AN IMAGE IS WORTH 16X16 WORDS: TRANSFORMERS FOR IMAGE RECOGNITION AT SCALE

SW융합대학 소프트웨어학과 김진호

목차

1) Image Classification 모델 구조

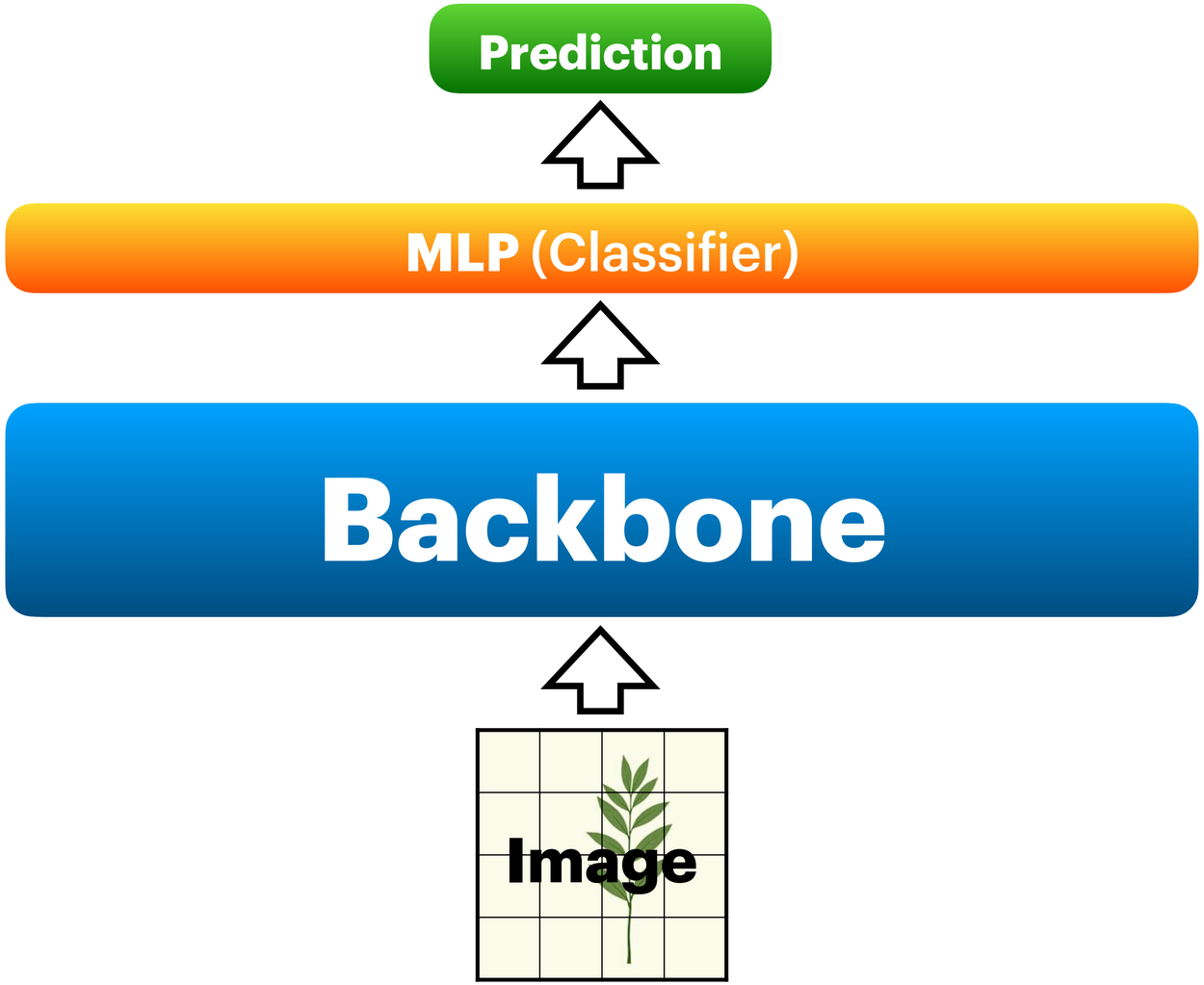
Rank	Model	Top 1 Accuracy	Number of params	GFLOPs	energy consumption	Extra Training Data	Paper	Code	Result	Year
1	**OmniVec** (ViT)	92.4%				×	OmniVec: Learning robust representations with cross modal sharing			2023
2	**CoCa** (finetuned)	91.0%	2100M			×	CoCa: Contrastive Captioners are Image-Text Foundation Models			2022
3	**Model soups** (BASIC-L)	90.98%	2440M			×	Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time			2022
4	**Model soups** (ViT-G/14)	90.94%	1843M			×	Model soups: averaging weights of multiple fine-tuned models improves			2022

1) Image Classification 모델 구조

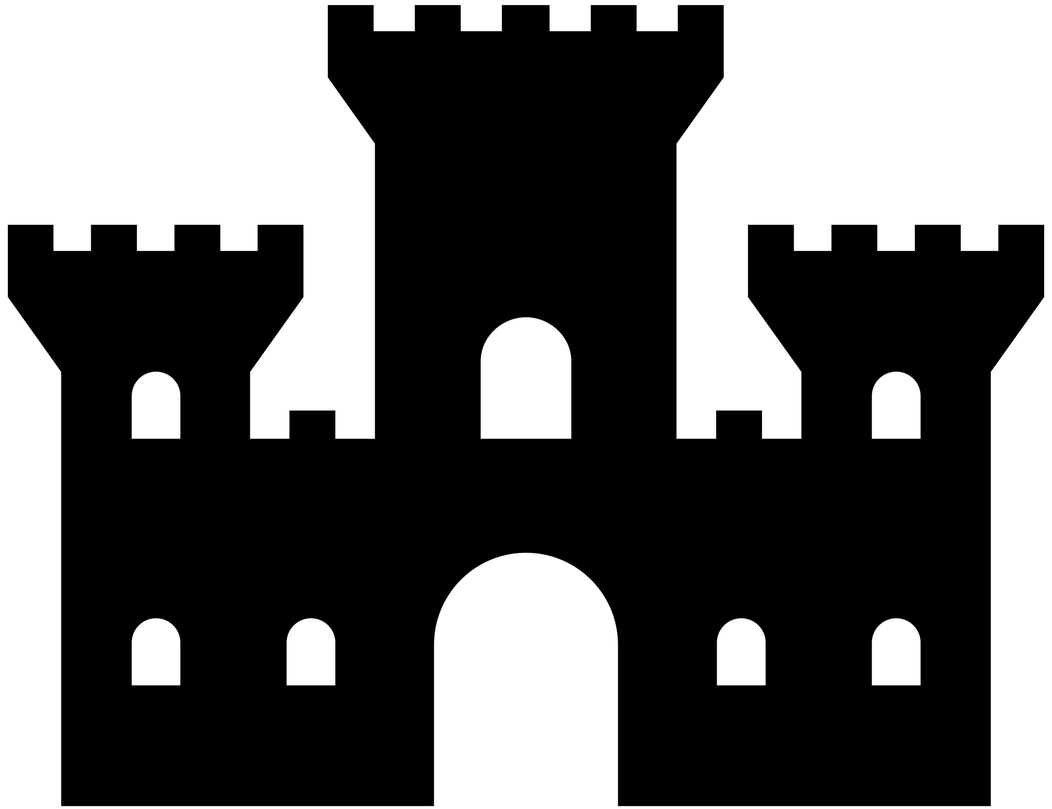
Rank	Model	Top 1 Accuracy	Number of params	GFLOPs	energy consumption	Extra Training Data	Paper	Code	Result	Year
1	**OmniVec** (ViT)	92.4%				×	OmniVec: Learning robust representations with cross-modal sharing			2023
2	**CoCa** (finetuned)	91.0%	2100M			×	CoCa: Contrastive Captioners are Image-Text Foundation Models			2022
3	**Model soups** (BASIC-L)	90.98%	2440M			×	Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time			2022
4	**Model soups** (ViT-G/14)	90.94%	1843M			×	Model soups: averaging weights of multiple fine-tuned models improves			2022

All of These Are Transformers

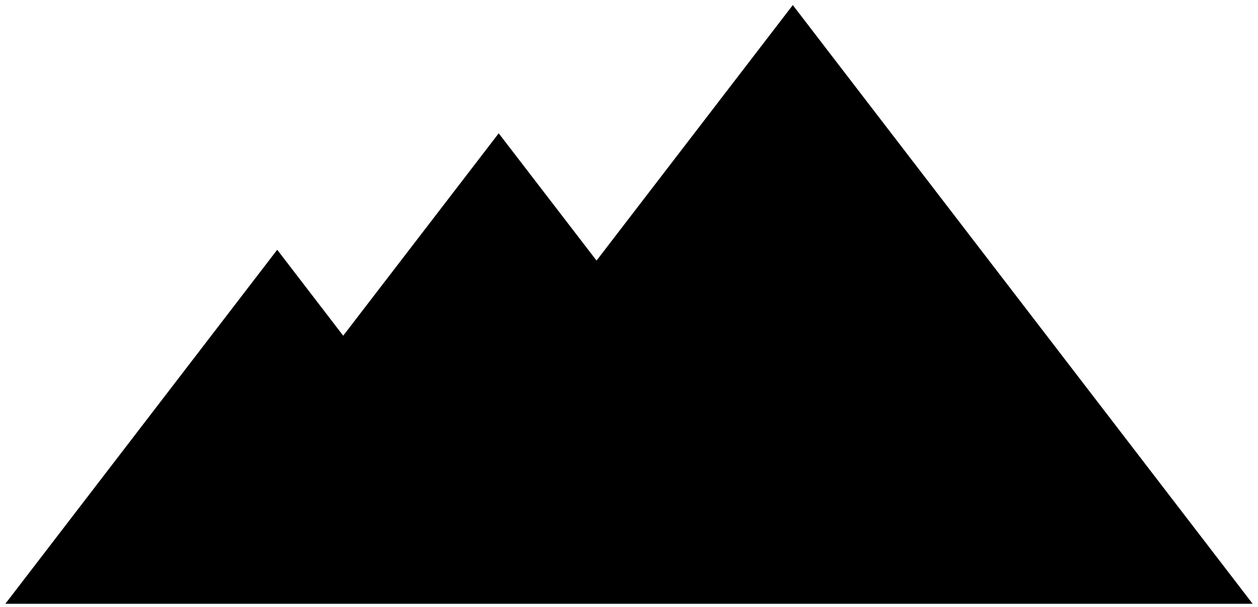
1) Image Classification 모델 구조



1) Image Classification 모델 구조



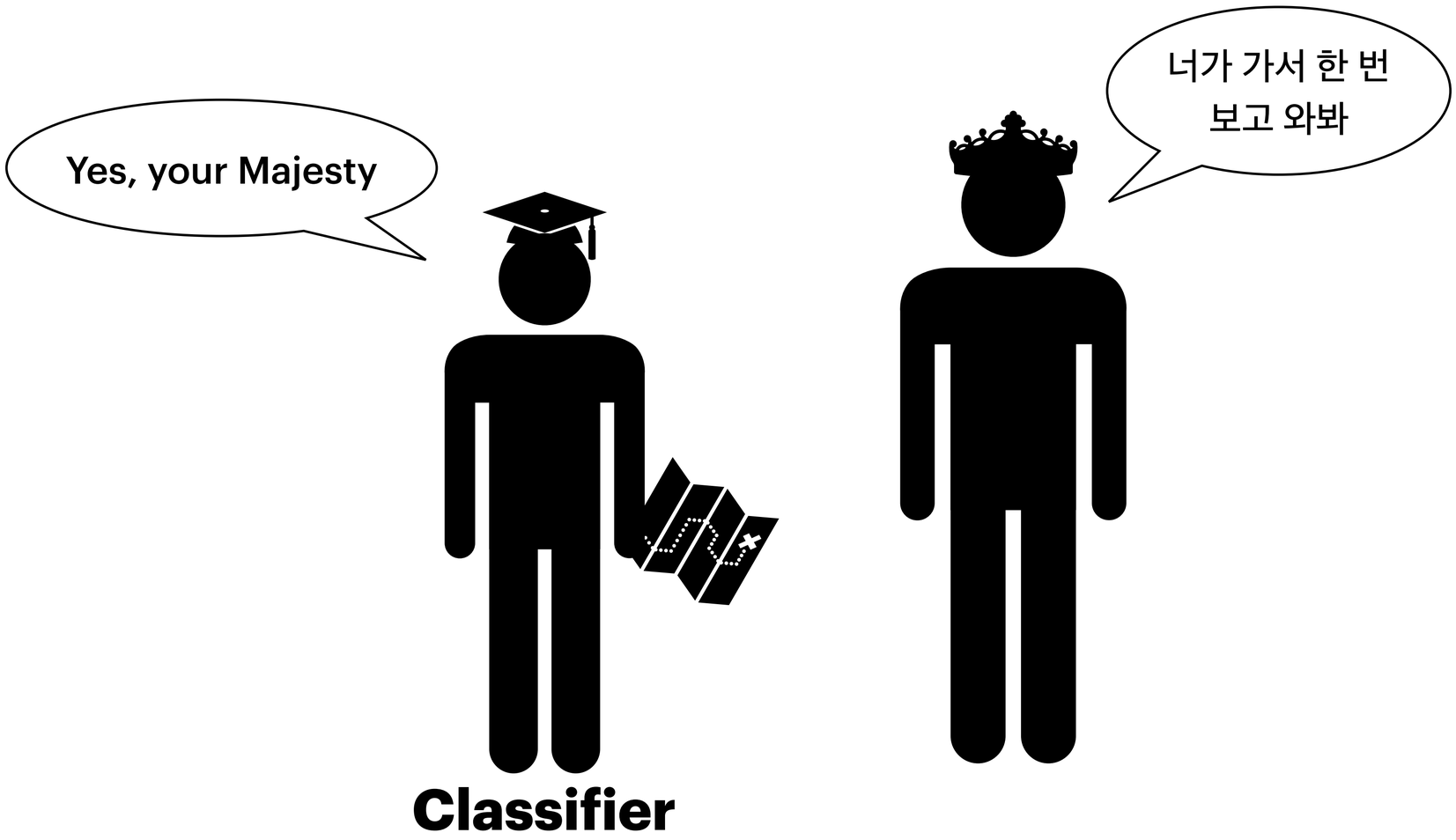
1) Image Classification 모델 구조



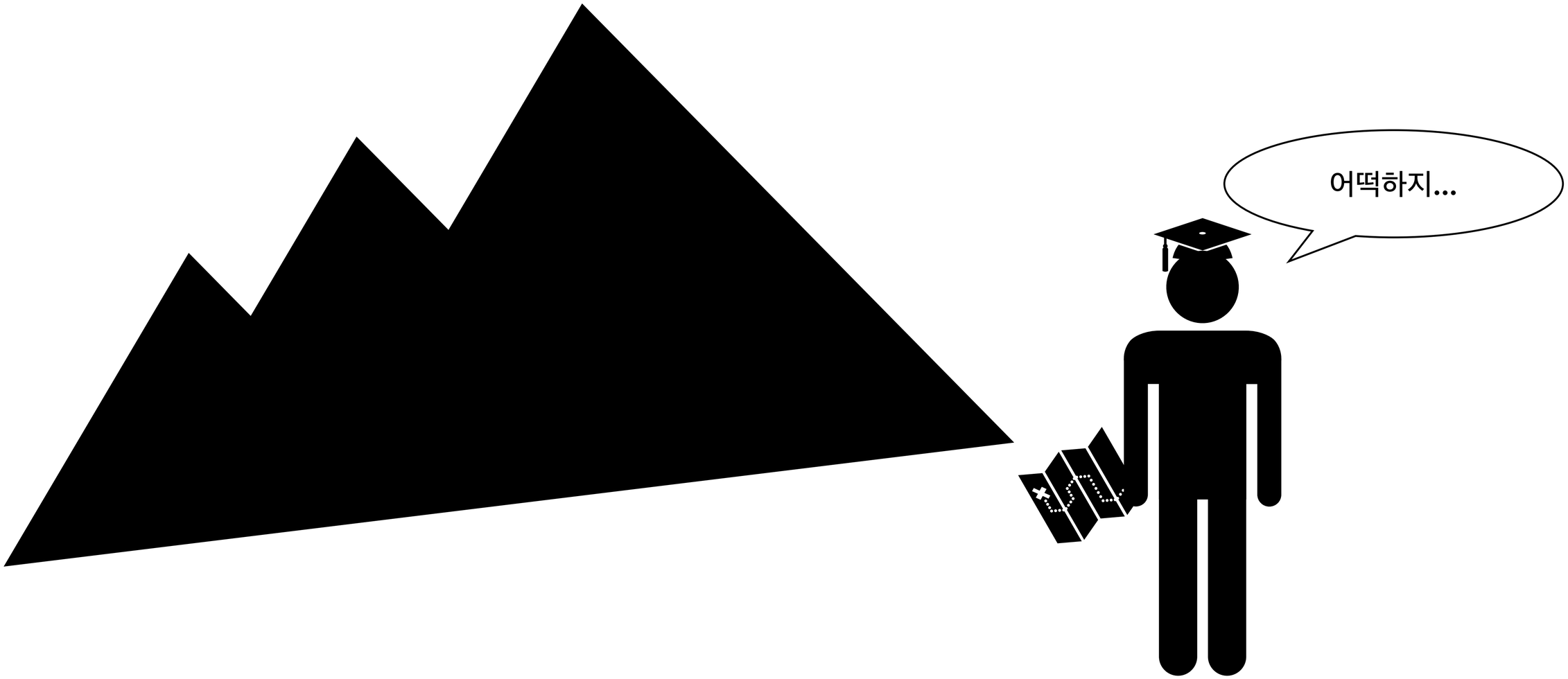
Dankook Mountain

1) Image Classification 모델 구조

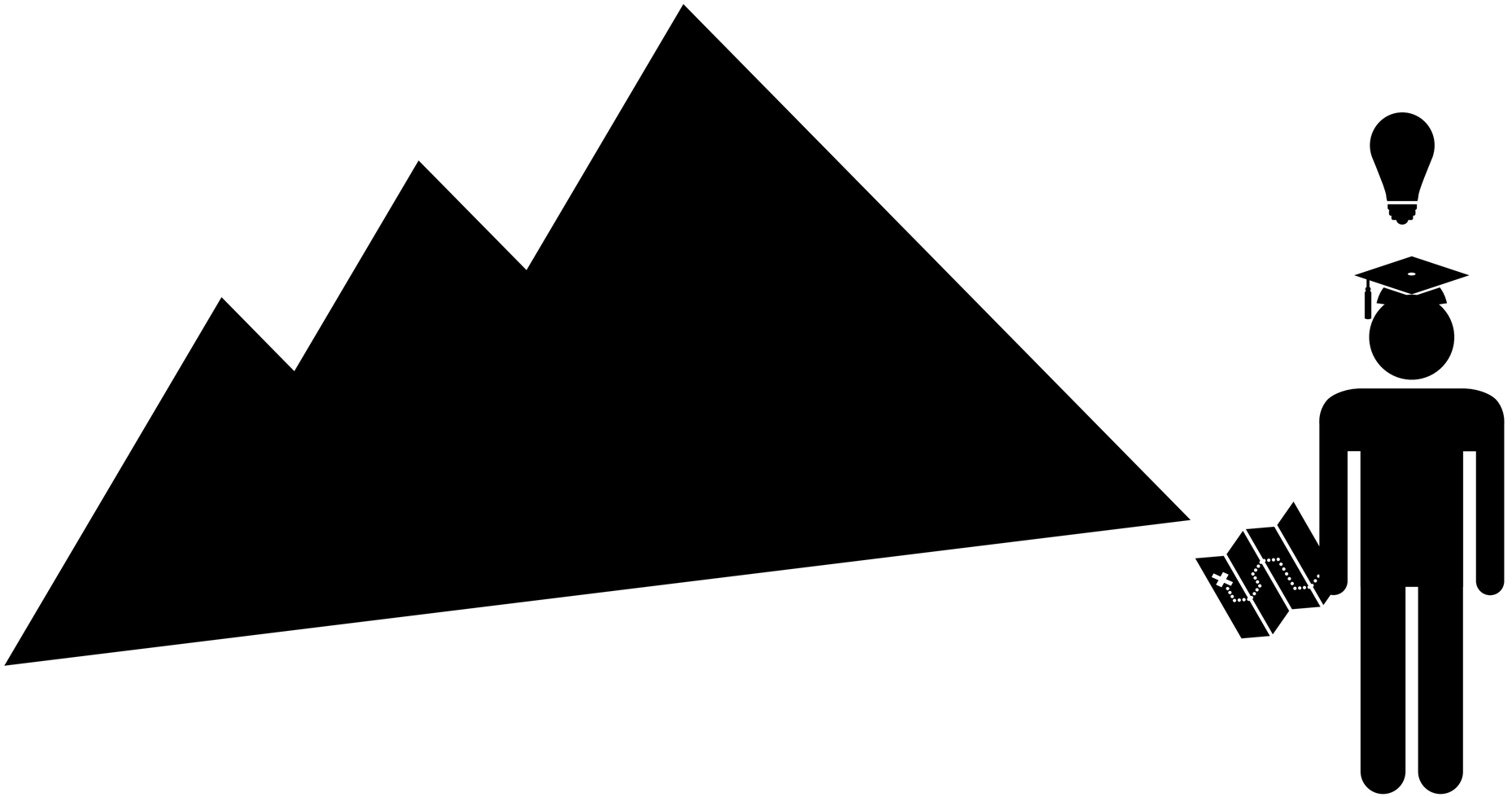
1) Image Classification 모델 구조



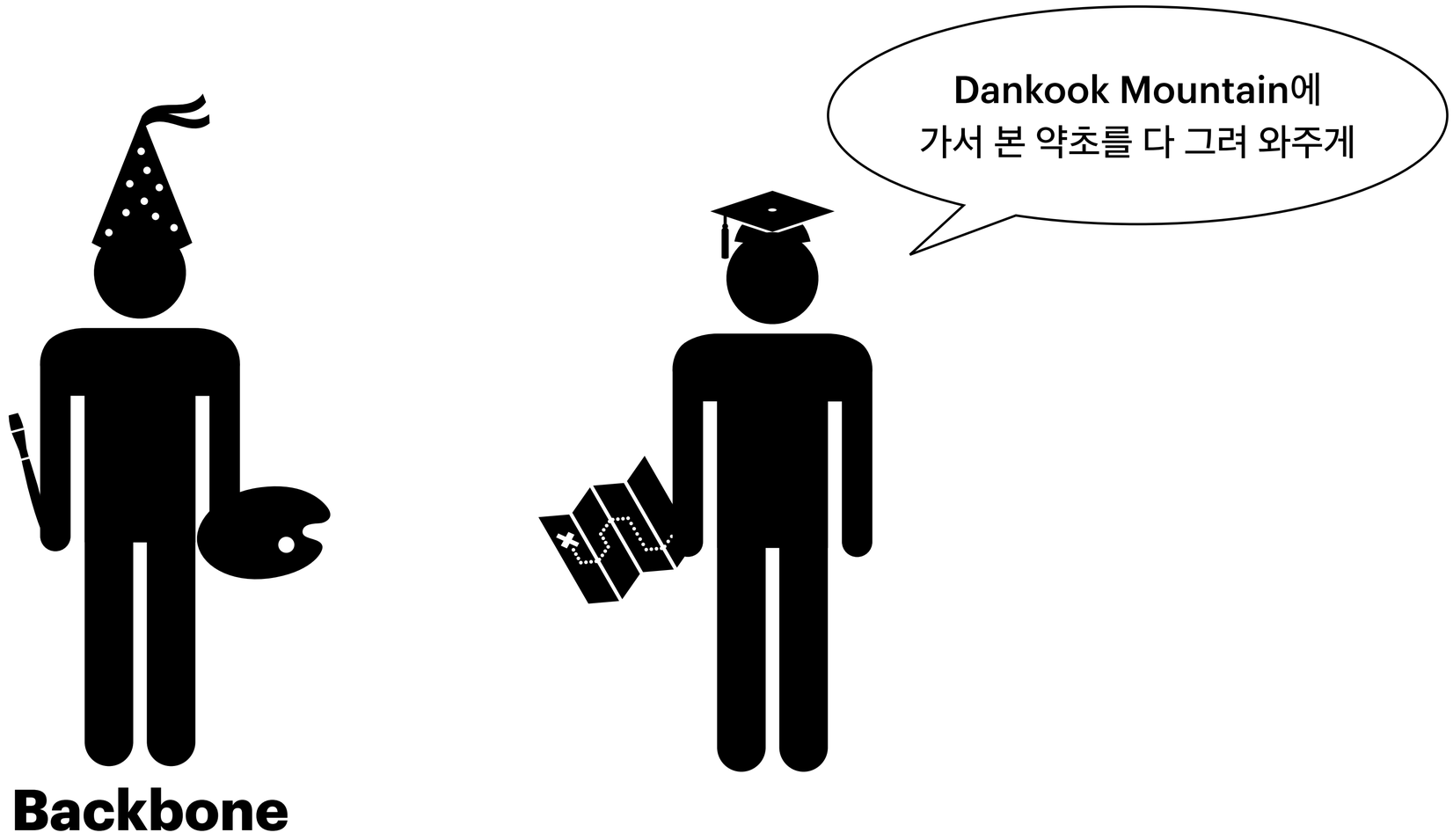
1) Image Classification 모델 구조



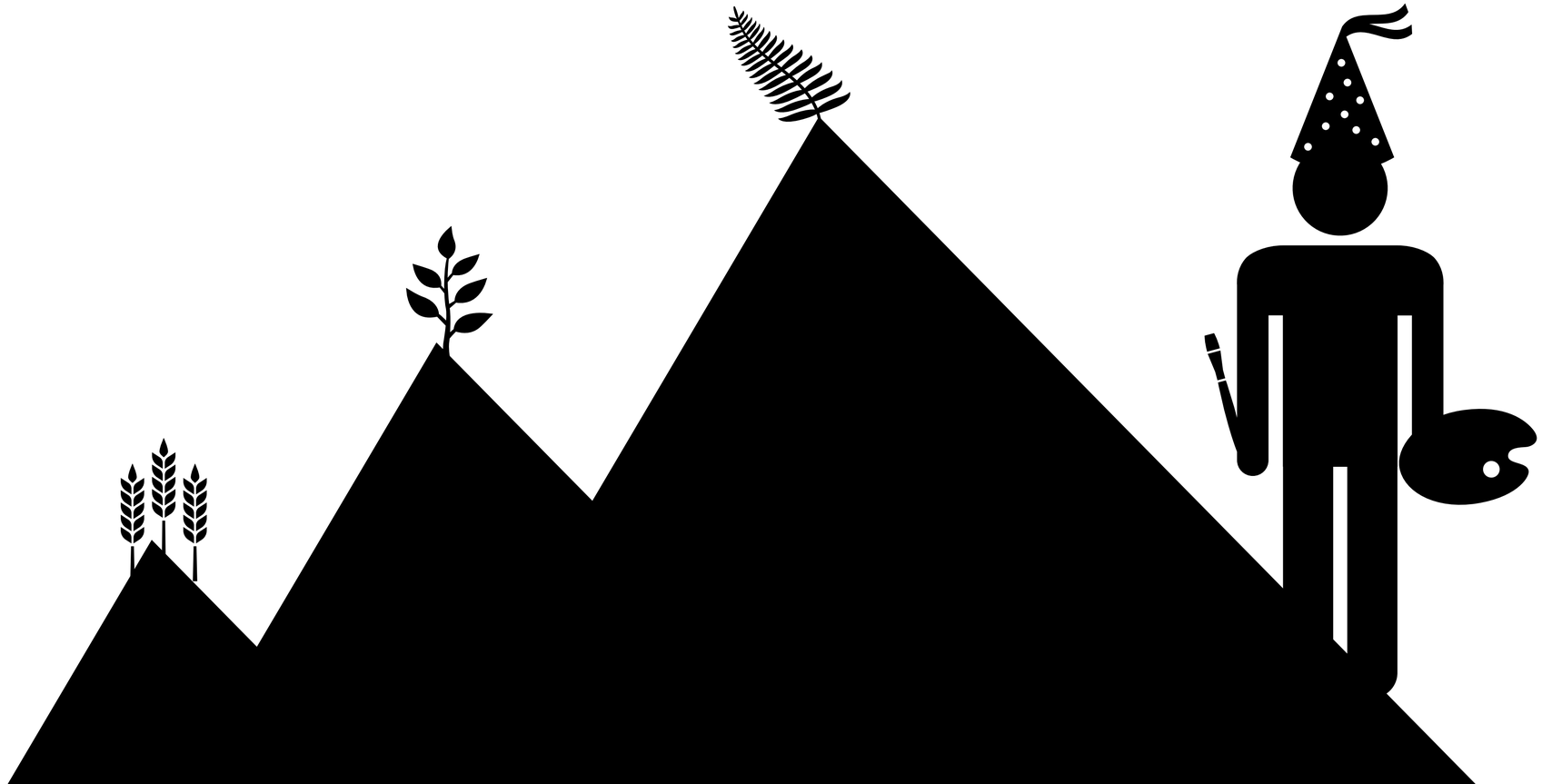
1) Image Classification 모델 구조



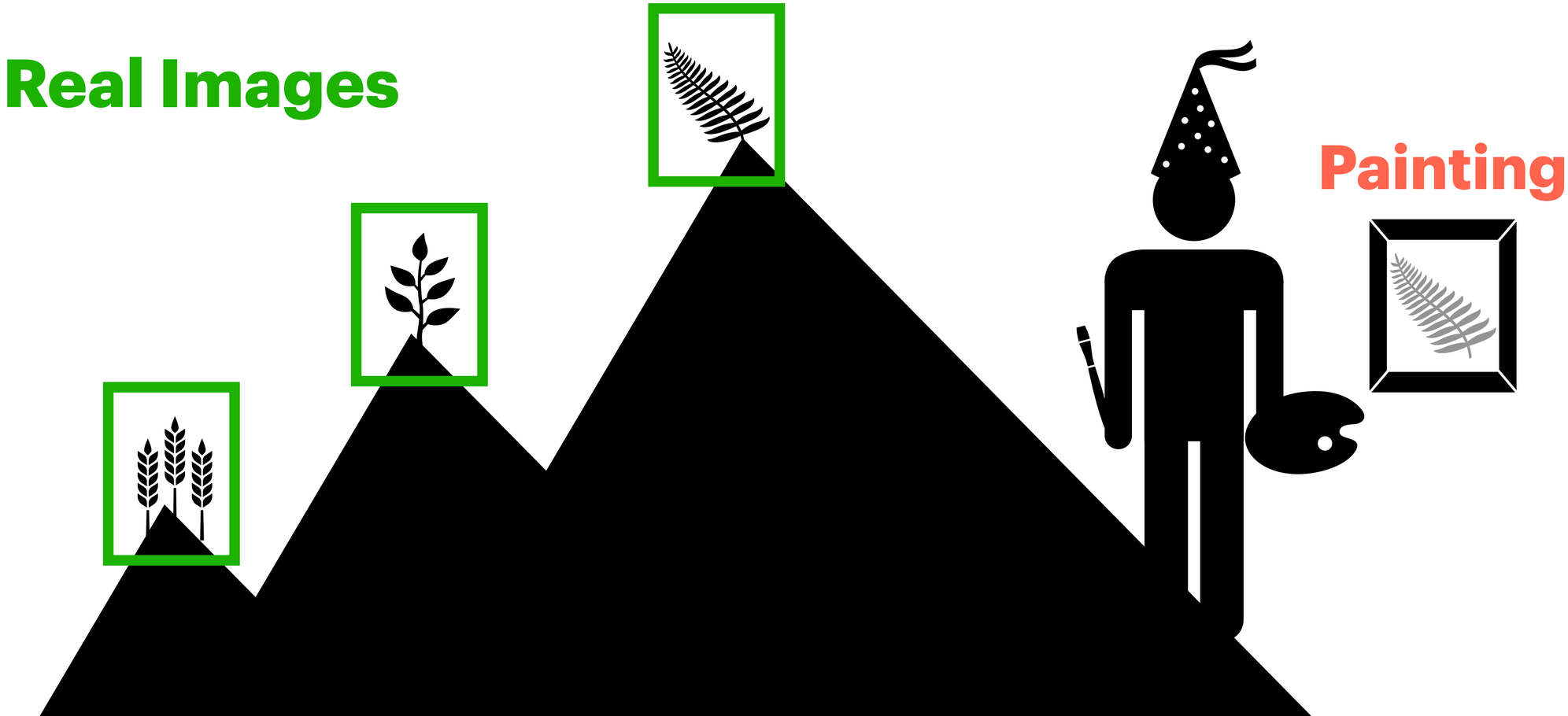
1) Image Classification 모델 구조



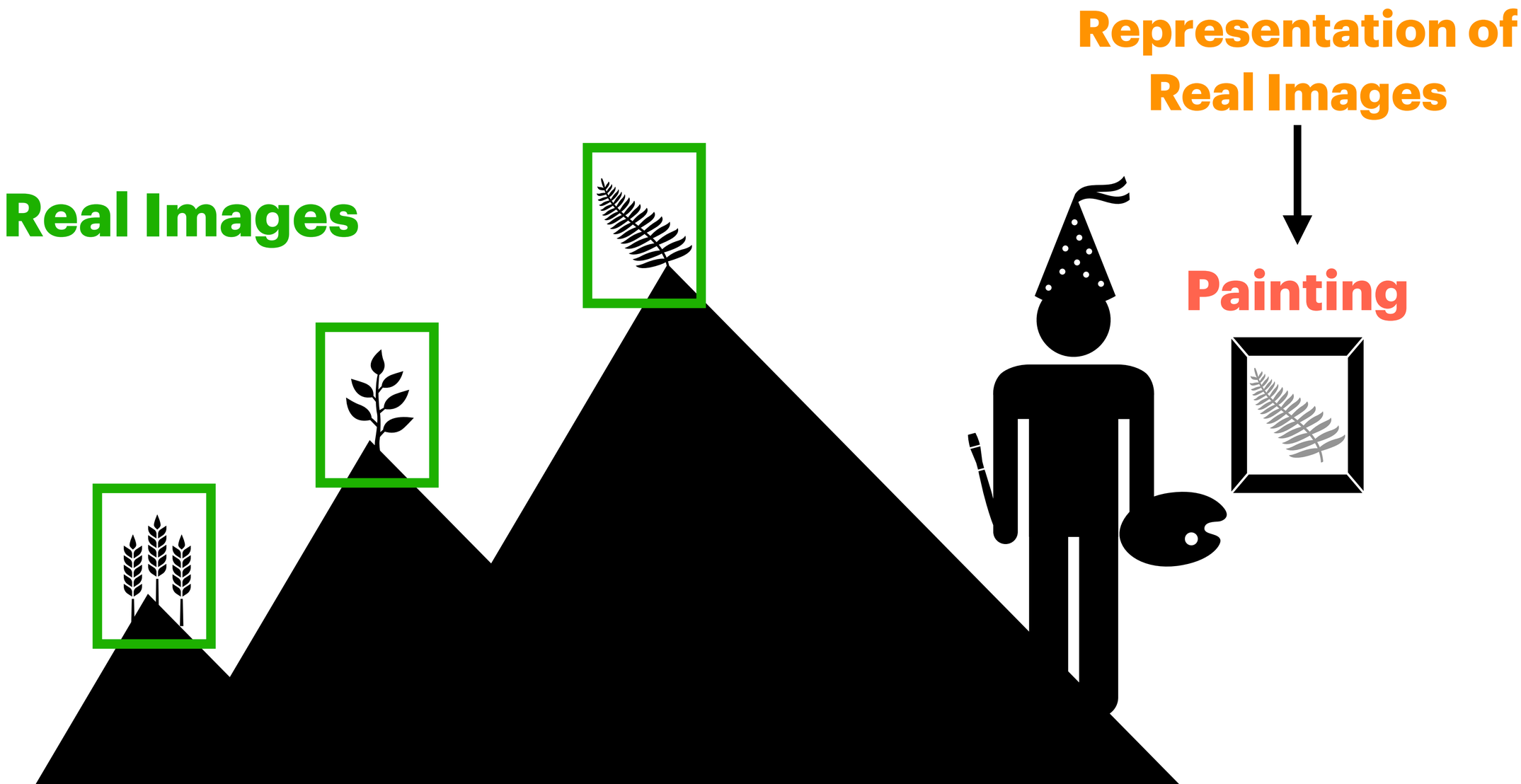
1) Image Classification 모델 구조



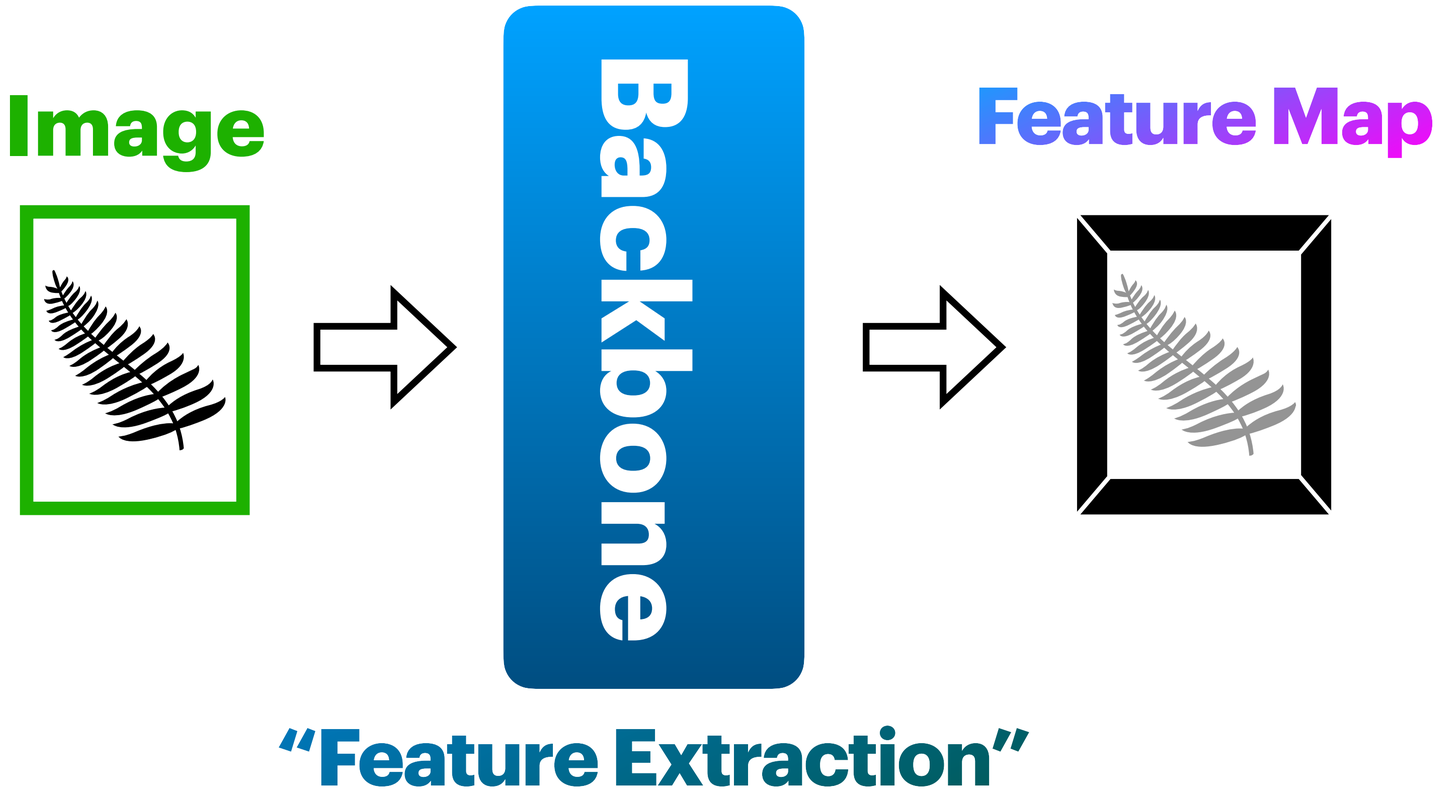
1) Image Classification 모델 구조



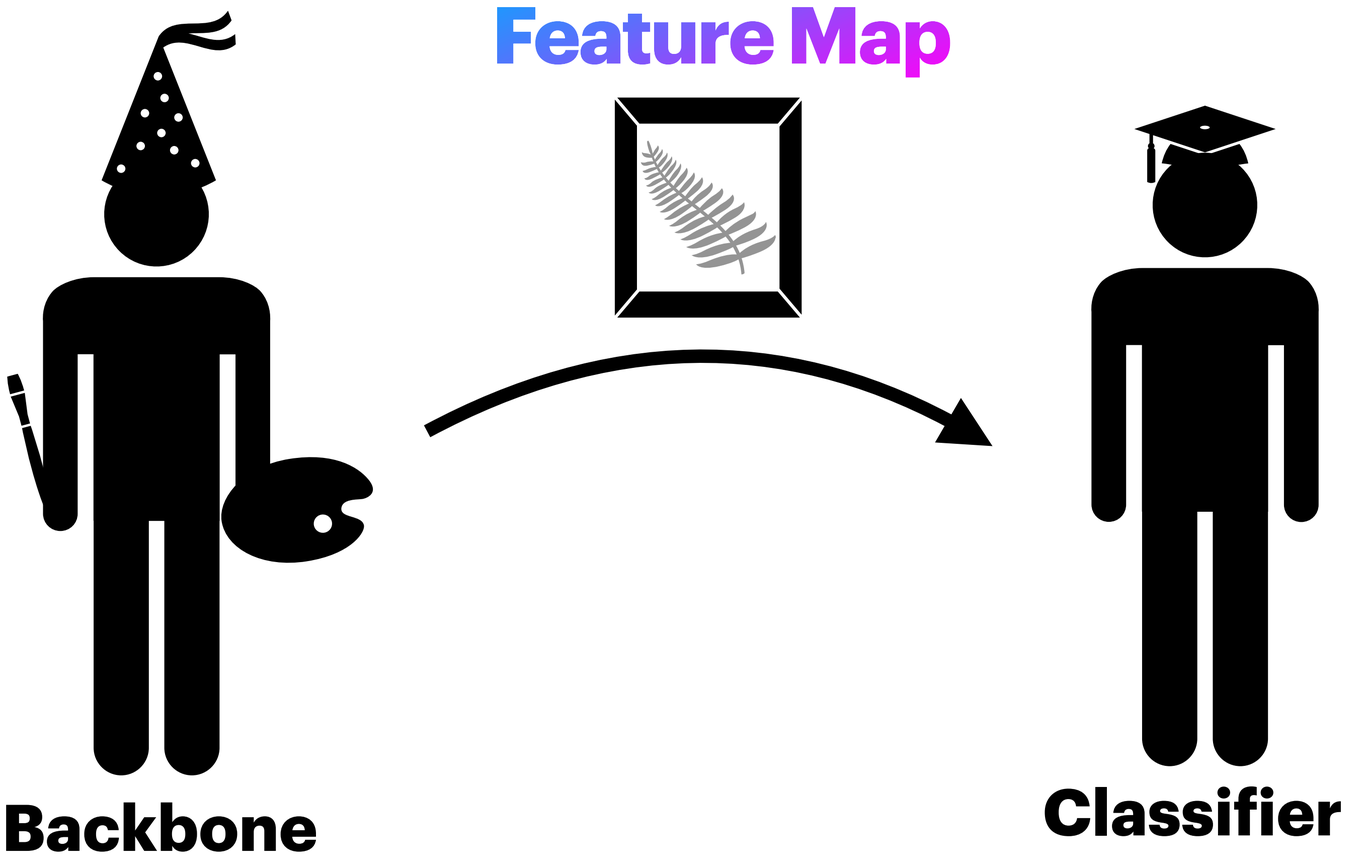
Vision Transformer (ViT) 란?



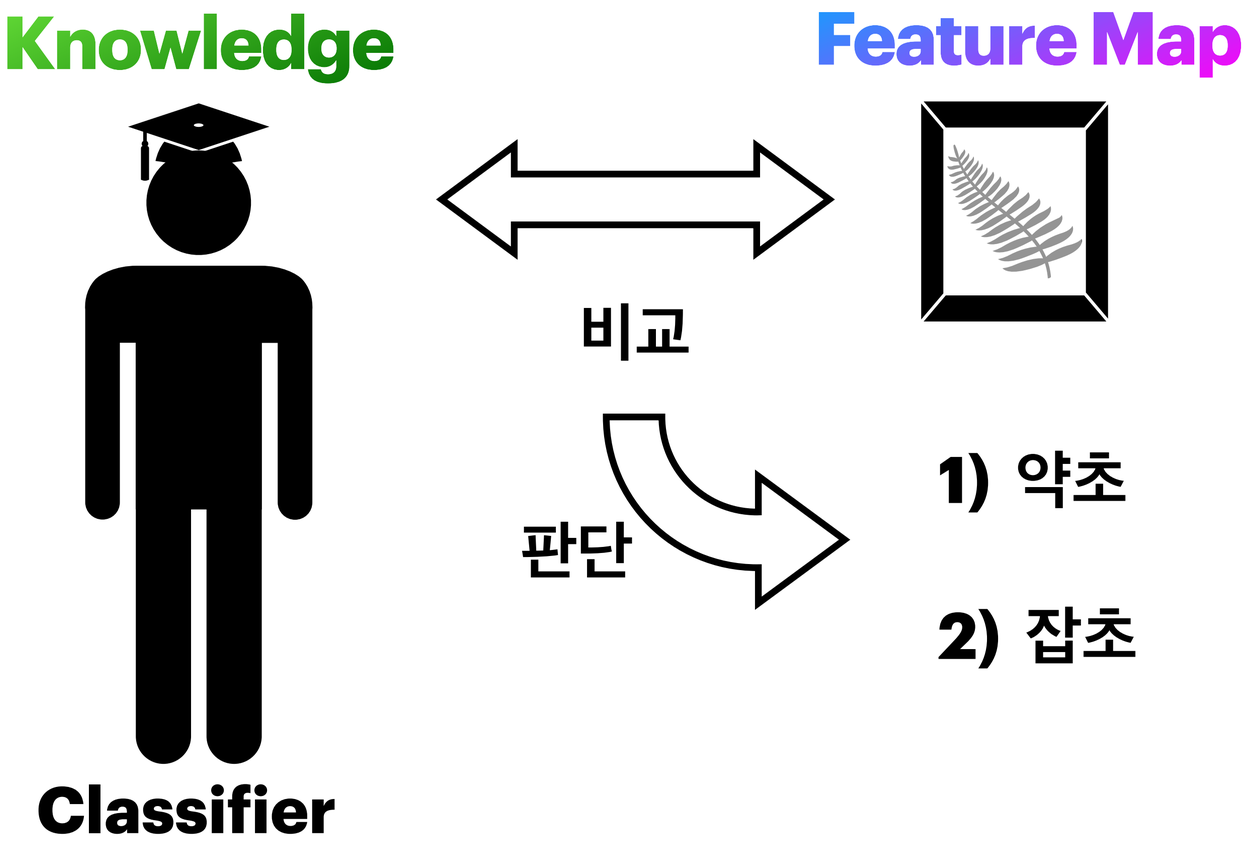
1) Image Classification 모델 구조



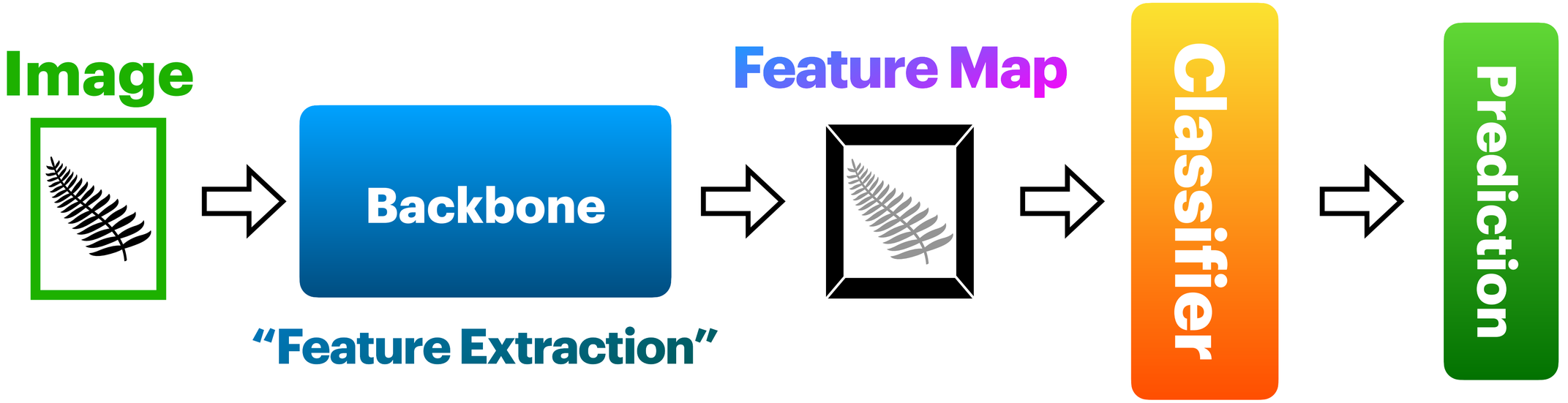
1) Image Classification 모델 구조



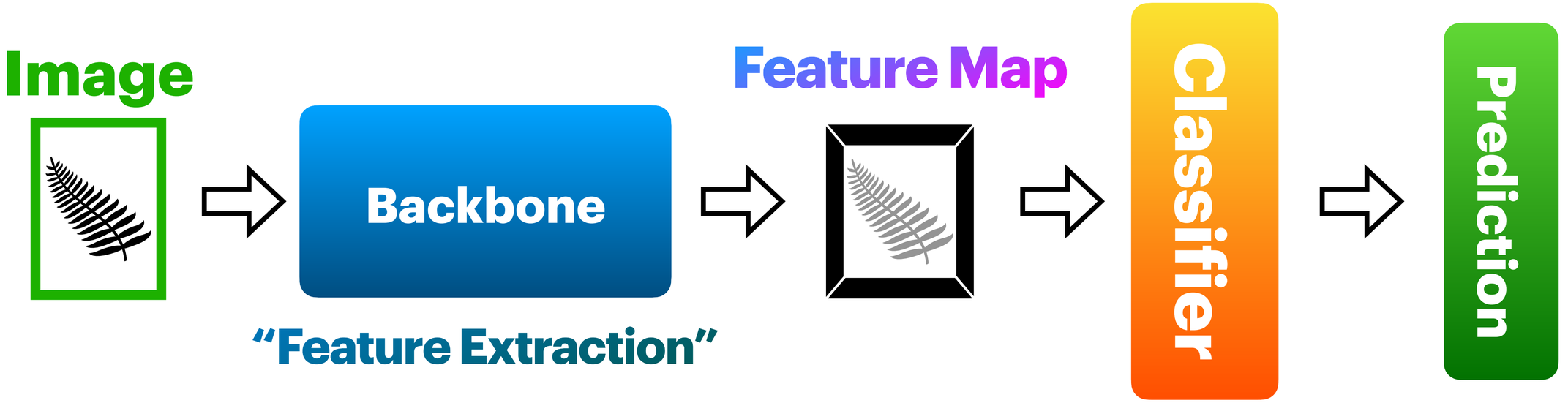
1) Image Classification 모델 구조



1) Image Classification 모델 구조

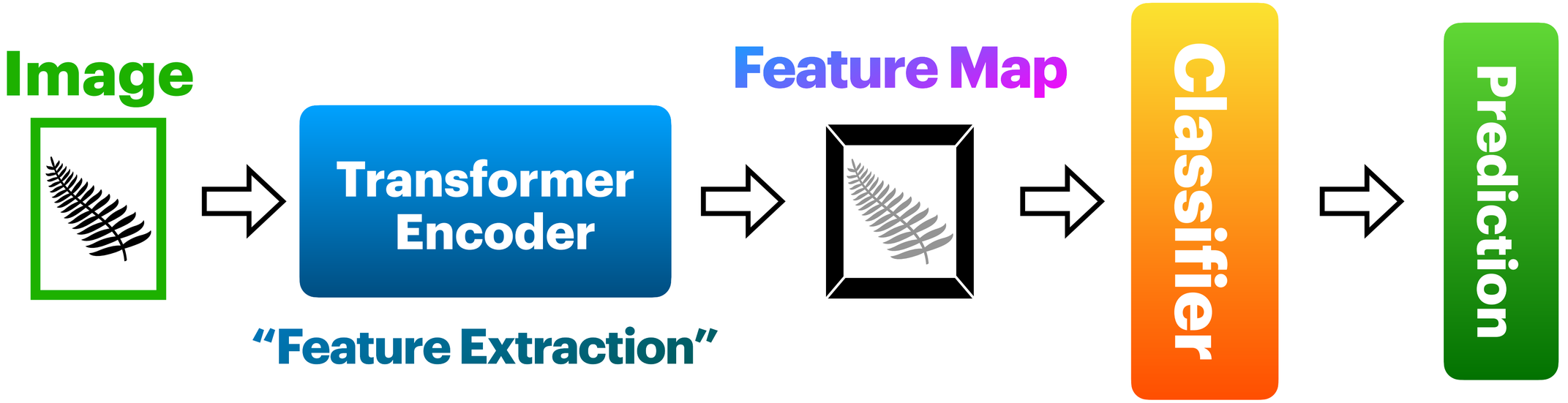


1) Image Classification 모델 구조



그림을 잘 그려줘야 합니다

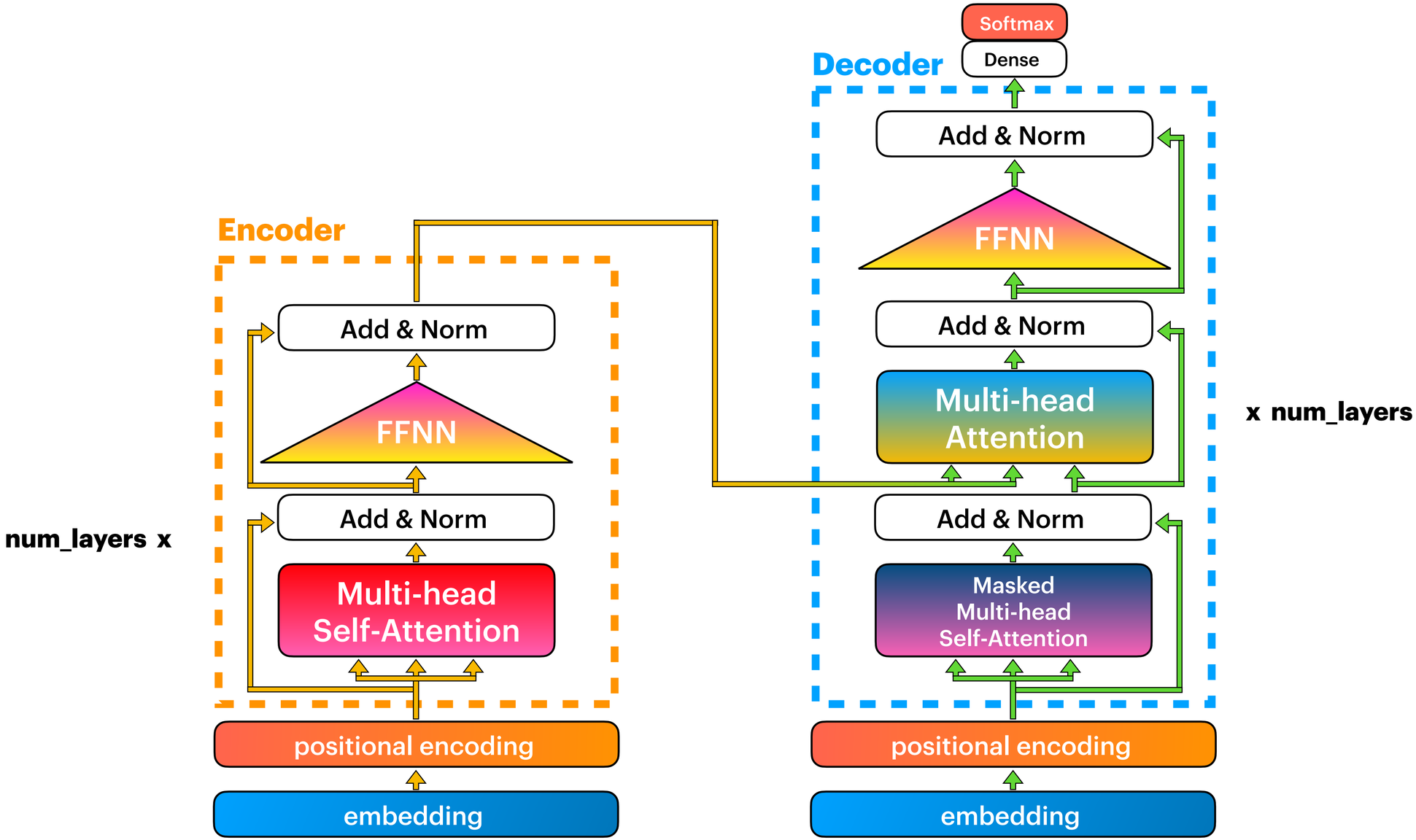
1) Image Classification 모델 구조



Transformer의 Encoder가 정말 잘 그려줍니다

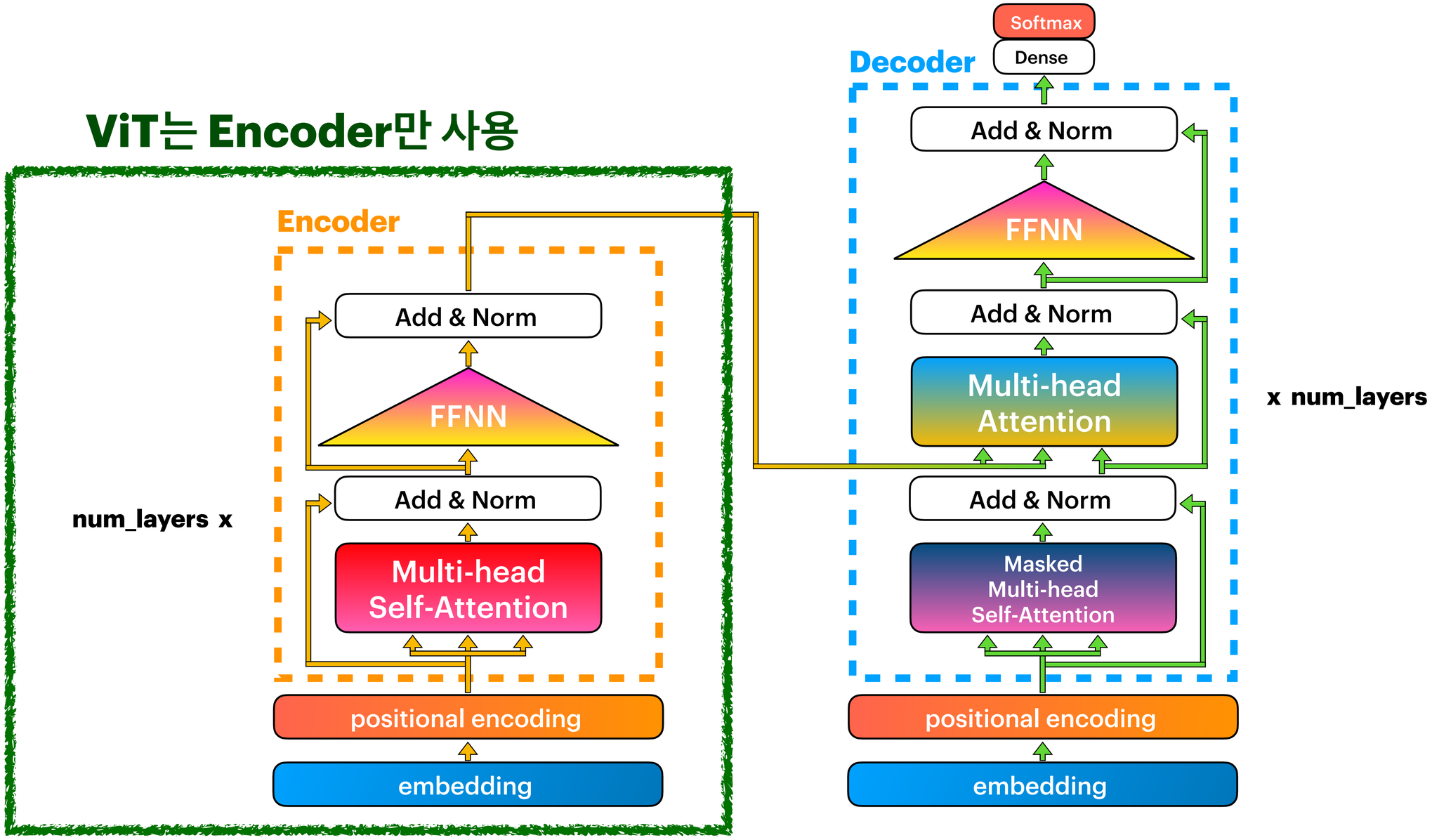
2) ViT 모델 구조

Architecture Overview: Transformer



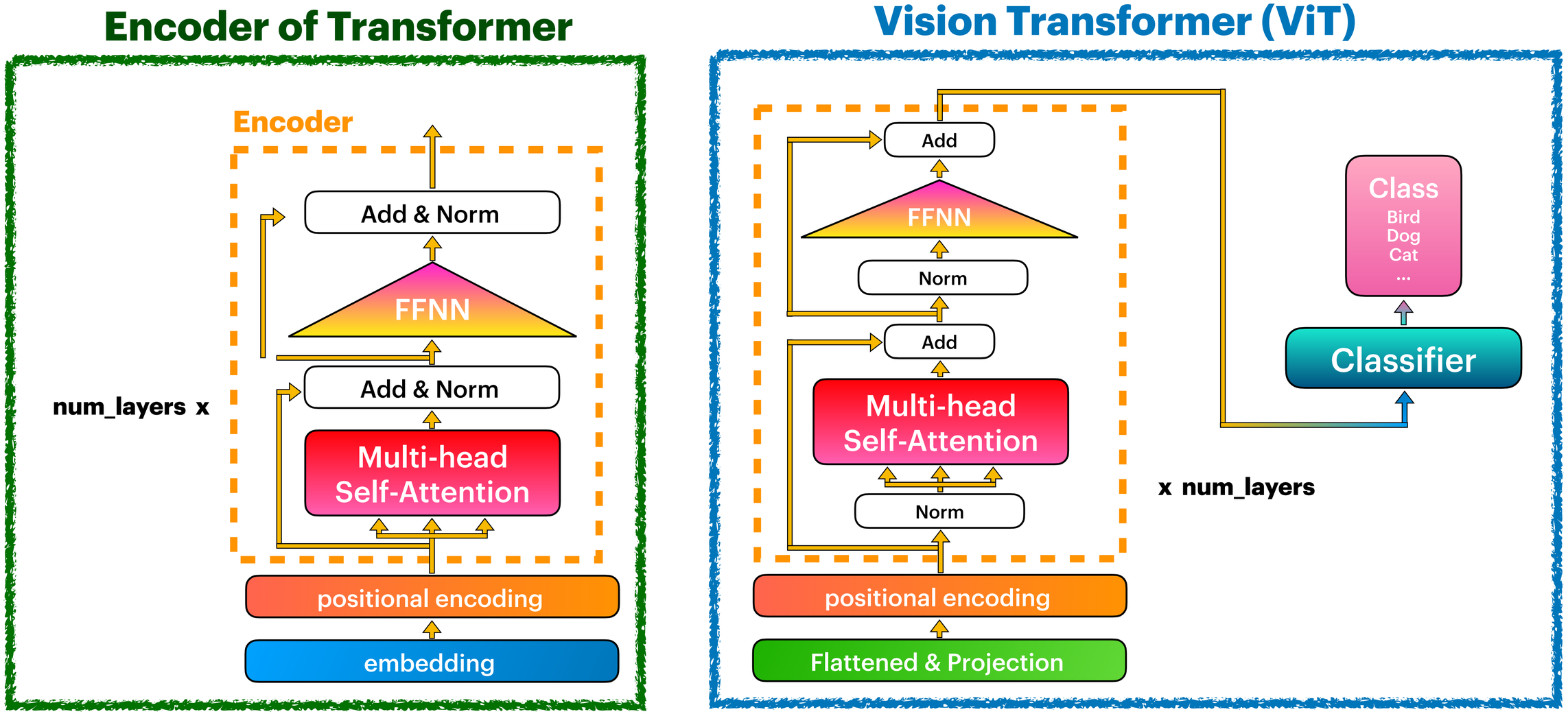
2) ViT 모델 구조

Architecture Overview: Transformer



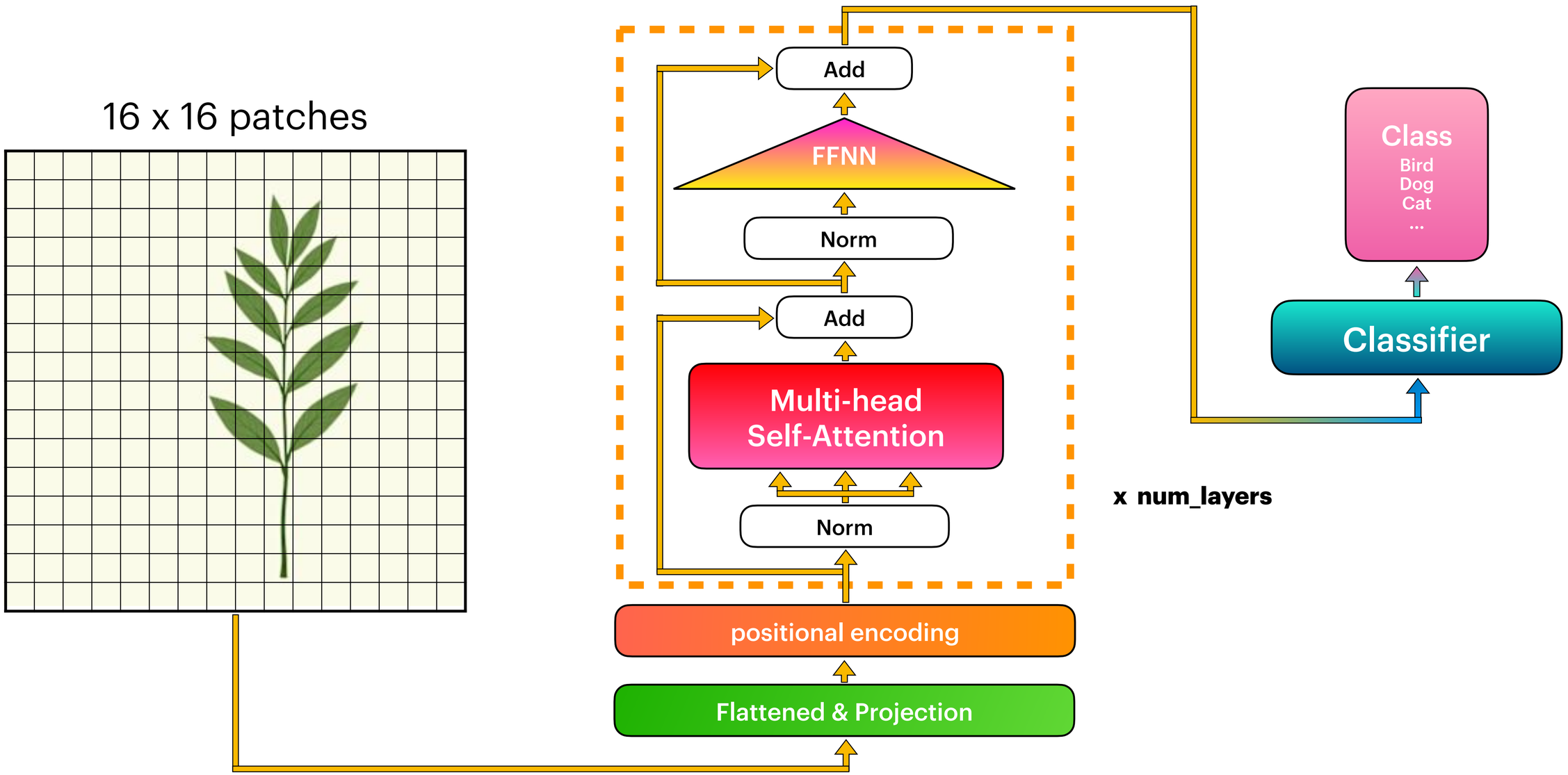
2) ViT 모델 구조

Architecture Overview: ViT



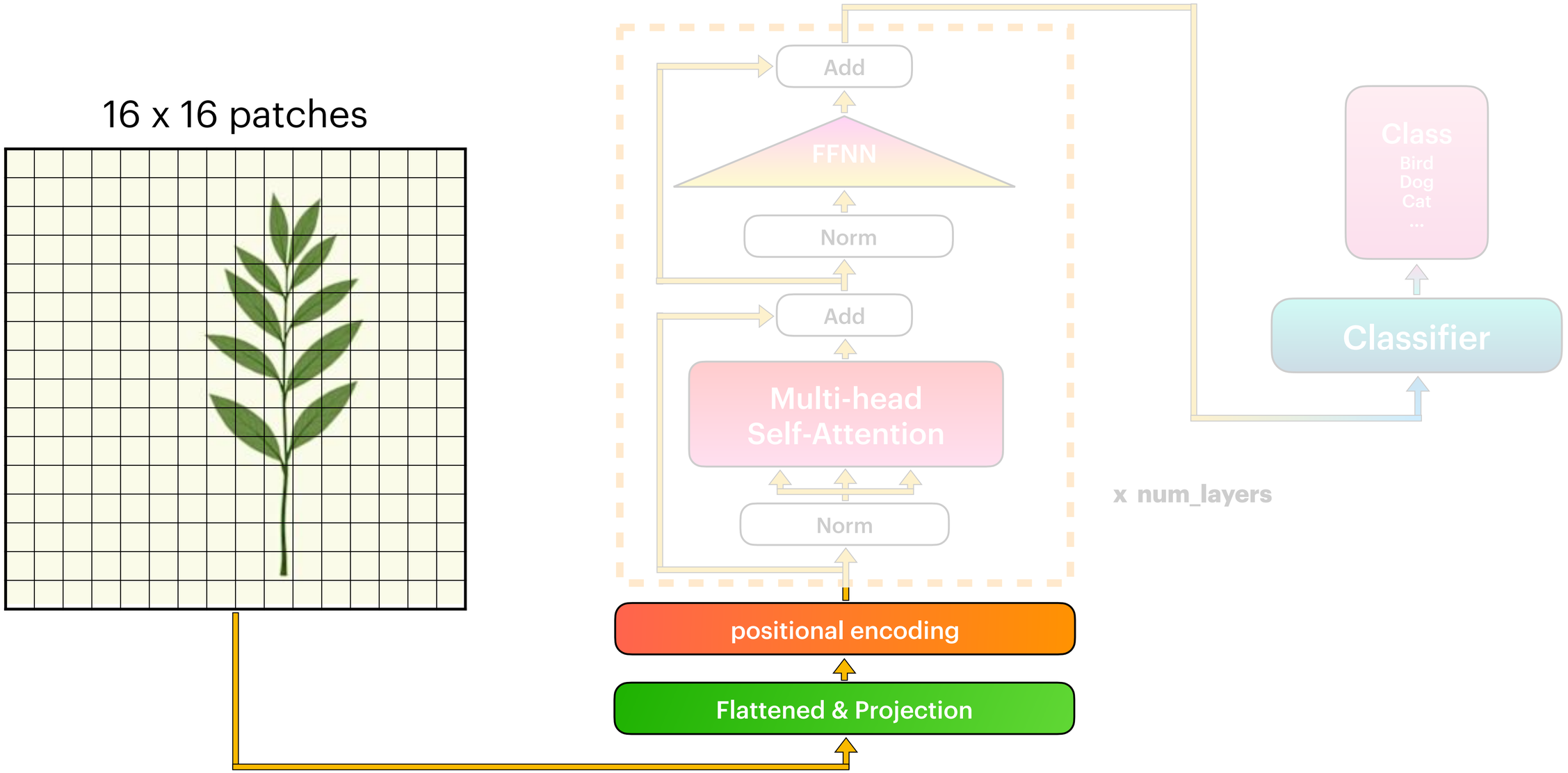
2) ViT 모델 구조

Architecture Overview: ViT



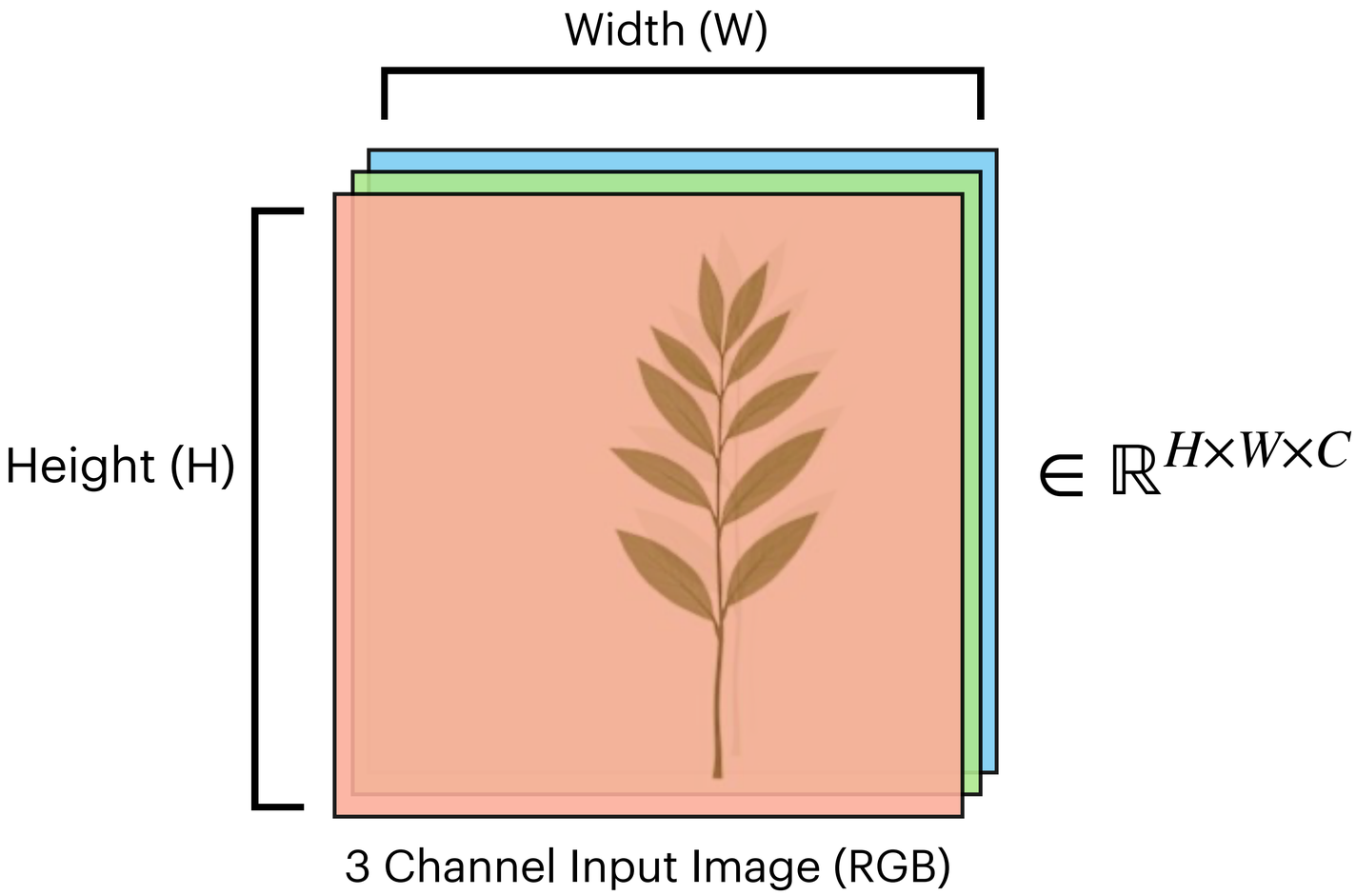
2) ViT 모델 구조

Architecture Detail: Patch Embedding



2) ViT 모델 구조

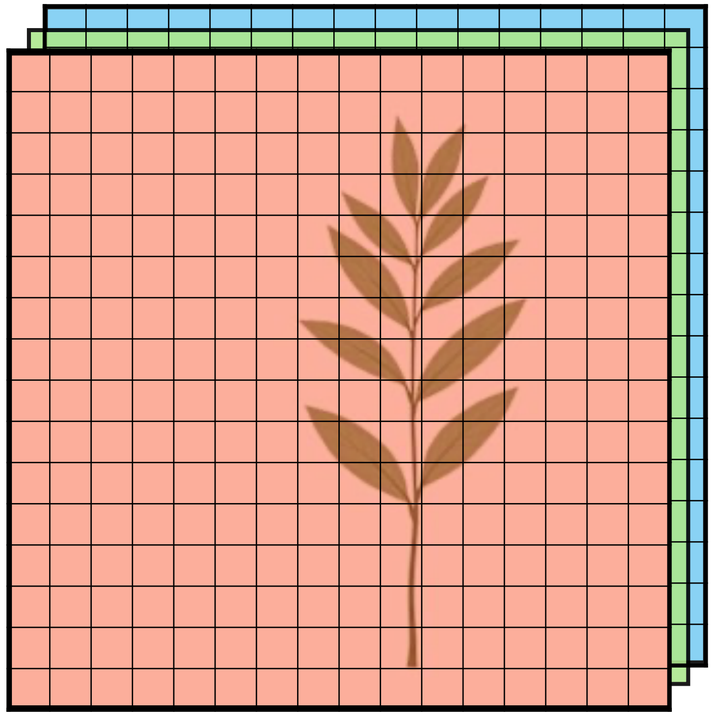
Architecture Detail: Patch Embedding



2) ViT 모델 구조

Architecture Detail: Patch Embedding

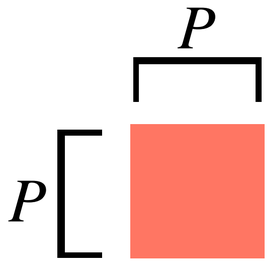
(1) Divide into 16 x 16 patches



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

Resolution of Each Patch

$= P \times P$

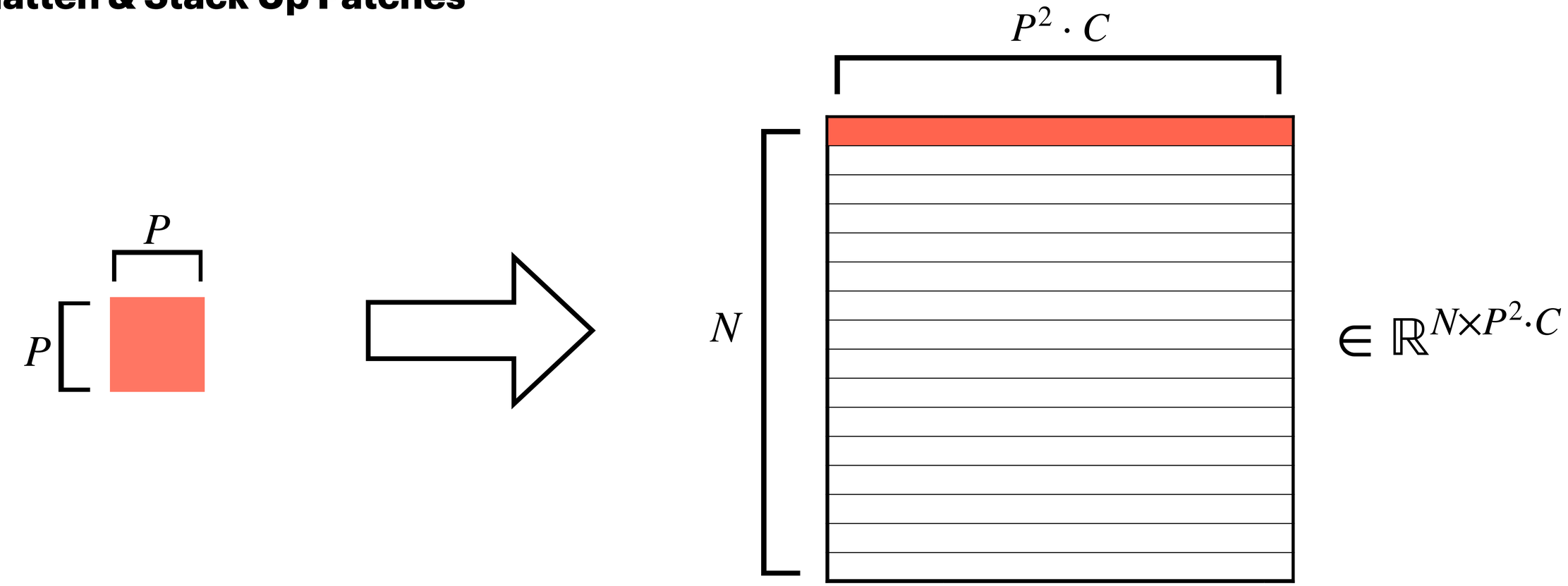


# of Patches: $N = \frac{H \times W}{P^2}$

2) ViT 모델 구조

Architecture Detail: Patch Embedding

(2) Flatten & Stack Up Patches

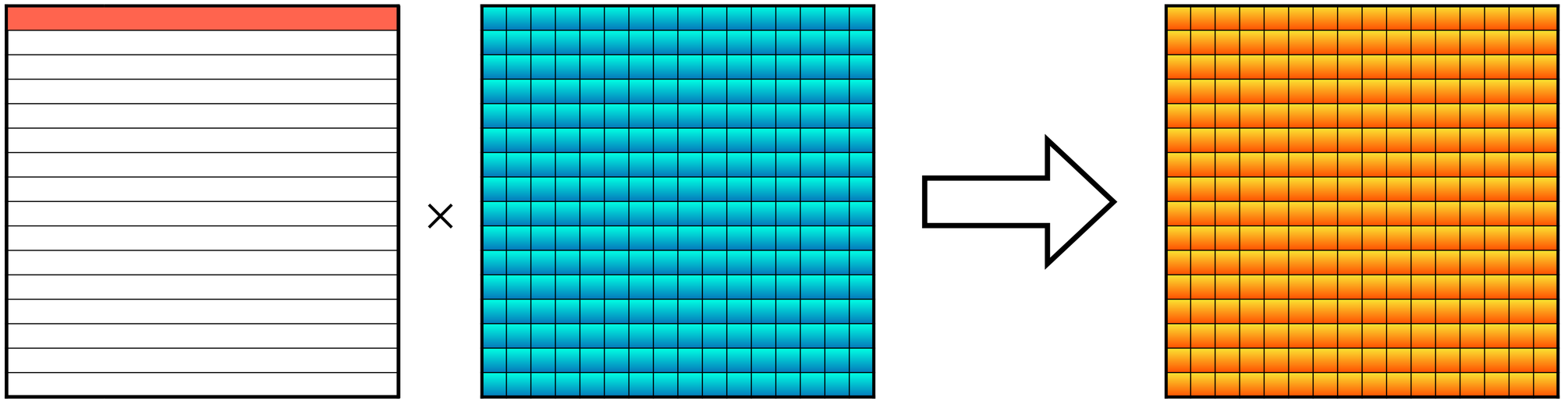


2) ViT 모델 구조

Architecture Detail: Patch Embedding

(3) Projection

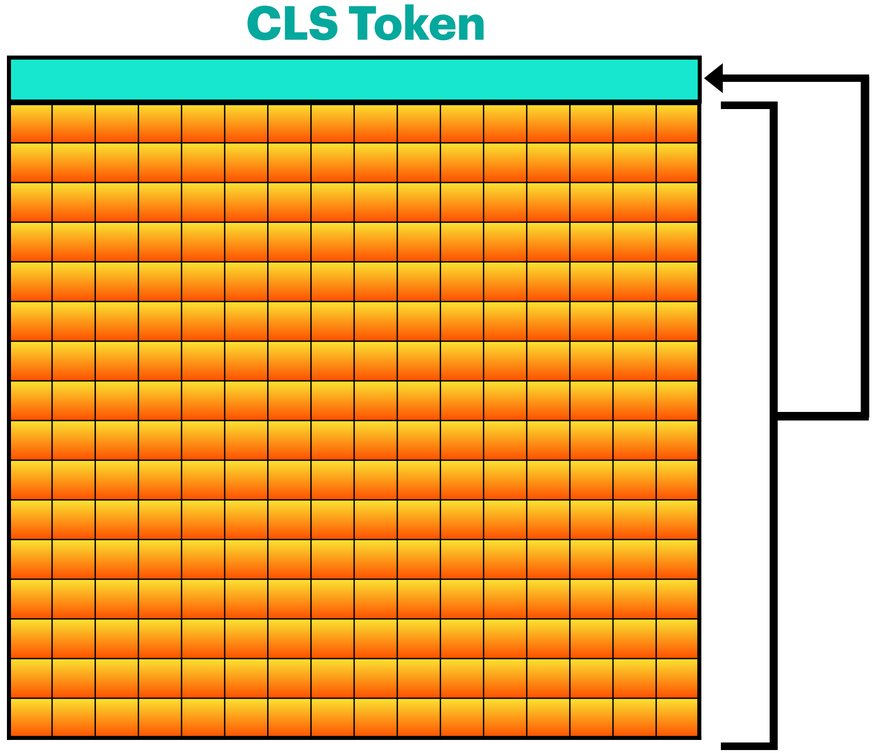
N개의 D차원 input vectors로 변환



2) ViT 모델 구조

Architecture Detail: Patch Embedding

(4) CLS Token 추가



$\in \mathbb{R}^{(N+1)\times D}$

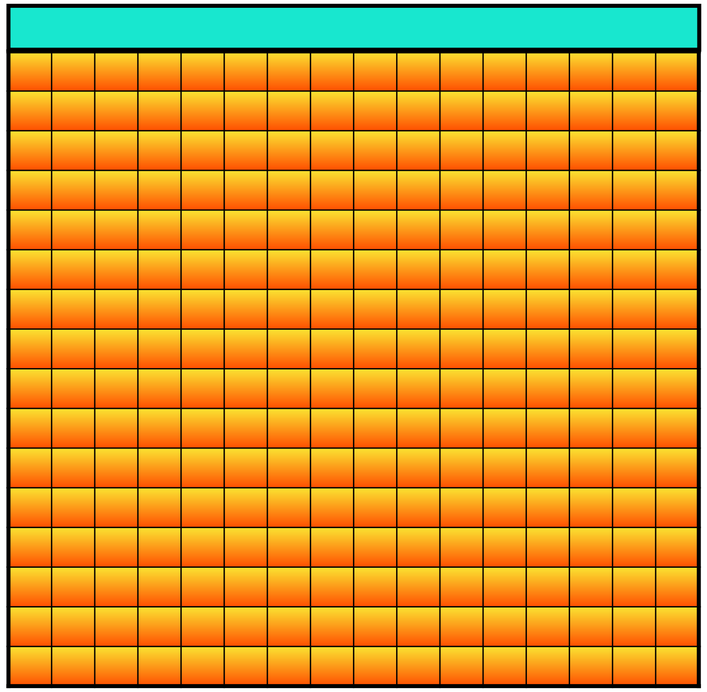
Image Patch 들의 모든 정보를
압축해 저장

2) ViT 모델 구조

Architecture Detail: Patch Embedding

(5) Positional Encoding

CLS Token



$\in \mathbb{R}^{(N+1)\times D}$

=

모델이 각 패치의
공간 상의 관계를 알 수가 없음

2) ViT 모델 구조

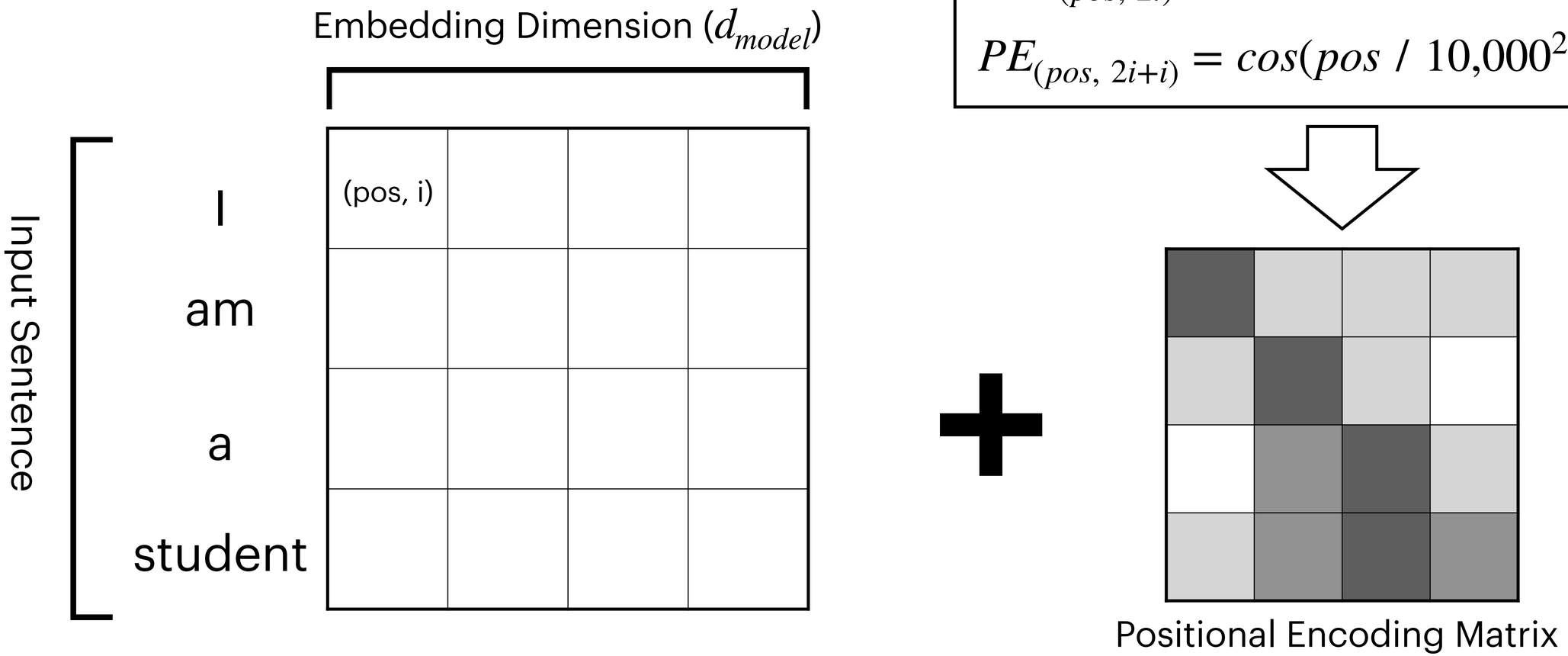
Architecture Detail: Patch Embedding

(5) Positional Encoding

Positional Encoding Function

$$PE_{(pos,\ 2i)} = sin(pos\ /\ 10{,}000^{2i\ /\ d_{model}})$$

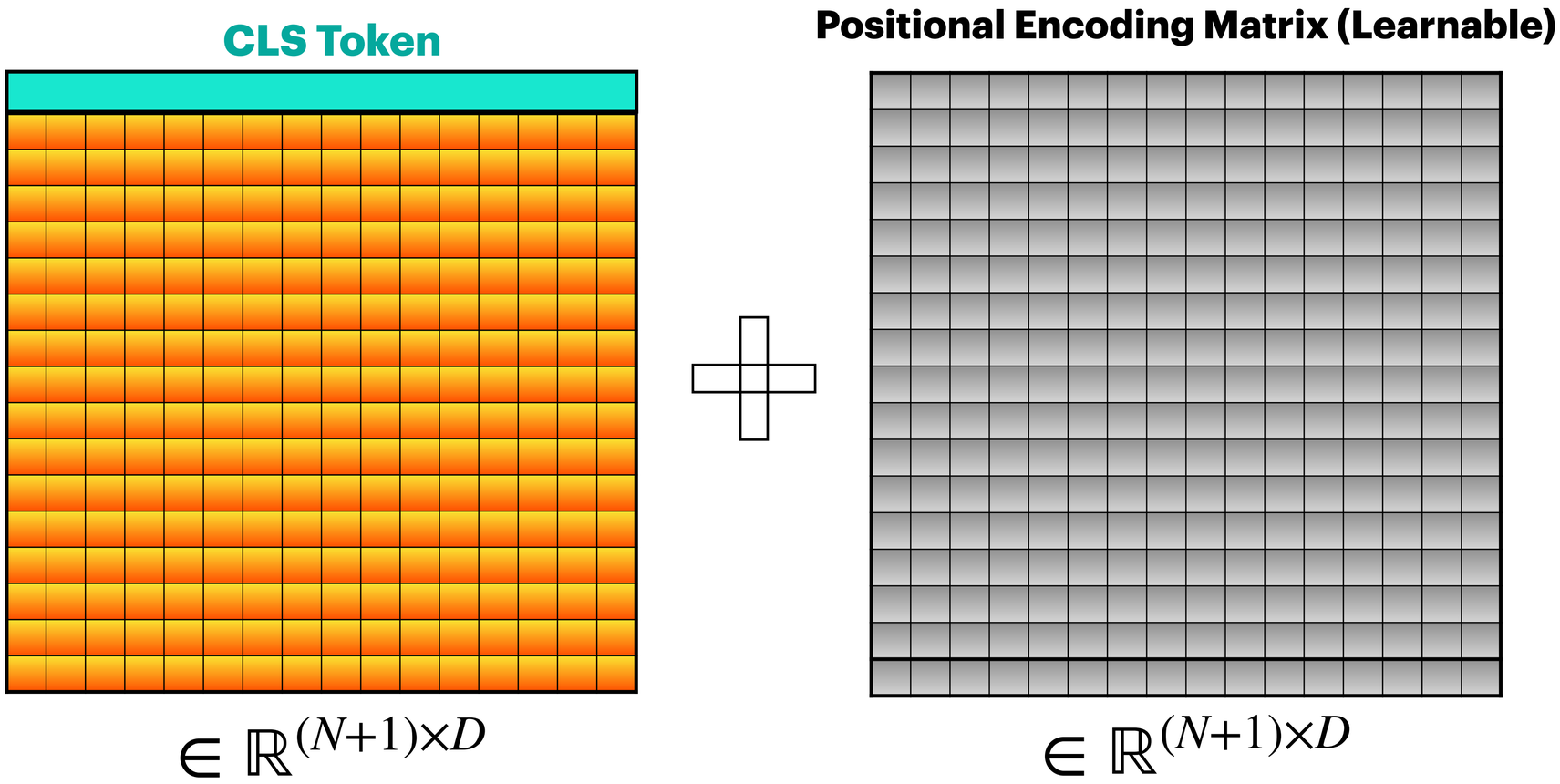
$$PE_{(pos,\ 2i+i)} = cos(pos\ /\ 10{,}000^{2i\ /\ d_{model}})$$



2) ViT 모델 구조

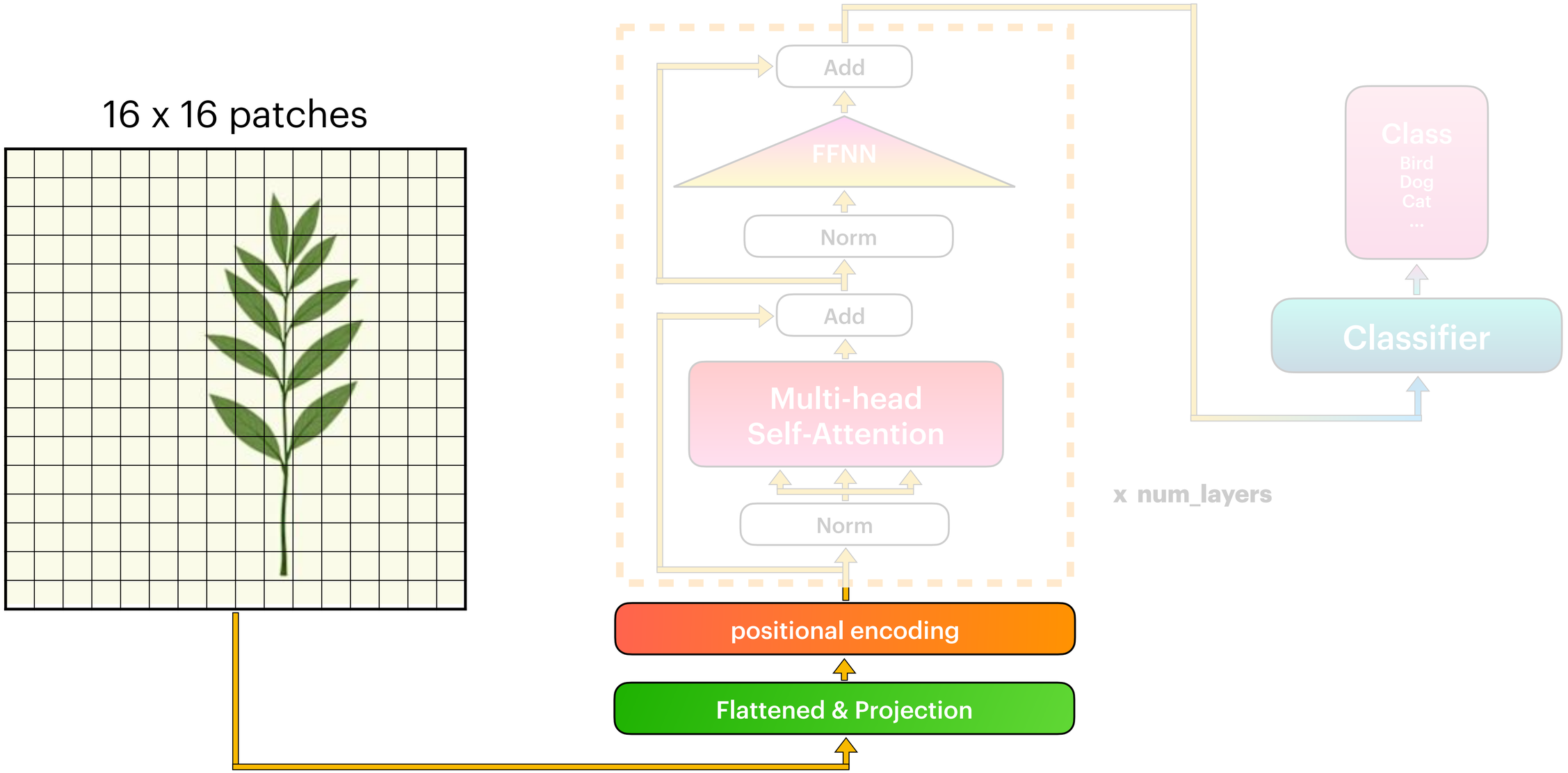
Architecture Detail: Patch Embedding

(5) Positional Encoding



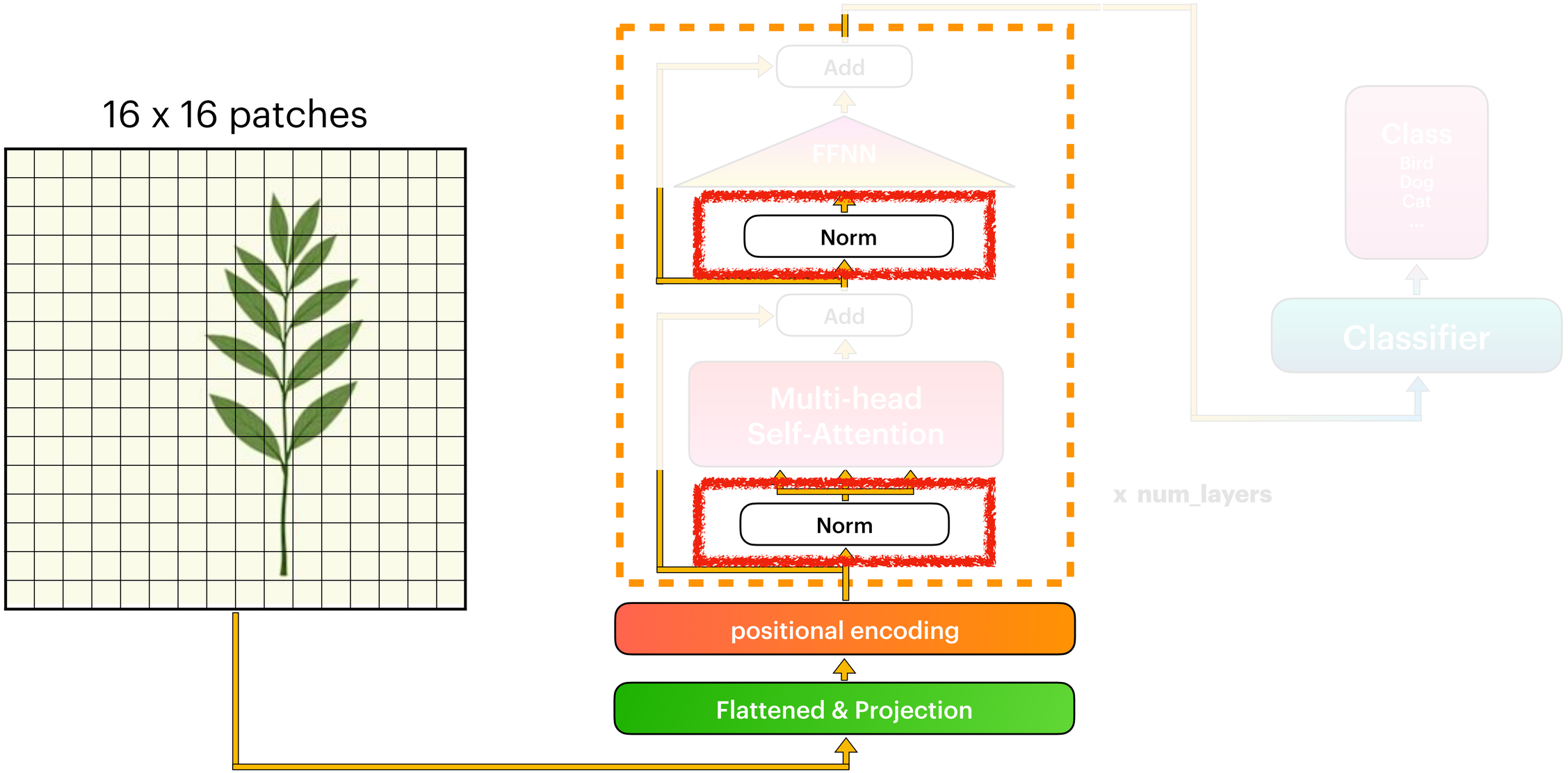
2) ViT 모델 구조

Architecture Detail: Patch Embedding



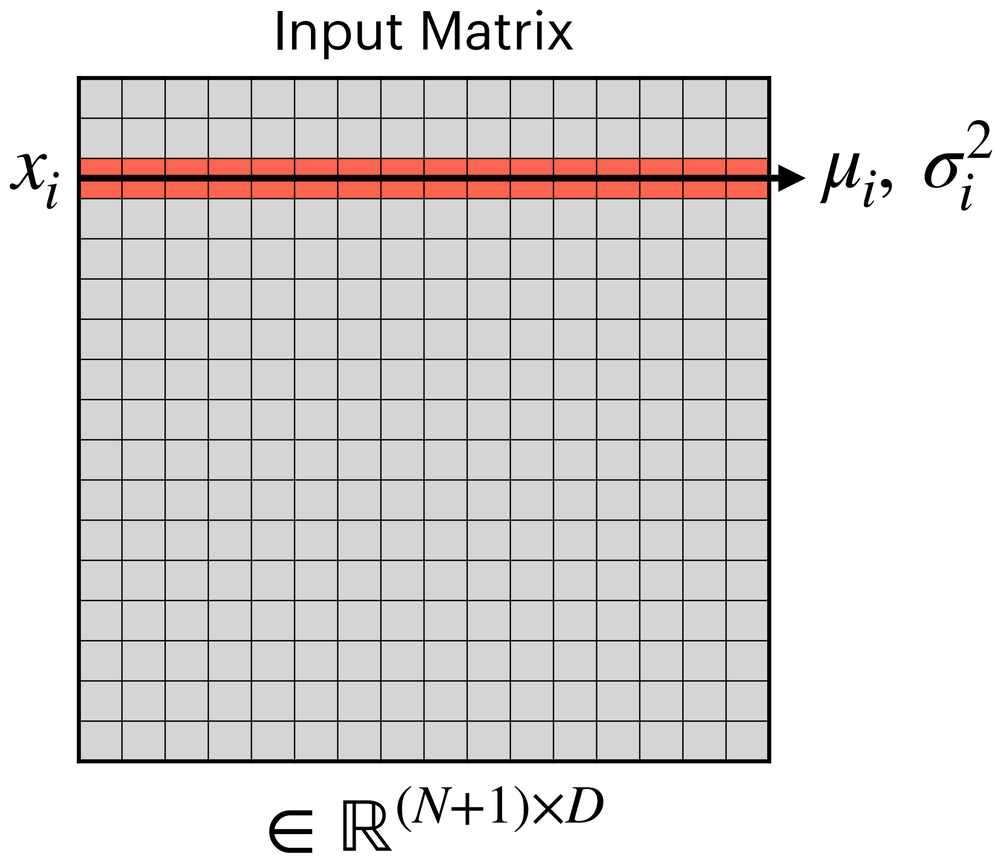
2) ViT 모델 구조

Architecture Detail: Layer Normalization



2) ViT 모델 구조

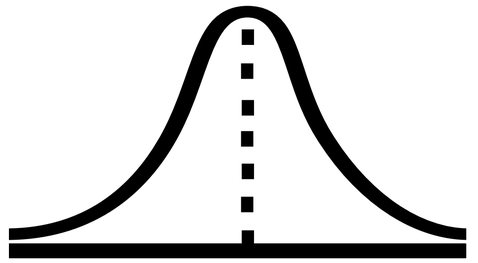
Architecture Detail: Layer Normalization



z-score 표준화

$$\hat{x}_i = \frac{x_i - \mu_i}{\sqrt{\sigma_i^2 + \epsilon}}$$

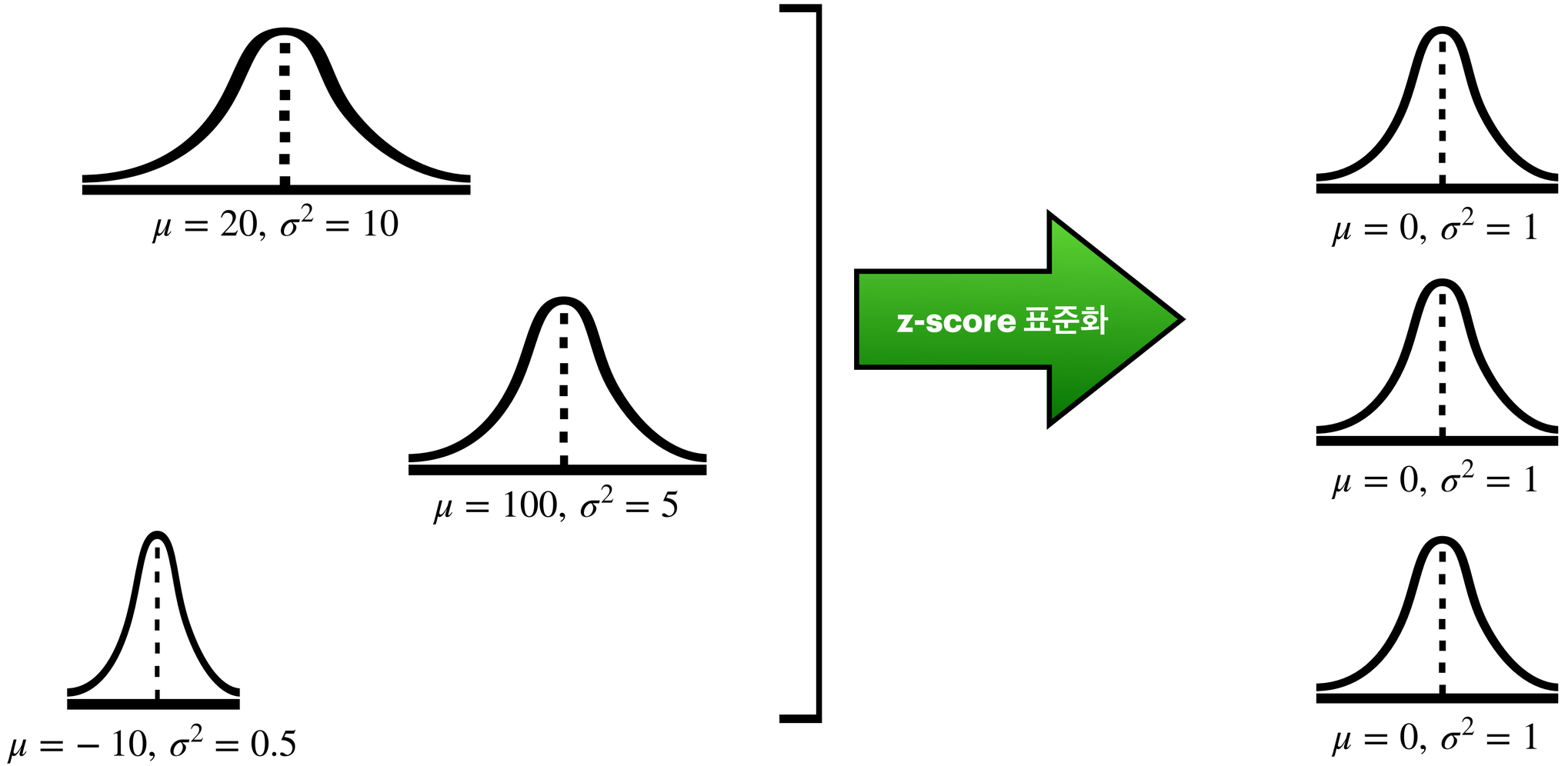
(ϵ: 분모가 0이 되지 않게 하는 아주 작은 상수)



$\mu = 0,\ \sigma^2 = 1$

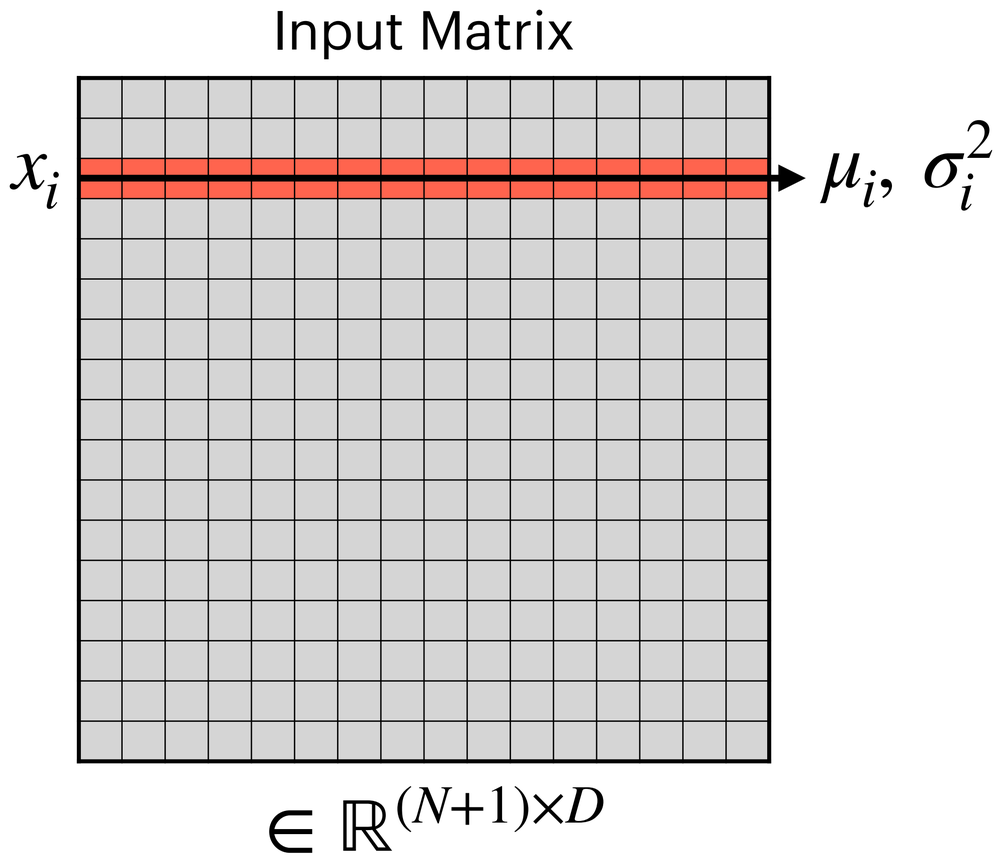
2) ViT 모델 구조

Architecture Detail: Layer Normalization



2) ViT 모델 구조

Architecture Detail: Layer Normalization



Layer Normalization

$$\hat{x}_i = \frac{x_i - \mu_i}{\sqrt{\sigma_i^2 + \epsilon}}$$

(ϵ: 분모가 0이 되지 않게 하는 아주 작은 상수)

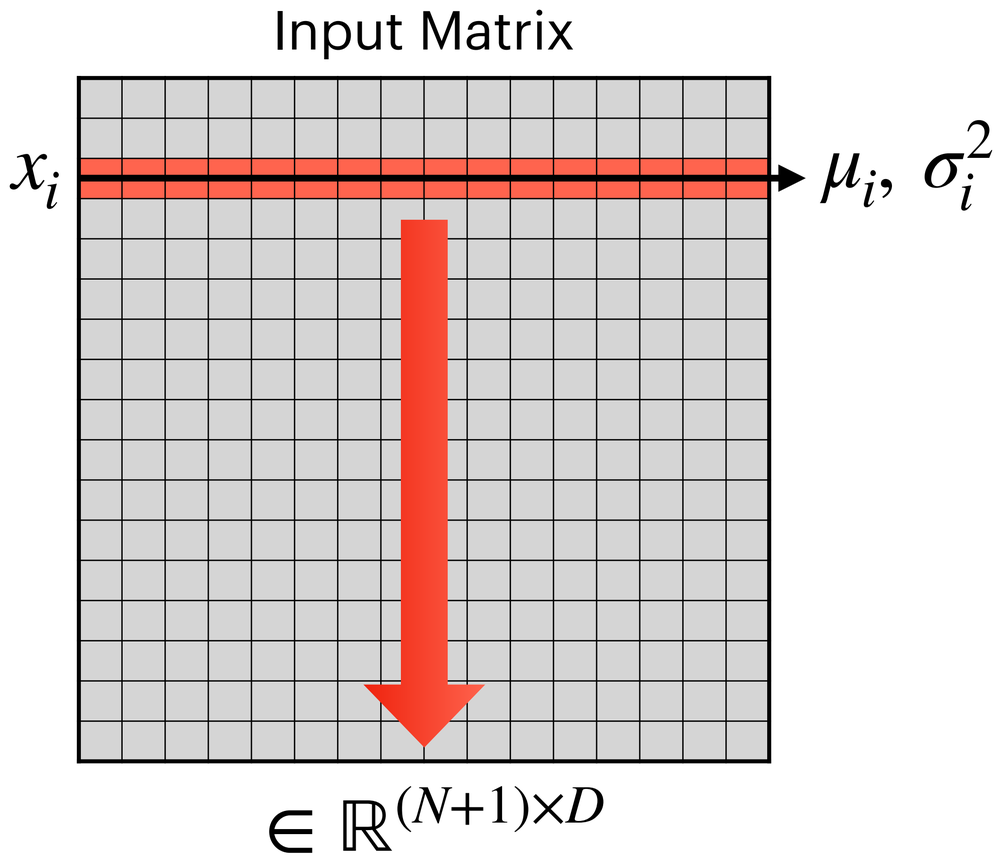
$$LayerNorm(x_i) = \gamma \hat{x}_i + \beta$$

γ: 초기값 1인 $1 \times D$ 벡터 (learnable)

β: 초기값 1인 $D \times 1$ 벡터 (learnable)

2) ViT 모델 구조

Architecture Detail: Layer Normalization



Layer Normalization

$$\hat{x}_i = \frac{x_i - \mu_i}{\sqrt{\sigma_i^2 + \epsilon}}$$

(ϵ: 분모가 0이 되지 않게 하는 아주 작은 상수)

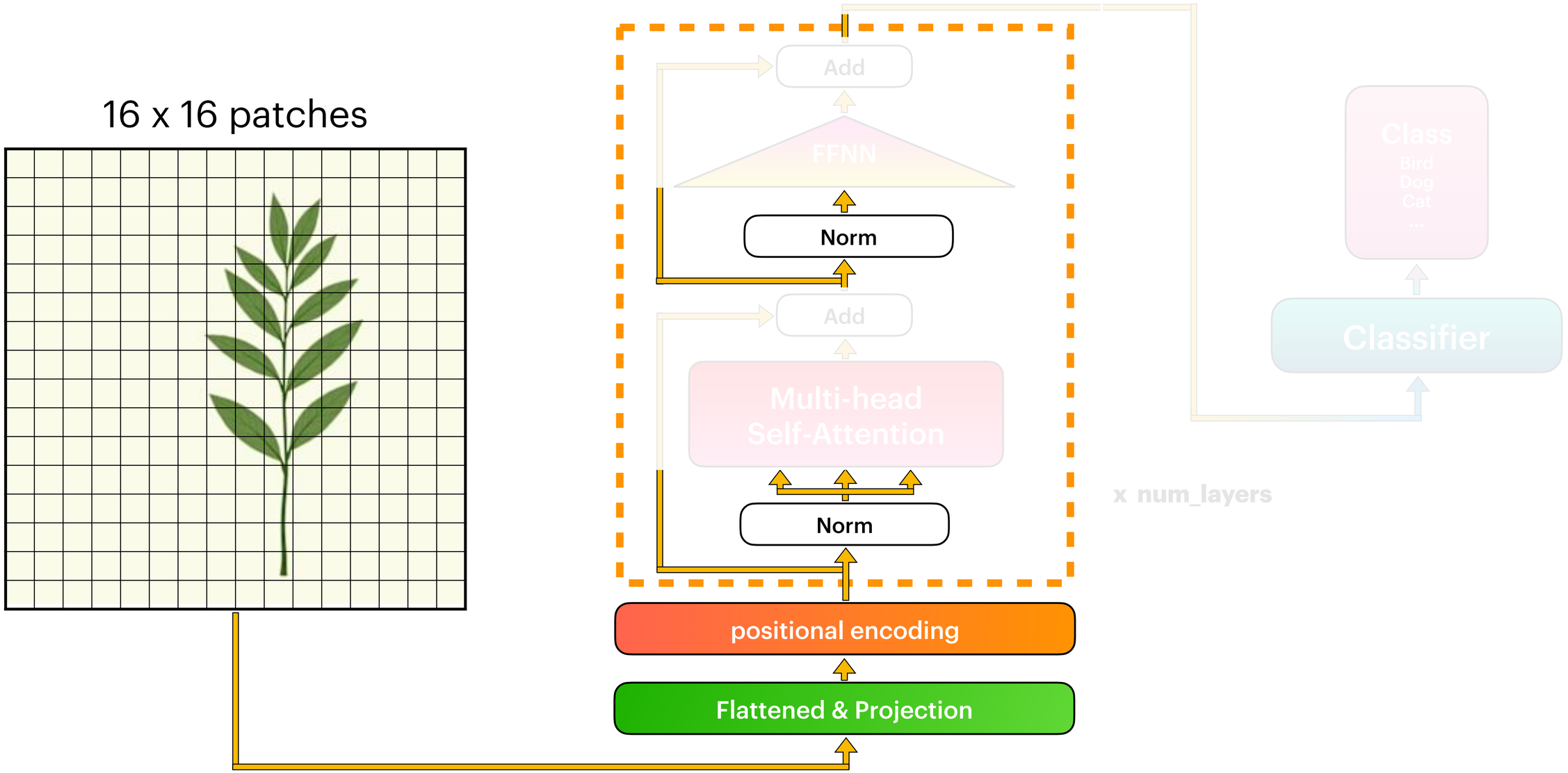
$$LayerNorm(x_i) = \gamma \hat{x}_i + \beta$$

γ: 초기값 1인 $1 \times D$ 벡터 (learnable)

β: 초기값 1인 $D \times 1$ 벡터 (learnable)

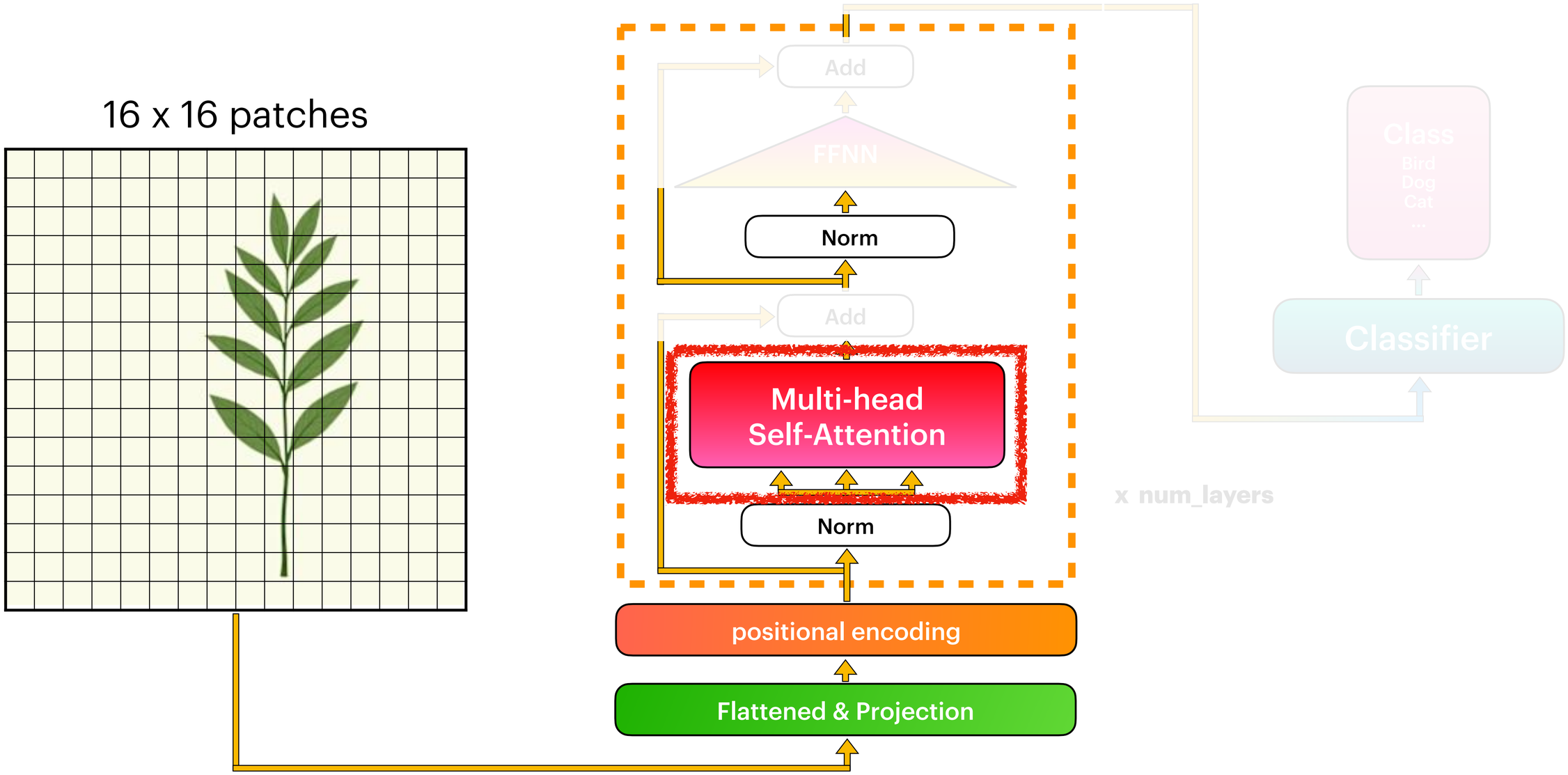
2) ViT 모델 구조

Architecture Detail: Layer Normalization



2) ViT 모델 구조

Architecture Detail: Multi-head Self-Attention



2) ViT 모델 구조

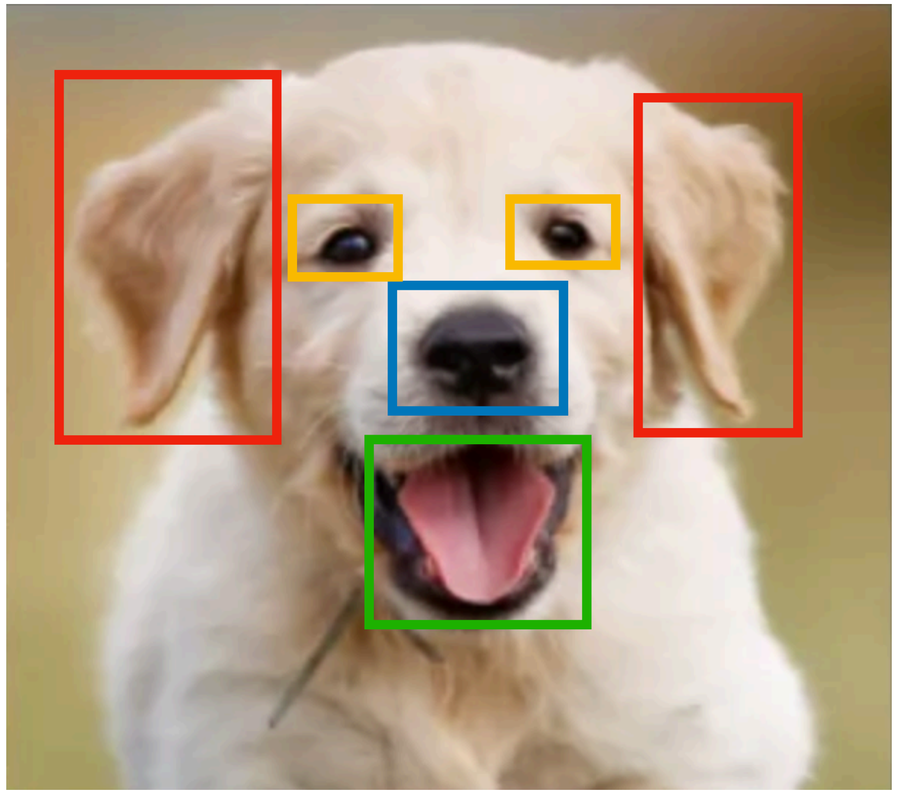
Architecture Detail: Multi-head Self-Attention

How do we see? 👁

2) ViT 모델 구조

Architecture Detail: Multi-head Self-Attention

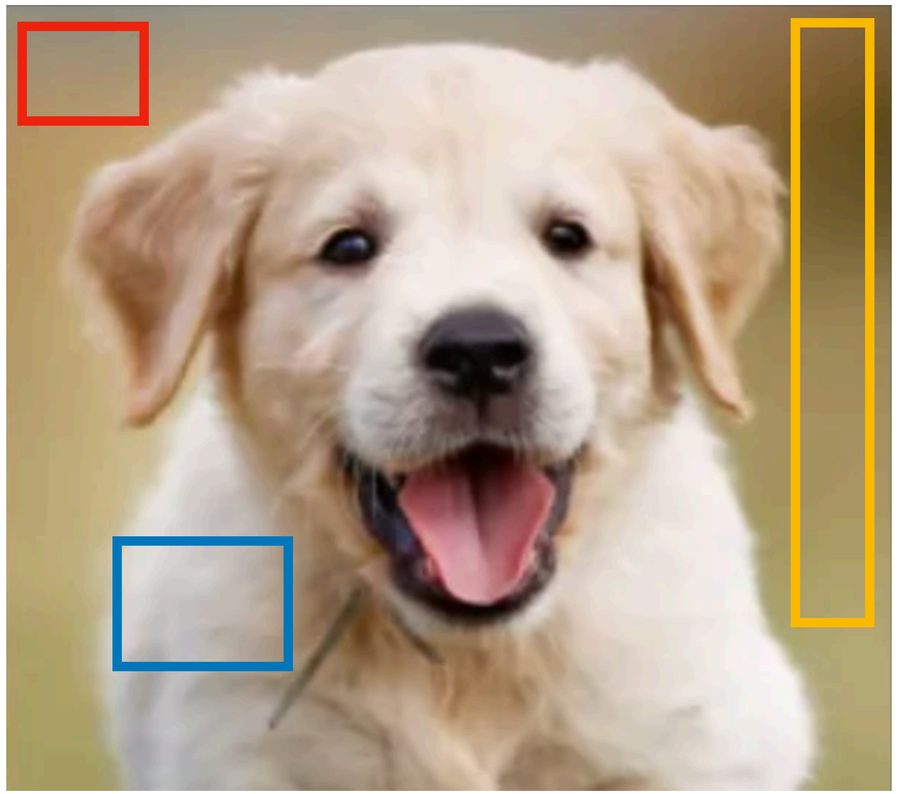
How do we see?



2) ViT 모델 구조

Architecture Detail: Multi-head Self-Attention

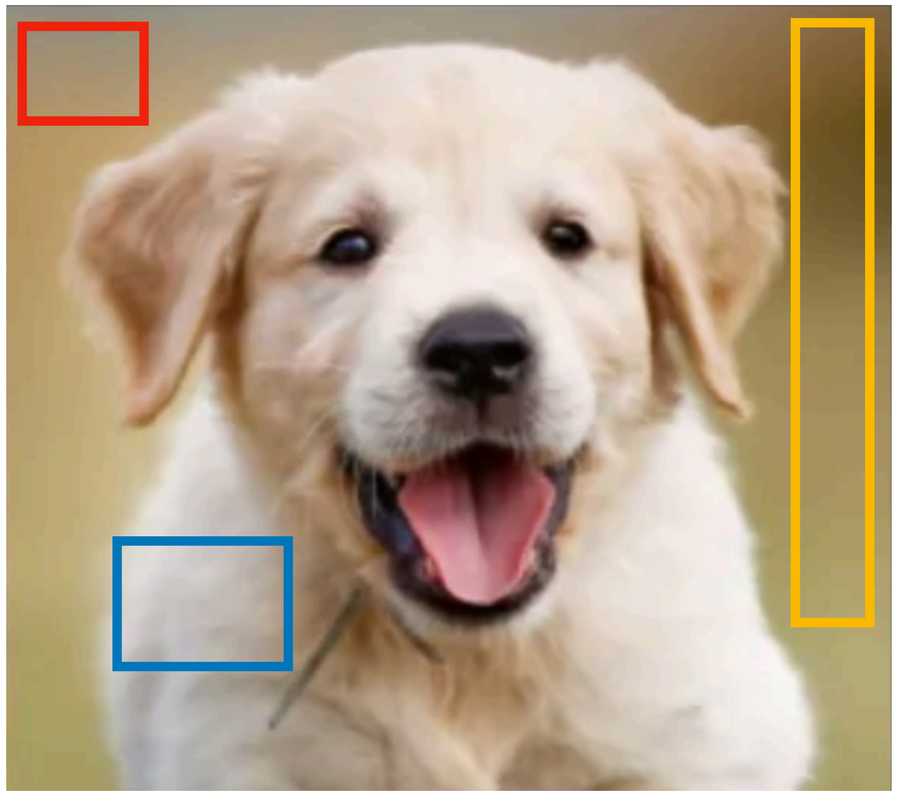
How do we see?



2) ViT 모델 구조

Architecture Detail: Multi-head Self-Attention

How do we see?



2) ViT 모델 구조

Architecture Detail: Multi-head Self-Attention

“이미지의 어느 부분을 더 집중해서 볼 것인가?”

2) ViT 모델 구조

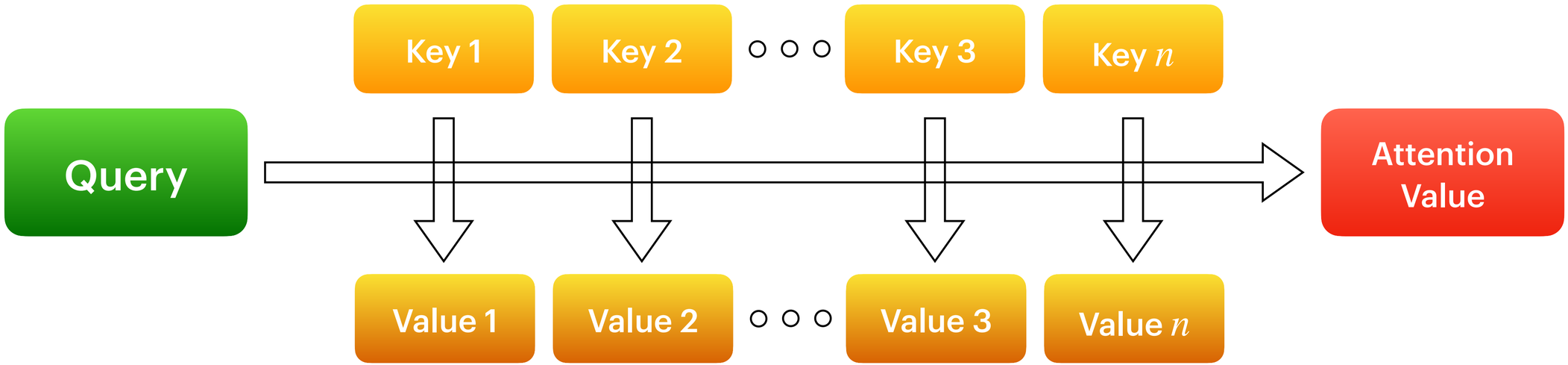
Architecture Detail: Multi-head Self-Attention

$$Attention(Q, K, V) = \text{Attention Value}$$

2) ViT 모델 구조

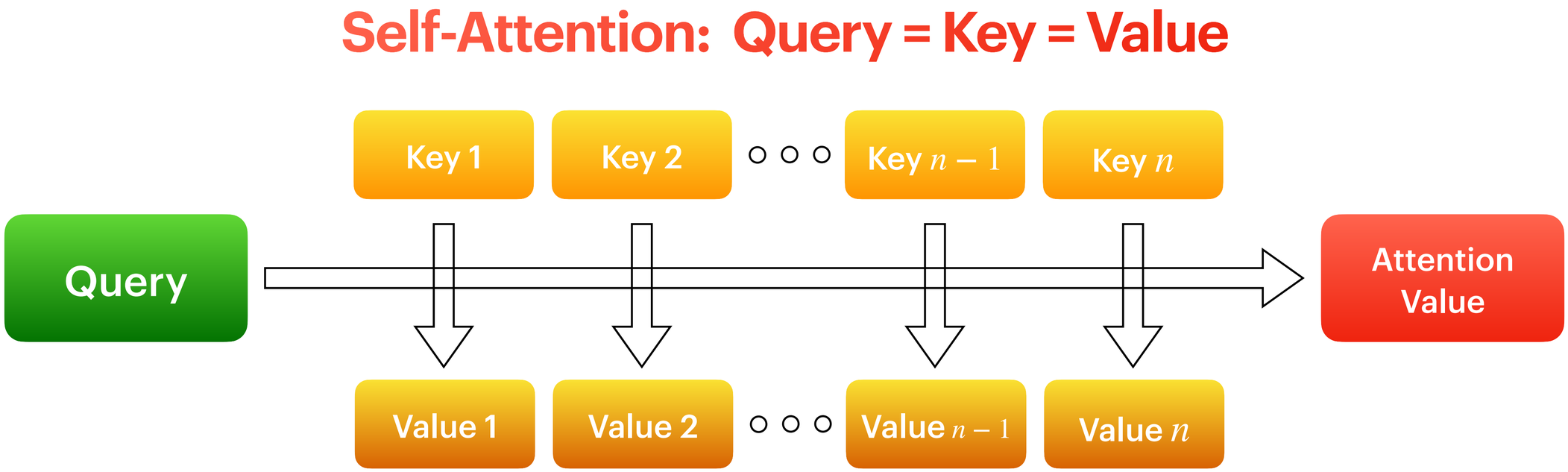
Architecture Detail: Multi-head Self-Attention

Attention: Query와 Keys 사이의 유사성 수치를 Value에 반영



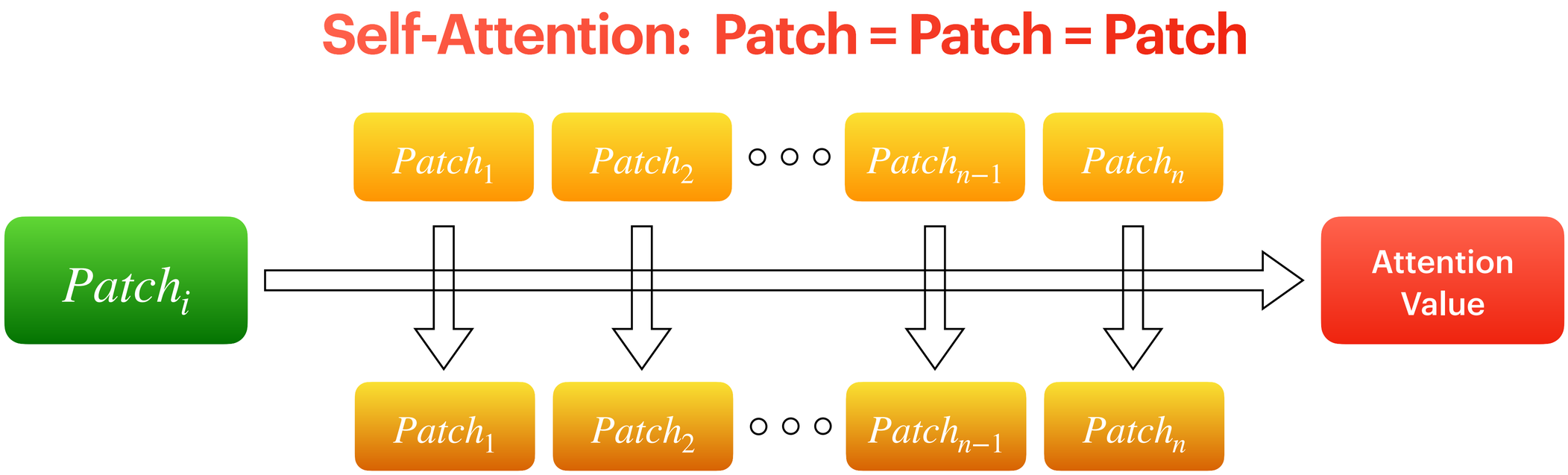
2) ViT 모델 구조

Architecture Detail: Multi-head Self-Attention



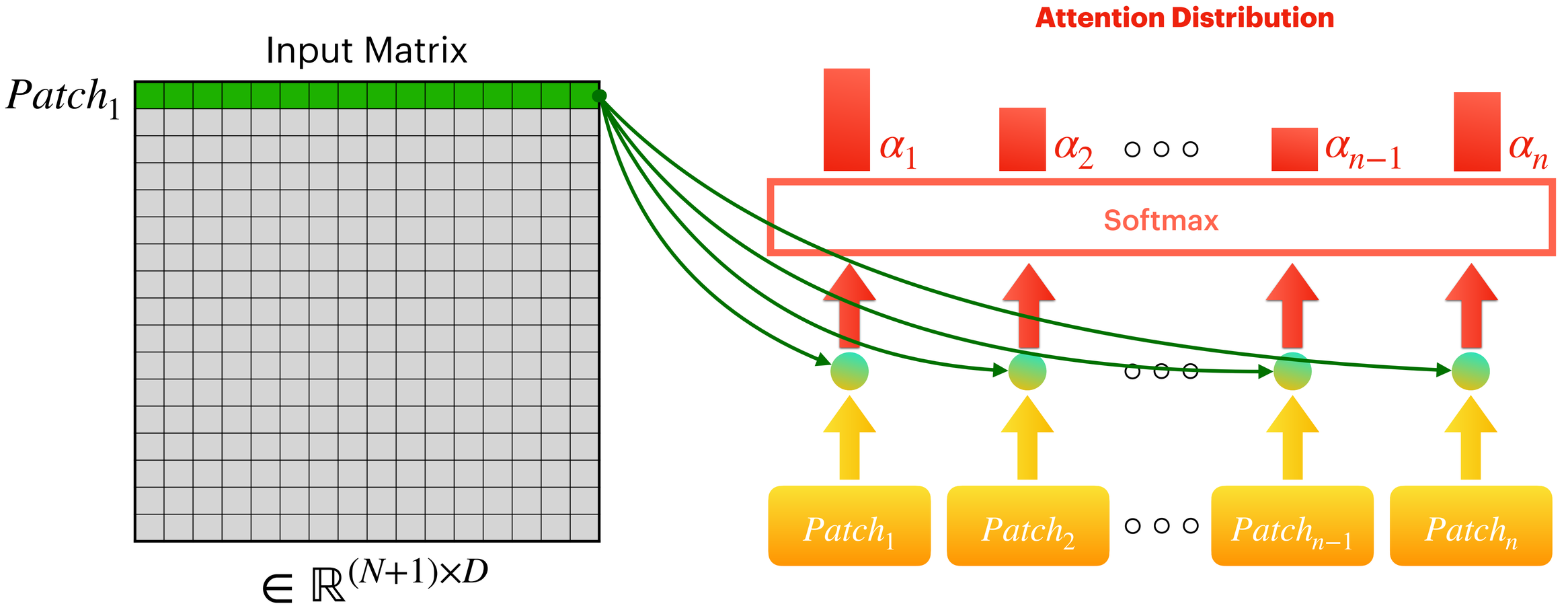
2) ViT 모델 구조

Architecture Detail: Multi-head Self-Attention



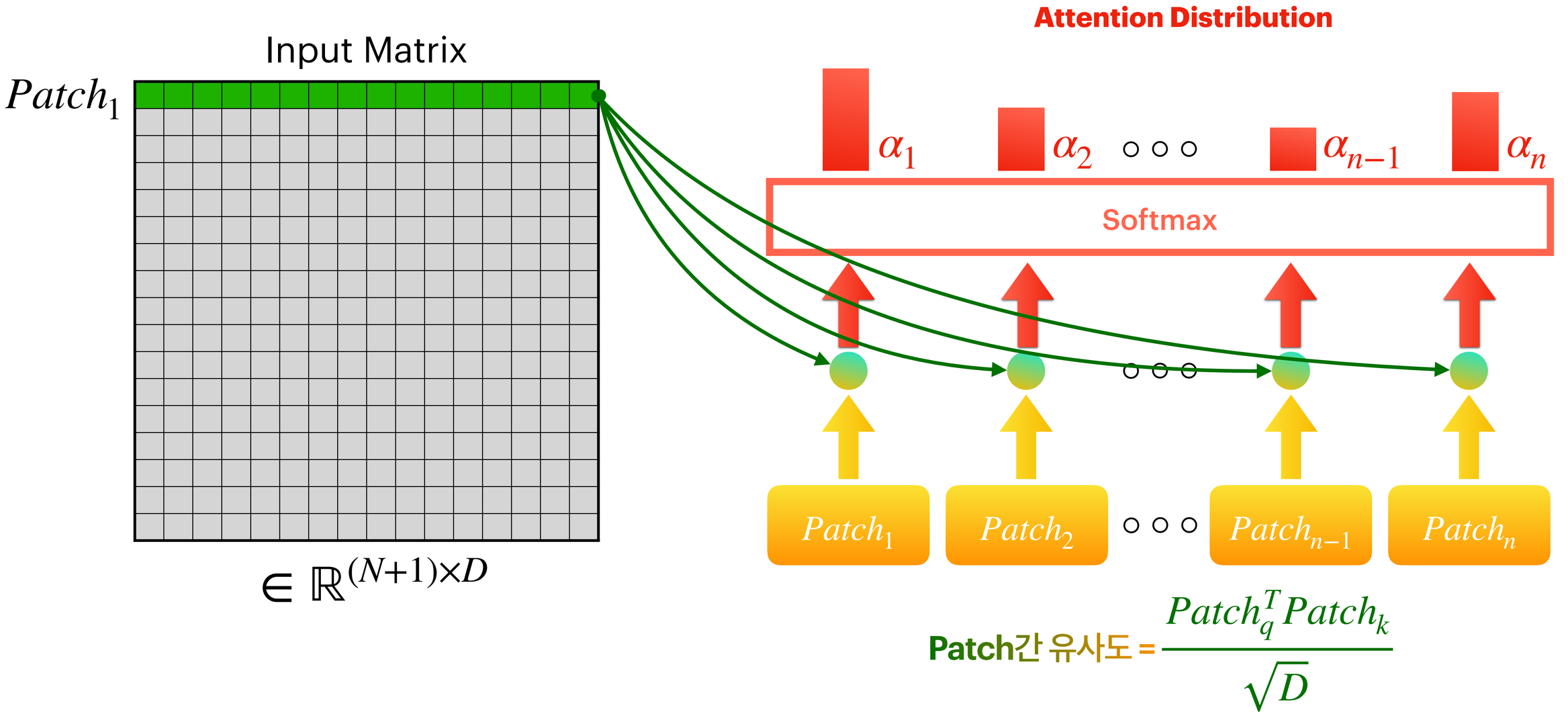
2) ViT 모델 구조

Architecture Detail: Multi-head Self-Attention



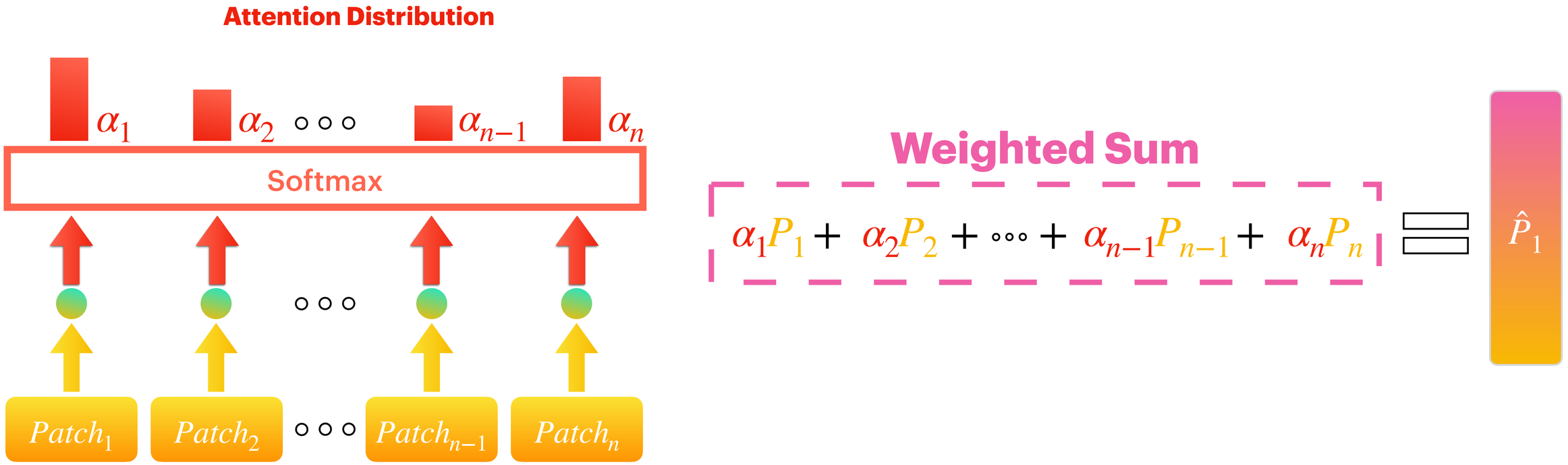
2) ViT 모델 구조

Architecture Detail: Multi-head Self-Attention



2) ViT 모델 구조

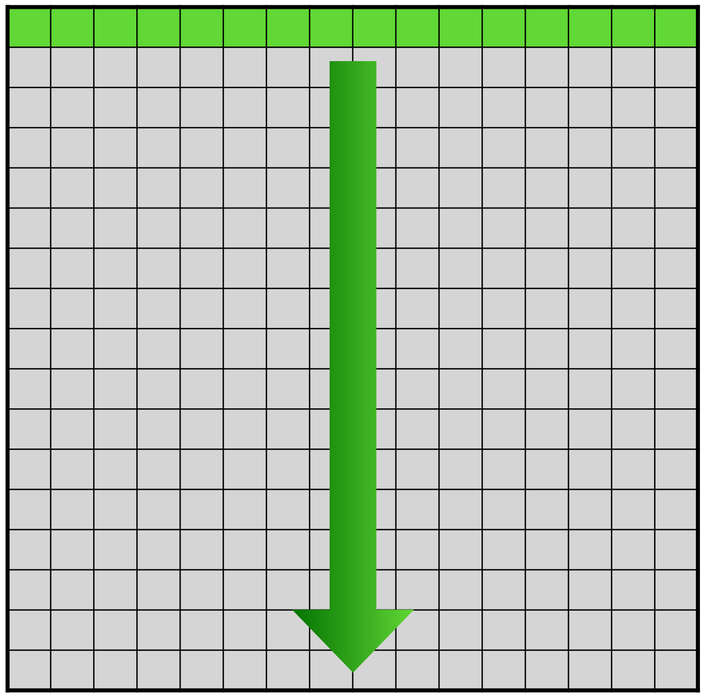
Architecture Detail: Multi-head Self-Attention



2) ViT 모델 구조

Architecture Detail: Multi-head Self-Attention

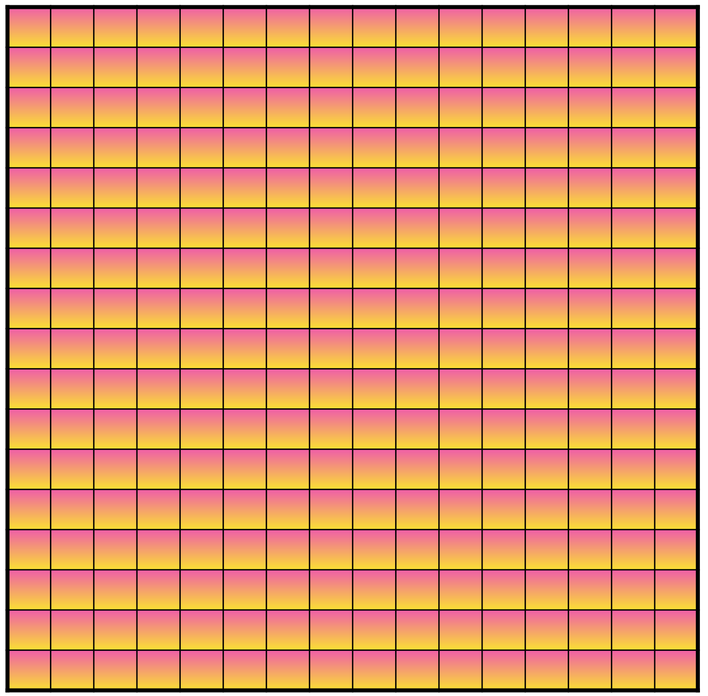
Input Matrix



$\in \mathbb{R}^{(N+1)\times D}$

모든 **Patch**에 대해 계산

Attention이 반영된 Matrix



$\in \mathbb{R}^{(N+1)\times D}$

2) ViT 모델 구조

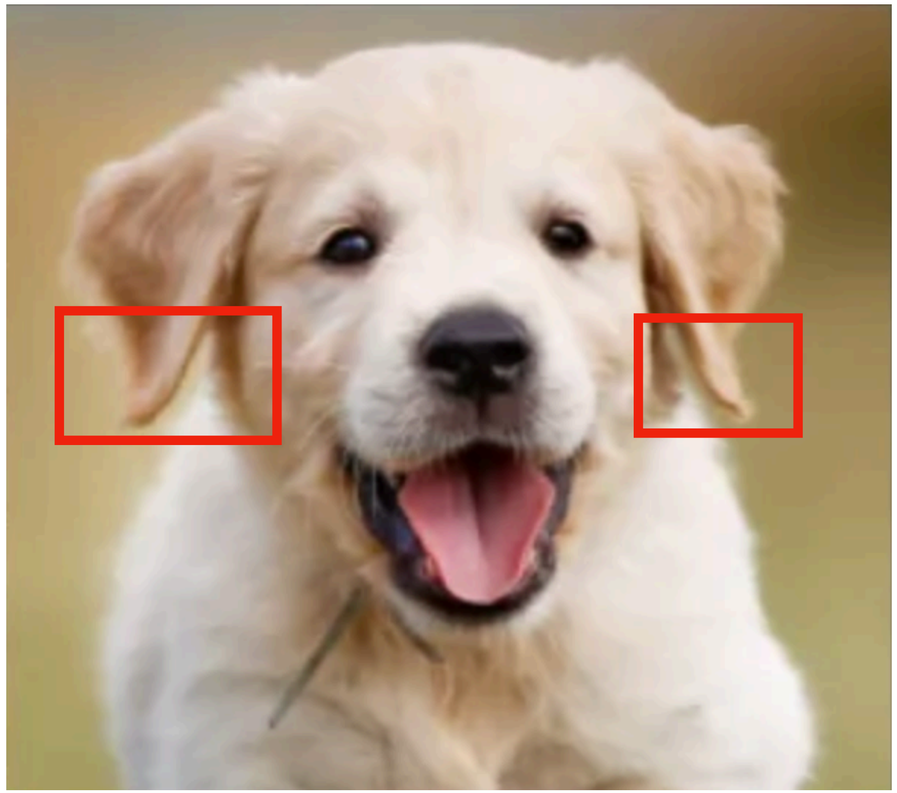
Architecture Detail: Multi-head Self-Attention

그럼 “Multi-head” 는?

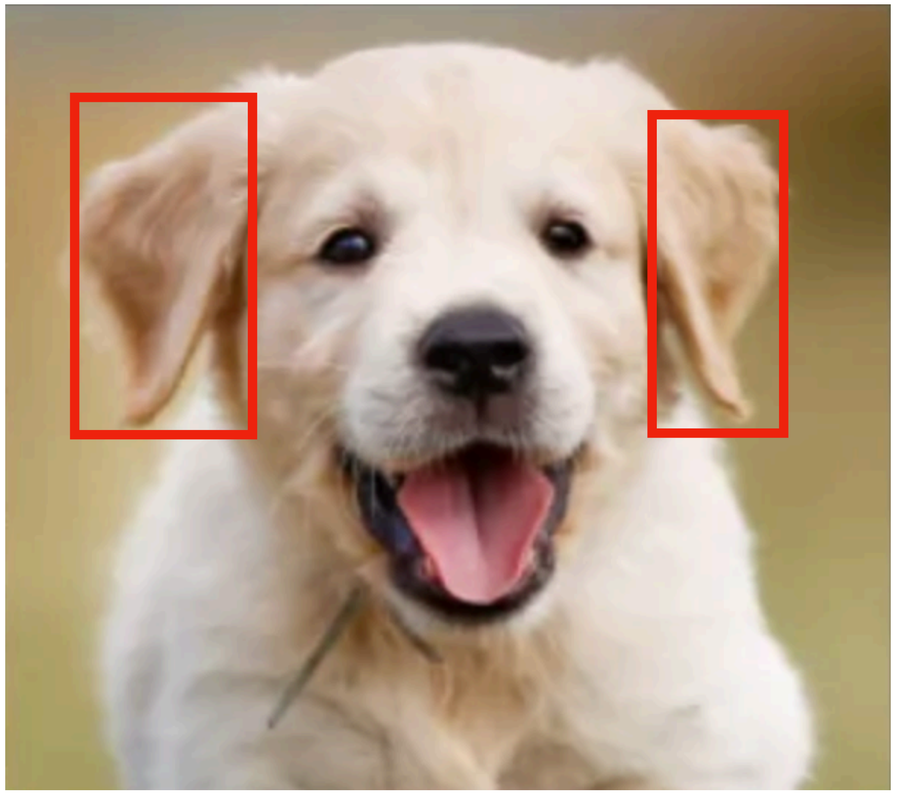
2) ViT 모델 구조

Architecture Detail: Multi-head Self-Attention

A: ‘귀 끝이 뾰족하네’



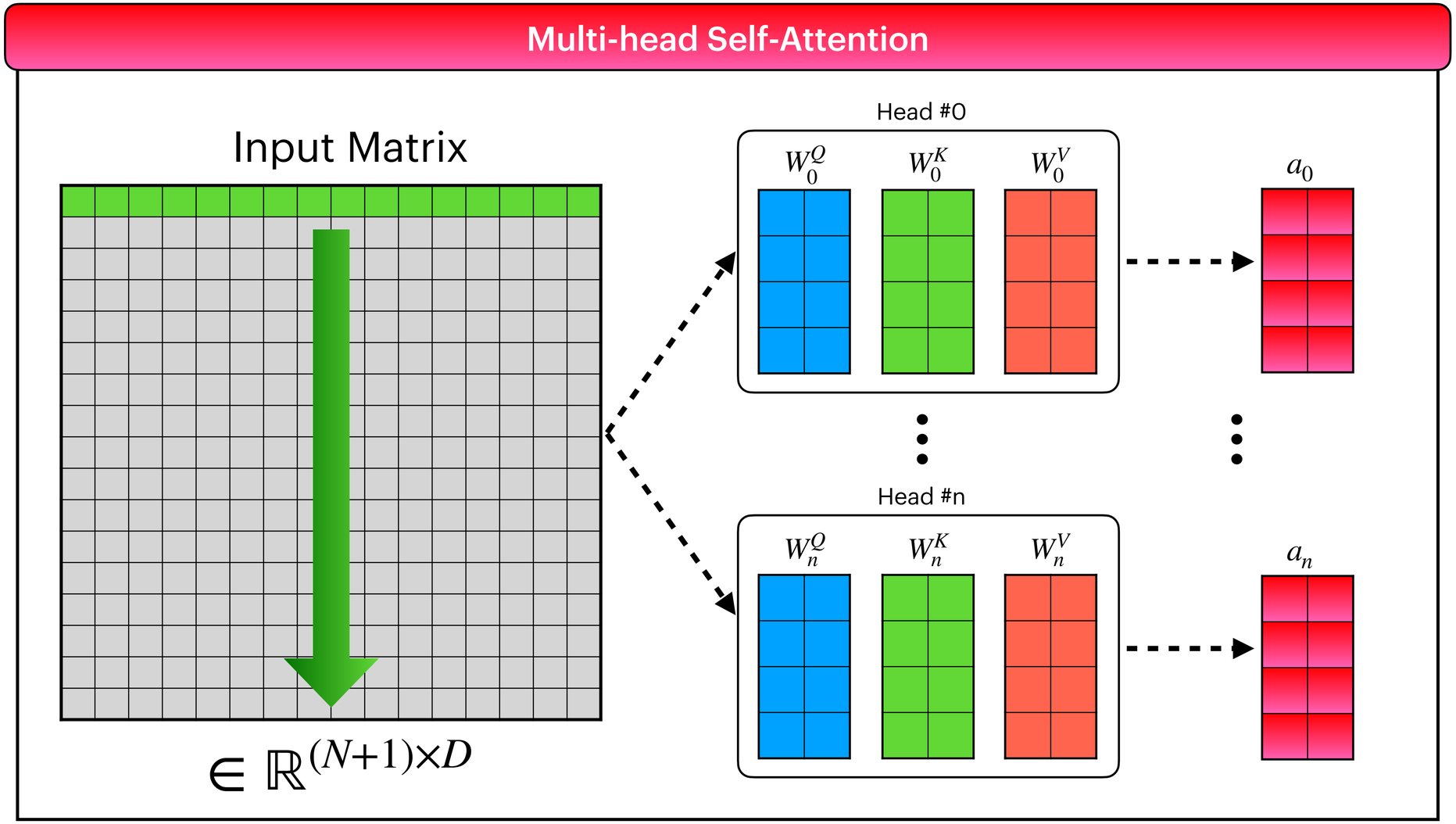
B: ‘귀 길이가 기네’



2) ViT 모델 구조

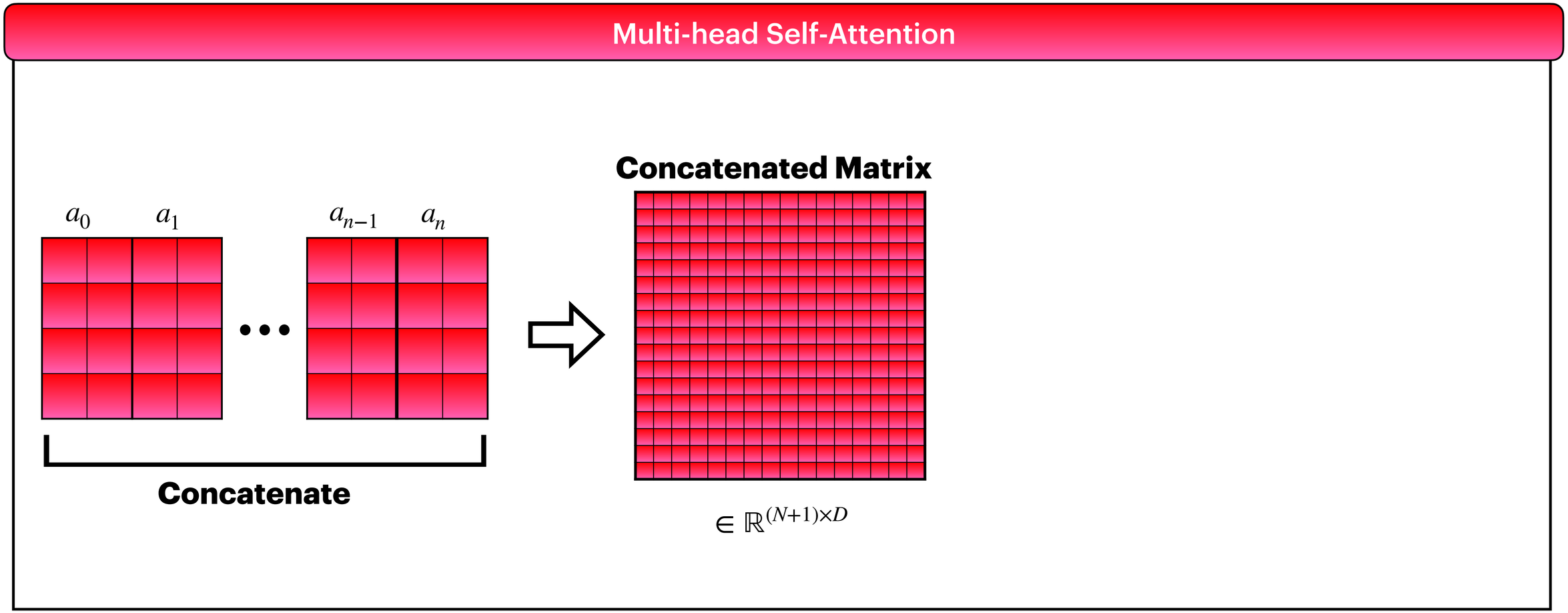
Architecture Detail: Multi-head Self-Attention

서로 다른 **Head**에서
각각 **Attention**을 계산



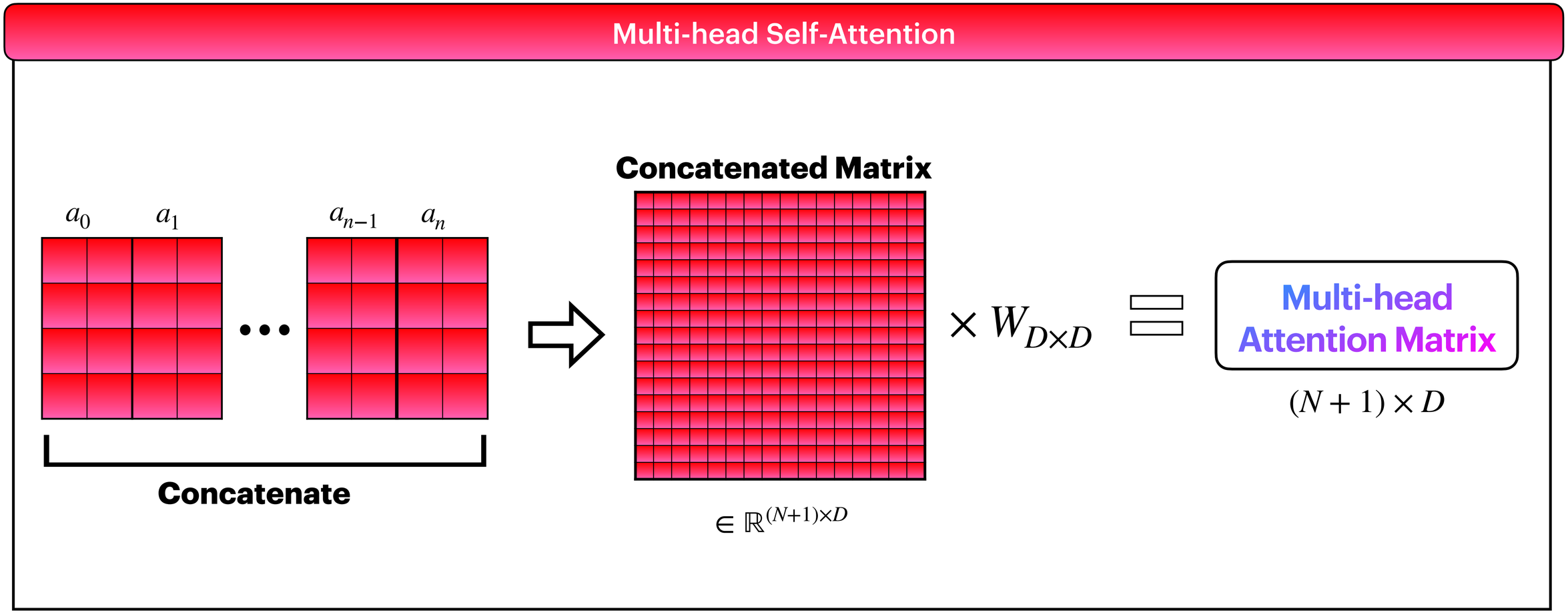
2) ViT 모델 구조

Architecture Detail: Multi-head Self-Attention



2) ViT 모델 구조

Architecture Detail: Multi-head Self-Attention

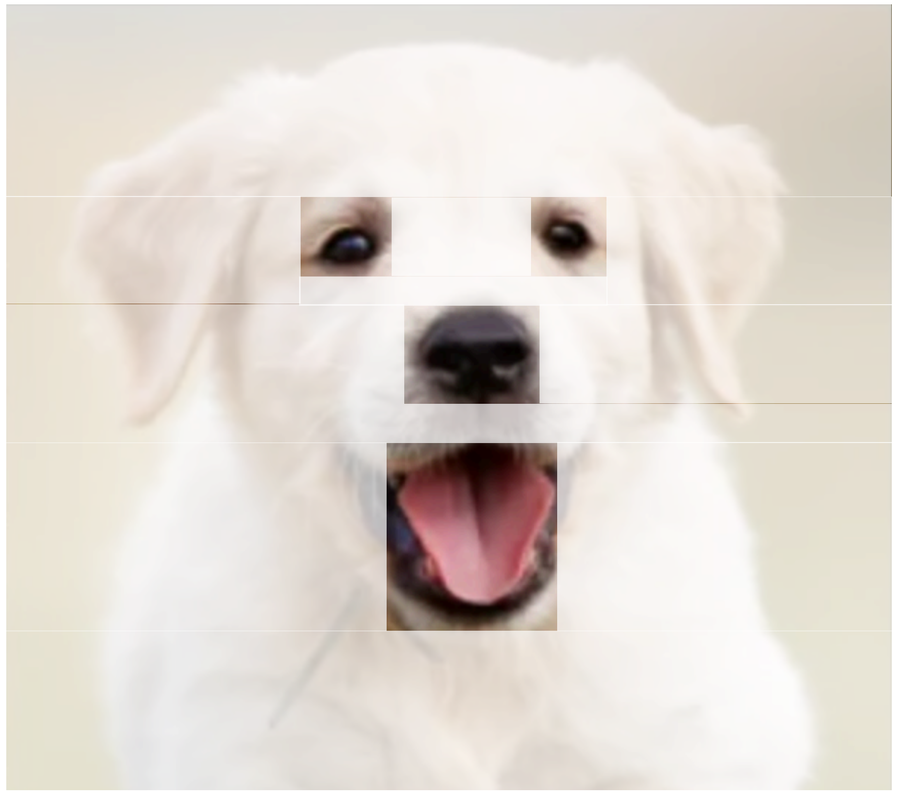


2) ViT 모델 구조

Architecture Detail: Multi-head Self-Attention

Multi-head Self-Attention

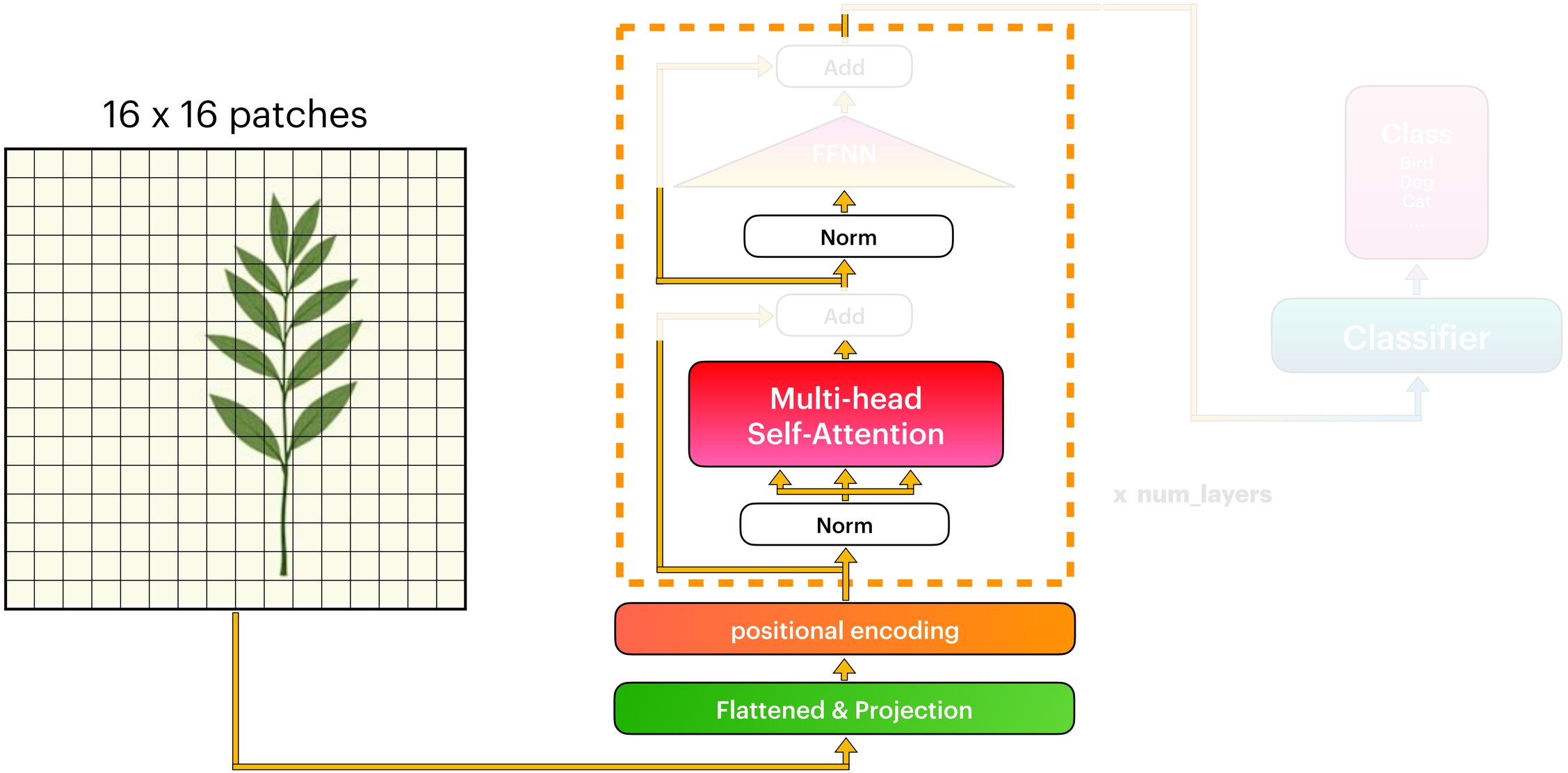
Head #1



Head #2

2) ViT 모델 구조

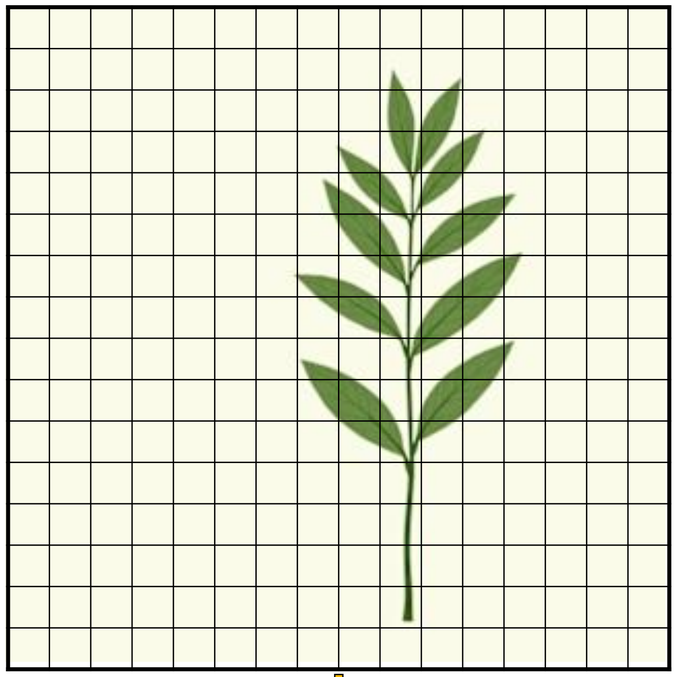
Architecture Detail: Multi-head Self-Attention

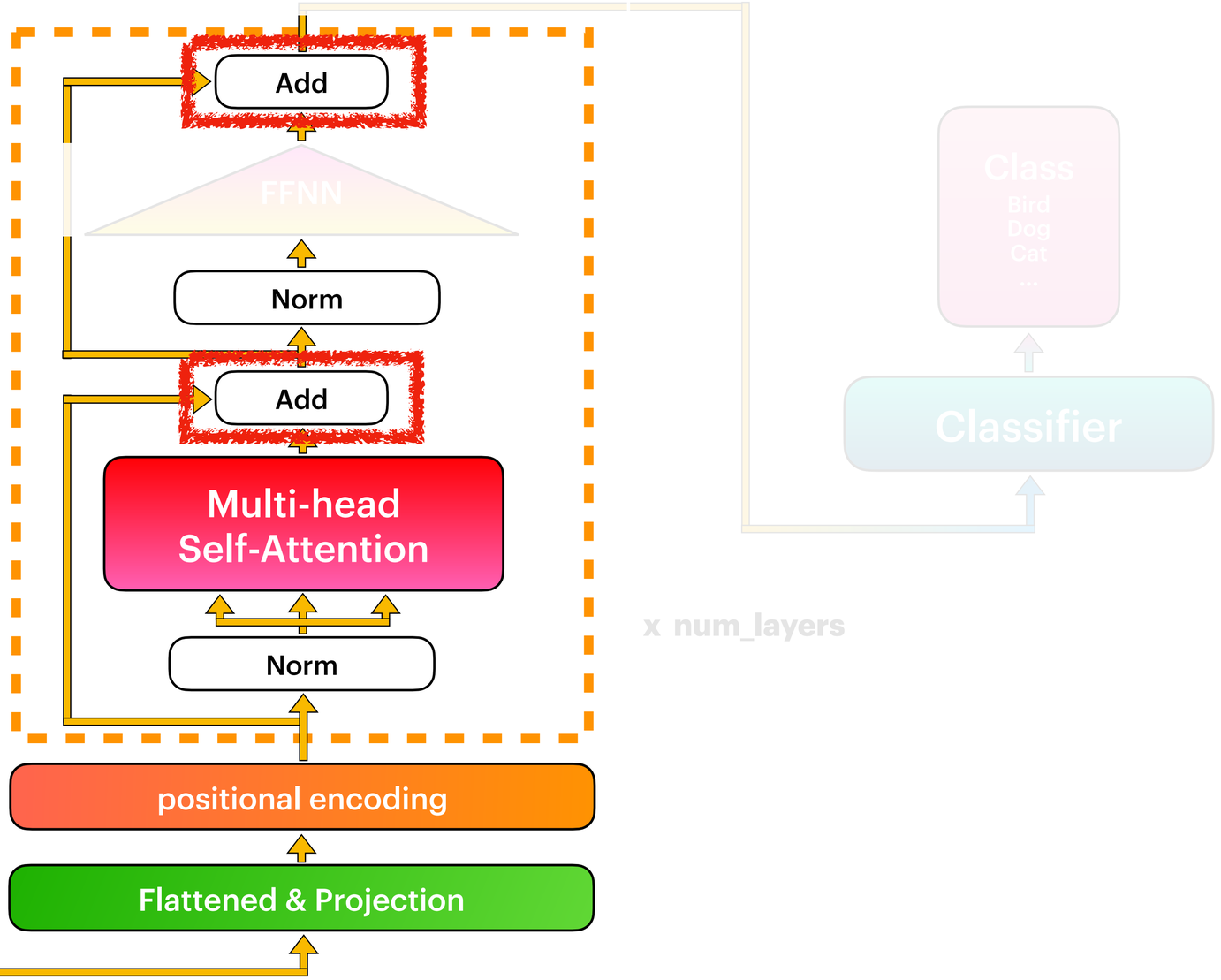


2) ViT 모델 구조

Architecture Detail: Residual Connection

16 x 16 patches





2) ViT 모델 구조

Architecture Detail: Residual Connection

Residual Connection = Input + Output

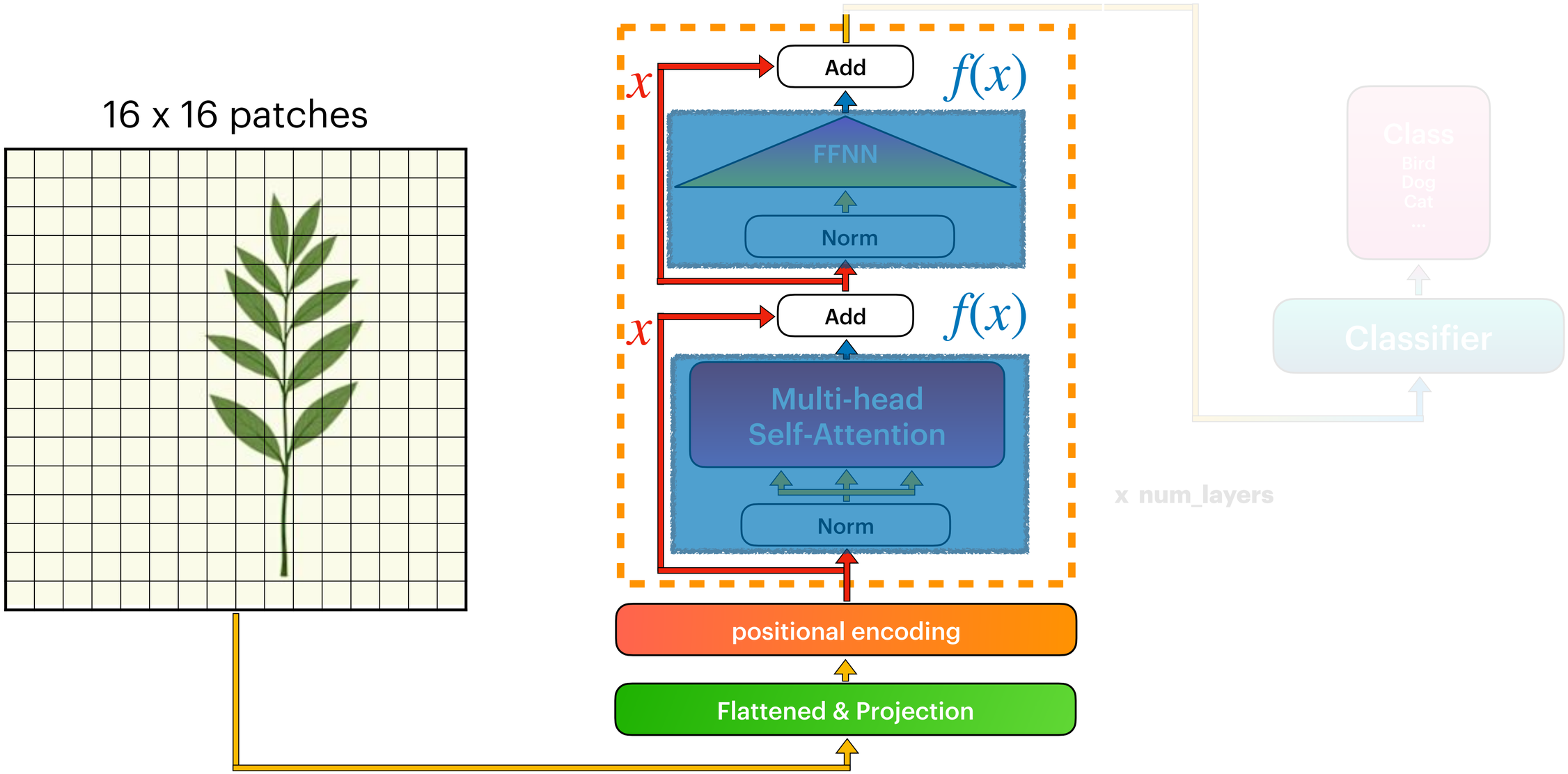
2) ViT 모델 구조

Architecture Detail: Residual Connection

$$residual\ \ connection(x) = x + f(x)$$

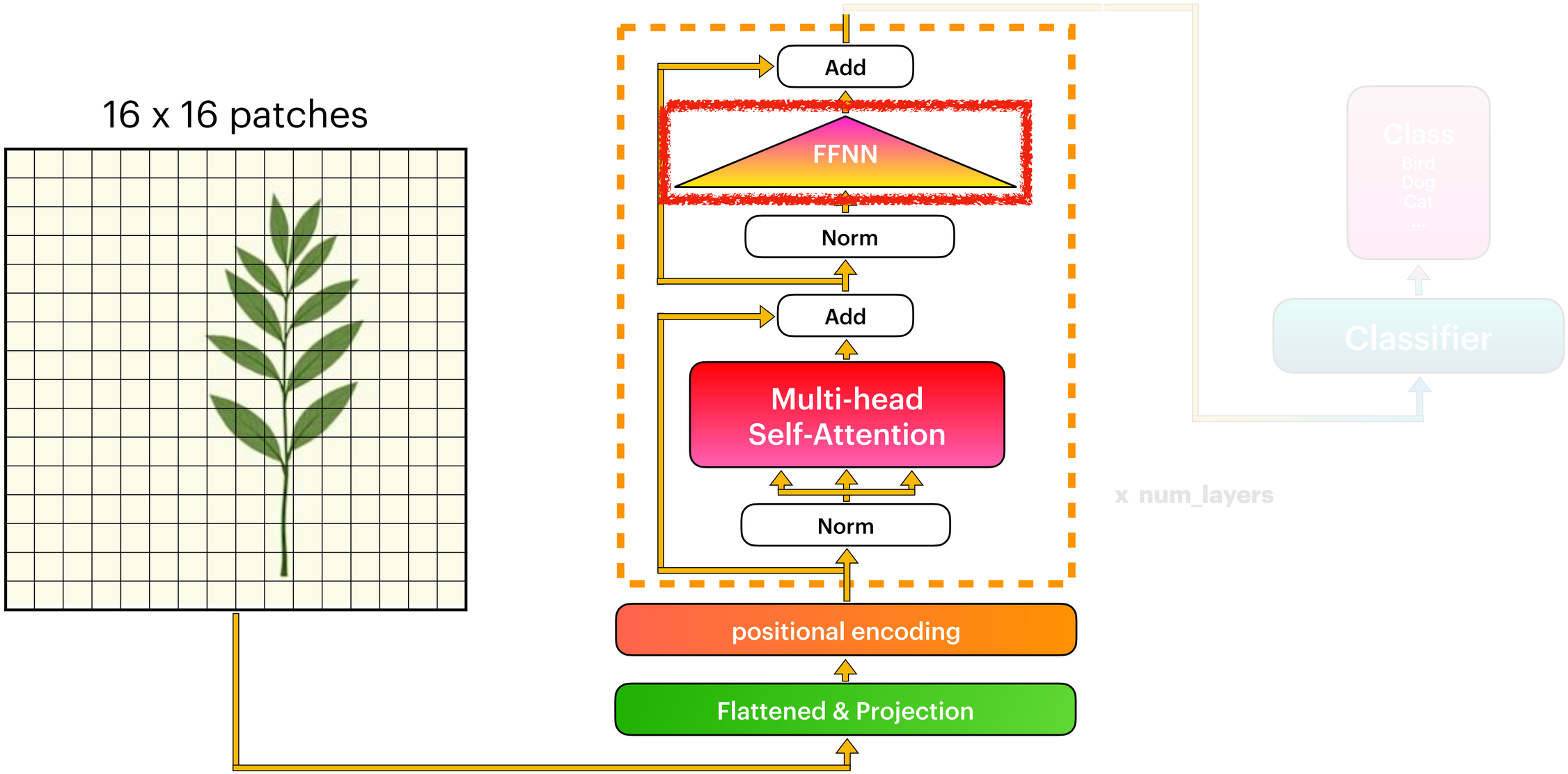
2) ViT 모델 구조

Architecture Detail: Residual Connection



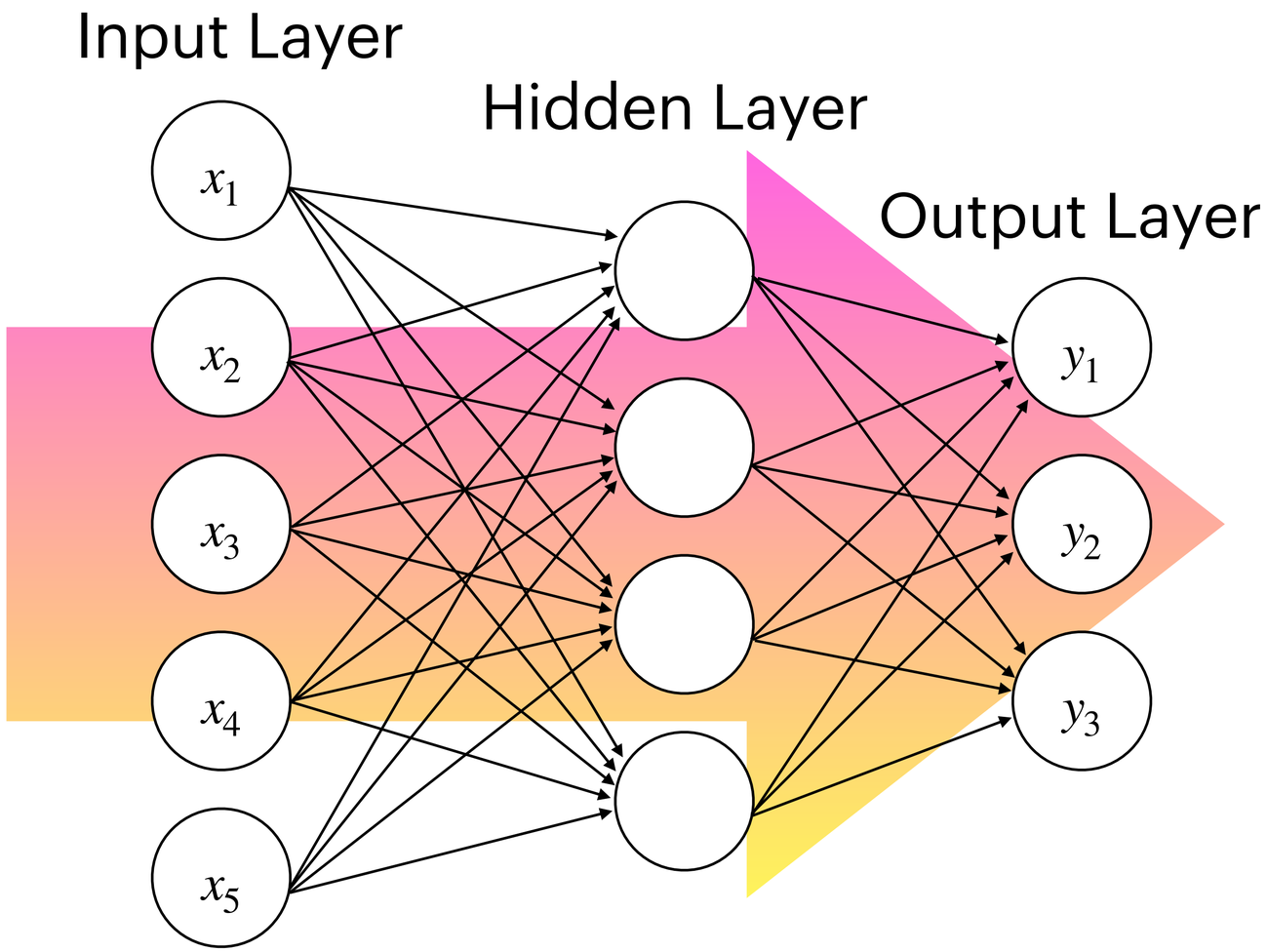
2) ViT 모델 구조

Architecture Detail: Feed-forward Neural Network



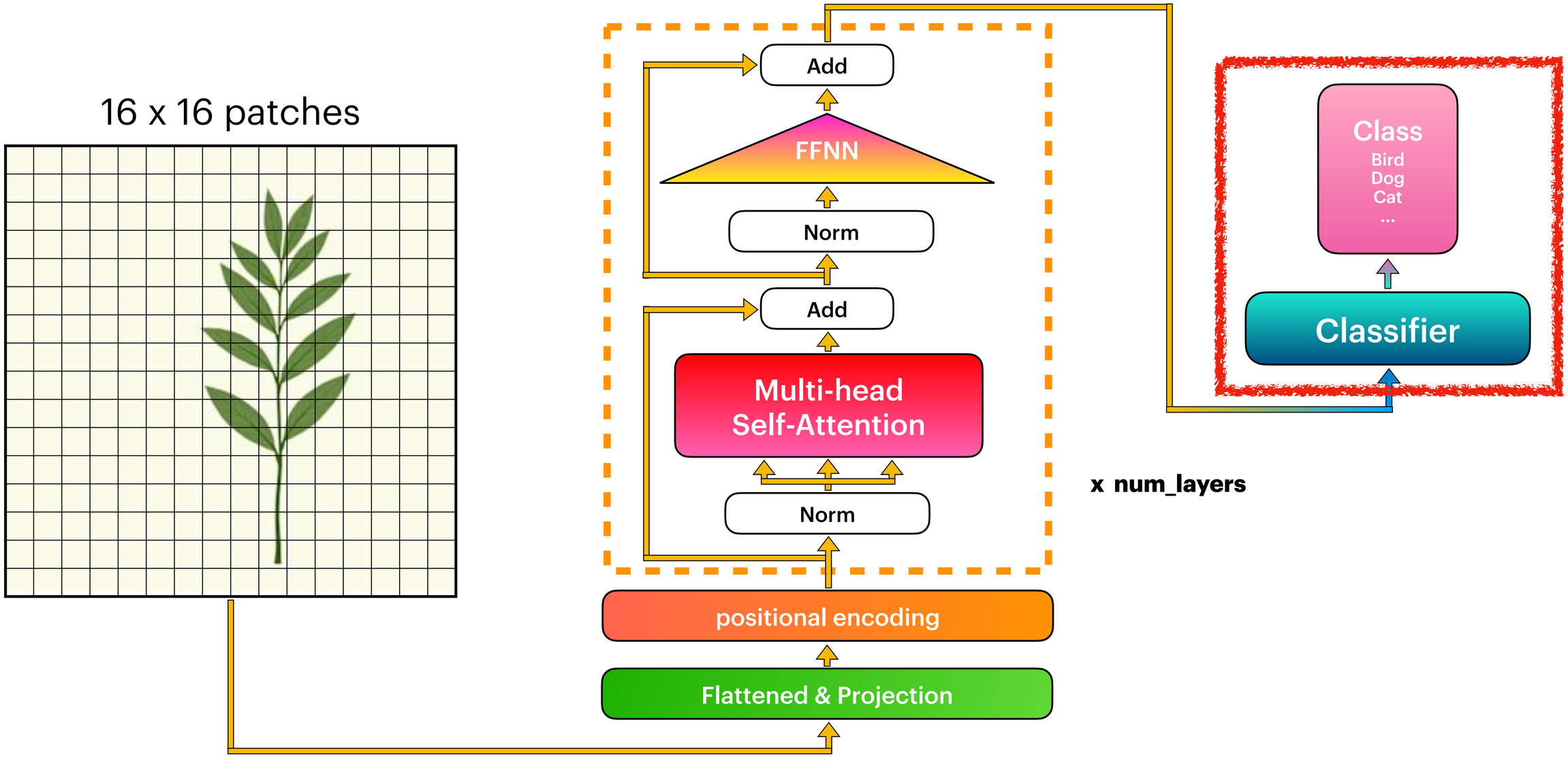
2) ViT 모델 구조

Architecture Detail: Feed-forward Neural Network



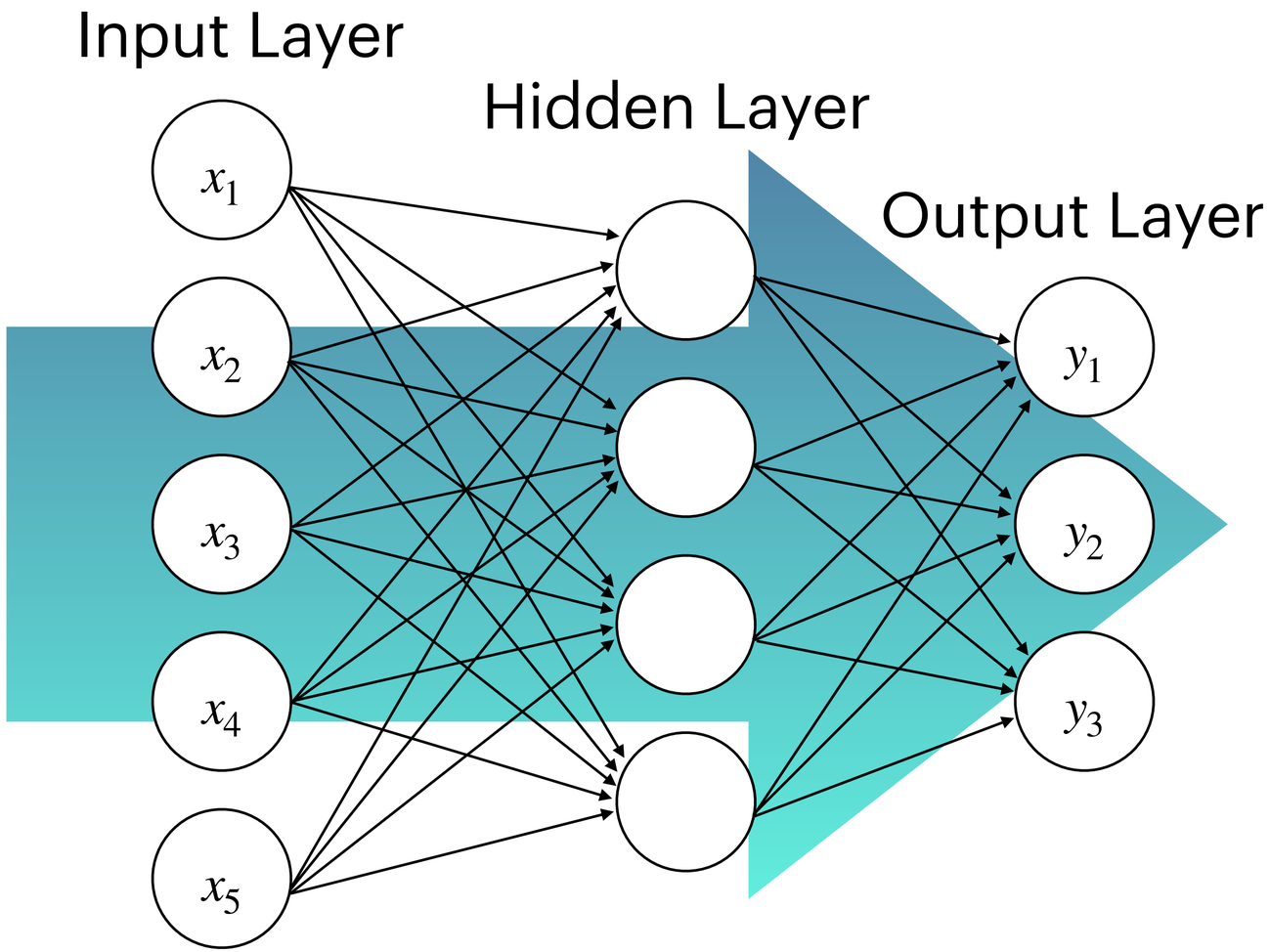
2) ViT 모델 구조

Architecture Detail: MLP Head



2) ViT 모델 구조

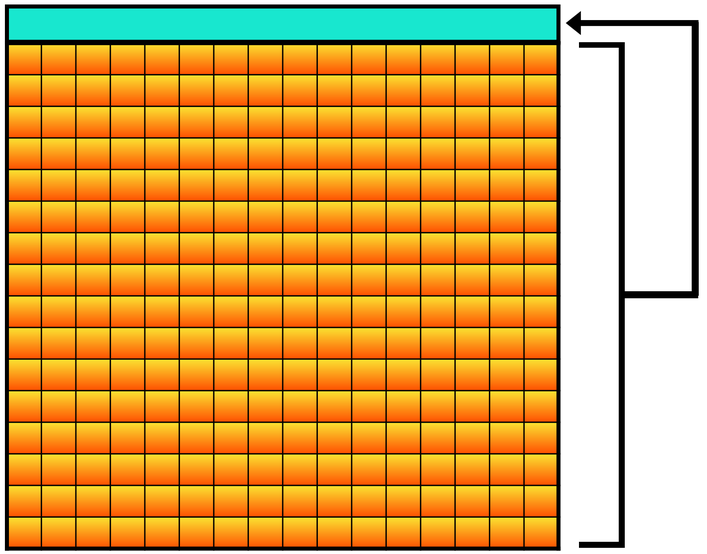
Architecture Detail: MLP Head



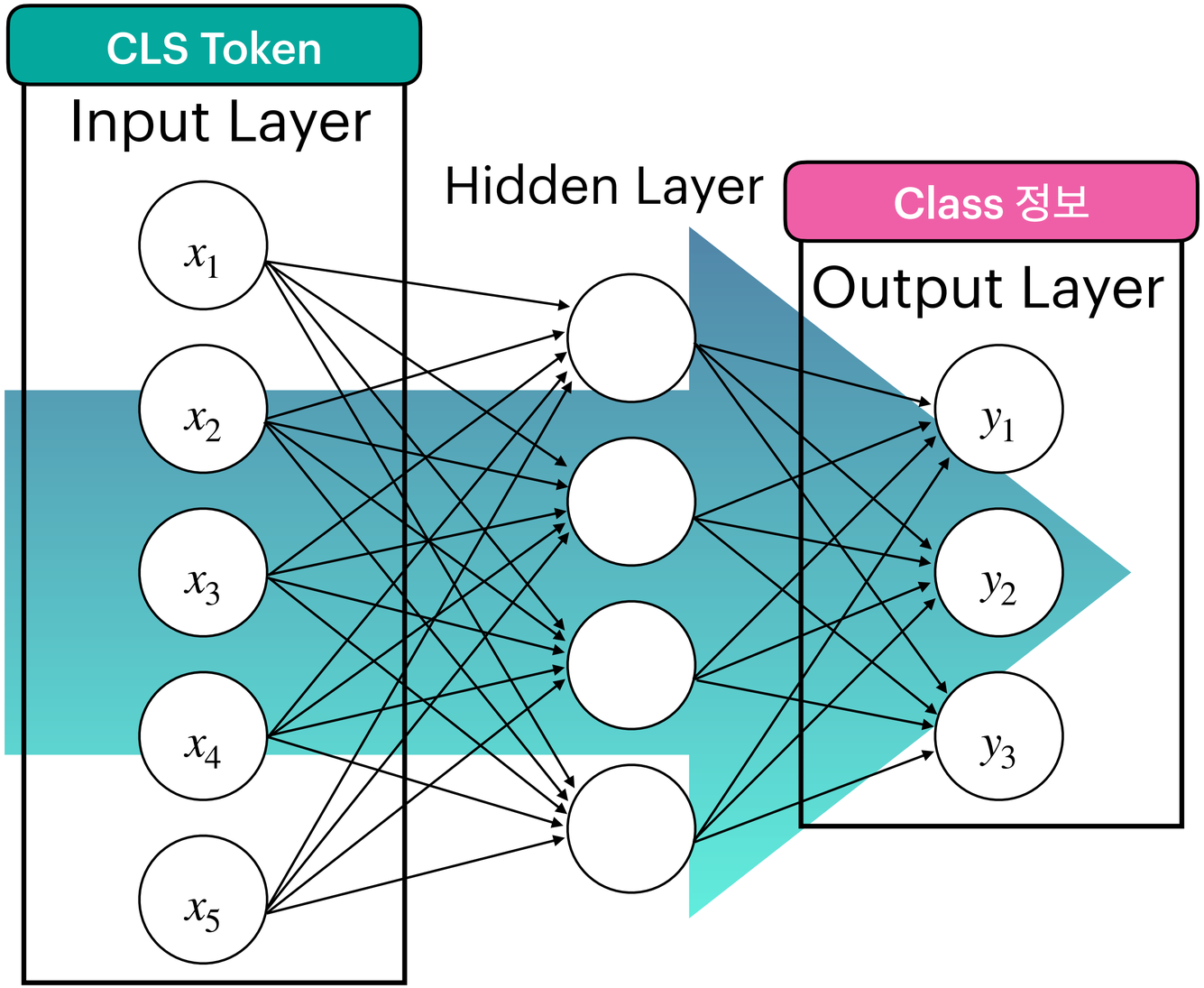
2) ViT 모델 구조

Architecture Detail: MLP Head

CLS Token: Image Patch 들의 모든 정보를 압축해 저장

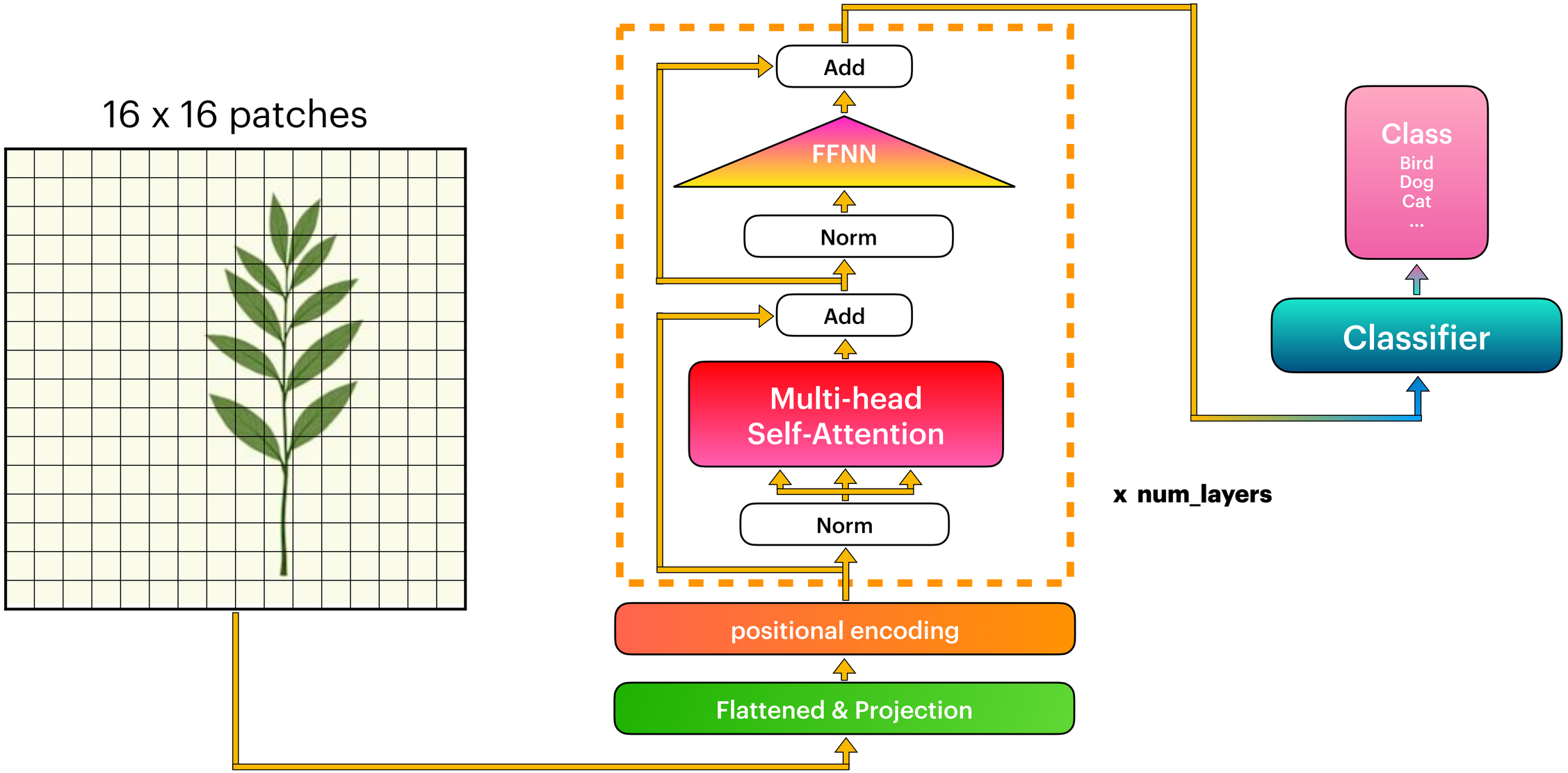


$\in \mathbb{R}^{(N+1)\times D}$



2) ViT 모델 구조

Architecture Detail: MLP Head



3) ViT의 특징

Large Scale Dataset의 필요

1,400만 ~ 3억개 이미지

3) ViT의 특징

Large Scale Dataset의 필요

Positional Encoding

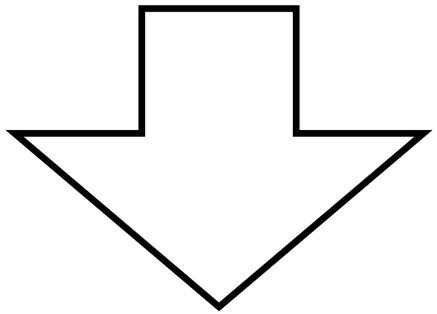
Multi-head Self-Attention

] **Data** 에 크게 의존

3) ViT의 특징

Large Scale Dataset의 필요

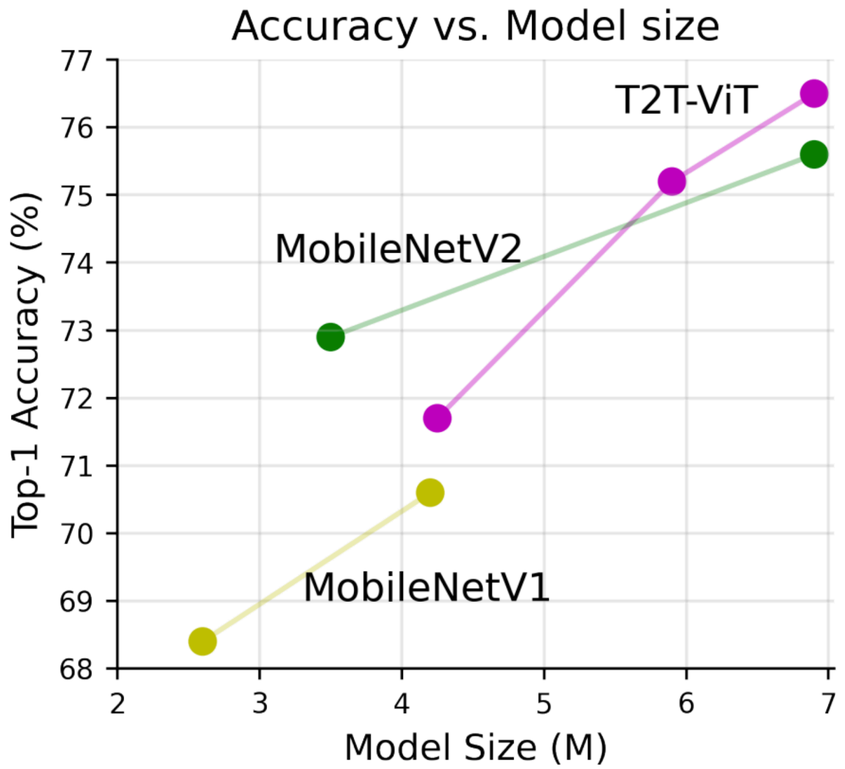
Large Scale Dataset 으로 Pre-training



Down-stream Task Dataset 으로 Fine-tuning

3) ViT의 특징

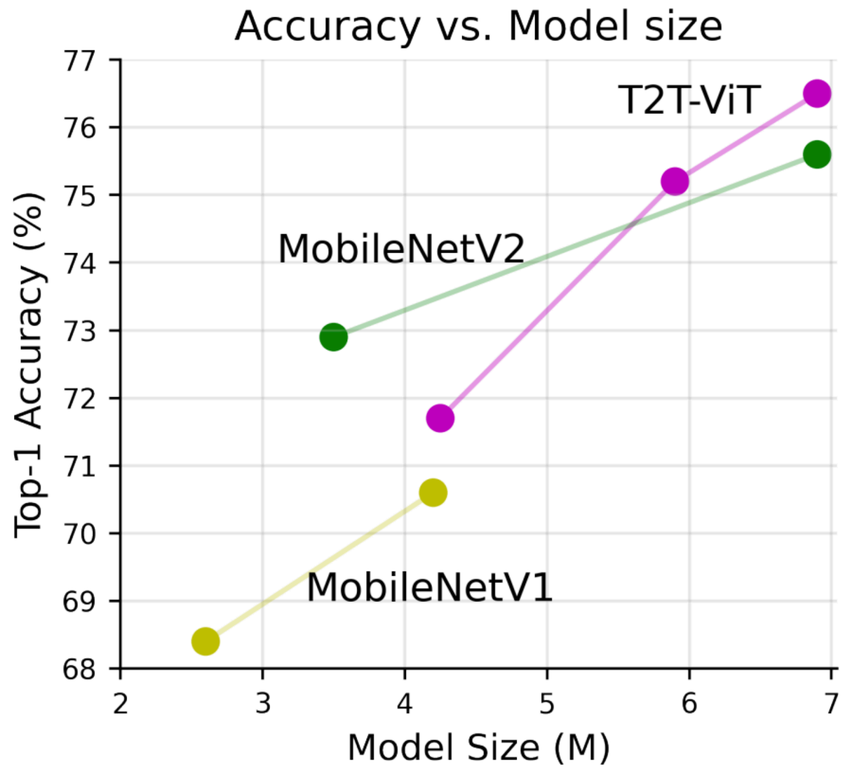
Scalability



모든 모델이 크기가 커진다고 해서 성능이 더 좋아지는 것은 아님

3) ViT의 특징

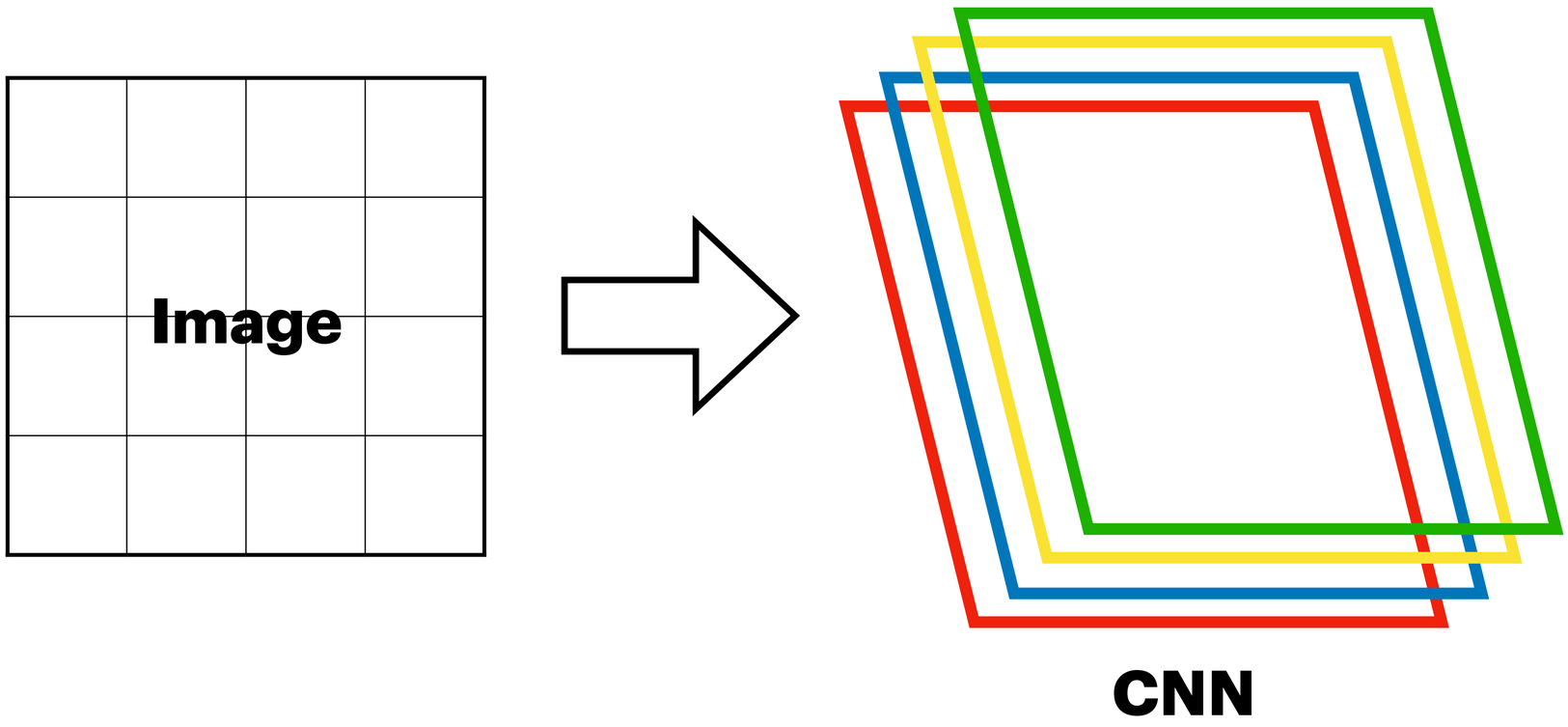
Scalability



ViT는 크기가 커져도 성능이 계속 좋아짐

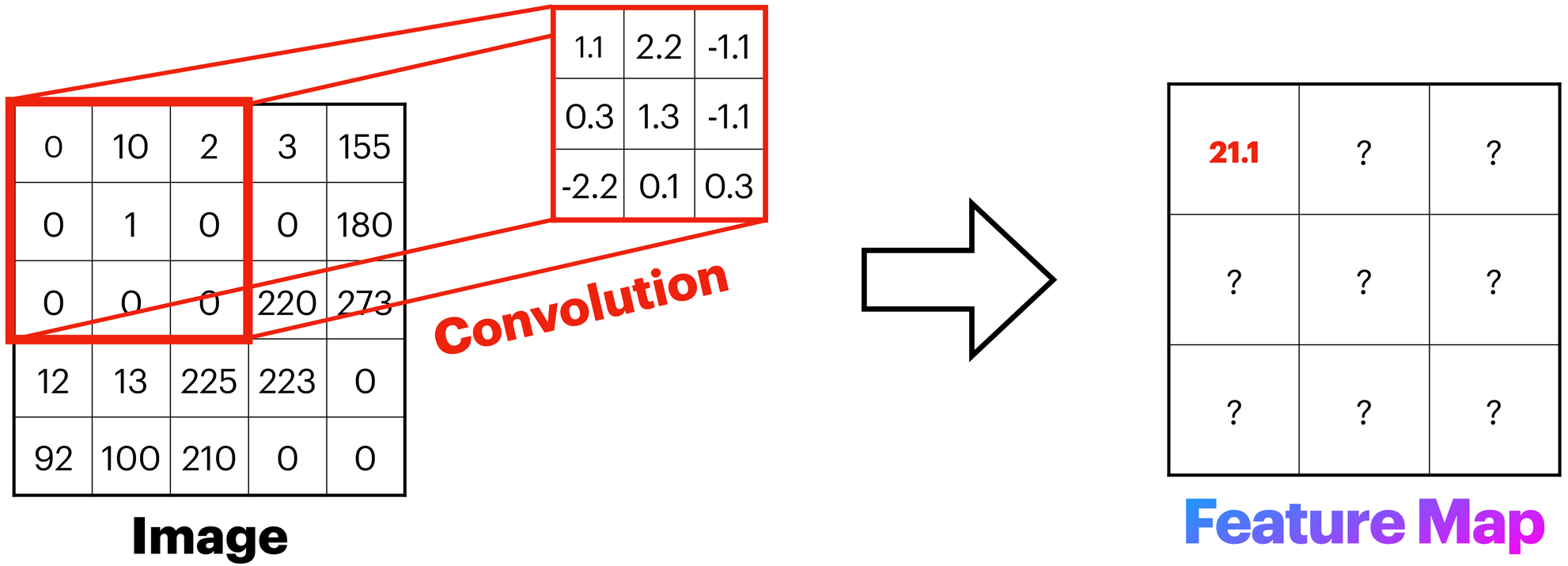
4) ViT와 CNN의 차이

Transformer Encoder 이전의 Backbone: CNN



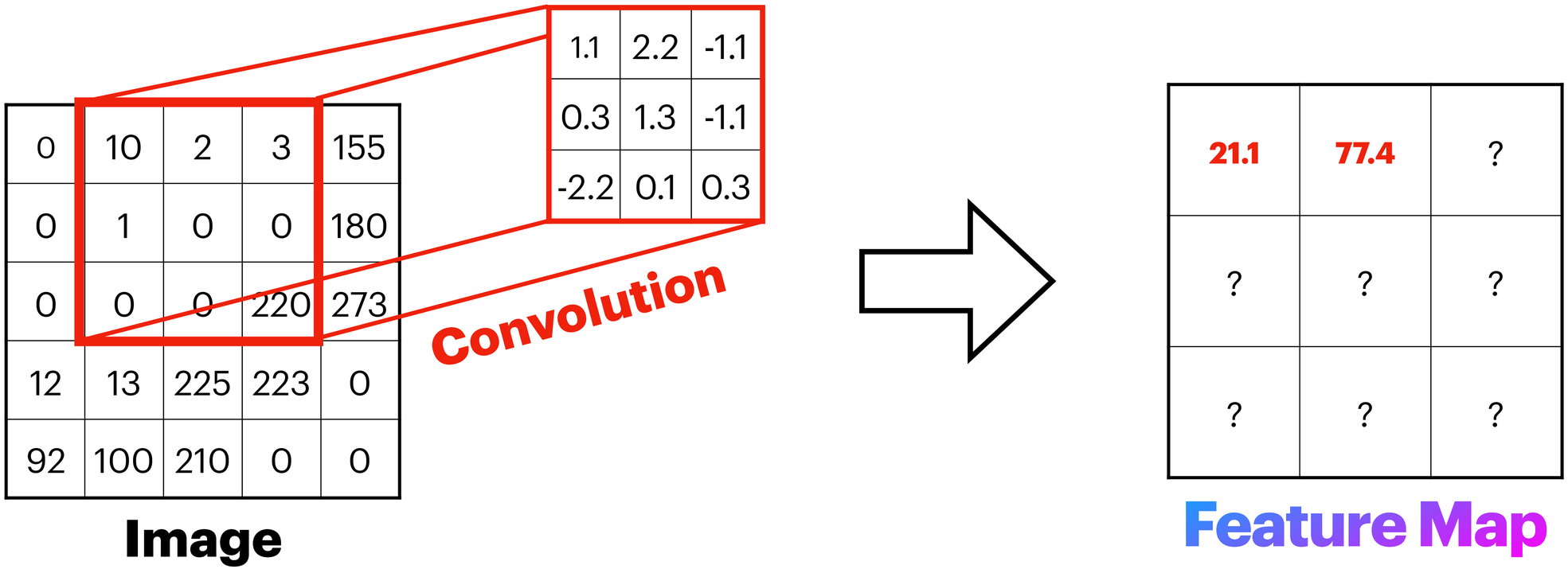
4) ViT와 CNN의 차이

Transformer Encoder 이전의 Backbone: CNN



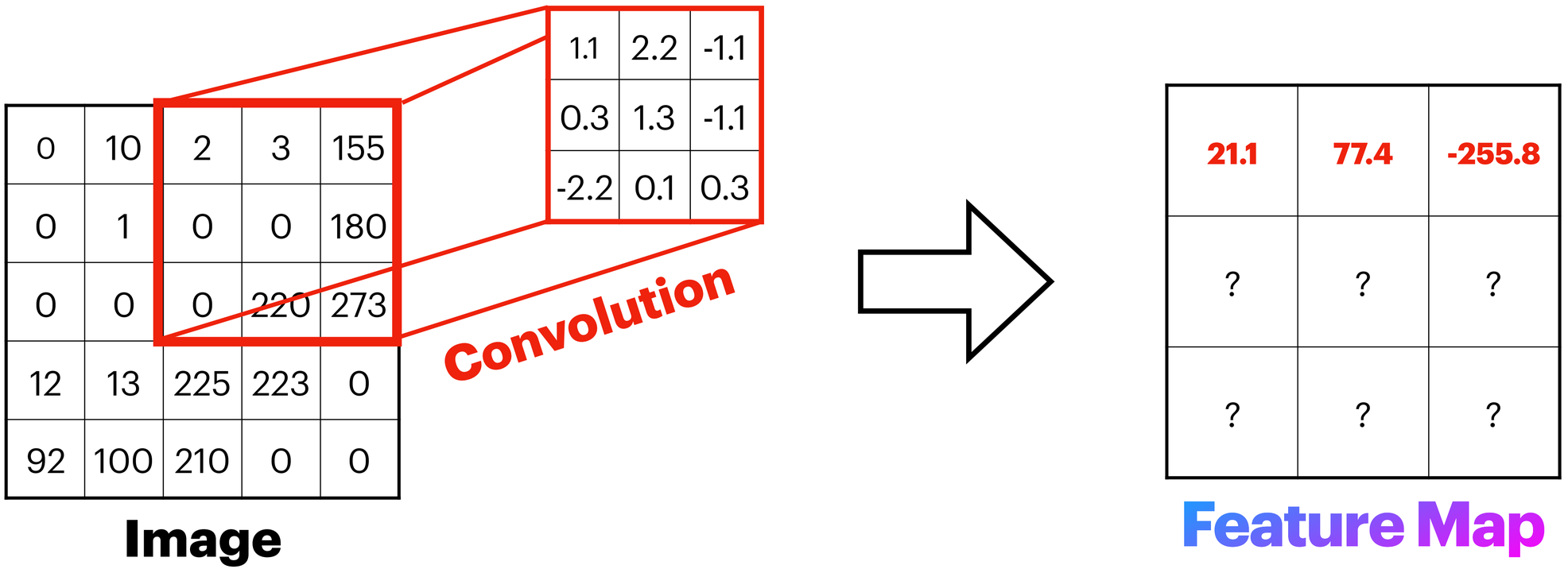
4) ViT와 CNN의 차이

Transformer Encoder 이전의 Backbone: CNN



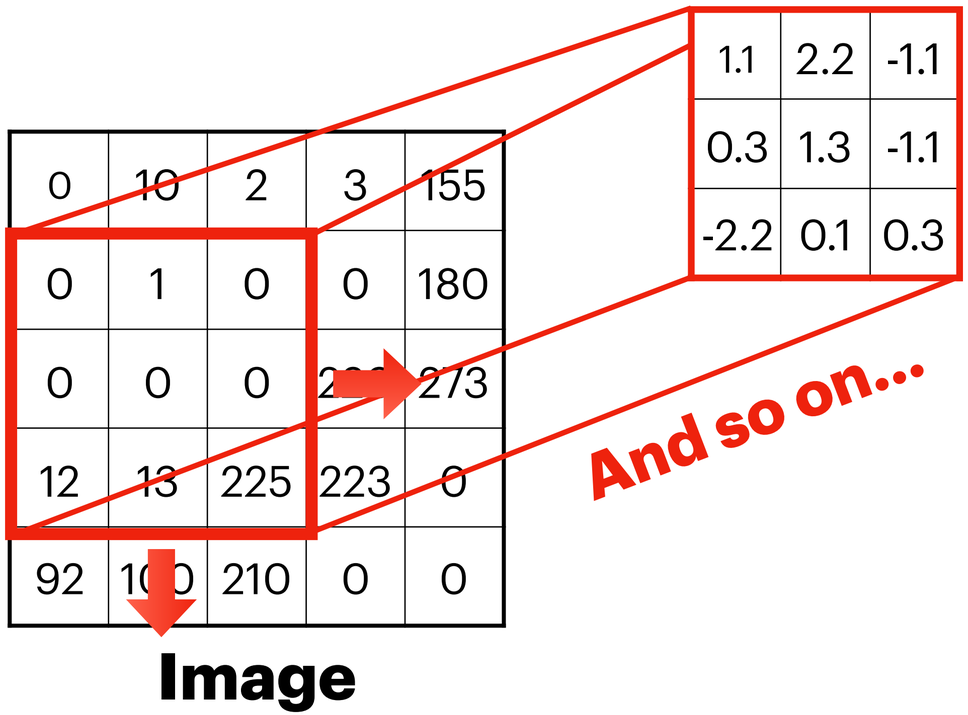
4) ViT와 CNN의 차이

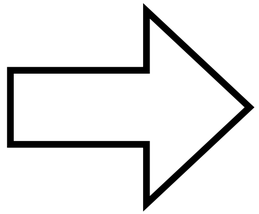
Transformer Encoder 이전의 Backbone: CNN



4) ViT와 CNN의 차이

Transformer Encoder 이전의 Backbone: CNN





21.1	**77.4**	**-255.8**
44.6	?	?
?	?	?

Feature Map

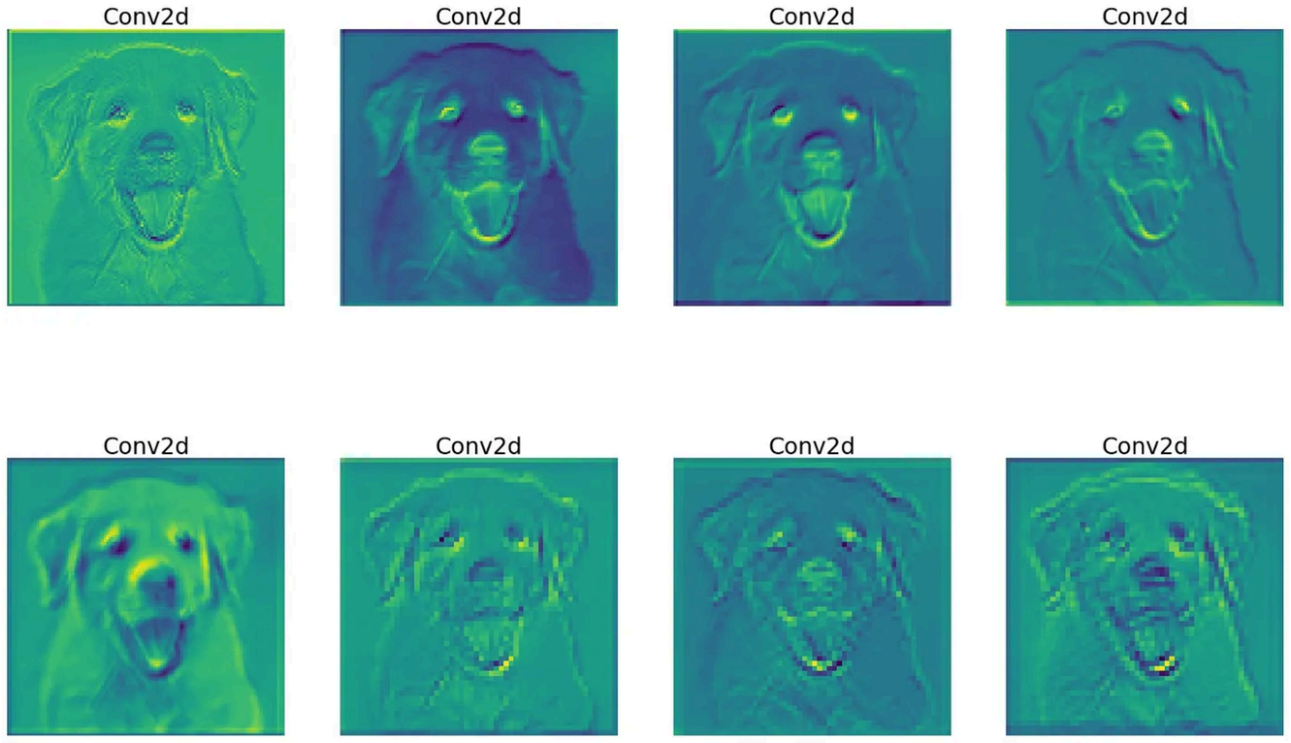
4) ViT와 CNN의 차이

Transformer Encoder 이전의 Backbone: CNN

Image

CNN

Feature Maps



images from https://ravivaishnav20.medium.com/visualizing-feature-maps-using-pytorch-12a48cd1e573

4) ViT와 CNN의 차이

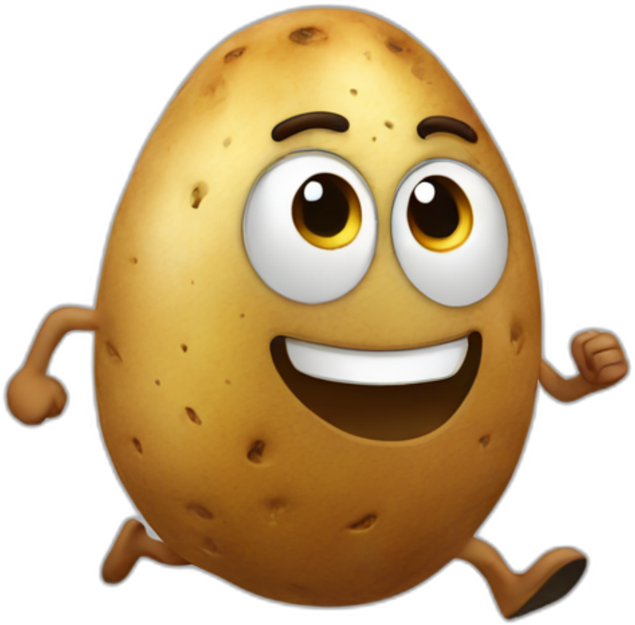
Inductive Bias

Inductive Bias 란?

4) ViT와 CNN의 차이

Inductive Bias

CNN: 대학생



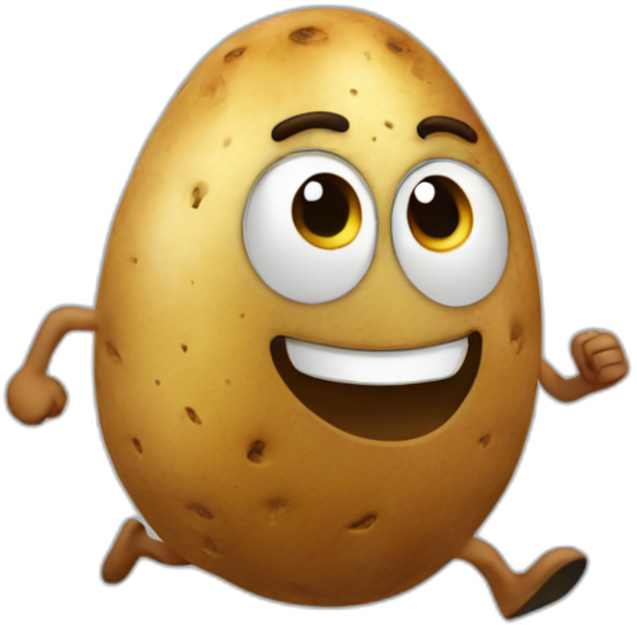
전문 지식 X

비상한 머리 X

4) ViT와 CNN의 차이

Inductive Bias

CNN: 대학생

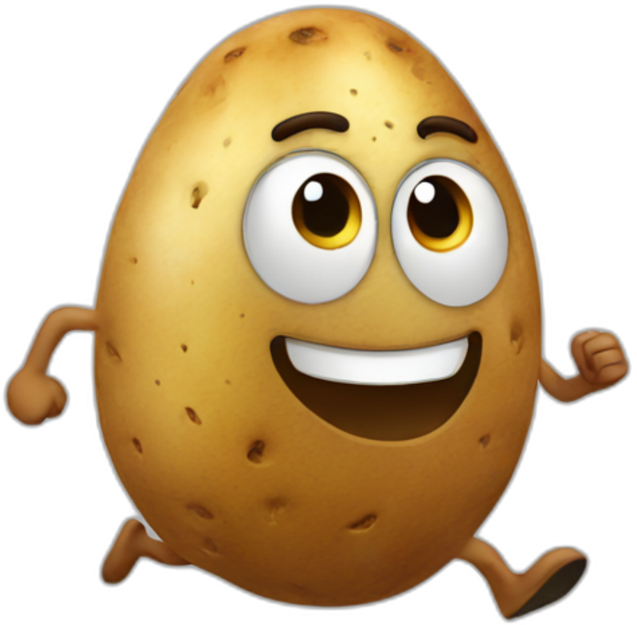


다음주가 시험인데...
어떻게 하면 시험을 잘 볼 수 있을까?

4) ViT와 CNN의 차이

Inductive Bias

CNN: 대학생

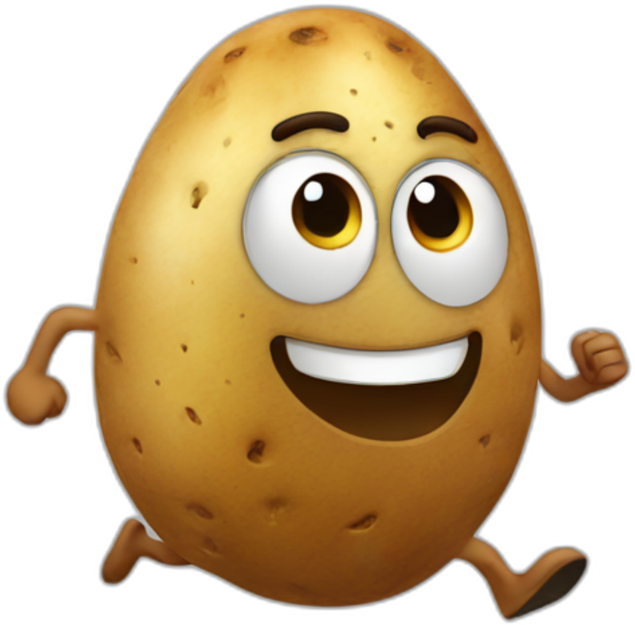


공부시간을 늘리면
성적이 오르지 않을까?

4) ViT와 CNN의 차이

Inductive Bias

CNN: 대학생

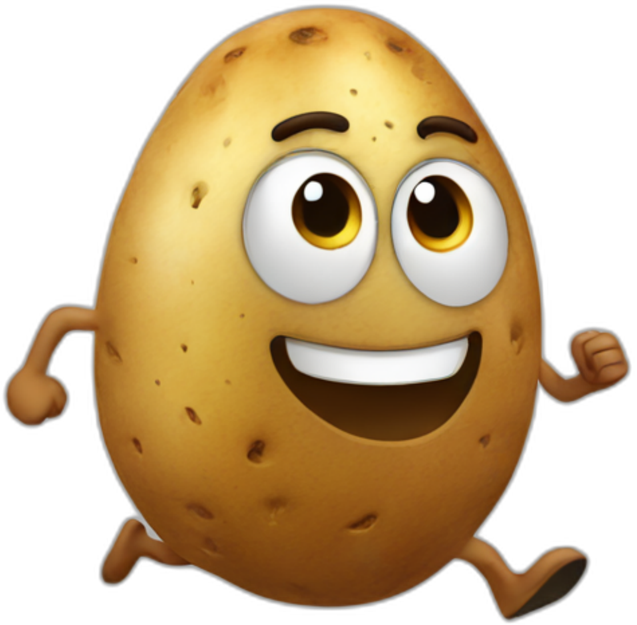


공부시간 ↑ → 성적 ↑ ?

4) ViT와 CNN의 차이

Inductive Bias

CNN: 대학생



가정, 편향된 생각 = bias

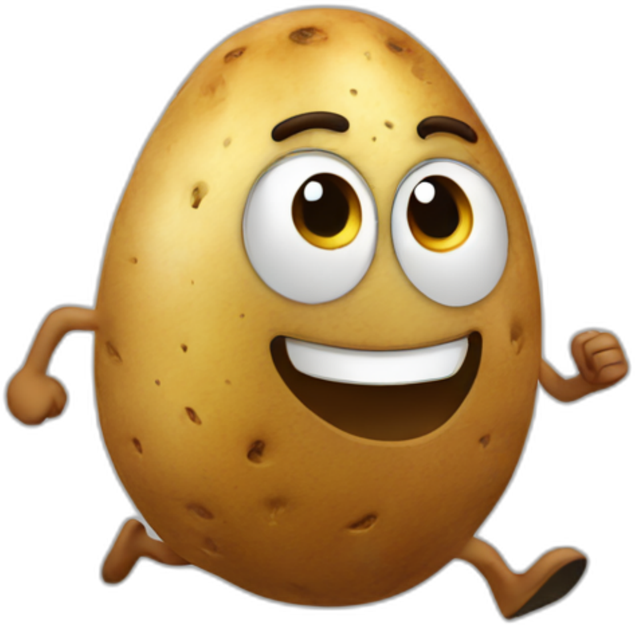
공부시간 ↑ → 성적 ↑ ?

→ 실제로도 그런지 검증이 필요

4) ViT와 CNN의 차이

Inductive Bias

CNN: 대학생



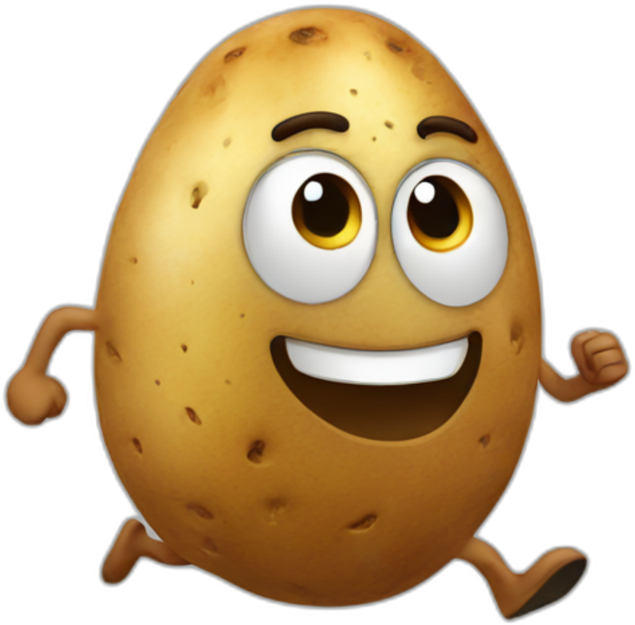
검증은 어떻게?

→ 직접 해봄

4) ViT와 CNN의 차이

Inductive Bias

CNN: 대학생



직접 해보니까 정말 그렇더라

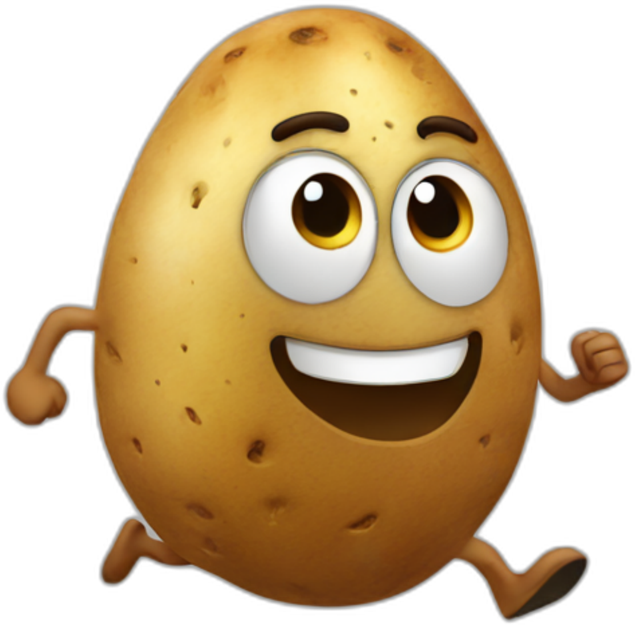
→ 공부시간이 많아지면 성적이 오른다

→ 귀납적 추론 결과 만들어진 편향 = **Inductive Bias**

4) ViT와 CNN의 차이

Inductive Bias

CNN: 대학생



1) Data가 적거나

2) Model이 작으면

효과적!

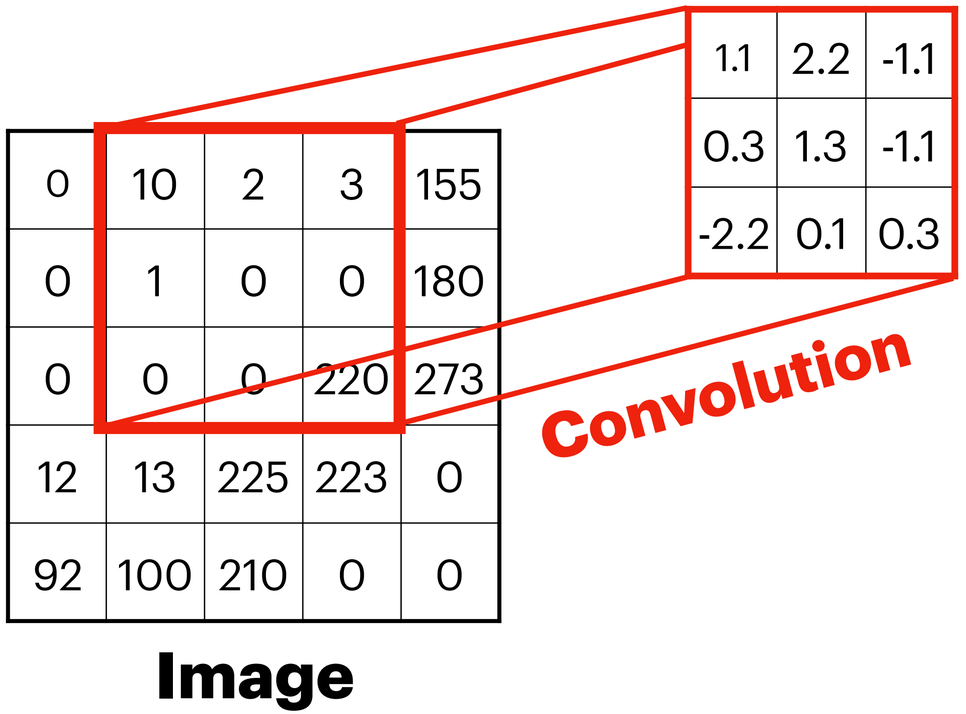
→ CNN이 작은 Dataset에 대해선 성능 더 좋음

4) ViT와 CNN의 차이

Inductive Bias

CNN의 대표적인 Inductive Bias

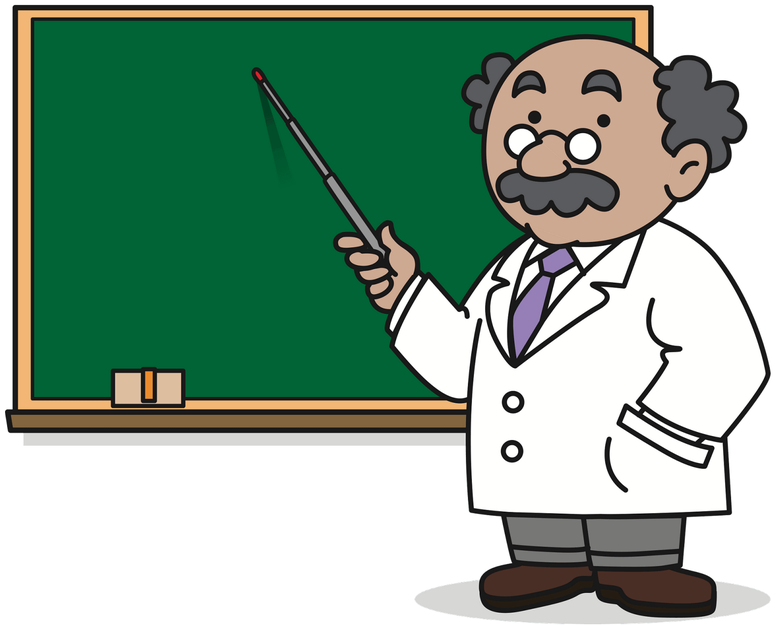
1) Locality
2) 2D Neighborhood Structure
3) Translation Equivariance



4) ViT와 CNN의 차이

Inductive Bias

ViT: 통계학 교수님



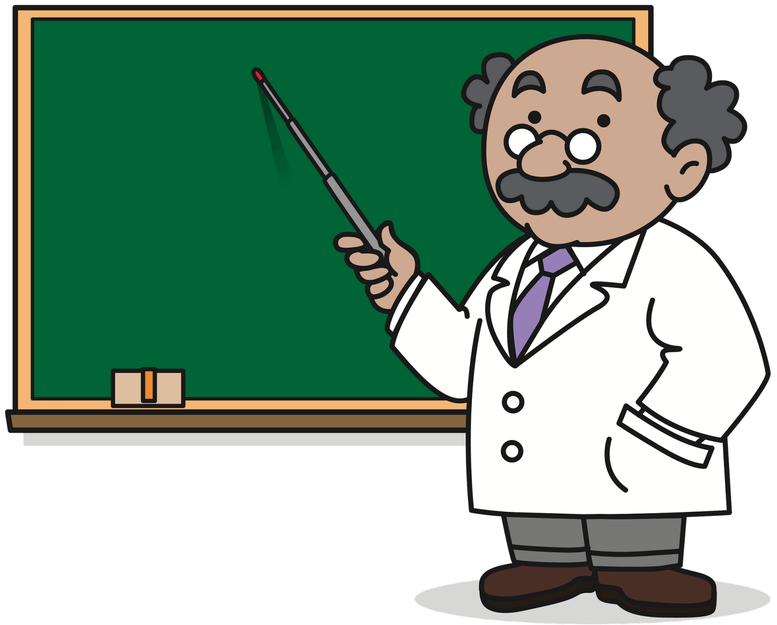
전문 지식 O

비상한 머리 O

4) ViT와 CNN의 차이

Inductive Bias

ViT: 통계학 교수님

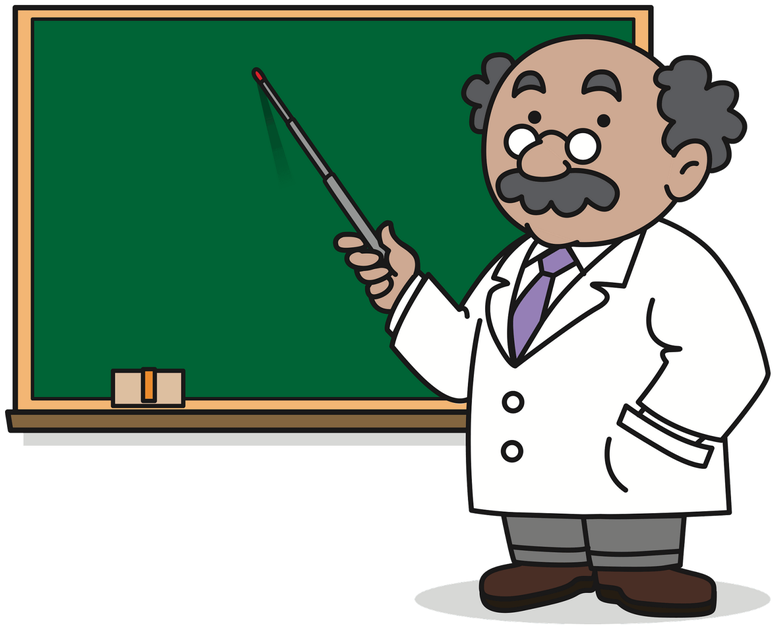


“과학적 분석”을 중시
귀납적 가정 거의 X
직접 데이터를 보고 분석

4) ViT와 CNN의 차이

Inductive Bias

ViT: 통계학 교수님



→ 실제 수 많은 데이터를 바탕으로 분석

→ 경험에 근거한 개인적 편향 없이 데이터에서 숨겨진 패턴 분석

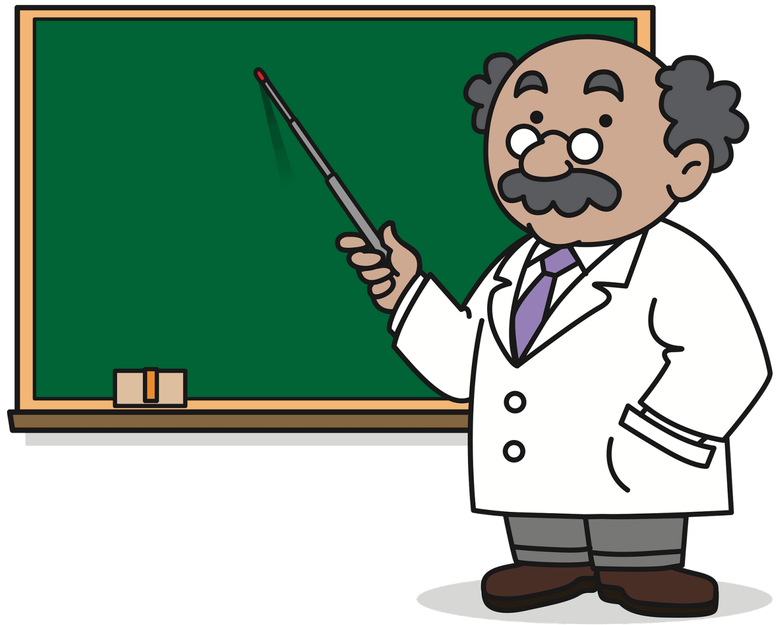
→ 데이터의 크기가 클 수록 분석한 결과가 정확할 확률 ↑

→ 일반화 성능 ↑

4) ViT와 CNN의 차이

Inductive Bias

ViT: 통계학 교수님



Dataset의 크기가 충분히 크면

ViT의 성능이 압도적

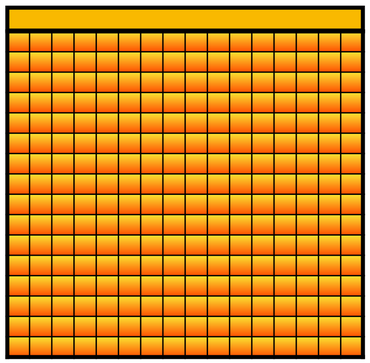
4) ViT와 CNN의 차이

Inductive Bias

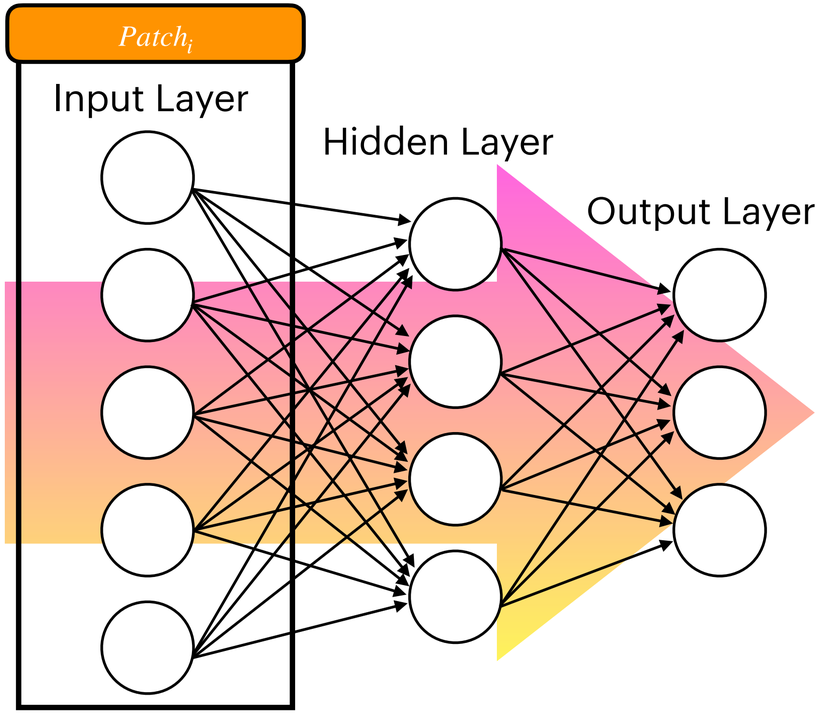
ViT의 Inductive Bias

1) Locality and Translation Equivariance in MLP Layers

2) 2D Neighborhood Structure in Patching & Fine-tuning Positional Encoding



$\in \mathbb{R}^{(N+1)\times D}$



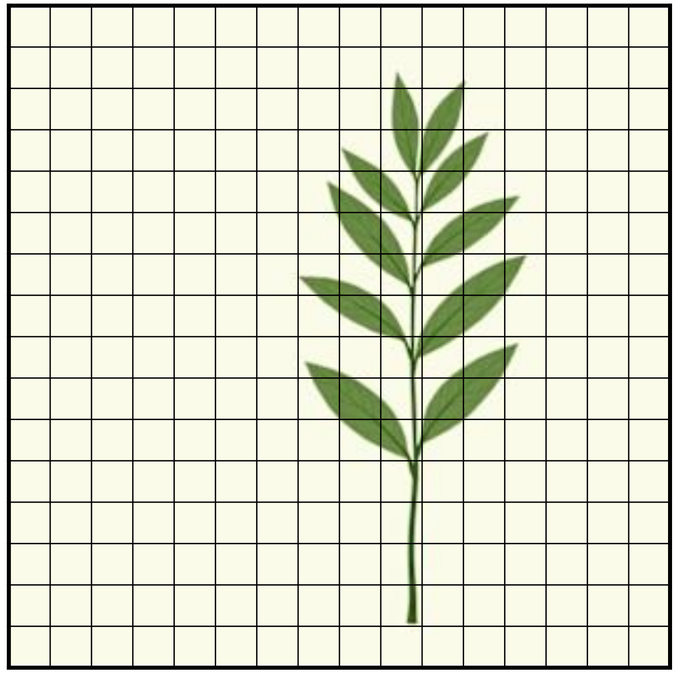
4) ViT와 CNN의 차이

Inductive Bias

ViT의 Inductive Bias

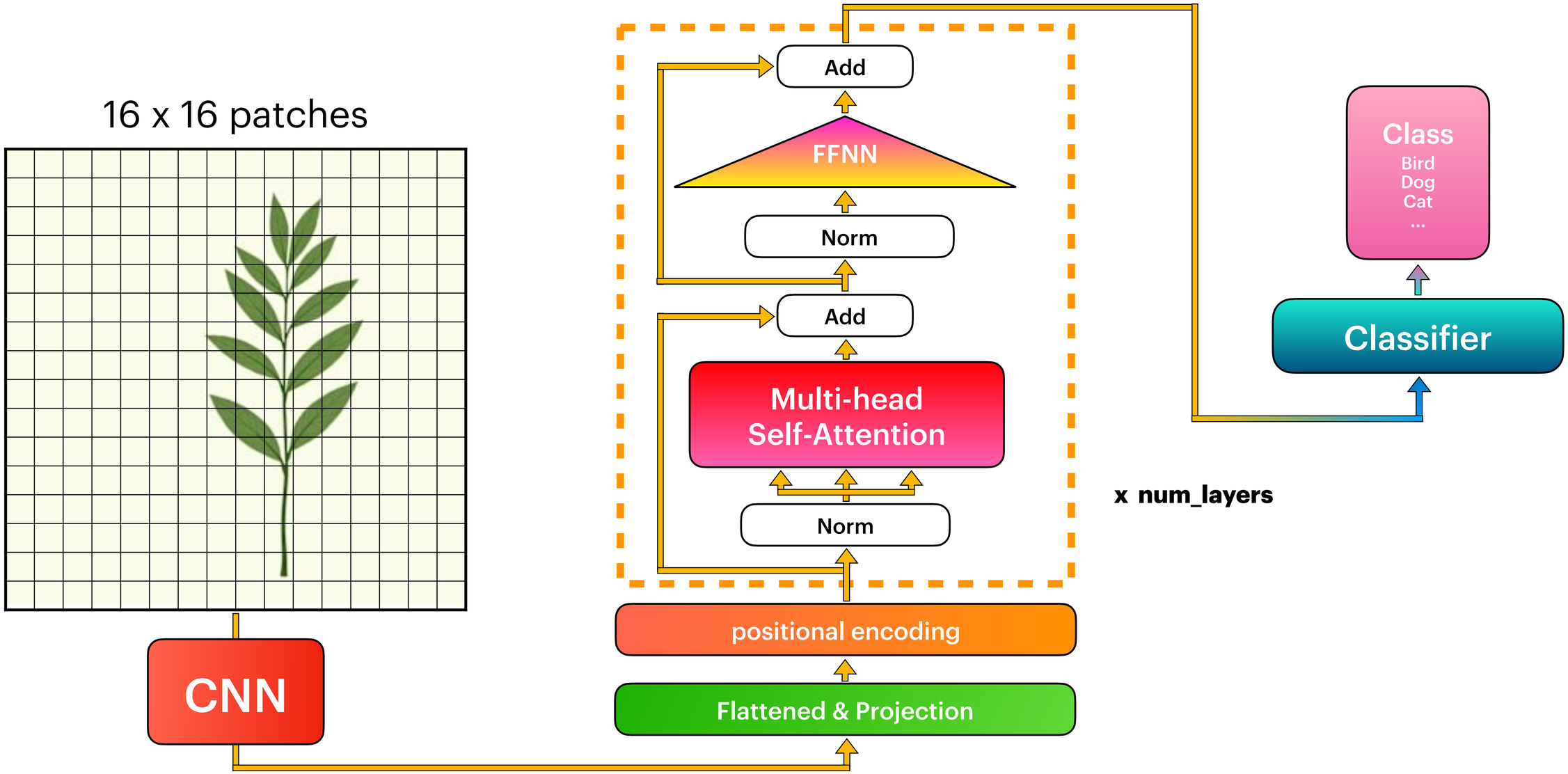
1) Locality and Translation Equivariance in MLP Layers

2) 2D Neighborhood Structure in Patching & Fine-tuning Positional Encoding



4) ViT 와 CNN의 차이

Hybrid Architecture: ViT + CNN



Q & A