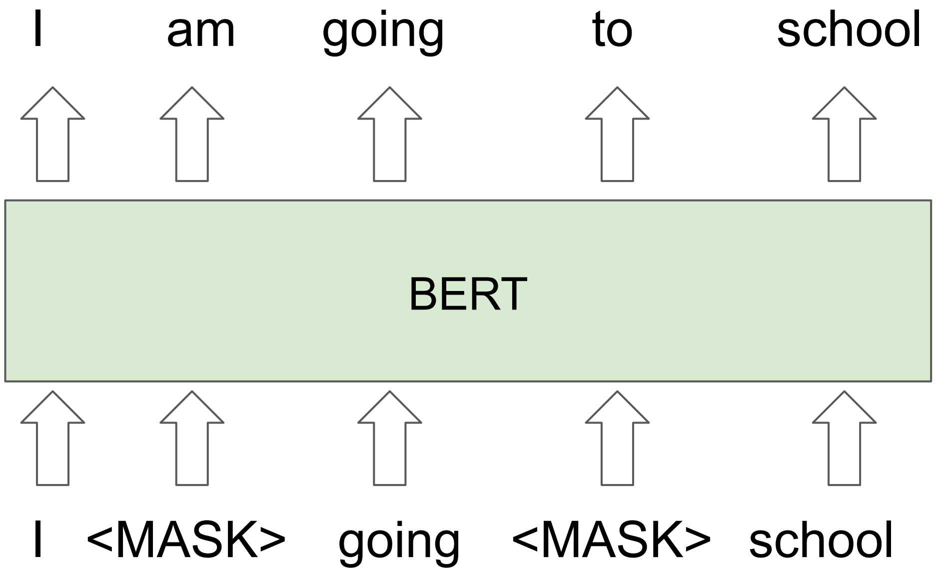
# BEiT: BERT Pre-training of Image Transformers

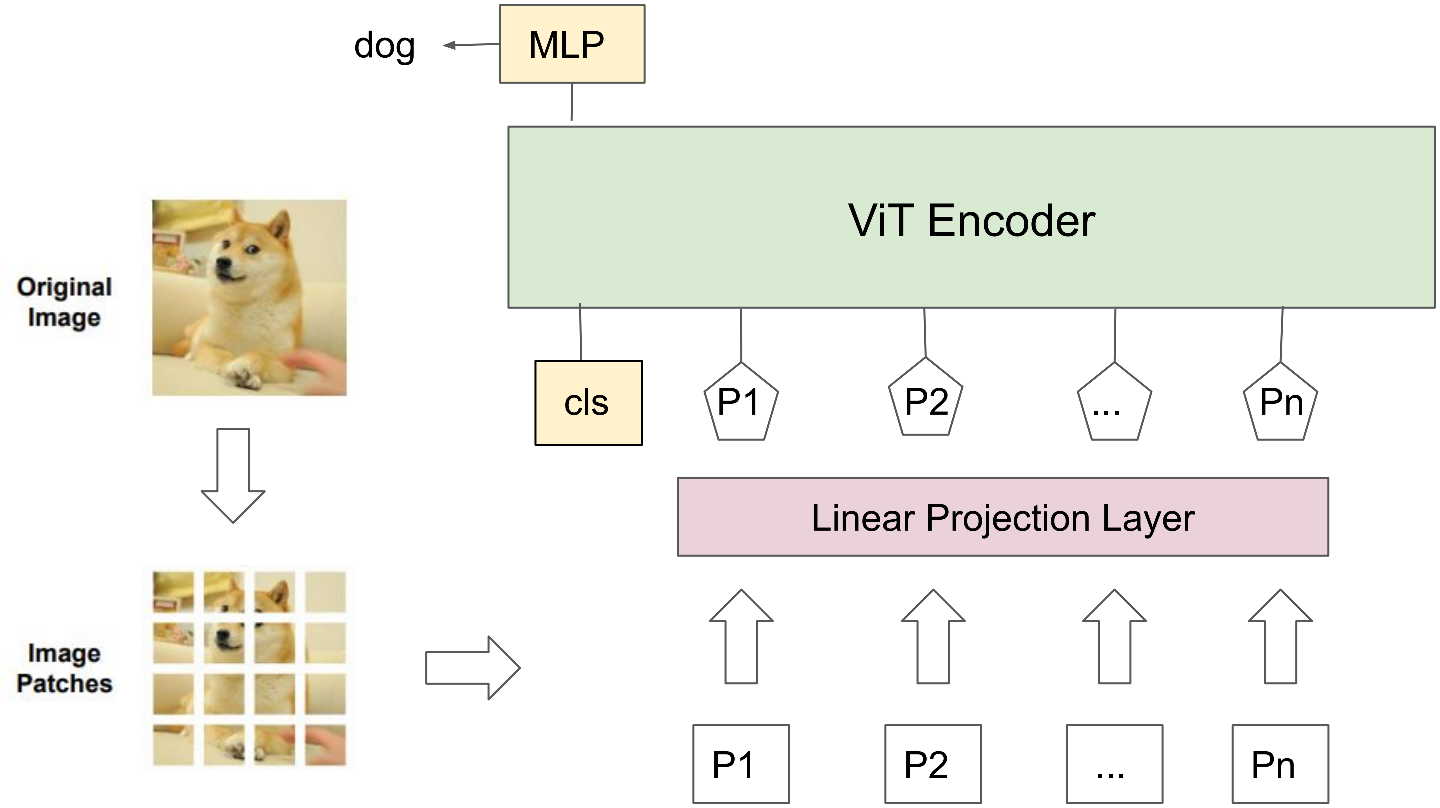
Presenter: Mert Kilickaya
15 June 2022

# BERT Pre-training



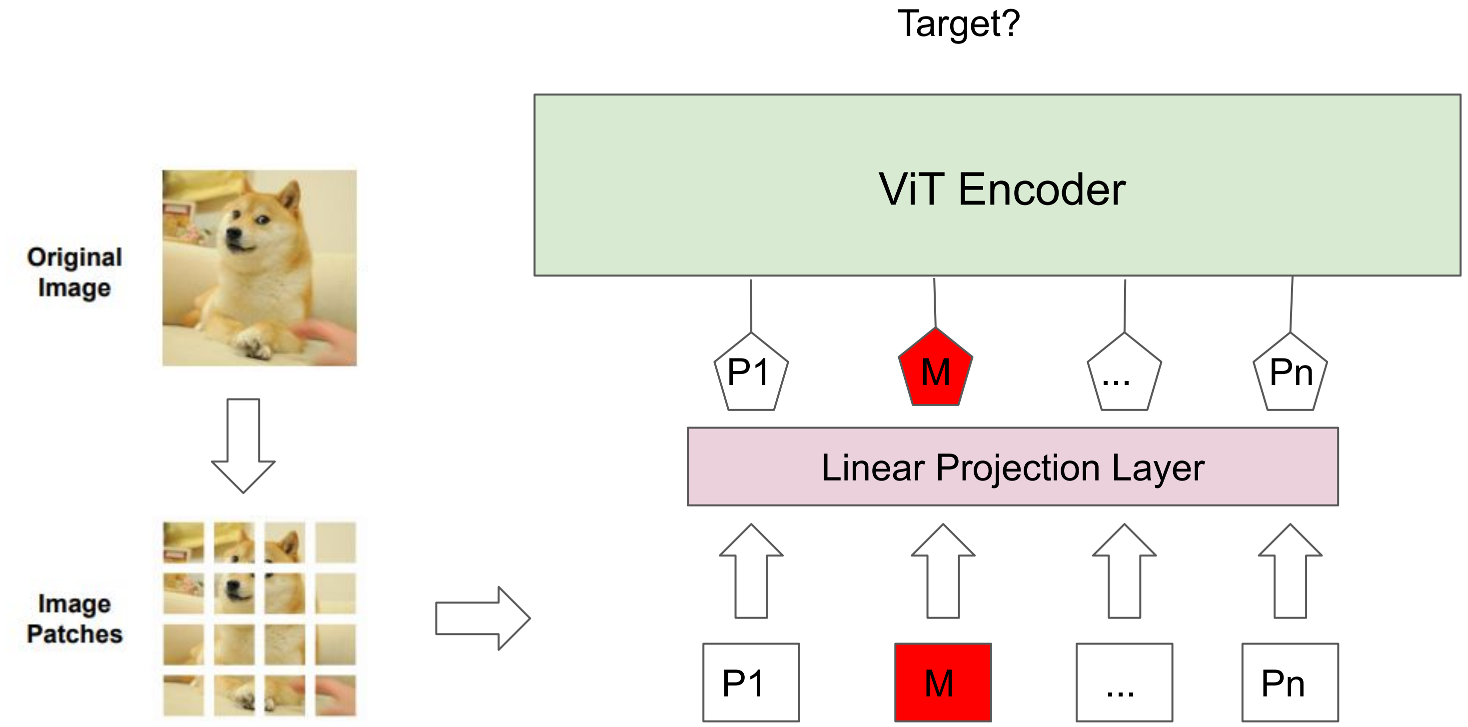

Masked Language Modelling predicts (randomly) maskes words for prediction as a pretext task.

# Image Transformers



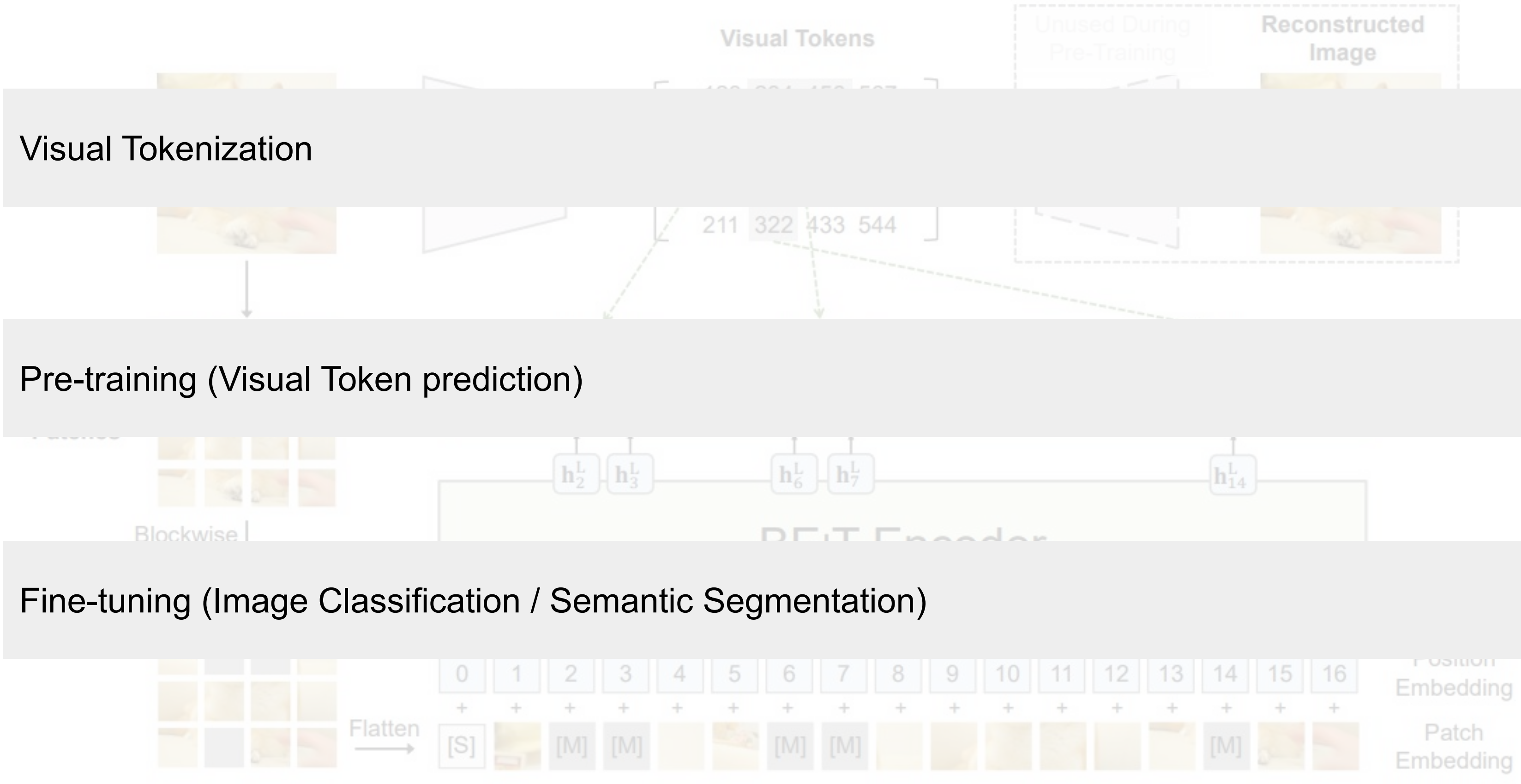

Vision Transformers (ViT) learn to encode non-overlapping image patch embeddings for classification

# BERT Pre-training of Image Transformers: Target?




Visual world -*unlike language*- is not tokenized: No well-defined discrete target for masking

# BERT Pre-training of Image Transformers: 3 Steps




Visual Tokenization

Pre-training (Visual Token prediction)

Fine-tuning (Image Classification / Semantic Segmentation)

# BERT Pre-training of Image Transformers-1: Tokenization

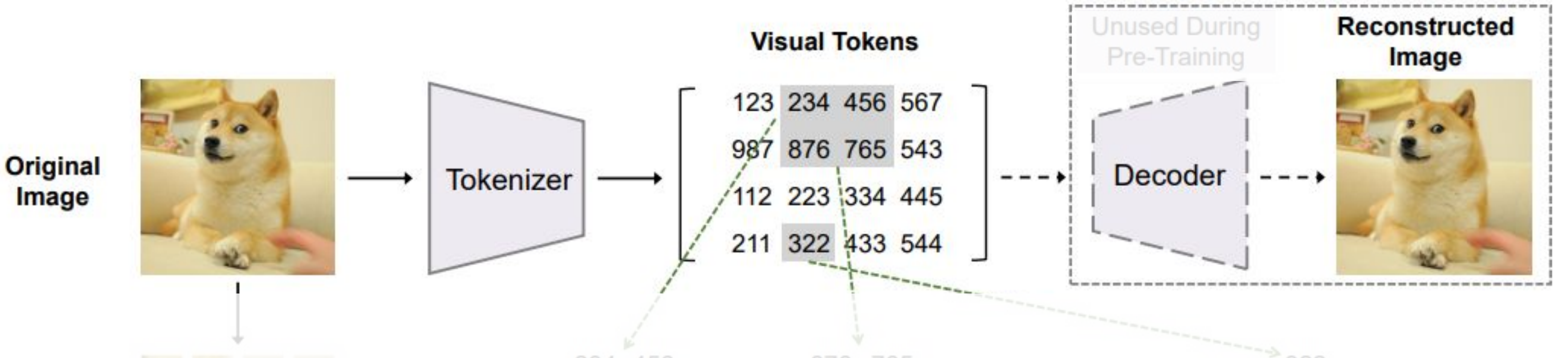


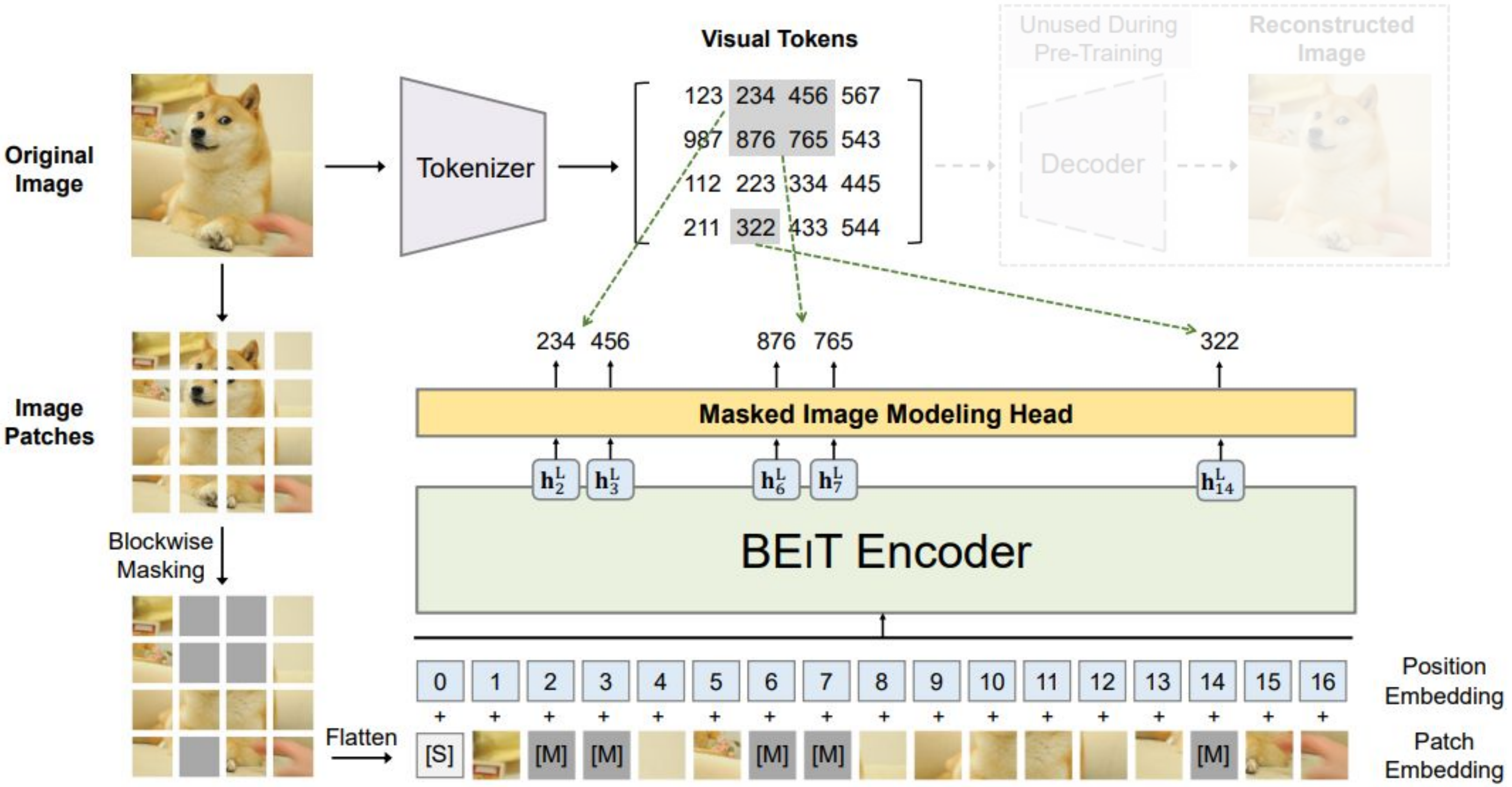

Image Tokenizer is an Off-the-Shelf discrete Variational Autoencoder (d-VAE*)

Compress an image into M x M discrete codes (Visual Tokens), then reconstruct it via Decoder

d-VAE needs to encode part-level object semantics (i.e. dog ear) to satisfy reconstruction

*[Zero-Shot Text-to-Image Generation](#) (OpenAI, ICML'21)

# BERT Pre-training of Image Transformers-2: Pre-training



# BERT Pre-training of Image Transformers-2: Pre-training

Architecture: ViT-Base | 12-layer Transformer | 768 hidden size | 12 attention head per-layer

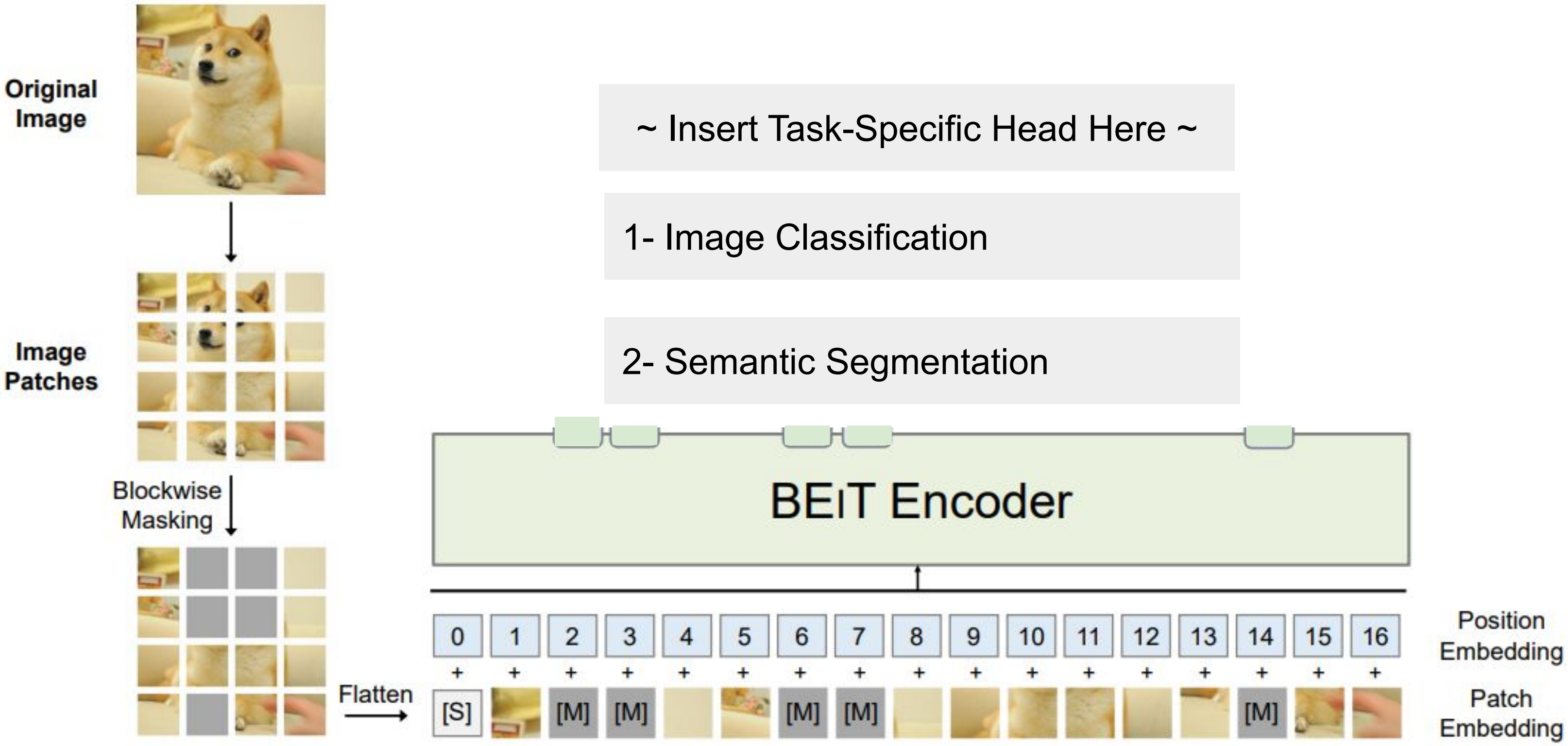
Input image resolution: (**224 x 224**) | #Patches: (**14 x 14**) | Patch resolution: (**16 x 16**)

Visual Token Vocabulary Size: **|V| = 8192**

**40%** of all image patches (**196 x 0.4 ~= 75**) are masked out during pre-training

**1.2M** ImageNet-1k | **800** epochs | **2k** Batch Size | 5 days on 16 Tesla V 100 32 GB

# BERT Pre-training of Image Transformers-3: Fine-tuning



# Experiment-1: ImageNet Classification

| Models | CIFAR-100 | ImageNet |
|---|---|---|
| *Training from scratch (i.e., random initialization)* | | |
| $\text{ViT}_{384}$ (Dosovitskiy et al., 2020) | 48.5* | 77.9 |
| DeiT (Touvron et al., 2020) | n/a | 81.8 |
| *Supervised Pre-Training on ImageNet-1K (using labeled data)* | | |
| $\text{ViT}_{384}$ (Dosovitskiy et al., 2020) | 87.1 | 77.9 |
| DeiT (Touvron et al., 2020) | 90.8 | 81.8 |
| *Self-Supervised Pre-Training on ImageNet-1K (without labeled data)* | | |
| iGPT-1.36B† (Chen et al., 2020a) | n/a | 66.5 |
| $\text{ViT}_{384}$-JFT300M‡ (Dosovitskiy et al., 2020) | n/a | 79.9 |
| DINO (Caron et al., 2021) | 91.7 | 82.8 |
| MoCo v3 (Chen et al., 2021) | 87.1 | n/a |
| BEiT (ours) | 90.1 | **83.2** |
| *Self-Supervised Pre-Training, and Intermediate Fine-Tuning on ImageNet-1K* | | |
| BEiT (ours) | **91.8** | **83.2** |

***Supervised-fine-tuning DeiT vs. BEiT***: BeiT improves 81.8 -> 83.2

***DINO vs. BEiT for Image Classification***: *BEiT is slightly better 82.8 -> 83.2*

# Experiment-2: Large-scale ImageNet Classification

| Models | Model Size | Image Size | ImageNet |
|---|---|---|---|
| *Self-Supervised Pre-Training on ImageNet-1K (without labeled data)* | | | |
| iGPT-1.36B† (Chen et al., 2020a) | 1.36B | $224^2$ | 66.5 |
| $\text{ViT}_{384}$-B-JFT300M‡ (Dosovitskiy et al., 2020) | 86M | $384^2$ | 79.9 |
| DINO-B (Caron et al., 2021) | 86M | $224^2$ | 82.8 |
| BEIT-B (ours) | 86M | $224^2$ | 83.2 |
| $\text{BEIT}_{384}$-B (ours) | 86M | $384^2$ | 84.6 |
| BEIT-L (ours) | 307M | $224^2$ | 85.2 |
| $\text{BEIT}_{384}$-L (ours) | 307M | $384^2$ | **86.3** |

***(224 x 224) vs (384 x 384)****:* Larger-resolution resolution helps BEiT 83.2 -> 84.6

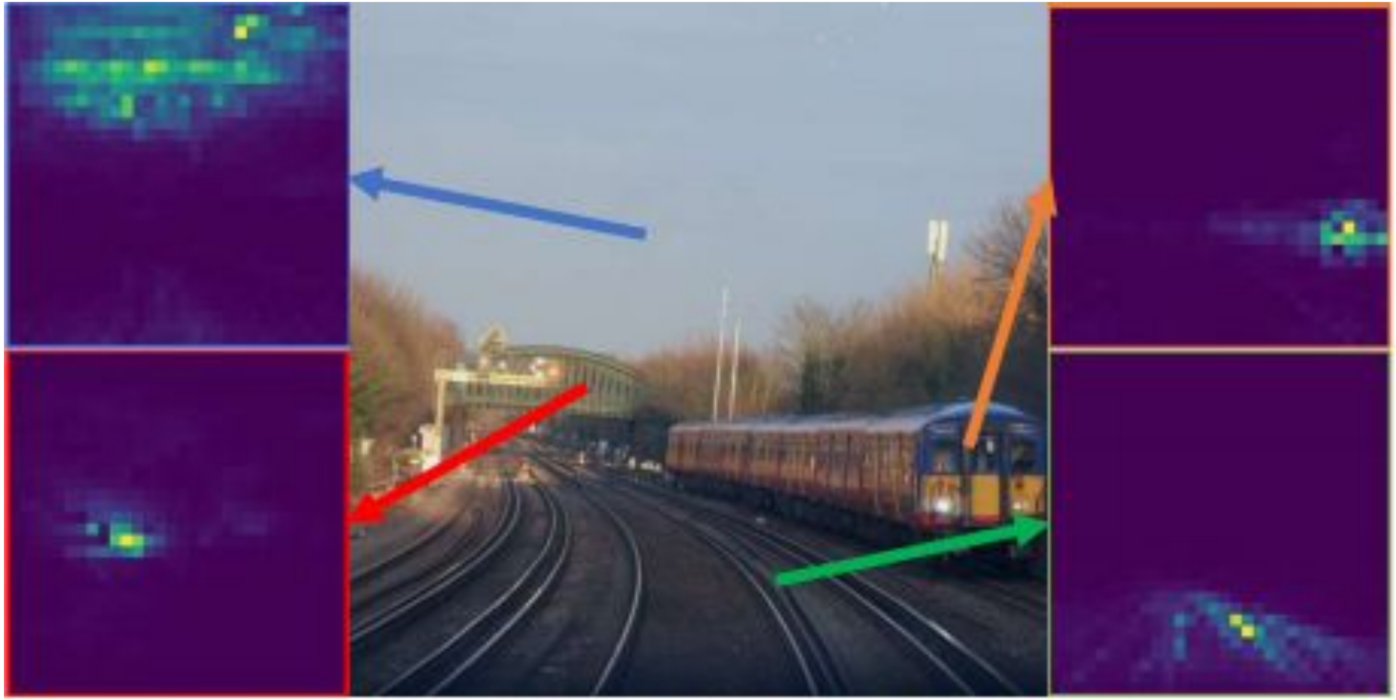
Larger embedding size helps BEiT even further 84.6 -> 86.3

# Experiment-3: Semantic Segmentation on ADE-20K

| Models | mIoU |
|---|---|
| Supervised Pre-Training on ImageNet | 45.3 |
| DINO (Caron et al., 2021) | 44.1 |
| BEiT (ours) | 45.6 |
| BEiT + Intermediate Fine-Tuning (ours) | **47.7** |

***DINO vs. BEiT for Semantic Segmentation****:* BEiT is significantly better 44.1 -> 45.6

# Qualitative: Segmentation Emerges from Pre-training



*Pixel-level attention separates/segments different object instances: Sky, Bridge, Rails, Train*

# Summary of BEiT

BEiT is pre-trained by predicting discrete visual tokens of masked patches

Visual tokens are obtained from an Off-the-Shelf compression model (d-VAE) via reconstruction

BEiT improves concurrent work of DINO for ImageNet classification and ADE-20K semantic segmentation

# Discussion

Is BEiT sensitive to the vocabulary size? Trade-off? (**|V| = 8192**)

Too high |V| -> Every patch is a token -> No abstraction.

Too low |V| -> Very high intra-token variation -> Difficulty in reconstruction/training.

What is a better way to represent Visual Token targets?

Patch-level prediction works much better than Pixel-level prediction.

Extension to Video?

Treat frames or temporal video segments as temporal visual tokens.