into patches of the same shape
  - $N = {}^{HW}/_{P^2}\,, number\ of\ patches$



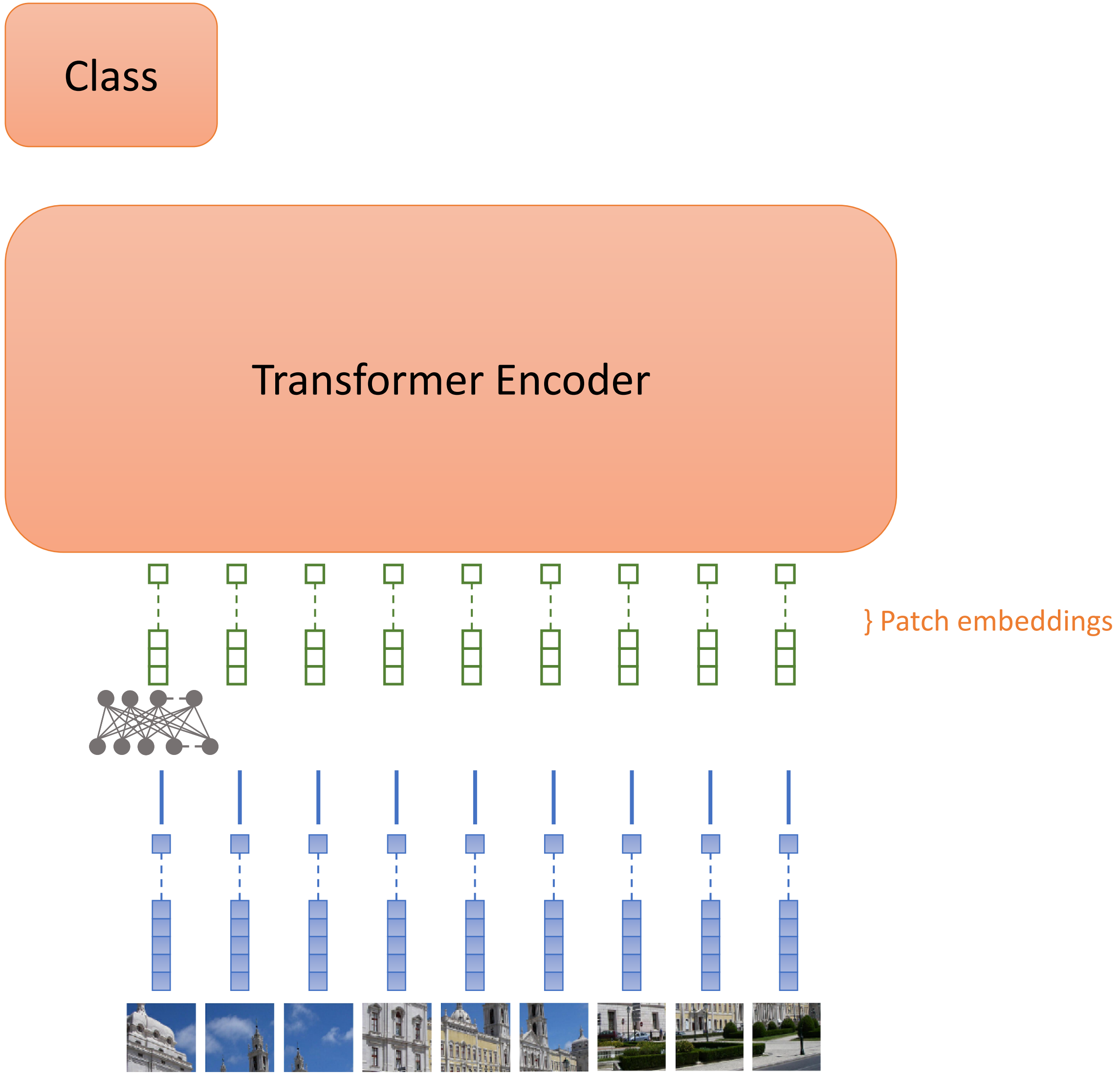
# Transformers for Image Recognition

- Split image into patches of the same shape
  - $N = {}^{HW}/_{P^2}, number\ of\ patches$
- Flatten patches to sequence of 1D vectors
  - $X \in \mathbb{R}^{H \times W \times C} \rightarrow X_P \in \mathbb{R}^{N \times (P^2 \cdot C)}$



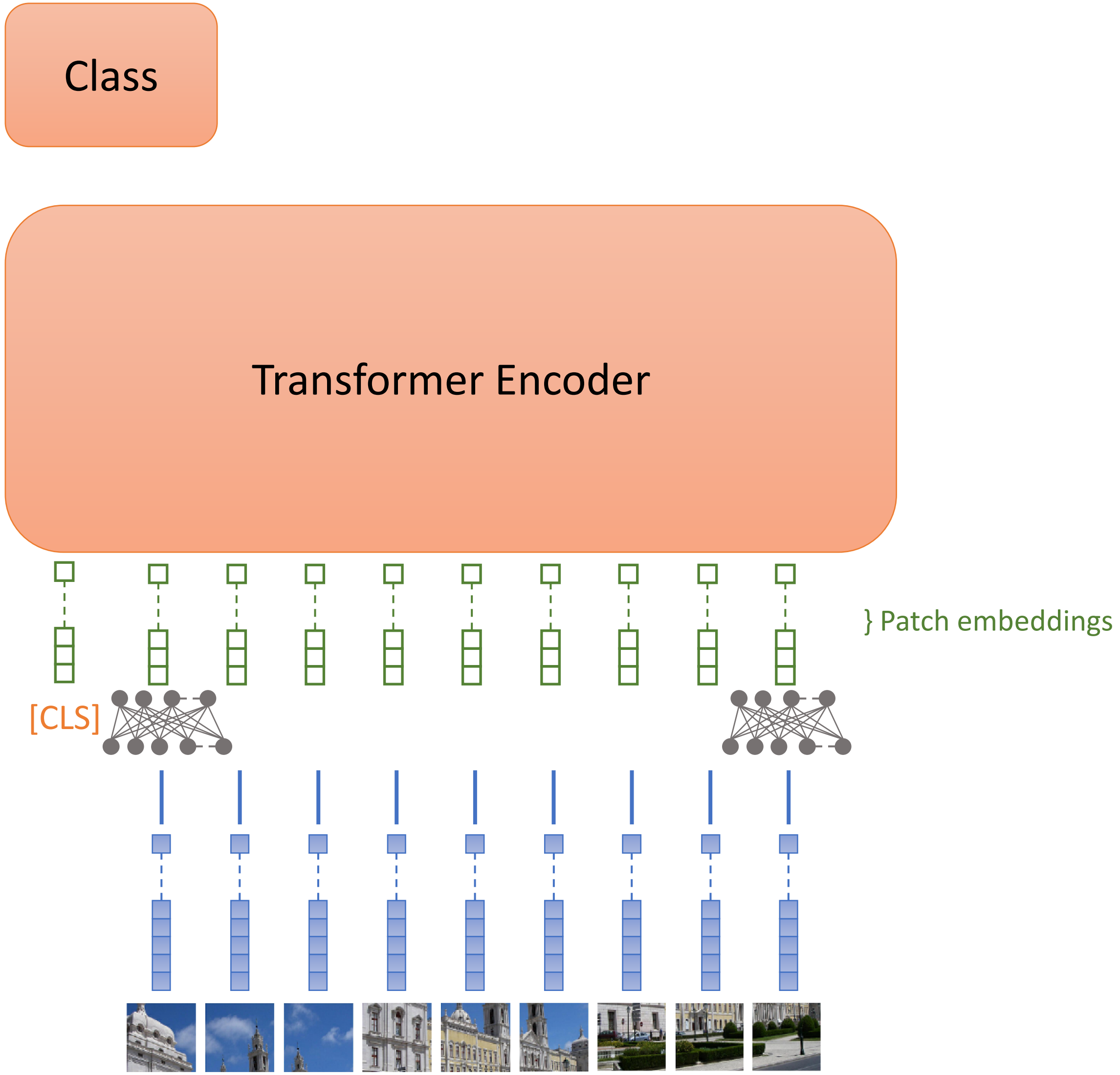
# Transformers for Image Recognition

- Split image into patches of the same shape
  - $N = {HW}/{P^2}, number\ of\ patches$
- Flatten patches to sequence of 1D vectors
  - $X \in \mathbb{R}^{H\times W\times C} \rightarrow X_p \in \mathbb{R}^{N\times(P^2\cdot C)}$
- Map to latent vector size $D$ with trainable linear projections



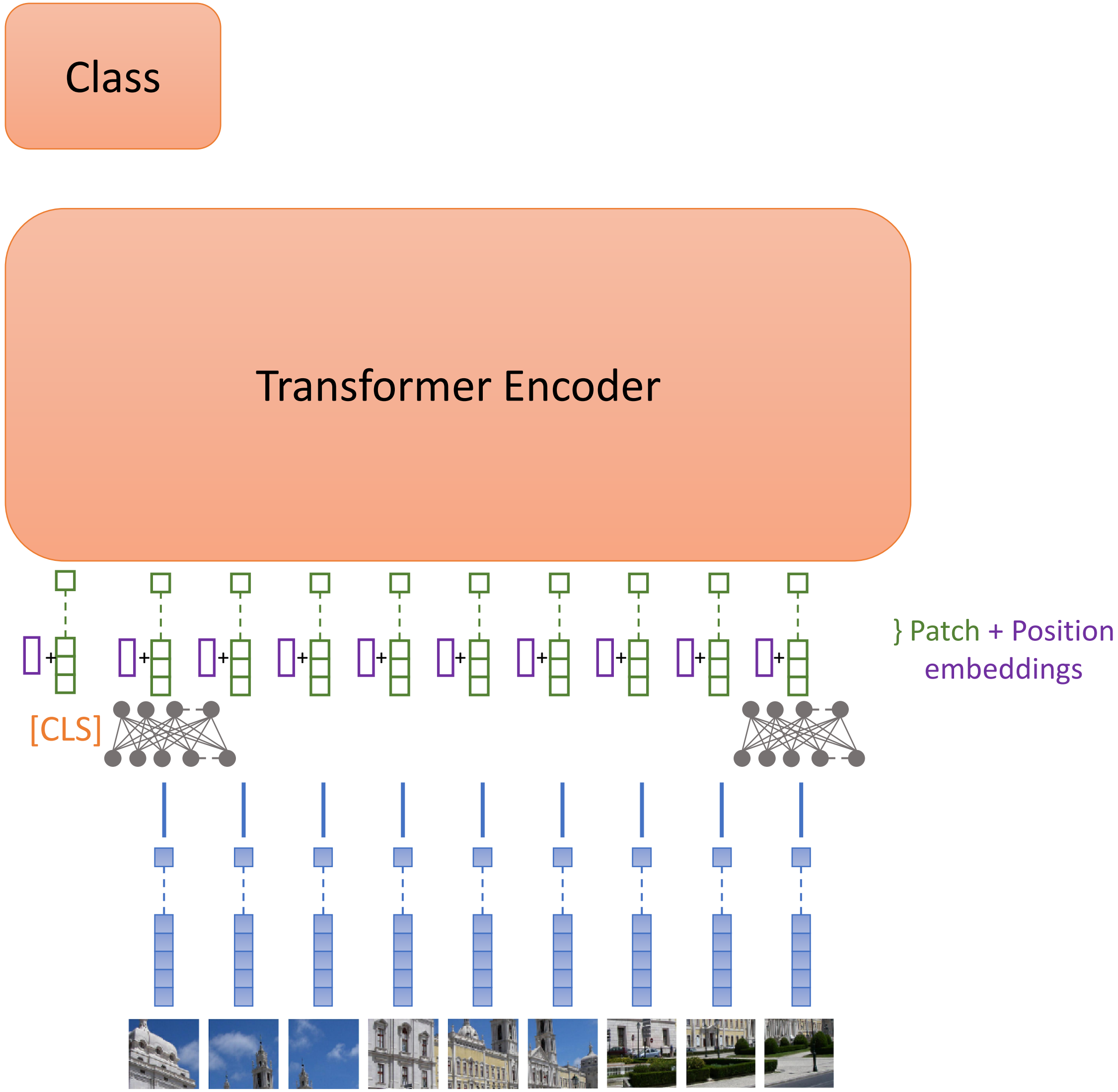
# Transformers for Image Recognition

- Prepend a learnable embedding for [class] to the sequence of embedded patches



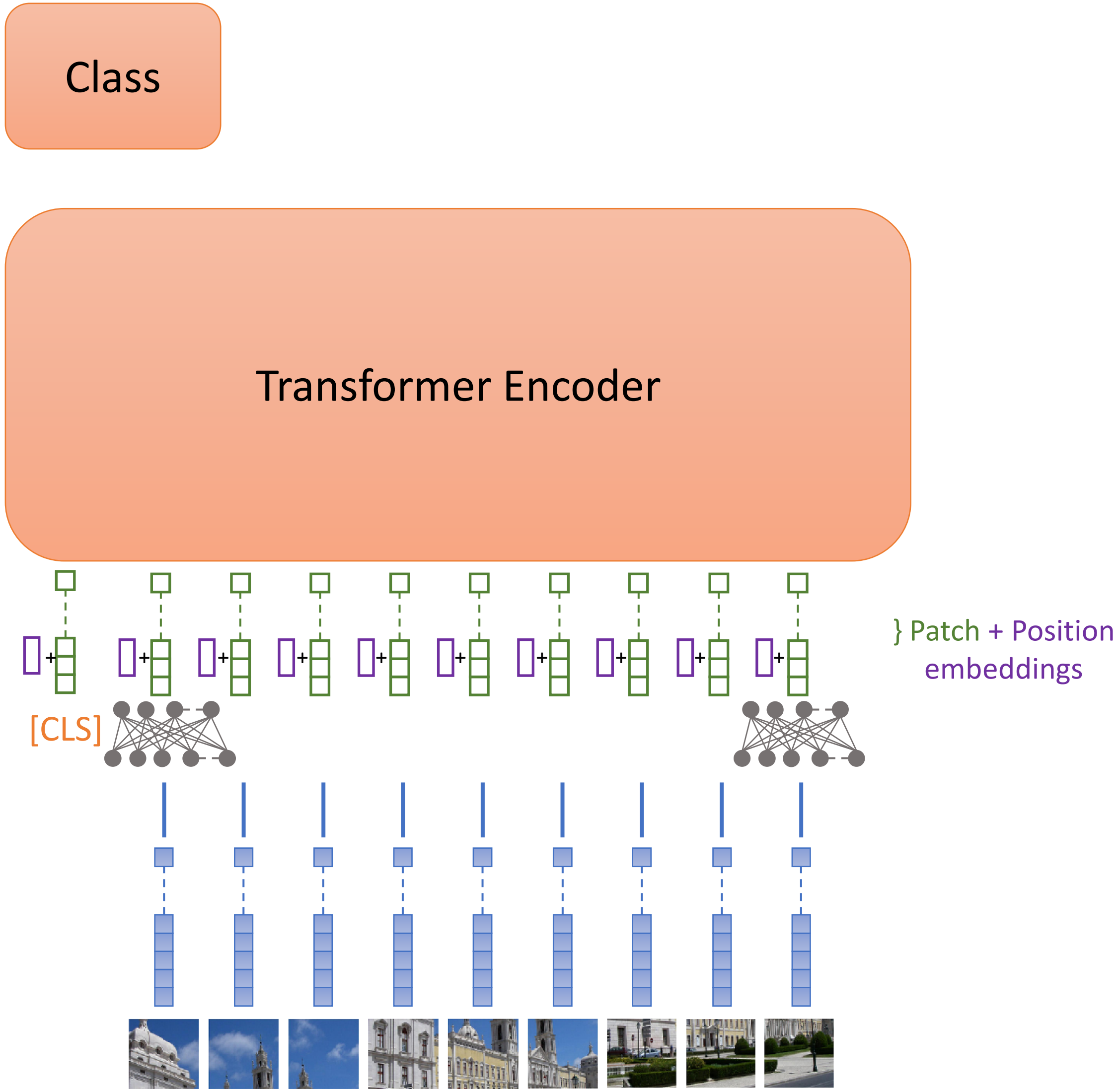
# Transformers for Image Recognition

- Prepend a learnable embedding for [class] to the sequence of embedded patches
- Add position embeddings



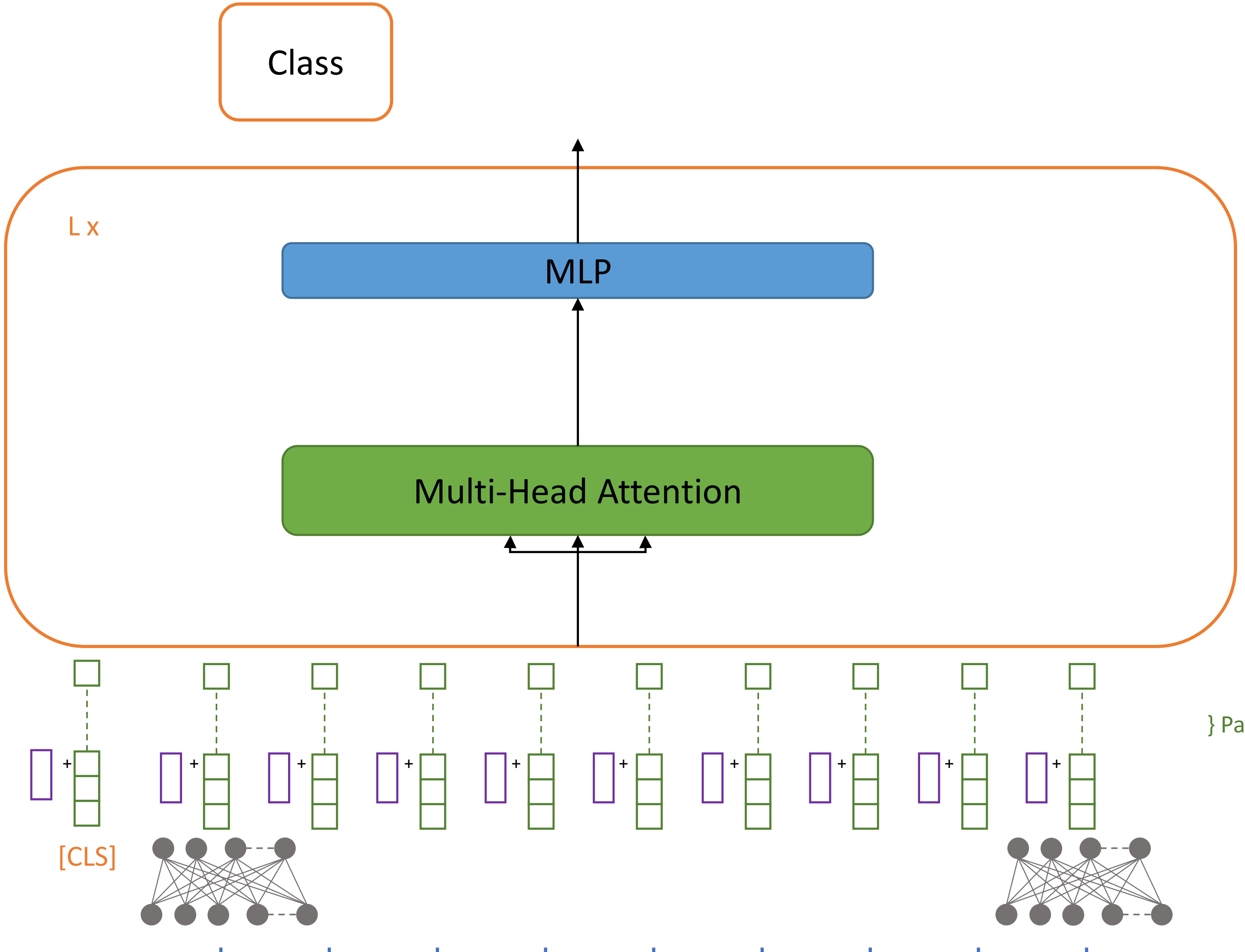
# Transformers for Image Recognition

- The transformer encoder is alternating layers of multiheaded self-attention and MLP blocks



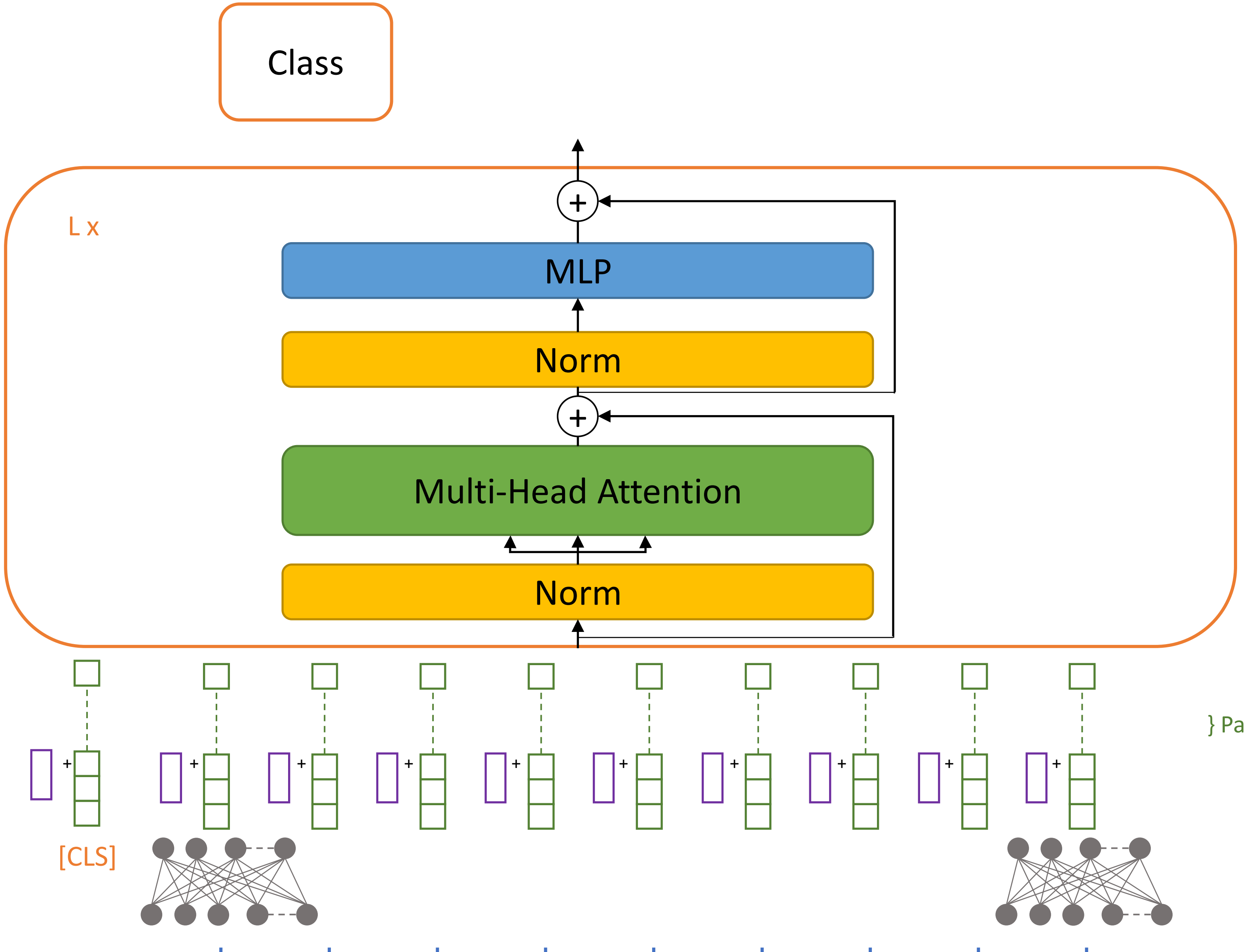
# Transformers for Image Recognition

- The transformer encoder is alternating layers of multiheaded self-attention and MLP blocks



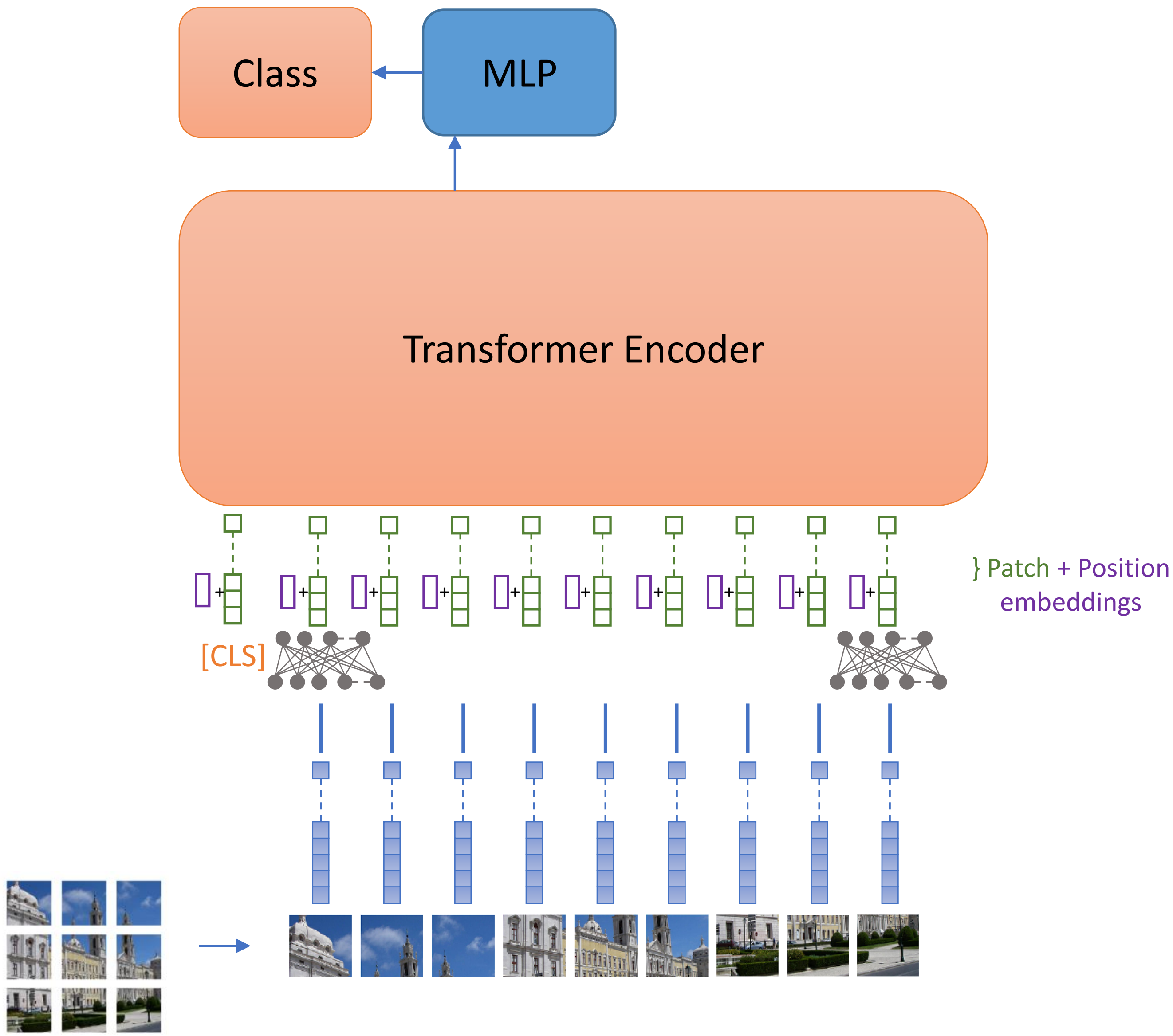
# Transformers for Image Recognition

- The transformer encoder is alternating layers of multiheaded self-attention and MLP blocks
- Layer normalization is applied before every block and residual connection after every block



# Transformers for Image Recognition

- The classification head is implemented by an MLP with one hidden layer during pretraining and a single linear layer during fine-tuning.



# Experiments and Results

- ViT is pretrained on large datasets and pre-train head is replaced with $D \times K$ feedforward layer
- ViT can handle arbitrary sequence length up to memory constraints, pre-trained position embeddings may no longer be meaningful

ImageNet
1k classes, 1.3M images

ImageNet-21k
21k classes, 14M images

JFT300M
(proprietary)
18k classes, 303M images

# Experiments and Results

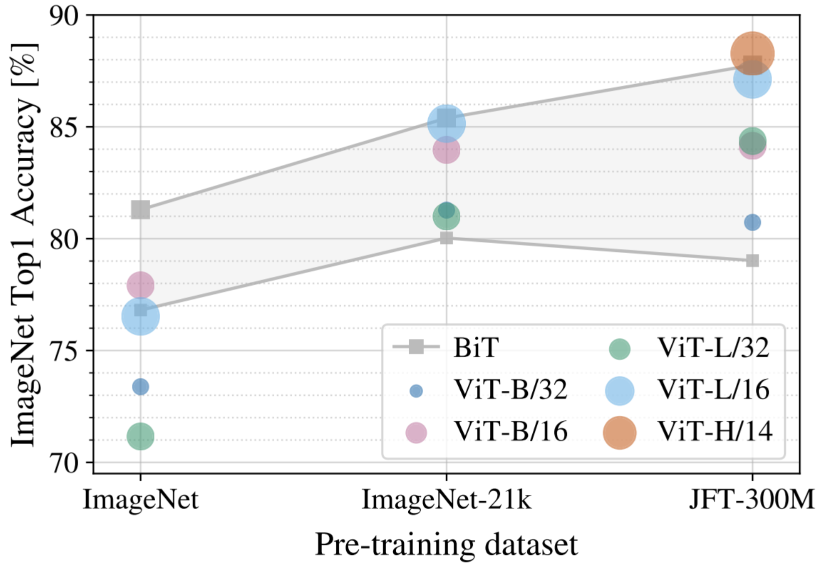
- Vision Transformer models pre-trained on the JFT-300M dataset outperform ResNet-based baselines on all datasets, while taking substantially less computational resources to pre-train

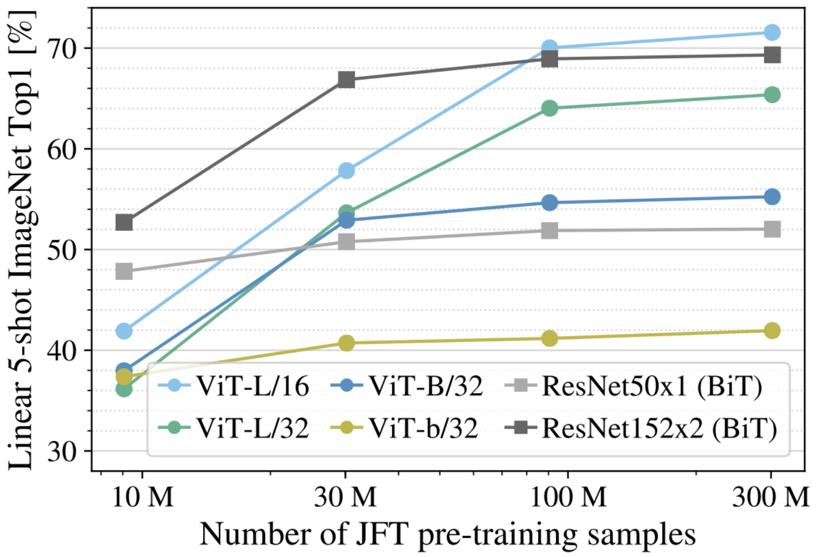
| | Ours-JFT (ViT-H/14) | Ours-JFT (ViT-L/16) | Ours-I21k (ViT-L/16) | BiT-L (ResNet152x4) | Noisy Student (EfficientNet-L2) |
|---|---|---|---|---|---|
| ImageNet | **88.55** $\pm$ 0.04 | 87.76 $\pm$ 0.03 | 85.30 $\pm$ 0.02 | 87.54 $\pm$ 0.02 | 88.4/88.5* |
| ImageNet ReaL | **90.72** $\pm$ 0.05 | 90.54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

# Experiments and Results

- Vision Transformer models pre-trained on the JFT-300M dataset outperform ResNet-based baselines on all datasets, while taking substantially less computational resources to pre-train
- ViT models perform worse than BiT ResNets (shaded area) when pre-trained on small datasets, ResNets perform better with smaller pre-training datasets but plateau sooner than ViT

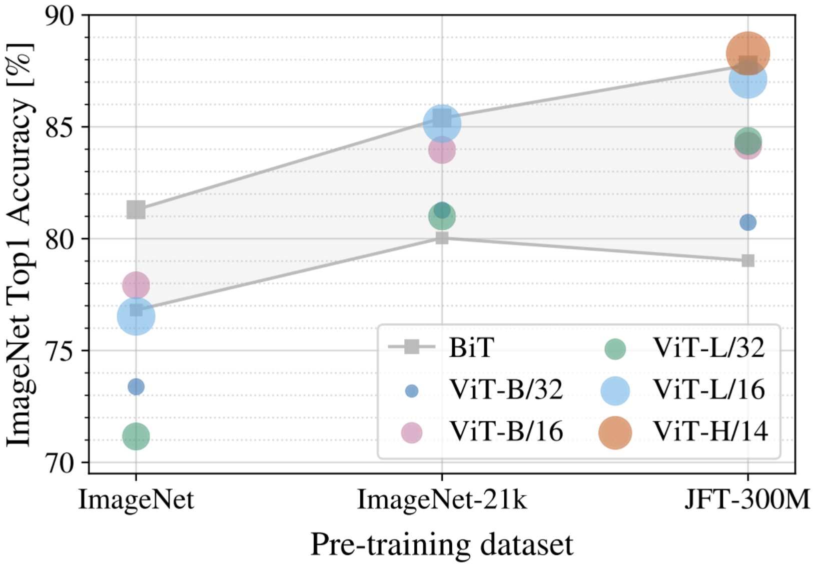
| | Ours-JFT (ViT-H/14) | Ours-JFT (ViT-L/16) | Ours-I21k (ViT-L/16) | BiT-L (ResNet152x4) | Noisy Student (EfficientNet-L2) |
|---|---|---|---|---|---|
| ImageNet | **88.55** $\pm$ 0.04 | 87.76 $\pm$ 0.03 | 85.30 $\pm$ 0.02 | 87.54 $\pm$ 0.02 | 88.4/88.5* |
| ImageNet ReaL | **90.72** $\pm$ 0.05 | 90.54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

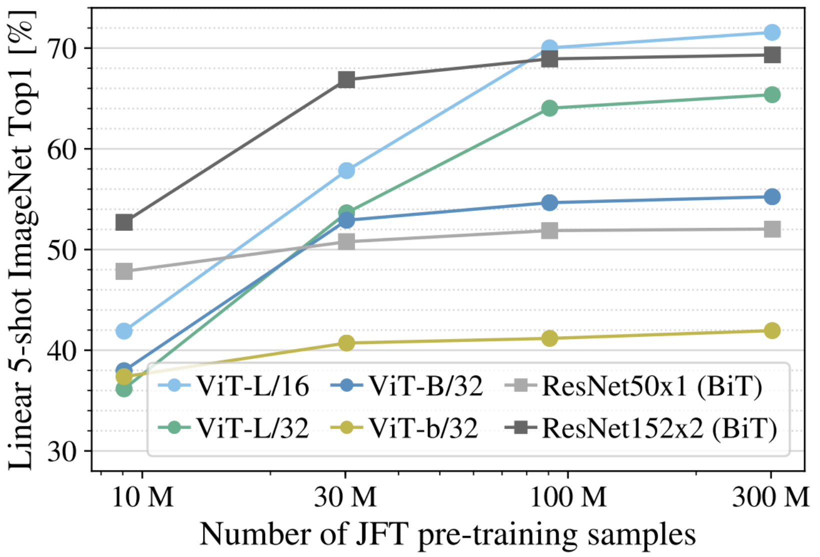


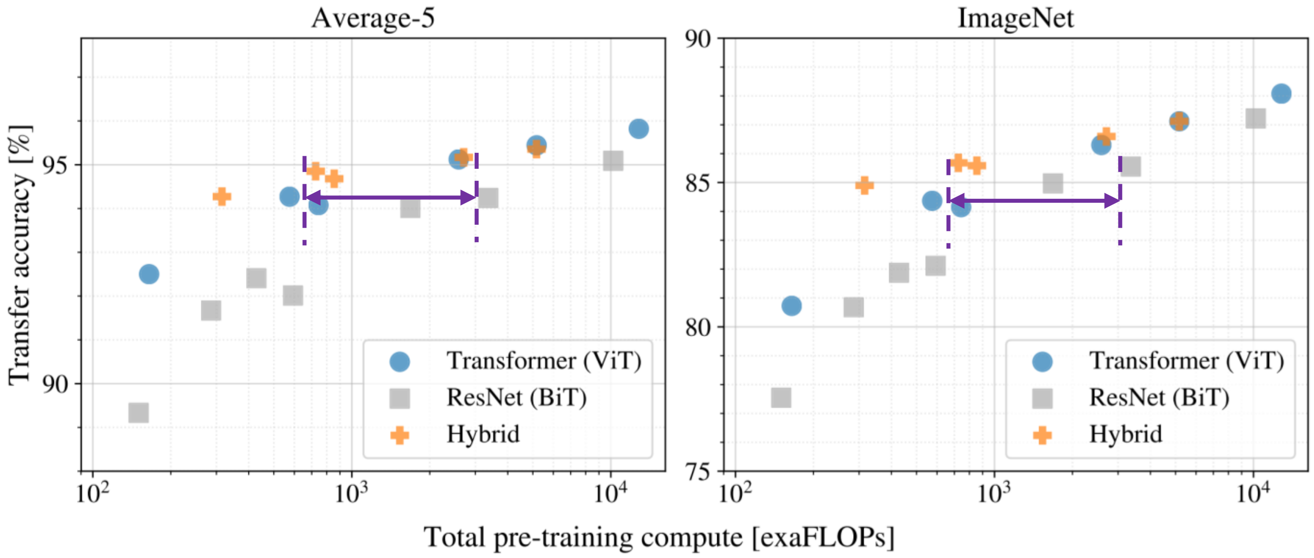



# Experiments and Results

- ViT models perform worse than BiT ResNets (shaded area) when pre-trained on small datasets, ResNets perform better with smaller pre-training datasets but plateau sooner than ViT
- Vision Transformers generally outperform ResNets with the same computational budget.

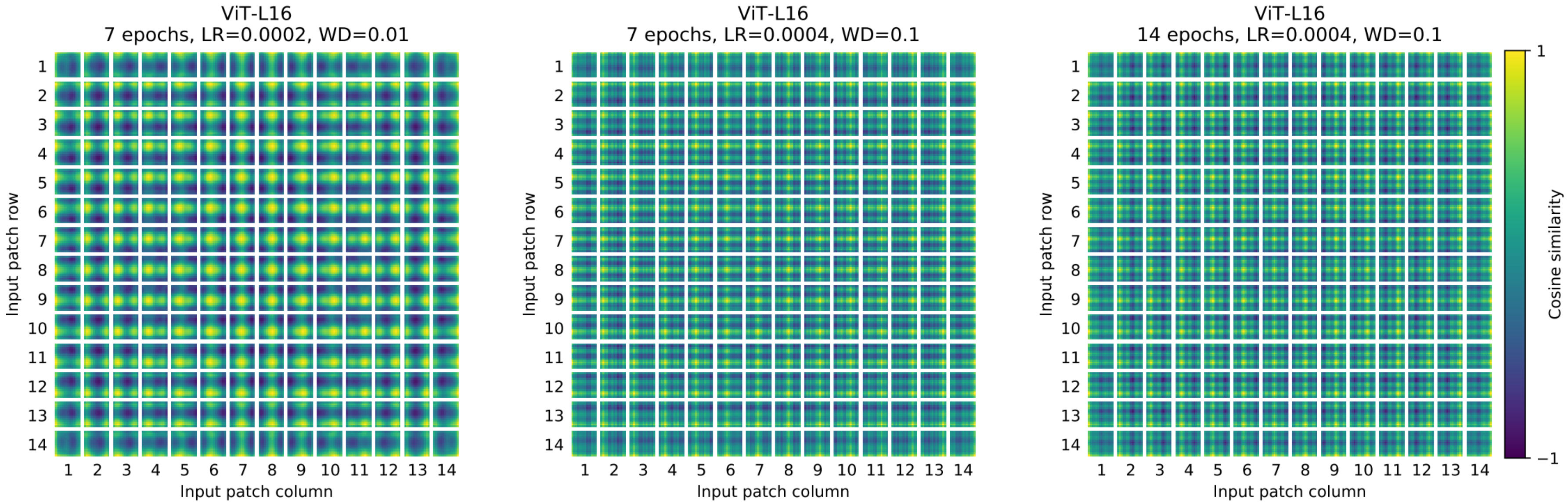


# Experiments and Results

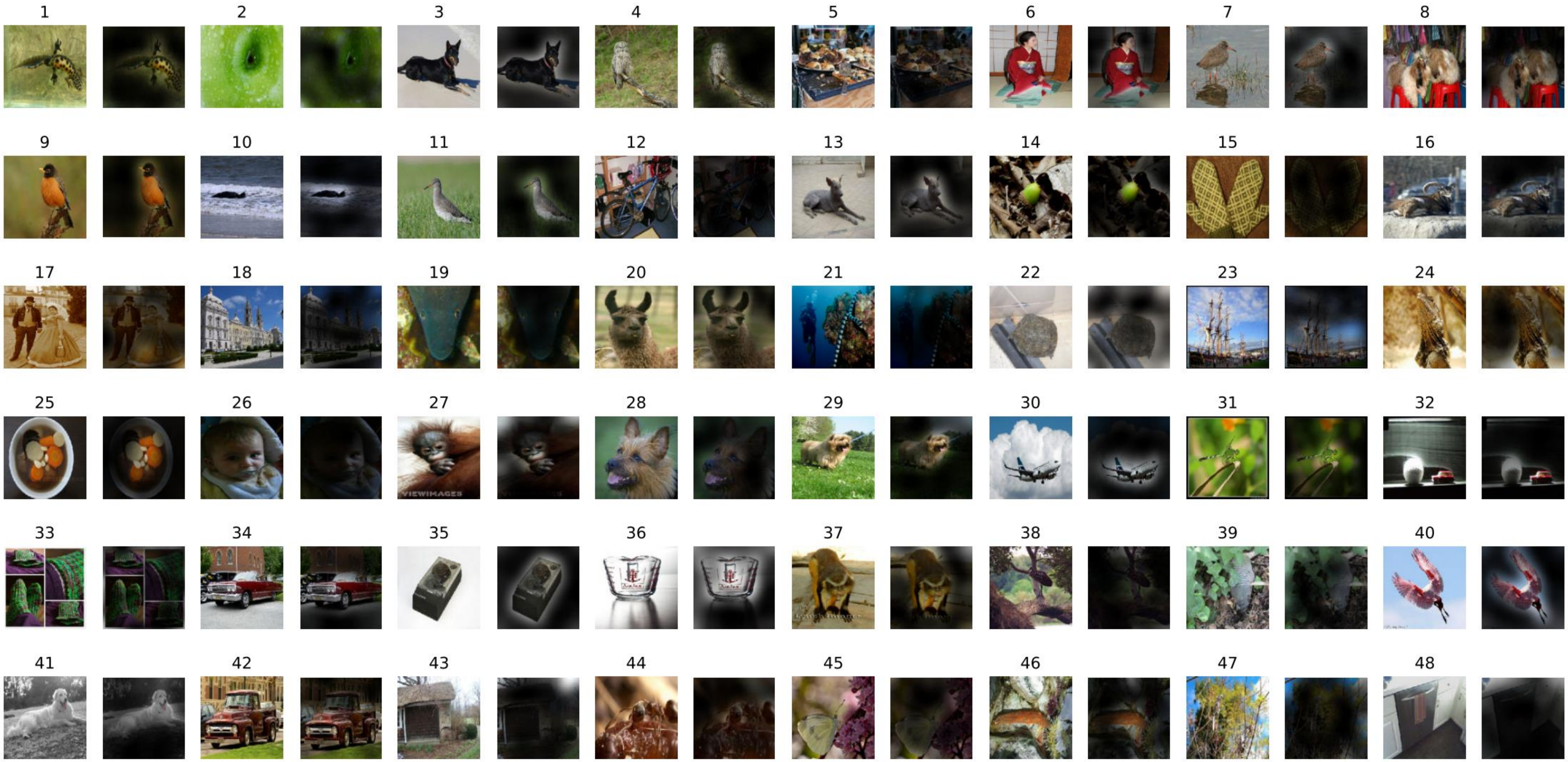
- No significant difference between implementations of 1D and 2D positional embeddings.
- The model learns to encode distance within the image in the similarity of position embeddings



# Final Points

- Transformers lack some inductive biases inherent to CNNs, such as translation equivariance and locality and therefore do not generalize well when trained on insufficient amount of data
- There are still unaddressed challenges
  - Application to other vision tasks; detection and segmentation
  - How to do large scale self-supervised pre-training

# Thank you


1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48