An Image is Worth 16x16 Words
Transformers for Image Recognition at Scale

ICLR 2021

Alexey Dosovitskiy, Lucas Beyer, Neil Houlsby

https://github.com/google-research/vision_transformer

Authors

Alexey Dosovitskiy
Google

Neural Networks | Computer Vision | Unsupervised Learning

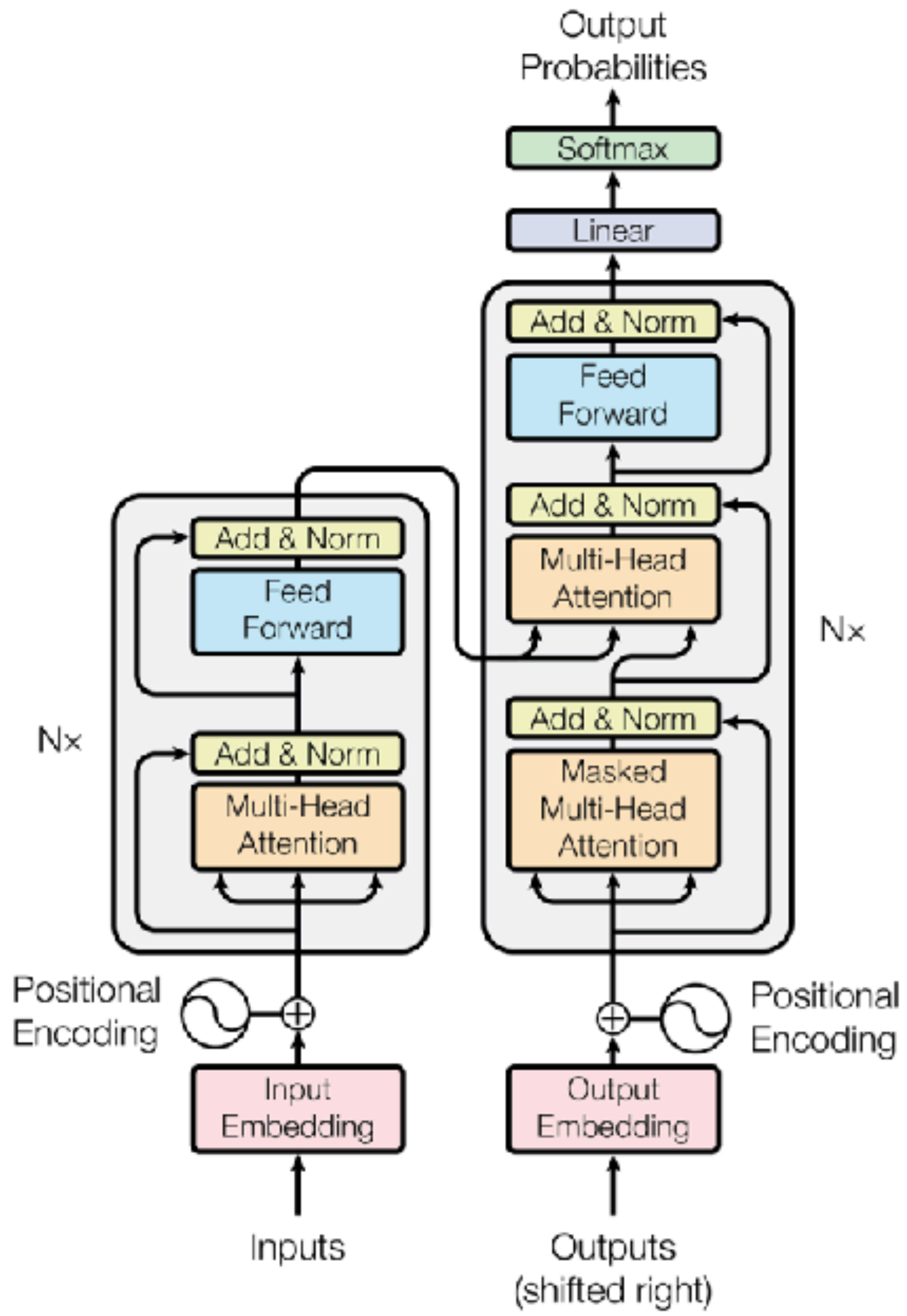
Lucas Beyer (Link)
Google DeepMind

Representation Learning | Robotics | Good Shift | Computer Vision

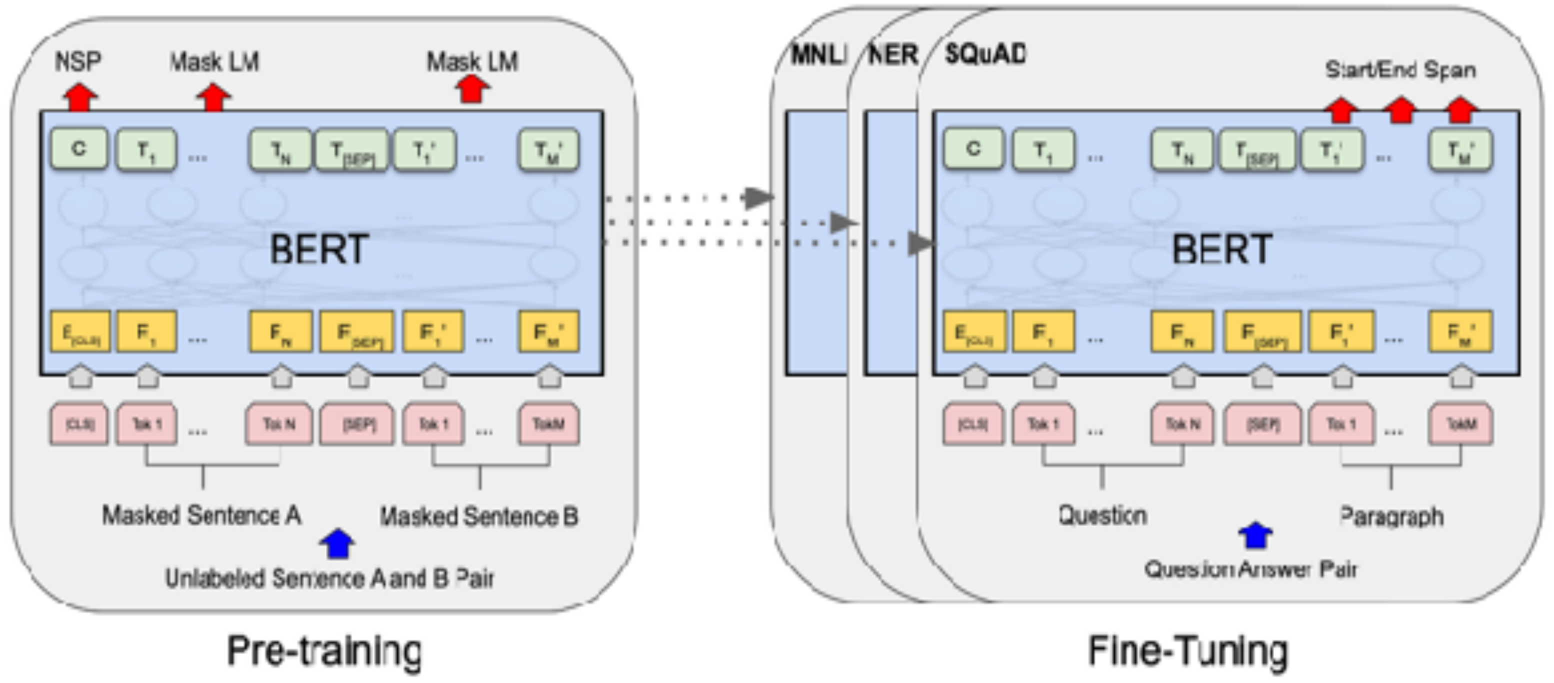
Neil Houlsby (Link)
Google

Artificial Intelligence | NLP | Machine Learning | Computer Vision

1. Introduction - NLP



Transformers
(Self-Attention-based Architecture)



Dominant Approach

[Pre-Train]
Large Text Corpus

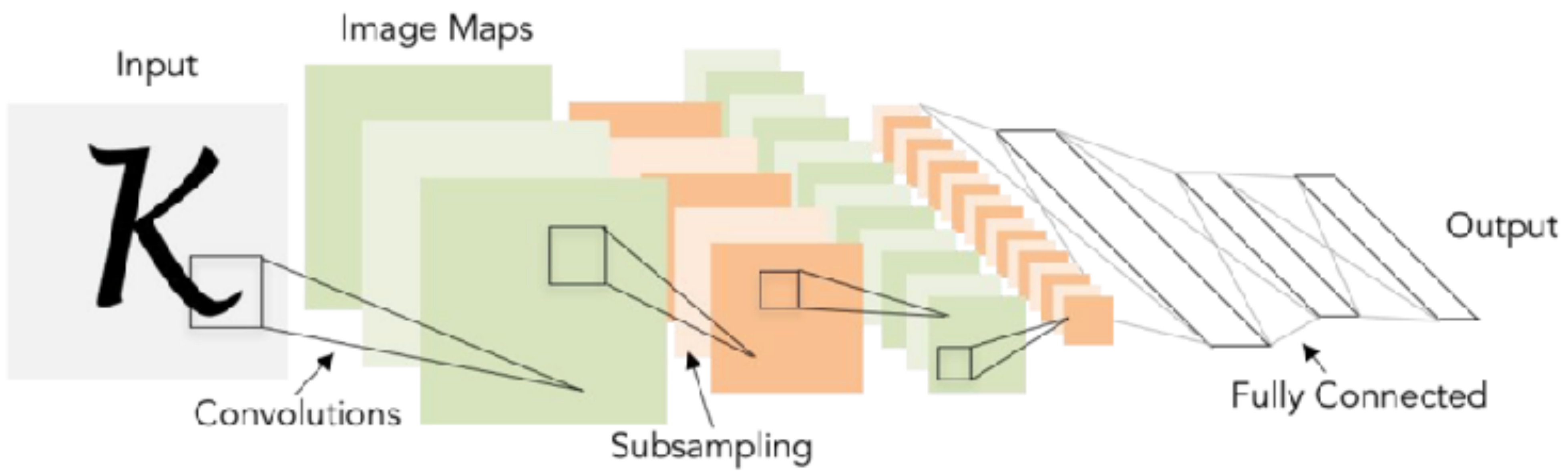
[Fine-Tune]
Smaller Task-Specific Dataset

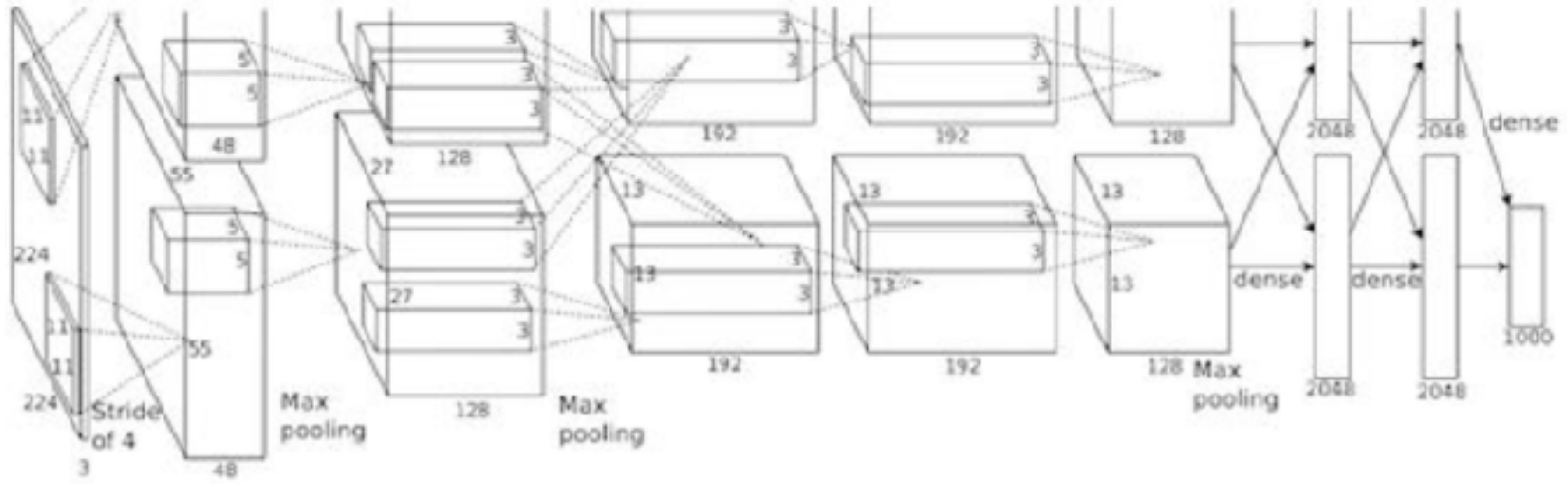
GPT-3

Efficiency & Scalability

Model of Unprecedented Size (e.g. over 100B parameters)

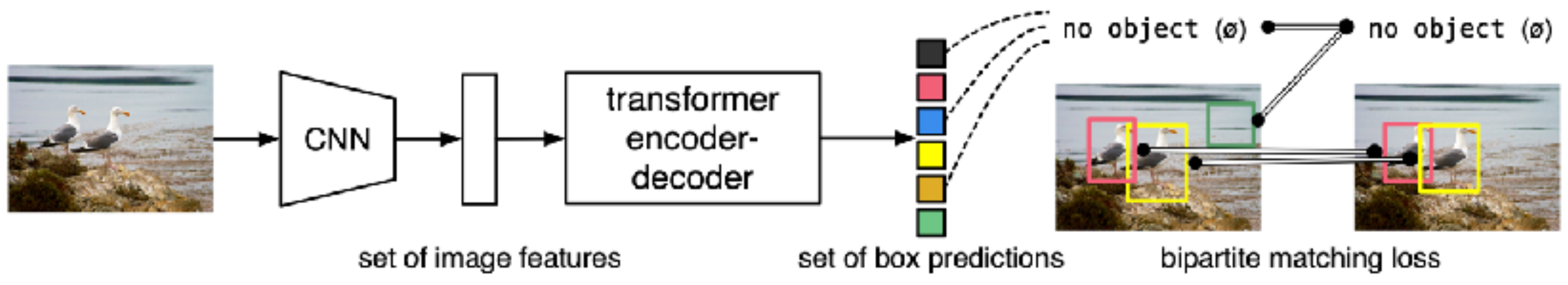
1. Introduction - CV



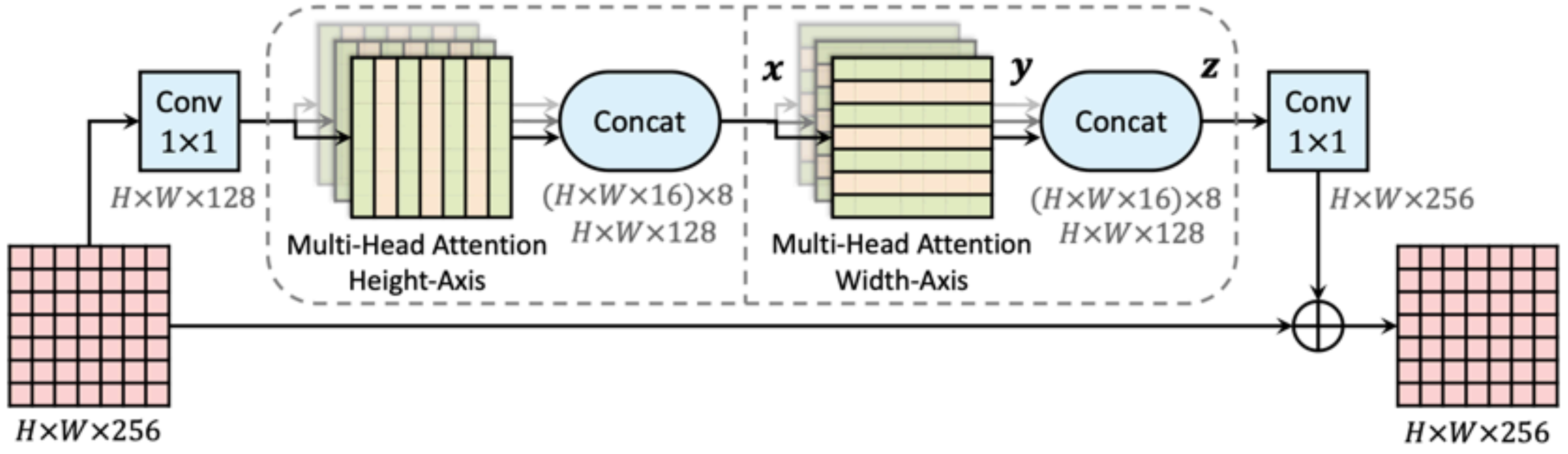


layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
conv2_x	56×56	3×3 max pool, stride 2				
		[3×3, 64; 3×3, 64] ×2	[3×3, 64; 3×3, 64] ×3	[1×1, 64; 3×3, 64; 1×1, 256] ×3	[1×1, 64; 3×3, 64; 1×1, 256] ×3	[1×1, 64; 3×3, 64; 1×1, 256] ×3
conv3_x	28×28	[3×3, 128; 3×3, 128] ×2	[3×3, 128; 3×3, 128] ×4	[1×1, 128; 3×3, 128; 1×1, 512] ×4	[1×1, 128; 3×3, 128; 1×1, 512] ×4	[1×1, 128; 3×3, 128; 1×1, 512] ×8
conv4_x	14×14	[3×3, 256; 3×3, 256] ×2	[3×3, 256; 3×3, 256] ×6	[1×1, 256; 3×3, 256; 1×1, 1024] ×6	[1×1, 256; 3×3, 256; 1×1, 1024] ×23	[1×1, 256; 3×3, 256; 1×1, 1024] ×36
conv5_x	7×7	[3×3, 512; 3×3, 512] ×2	[3×3, 512; 3×3, 512] ×3	[1×1, 512; 3×3, 512; 1×1, 2048] ×3	[1×1, 512; 3×3, 512; 1×1, 2048] ×3	[1×1, 512; 3×3, 512; 1×1, 2048] ×3
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

ures for ImageNet. Building blocks are shown in brackets (see also Fig. 5), with the numbers of block



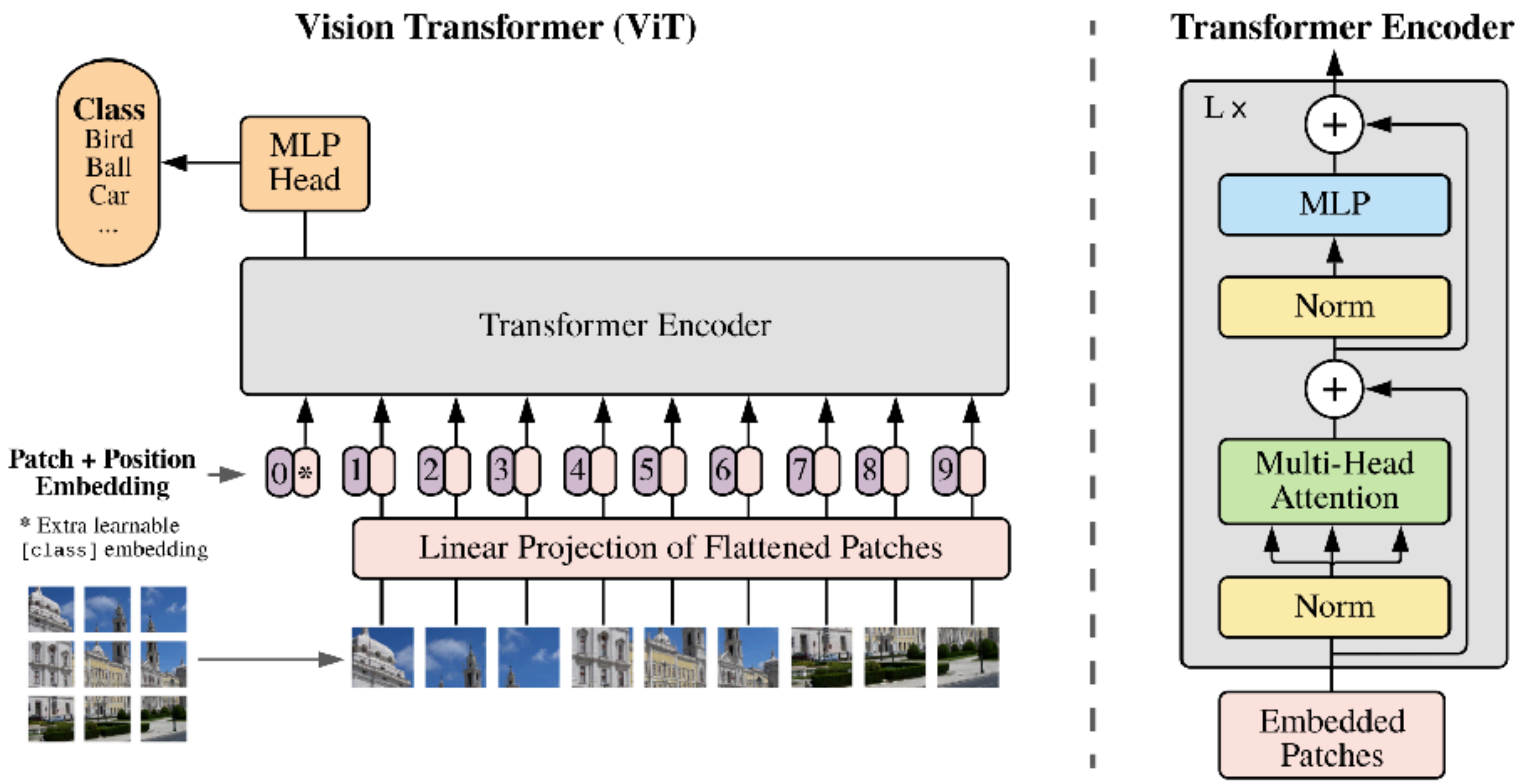
CNN-Like Architecture with Self-Attention



Replacing the Convolutions entirely

특수한 Attention Pattern의 사용 때문에 Modern HW에 효과적으로 적용되지 못함

1. Introduction - Contributions



약간의 변화로 기존의 Transformer의 직접적인 적용

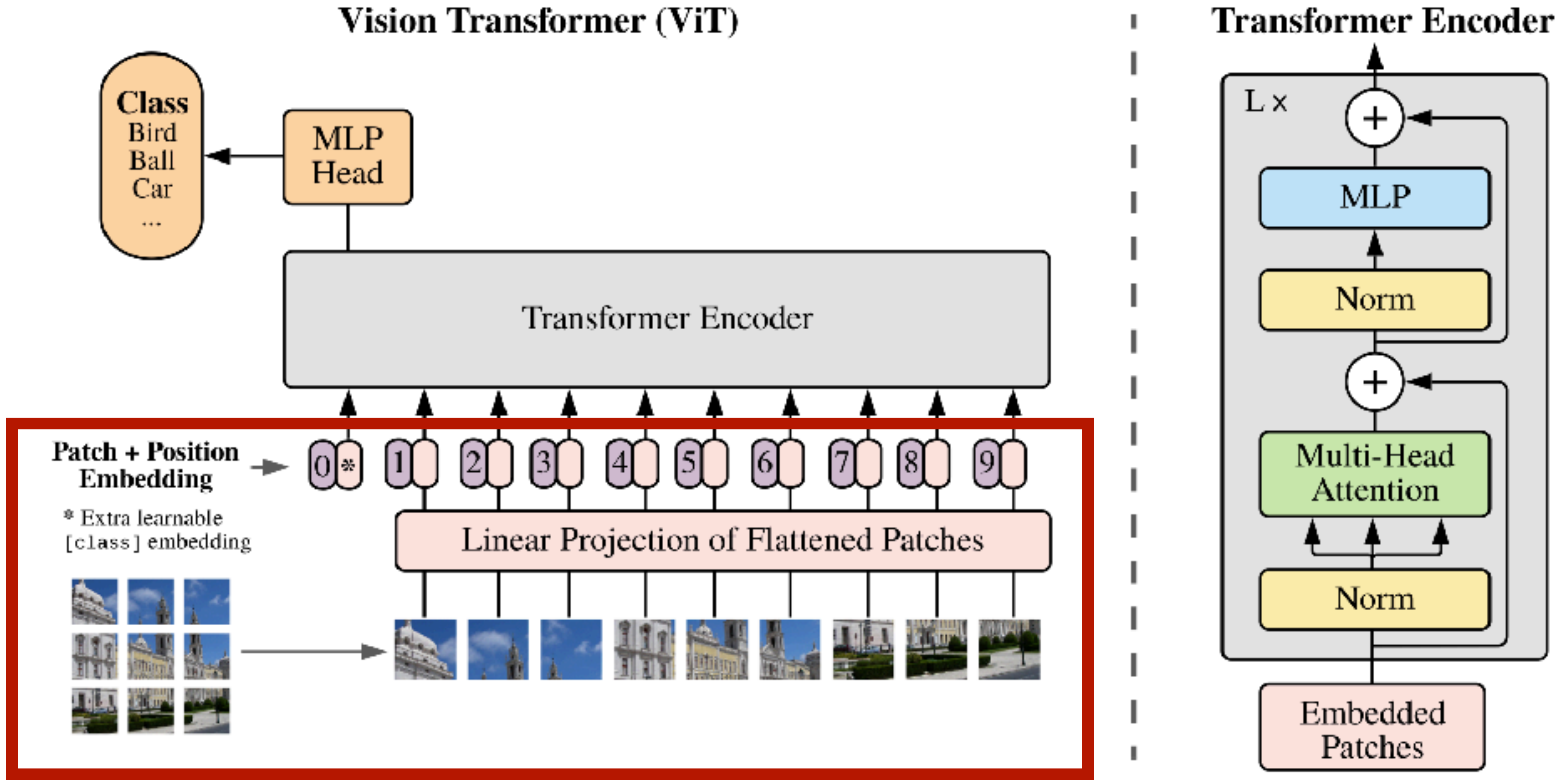
- Image를 여러 Patch의 형태로 분할
- Transformer의 입력으로 일련의 Patch의 Embedding 사용
- Image Classification Task와 Supervised Fashion

대용량의 Dataset을 통한 Pre-Trained Model

- CNN에 비해 낮은 Inductive Bias(e.g. Locality, Translation Equivariance)로 인한 낮은 성능 향상을 개선

https://moon-walker.medium.com/transformer는-inductive-bias이-부족하다라는-의미는-무엇일까-4f6005d32558

2. Method



2D Images

$$x \in \mathbb{R}^{H \times W \times C}$$

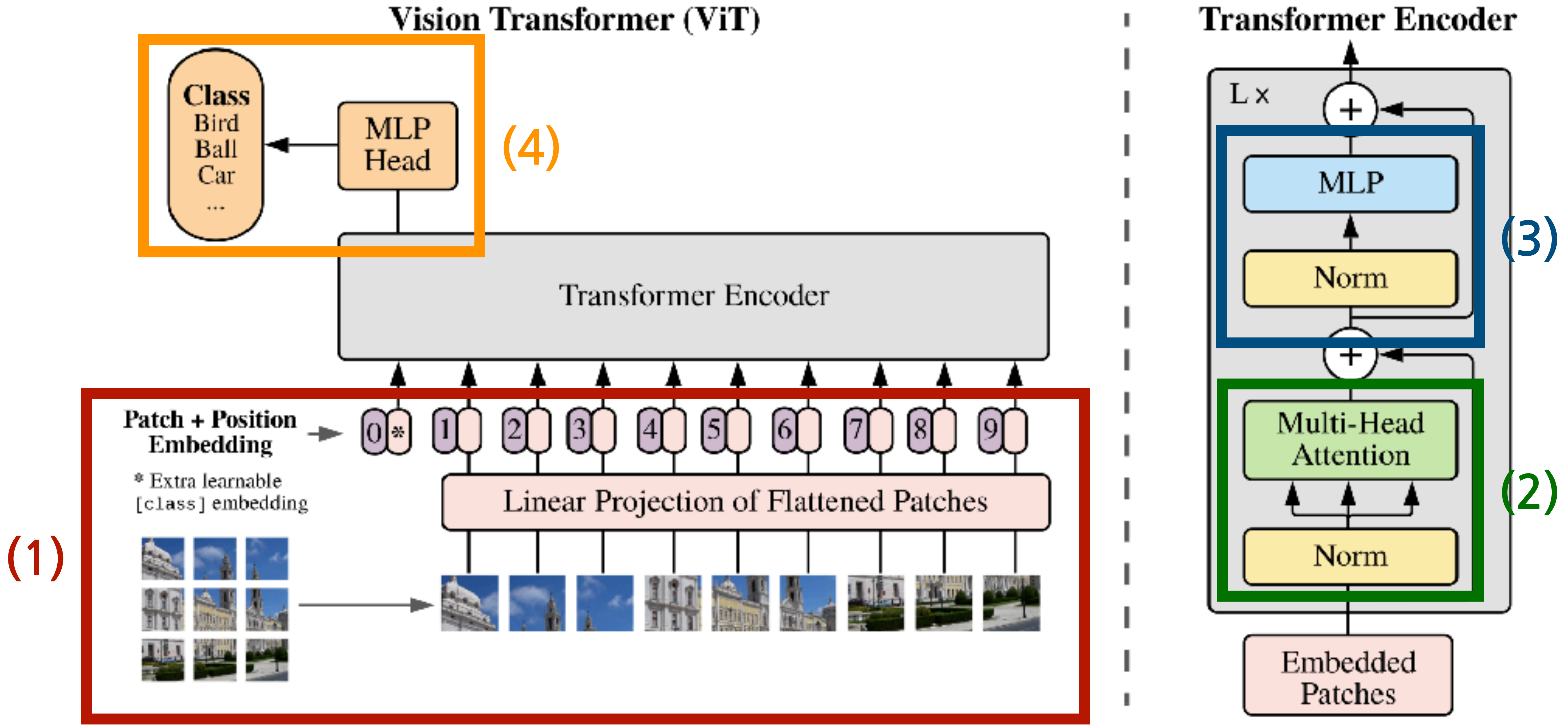
1D Token Embedding

$$x_p \in \mathbb{R}^{N \times (P^2 C)}$$

Resolution of Image Patch (P, P)

Number of Patches $N = HW/P^2$

2. Method



$$\mathbf{z}_0 = [\mathbf{x}_{\text{class}};\, \mathbf{x}_p^1\mathbf{E};\, \mathbf{x}_p^2\mathbf{E}; \cdots;\, \mathbf{x}_p^N\mathbf{E}] + \mathbf{E}_{pos}, \quad \mathbf{E} \in \mathbb{R}^{(P^2 \cdot C)\times D},\, \mathbf{E}_{pos} \in \mathbb{R}^{(N+1)\times D} \tag{1}$$

$$\mathbf{z}'_\ell = \mathrm{MSA}(\mathrm{LN}(\mathbf{z}_{\ell-1})) + \mathbf{z}_{\ell-1}, \quad \ell = 1 \ldots L \tag{2}$$

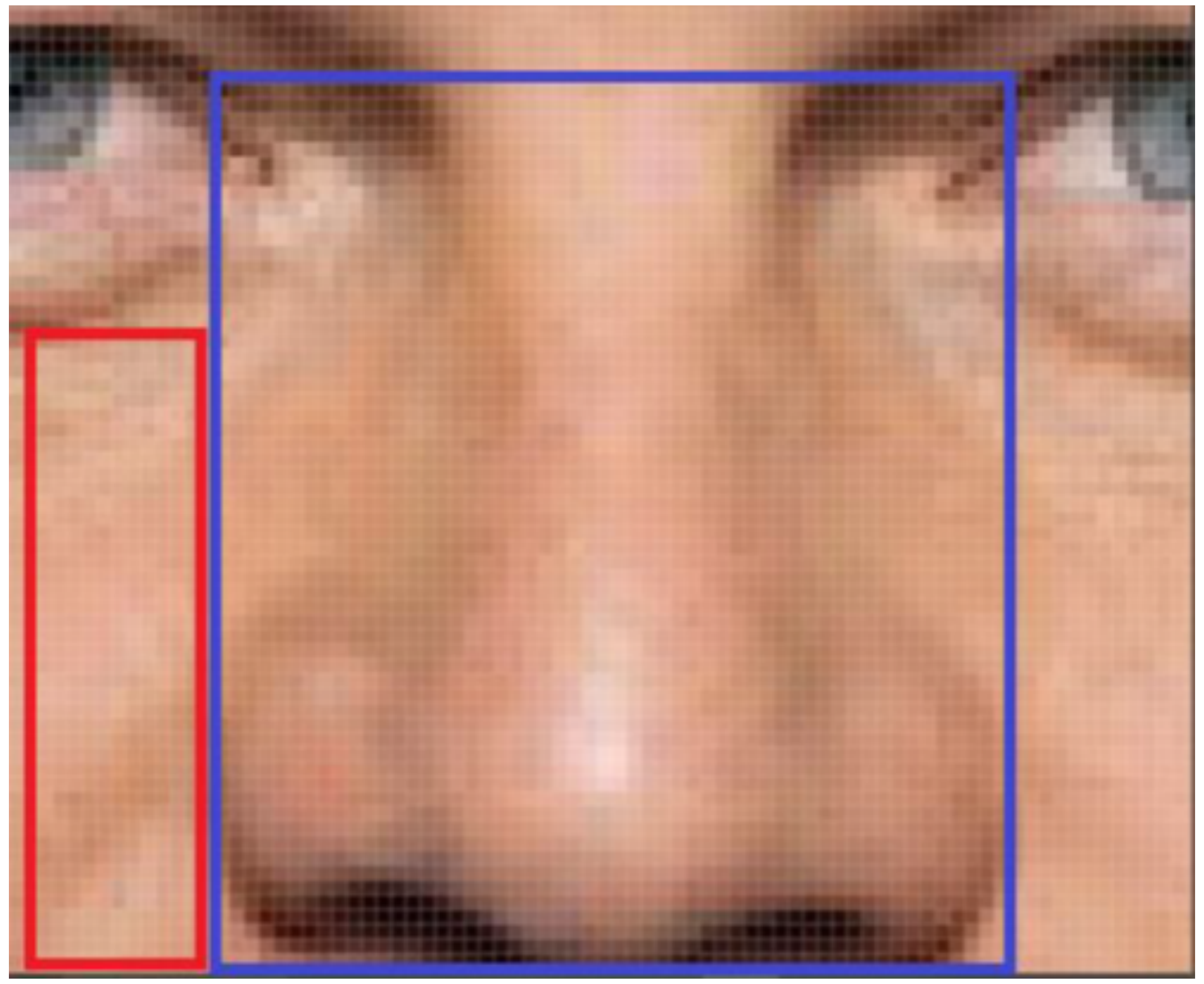
$$\mathbf{z}_\ell = \mathrm{MLP}(\mathrm{LN}(\mathbf{z}'_\ell)) + \mathbf{z}'_\ell, \quad \ell = 1 \ldots L \tag{3}$$

With GELU

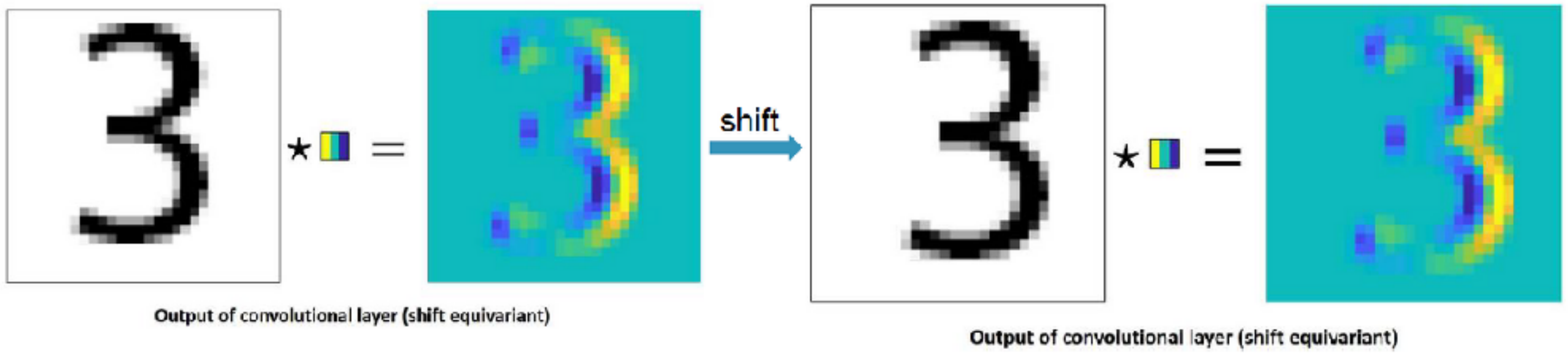
$$\mathbf{y} = \mathrm{LN}(\mathbf{z}_L^0) \tag{4}$$

2. Method

Inductive Bias: CNN > Transformer > FC



Locality

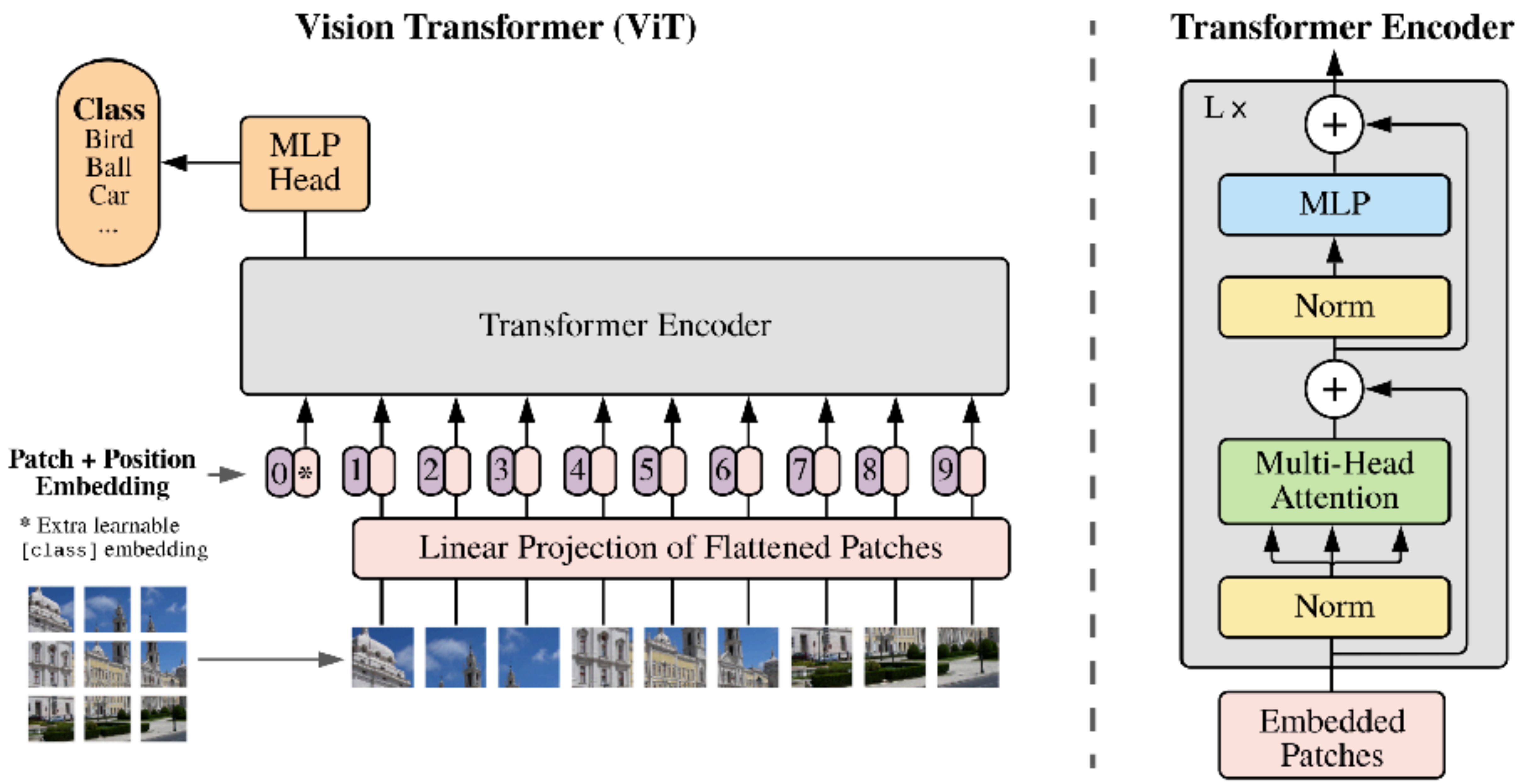


Translation Equivariance

https://daebaq27.tistory.com/108

2. Method

Inductive Bias: CNN > Transformer > FC



Patch/(Positional)Embedding: 2D Neighborhood Structure

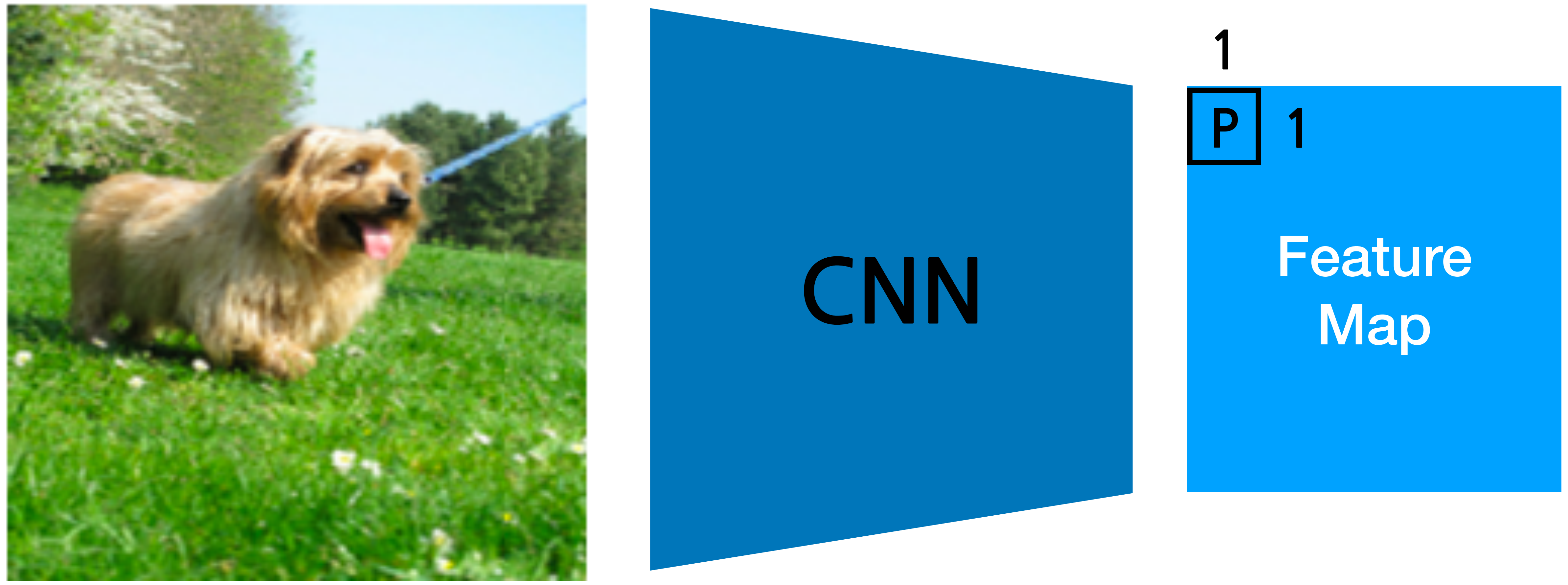
MLP: Locality + Translation Equivariance

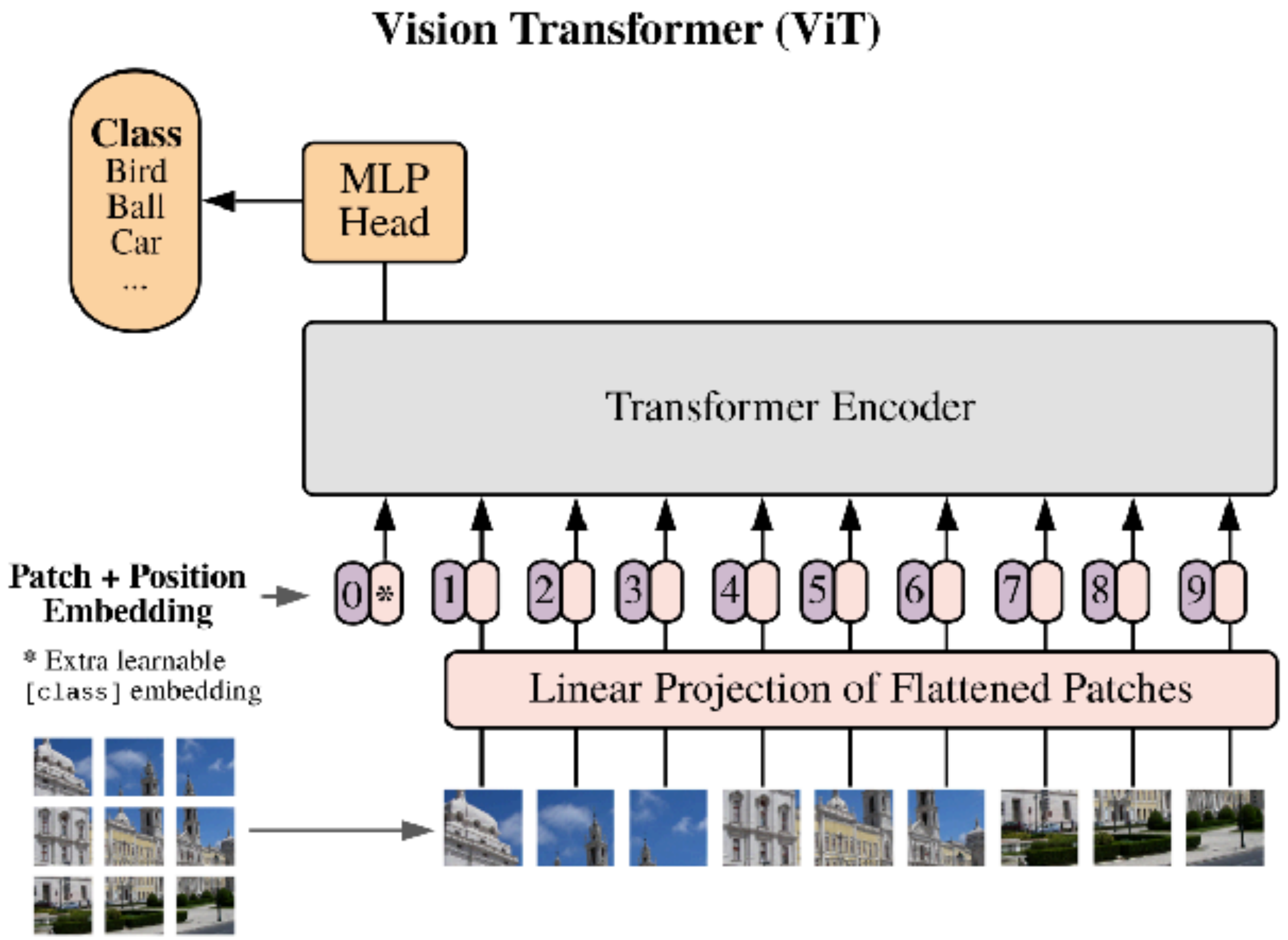
MSA: Global

https://daebaq27.tistory.com/108

2. Method

Hybrid Architecture

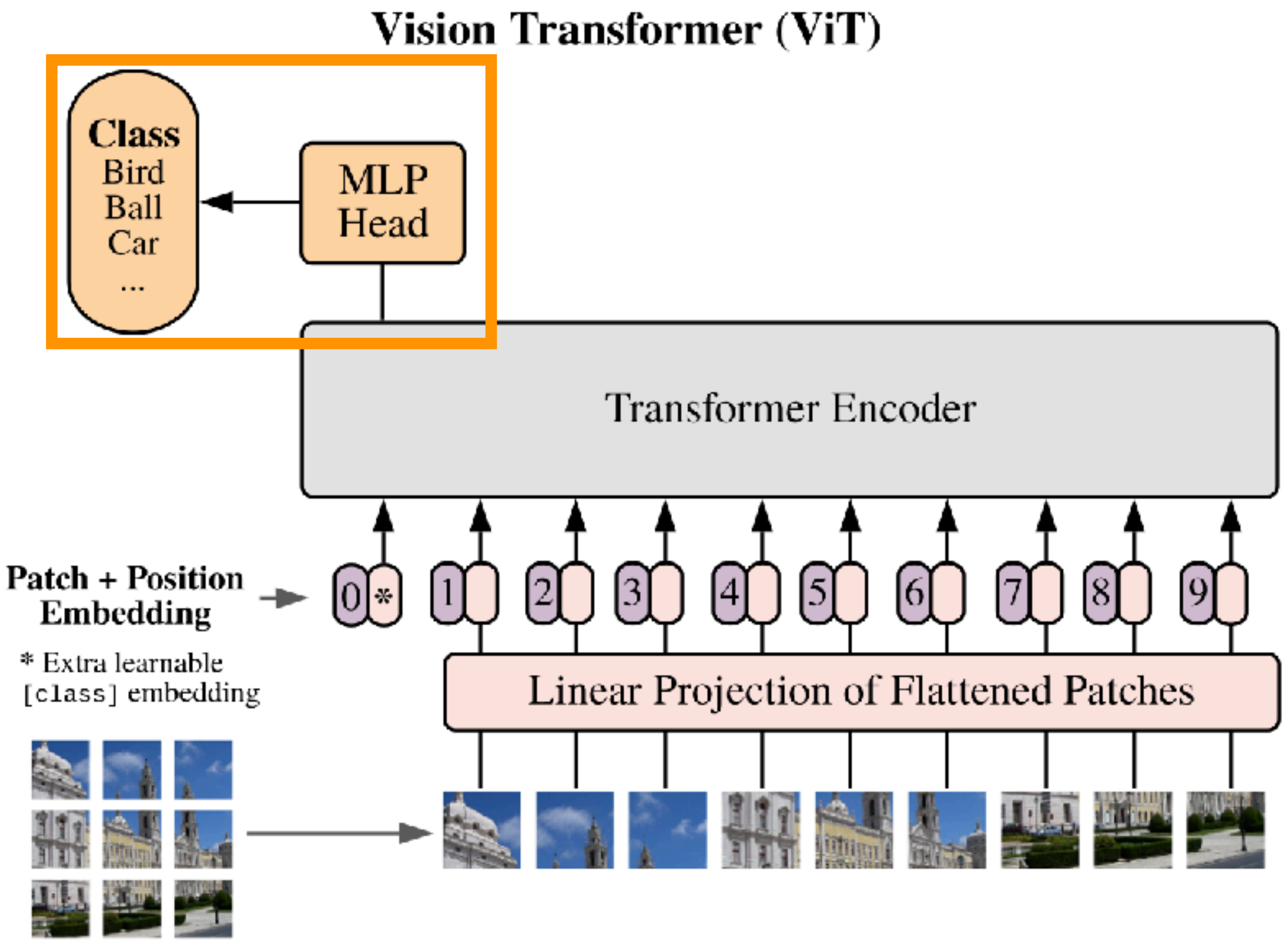




https://daebaq27.tistory.com/108

2. Method

Fine-Tuning and Higher Resolution



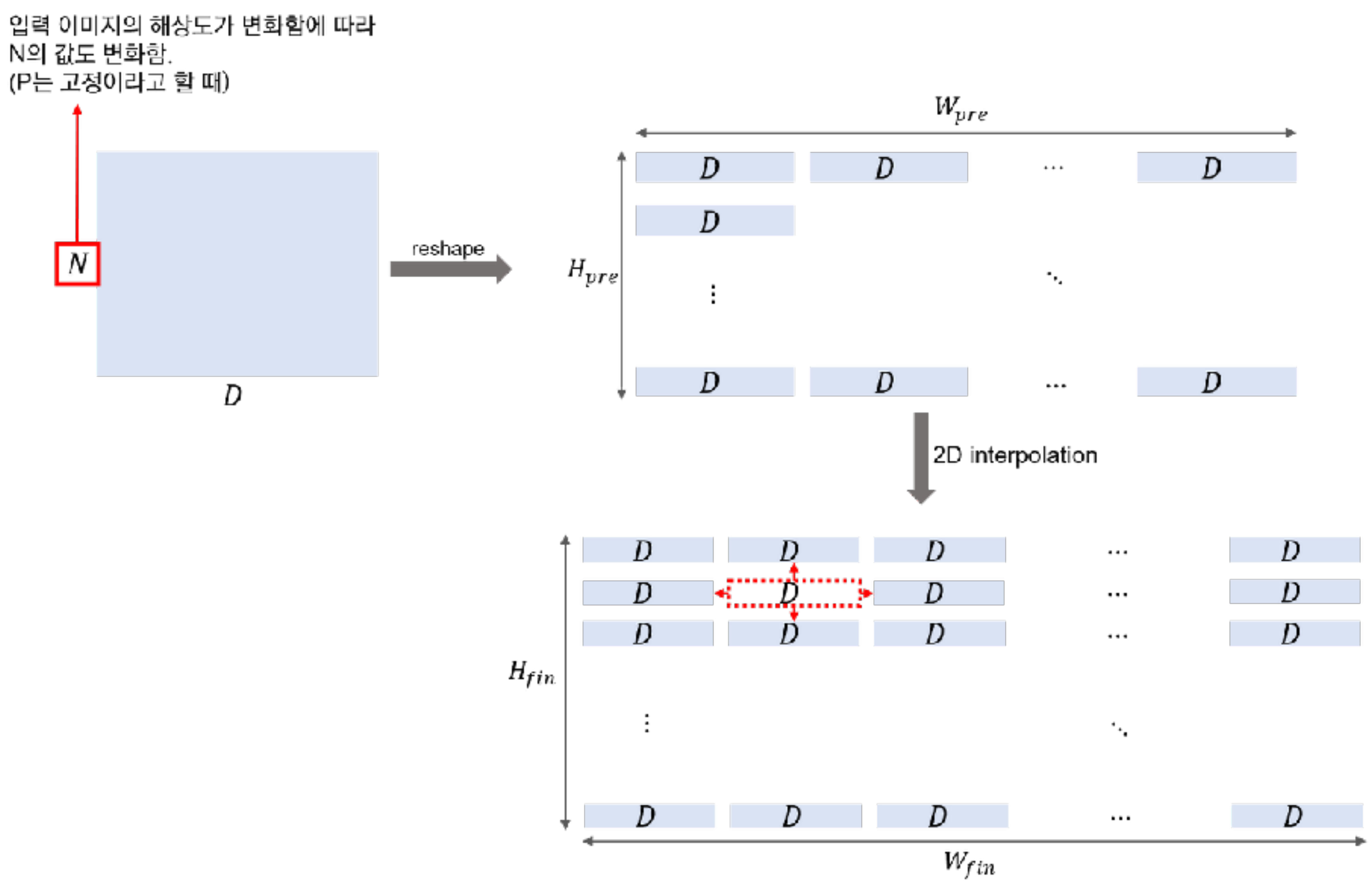
Pre-Training: MLP with One Hidden Layer

Fine-Tuning: Single Linear Layer(DxK)

https://daebaq27.tistory.com/108

2. Method

Fine-Turning and Higher Resolution



Pre-Training 과정에서보다 높은 Resolution 이미지로 Fine-Tuning 하는 것이 효과적

입력 데이터의 크기가 달라짐에 따라 기존의 Positional Embedding의 의미가 없어짐

- 새로운 입력 이미지의 크기에 맞게 2D Interpolation 수행

https://daebaq27.tistory.com/108

3. Conclusion

Image Recognition Task에 직접적으로 Transformer를 적용한 첫 번째 사례

간단하면서 높은 확장성을 가진 Transformer 구조를 적용하면서 SOTA 성능을 이룸

이미지에 특정된 Inductive Bias를 아키텍처에 적용하지 않음 (초기의 Patch 분할 과정 제외)

- 대신, 이미지를 일련의 Patch로서 NLP에서 사용하는 것처럼 Transformer에서 활용함
- Inductive Bias가 CNN-Like 아키텍처에 비해 약하기 때문에 Generalization 성능이 높음
- Inductive Bias가 낮음에 따라 대용량 Dataset을 활용