Masked Autoencoders Are Scalable Vision Learners

Kaiming He[*,†] Xinlei Chen[*] Saining Xie Yanghao Li Piotr Dollár Ross Girshick

[*]equal technical contribution [†]project lead

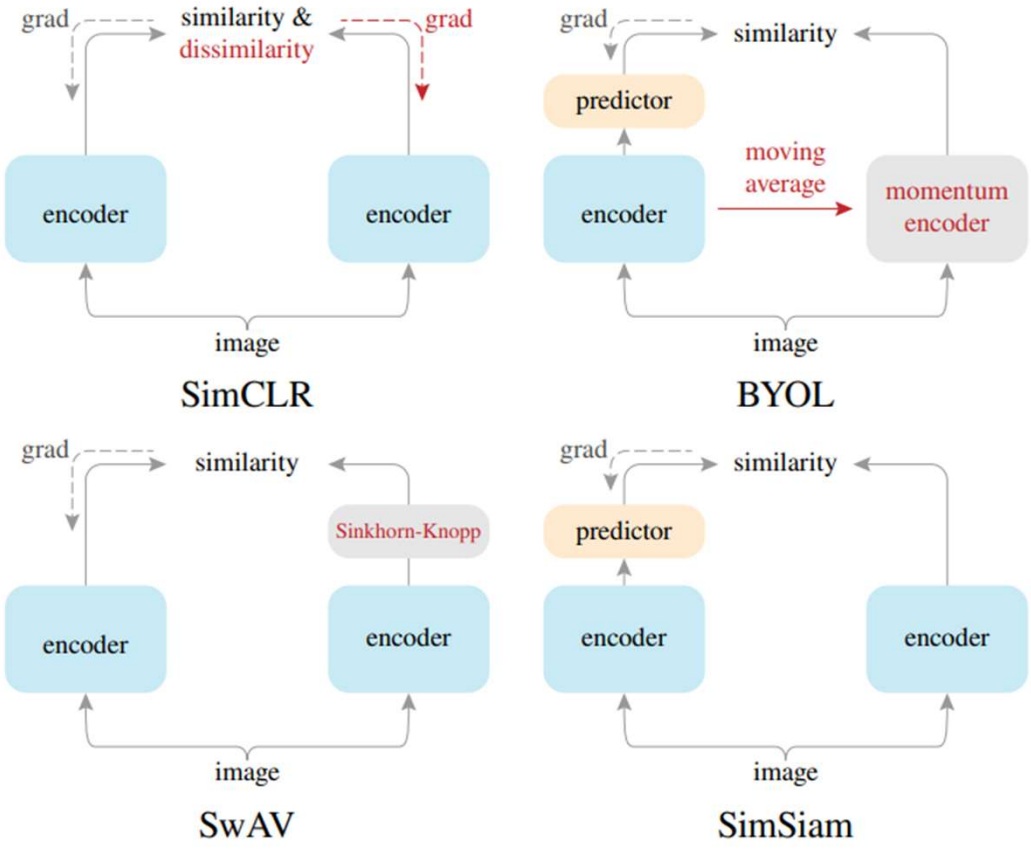
Facebook AI Research (FAIR)

Background

- Contrastive Learning (Moco, SimCLR, BYOL/SwAV/DINO, SimSiam)
- Masked language modeling (BERT)
- Generative method in CV: iGPT, MPP(ViT), BEiT
- Backbone: transformer / Vision transformer

Contrastive learning

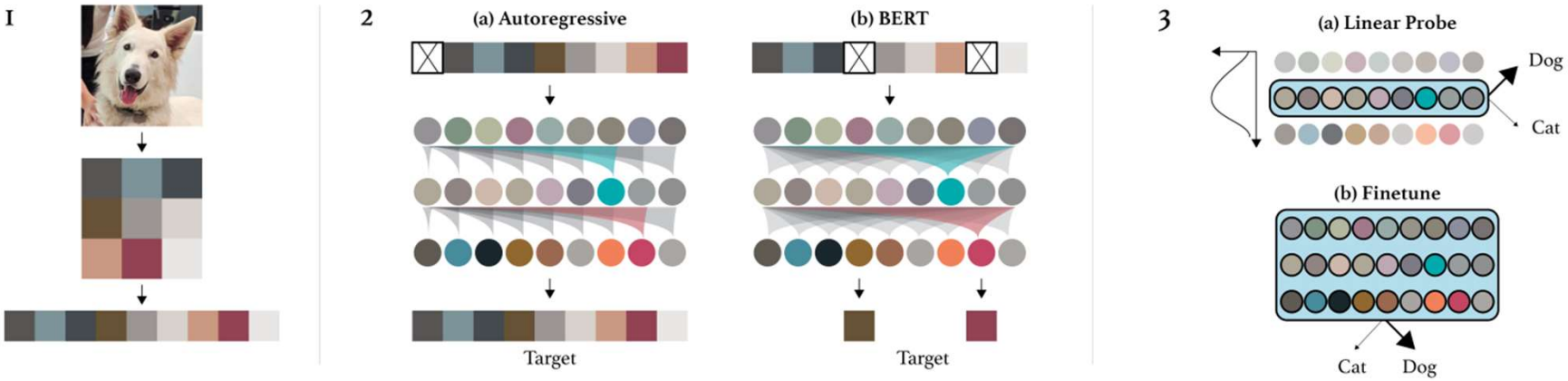
- Linear probe or transfer learning on COCO as benchmark

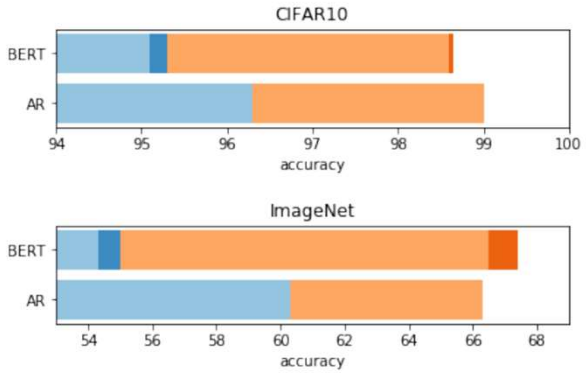


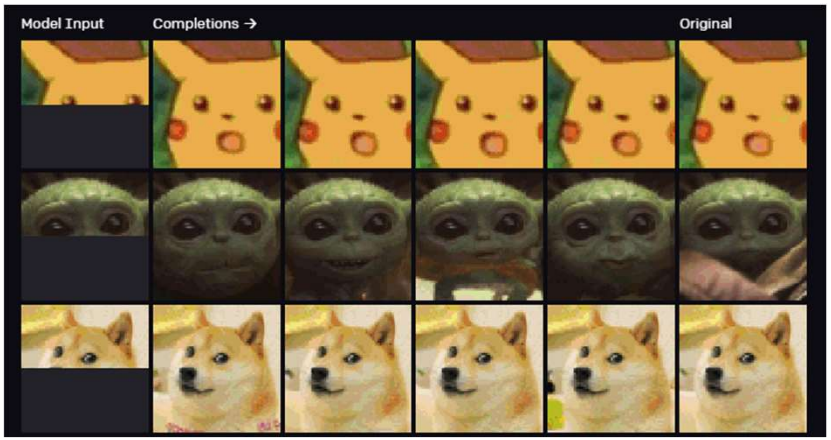
AE & DAE & MIM

- Autoencoding (AE)
  - An encoder that maps an input to a latent representation and a decoder that reconstructs the input.
- Denoising autoencoders (DAE)
  - A class of autoencoders that corrupt an input signal and learn to reconstruct the original, uncorrupted signal.
  - Like BERT
  - MAE is a form of DAE
- Masked Image Modeling (MIM)
  - learn representations from images corrupted by masking

iGPT


1
2
(a) Autoregressive
(b) BERT
Target
Target
3
(a) Linear Probe
Dog
Cat
(b) Finetune
Cat
Dog


CIFAR10
BERT
AR
94
95
96
97
98
99
100
accuracy
ImageNet
BERT
AR
54
56
58
60
62
64
66
68
accuracy

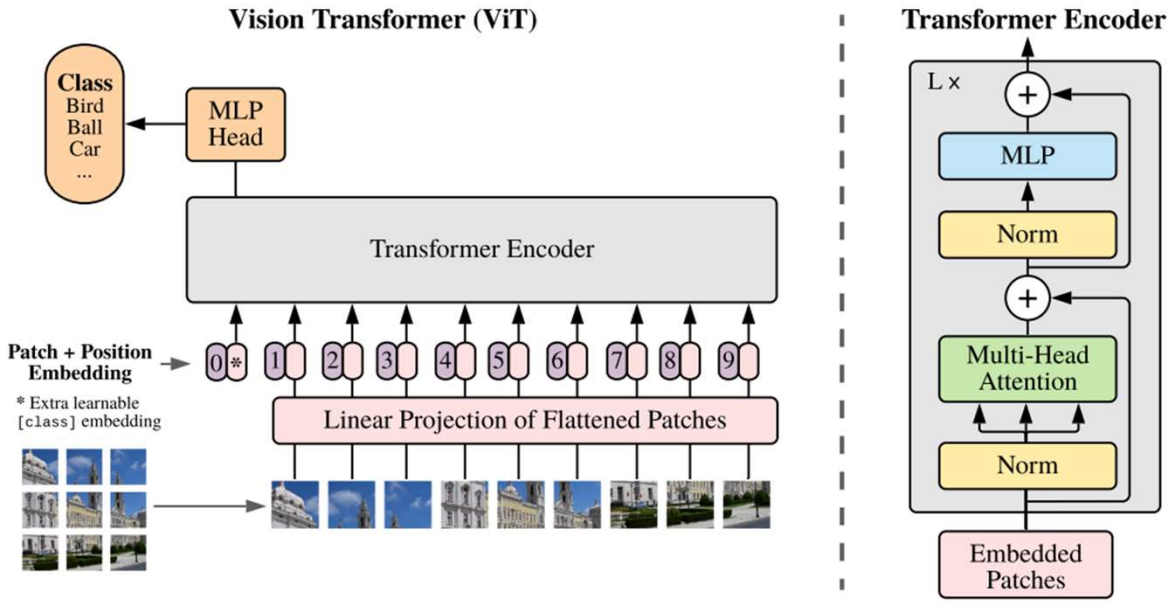

Model Input
Completions →
Original

iGPT

- Before ViT
- Input & Output is simplified
- Need a big model
- Linear probe & Finetuning

Method	IR	Params (M)	Features	Acc
Rotation	orig.	86	8192	55.4
iGPT-L	$32^2 \cdot 3$	1362	1536	60.3
BigBiGAN	orig.	86	8192	61.3
iGPT-L	$48^2 \cdot 3$	1362	1536	65.2
AMDIM	orig.	626	8192	68.1
MoCo	orig.	375	8192	68.6
iGPT-XL	$64^2 \cdot 3$	6801	3072	68.7
SimCLR	orig.	24	2048	69.3
CPC v2	orig.	303	8192	71.5
iGPT-XL	$64^2 \cdot 3$	6801	15360	72.0
SimCLR	orig.	375	8192	76.5

ViT



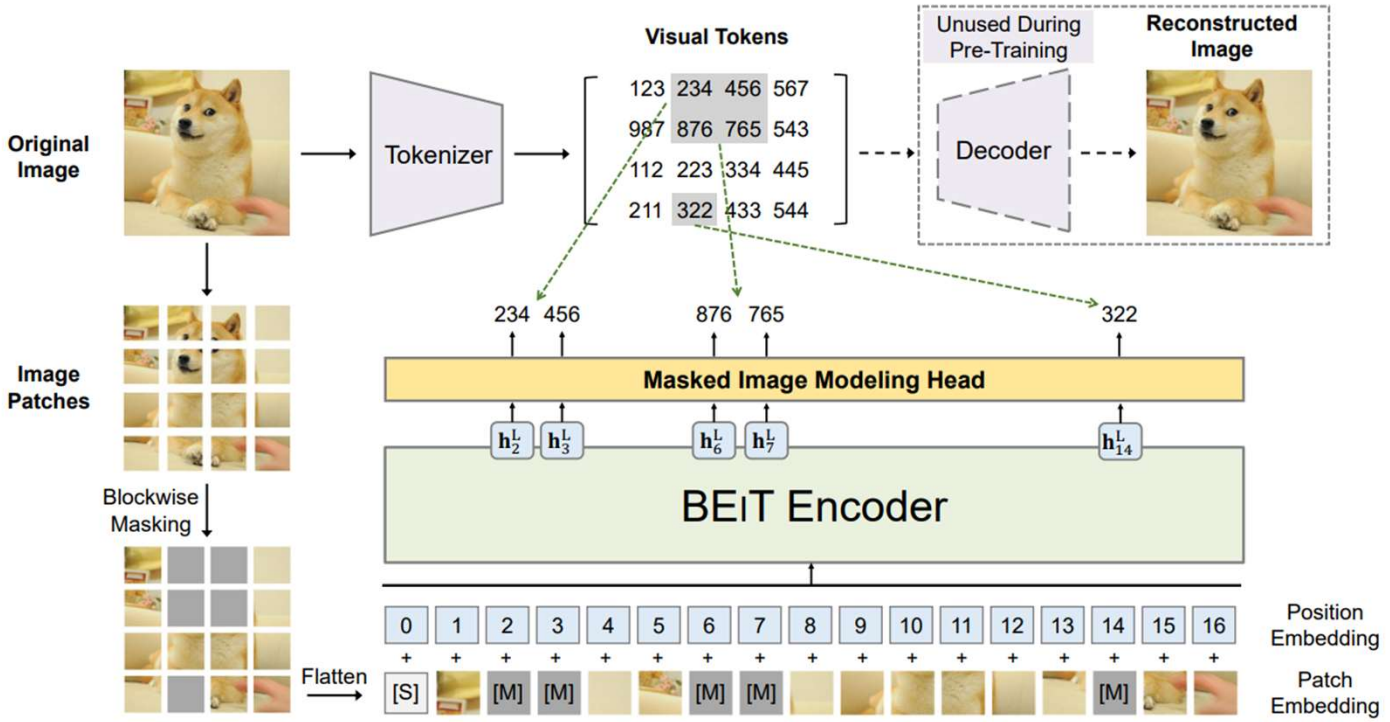
- Divide an image into regular non-overlapping patches
- Add position embedding and through vanilla transformer

ViT: Masked Patch Prediction

- Corrupt 50% patch
- 1. Predict 3-bit, mean color of every corrupted patch
- 2. Predicting a 4 × 4 downsized version of the 16 × 16 patch with 3bit colors in parallel
- 3. Regression on the full patch using L2

- Perform finetuning on downstream task!

BEiT: BERT Pre-Training of Image Transformers



BEiT

- Report finetuning performance on downstream tasks (IN1k & ADE20k & Cifar100)

Models	ImageNet	ADE20K
BEIT (300 Epochs)	82.86	44.65
− Blockwise masking	82.77	42.93
− Visual tokens (i.e., recover masked pixels)	81.04	41.38
− Visual tokens − Blockwise masking	80.50	37.09
+ Recover 100% visual tokens	82.59	40.93
Pretrain longer (800 epochs)	83.19	45.58

Table 4: Ablation studies for BEIT pre-training on image classification and semantic segmentation.

$$\sum_{(x_i, \tilde{x}_i) \in \mathcal{D}} \Big(\underbrace{\mathbb{E}_{z_i \sim q_\phi(z|x_i)}[\log p_\psi(x_i|z_i)]}_{\text{Stage 1: Visual Token Reconstruction}} + \underbrace{\log p_\theta(\hat{z}_i|\tilde{x}_i)}_{\text{Stage 2: Masked Image Modeling}} \Big)$$

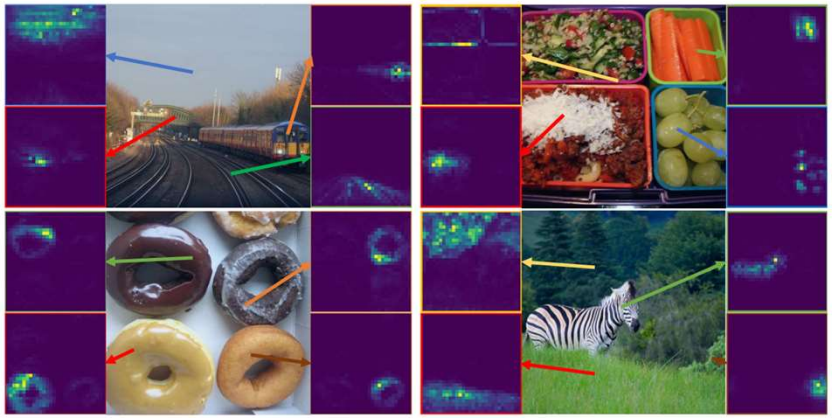


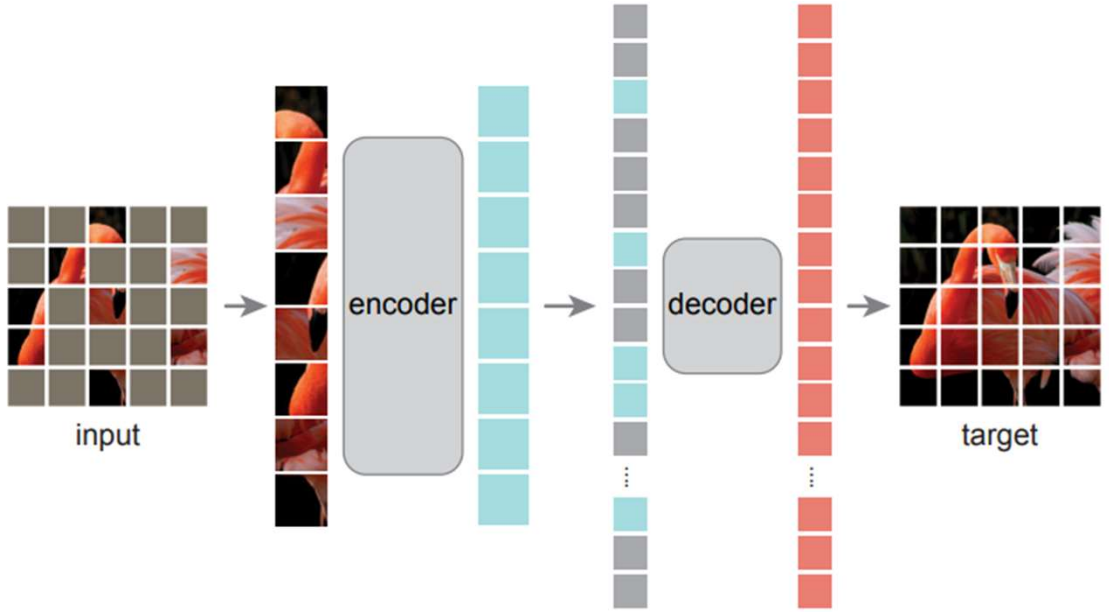
Figure 3: Self-attention map for different reference points. The self-attention mechanism in BEIT is able to separate objects, although self-supervised pre-training does not use manual annotations.

MAE

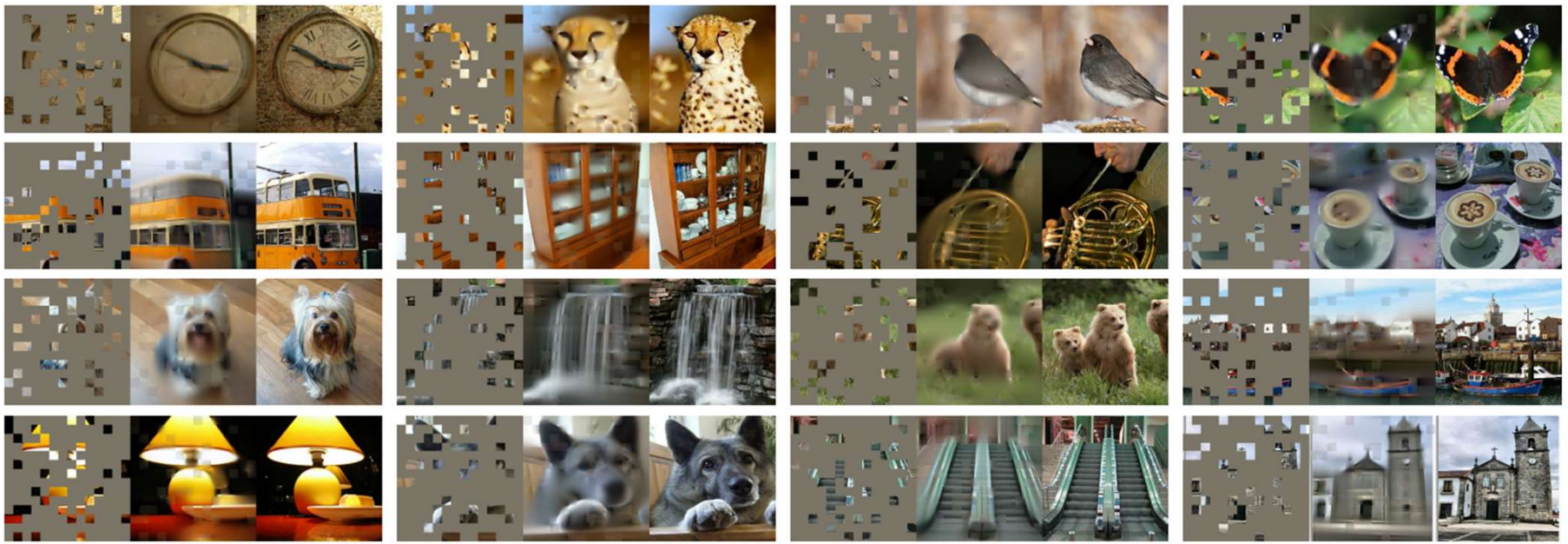
- Masked Autoencoders Are Scalable Vision Learners
- 1) Architectures
  - ViT as backbone (like iGPT, ViT-MPP, BEiT)
- 2) Information density
  - Mask 15% in NLP
  - Mask 75% in MAE
- 3) The autoencoder’s decoder
- Reconstruct pixel level information

MAE: Method

- Backbone: vit
- Masking: 75% patches
- Encoder: w/o [M]
- Decoder: w/[M] narrower and lower
- Reconstruction target
  - Patch level norm

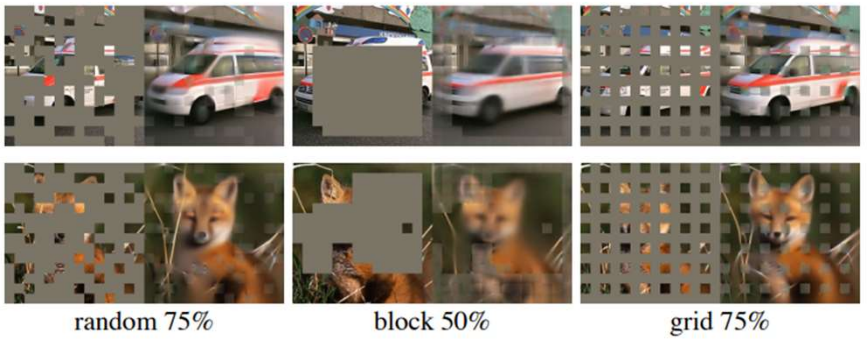


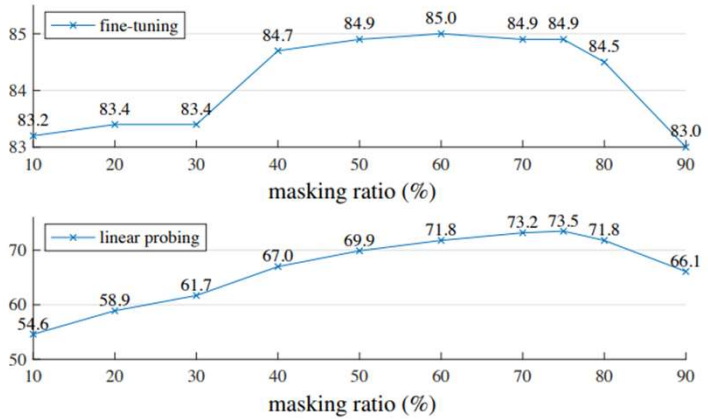
Mask and reconstruction



Masking


original
mask 75%
mask 85%
mask 95%


random 75%
block 50%
grid 75%


fine-tuning
85
84
83
83.2
83.4
83.4
84.7
84.9
85.0
84.9
84.9
84.5
83.0
10
20
30
40
50
60
70
80
90
masking ratio (%)

linear probing
70
60
50
54.6
58.9
61.7
67.0
69.9
71.8
73.2
73.5
71.8
66.1
10
20
30
40
50
60
70
80
90
masking ratio (%)

Ablation

blocks	ft	lin
1	84.8	65.5
2	**84.9**	70.0
4	**84.9**	71.9
8	**84.9**	**73.5**
12	84.4	73.3

(a) **Decoder depth**. A deep decoder can improve linear probing accuracy.

dim	ft	lin
128	**84.9**	69.1
256	84.8	71.3
512	**84.9**	**73.5**
768	84.4	73.1
1024	84.3	73.1

(b) **Decoder width**. The decoder can be narrower than the encoder (1024-d).

case	ft	lin	FLOPs
encoder w/ [M]	84.2	59.6	3.3×
encoder w/o [M]	**84.9**	**73.5**	**1**×

(c) **Mask token**. An encoder without mask tokens is more accurate and faster (Table 2).

case	ft	lin
pixel (w/o norm)	84.9	73.5
pixel (w/ norm)	**85.4**	**73.9**
PCA	84.6	72.3
dVAE token	85.3	71.6

(d) **Reconstruction target**. Pixels as reconstruction targets are effective.

case	ft	lin
none	84.0	65.7
crop, fixed size	84.7	73.1
crop, rand size	**84.9**	**73.5**
crop + color jit	84.3	71.9

(e) **Data augmentation**. Our MAE works with minimal or no augmentation.

case	ratio	ft	lin
random	75	**84.9**	**73.5**
block	50	83.9	72.3
block	75	82.8	63.9
grid	75	84.0	66.0

(f) **Mask sampling**. Random sampling works the best. See Figure 6 for visualizations.

- (a) (b) Decoder (c) w/o [M] (e) Data aug
- (d)Reconstruct target (f) masking

Speed Up

- Without mask tokens
- Drop 75% tokens in encoder
- Decoder is light
- Performance ?
  - Sample epoch
  - Sample FLOPS

encoder	dec. depth	ft acc	hours	speedup
ViT-L, w/ [M]	8	84.2	42.4	-
ViT-L	8	84.9	15.4	2.8×
ViT-L	1	84.8	11.6	**3.7**×
ViT-H, w/ [M]	8	-	119.6†	-
ViT-H	8	85.8	34.5	3.5×
ViT-H	1	85.9	29.3	**4.1**×

Reconstruction target

- Ref: Our results thus far are based on pixels without (per-patch) normalization. Using pixels with normalization improves accuracy.
- Image as input: 3 channels: RGB (0~255)
- Traditional method (input & label):
  - Compute mean and var of full image in training set
- MAE (only label):
  - Compute mean and var for each patch

case	ft	lin
pixel (w/o norm)	84.9	73.5
pixel (w/ norm)	**85.4**	**73.9**
PCA	84.6	72.3
dVAE token	85.3	71.6

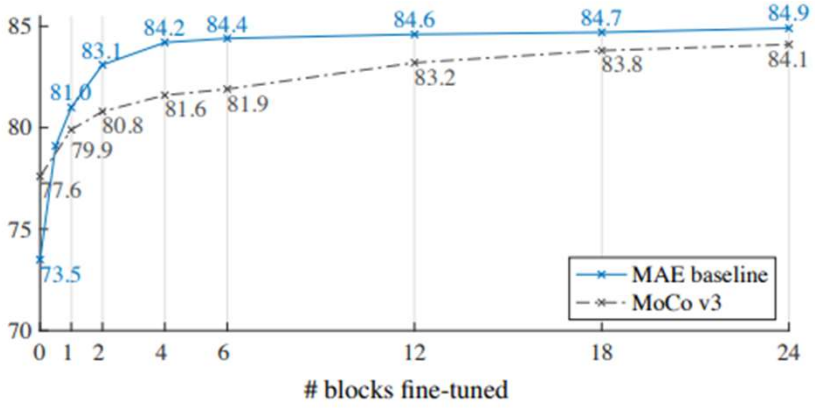
(d) **Reconstruction target**. Pixels as reconstruction targets are effective.

	IN1K			COCO		ADE20K	
	ViT-B	ViT-L	ViT-H	ViT-B	ViT-L	ViT-B	ViT-L
pixel (w/o norm)	83.3	85.1	86.2	49.5	52.8	48.0	51.8
pixel (w/ norm)	83.6	85.9	86.9	50.3	53.3	48.1	53.6
dVAE token	83.6	85.7	86.9	50.3	53.2	48.1	53.4
△	0.0	-0.2	0.0	0.0	-0.1	0.0	-0.2

Table 6. **Pixels *vs.* tokens** as the MAE reconstruction target. △ is the difference between using dVAE tokens and using normalized pixels. The difference is statistically insignificant.

Linear probe or finetuning?

- Linear probe
  - Freeze model and only train a linear layer for downstream task
  - Mainstream and important metric for contrastive Learning
- Finetuning
  - GPT/BERT/iGPT/BEiT
  - Also in MAE



method	model	params	acc
iGPT [6]	iGPT-L	1362 M	69.0
iGPT [6]	iGPT-XL	6801 M	72.0
BEiT [2]	ViT-L	304 M	52.1†
MAE	ViT-B	86 M	68.0
MAE	ViT-L	304 M	75.8
MAE	ViT-H	632 M	76.6

Scaling Up

method	pre-train data	ViT-B	ViT-L	ViT-H	ViT-H_{448}
scratch, our impl.	-	82.3	82.6	83.1	-
DINO [5]	IN1K	82.8	-	-	-
MoCo v3 [9]	IN1K	83.2	84.1	-	-
BEiT [2]	IN1K+DALLE	83.2	85.2	-	-
MAE	IN1K	83.6	85.9	86.9	**87.8**

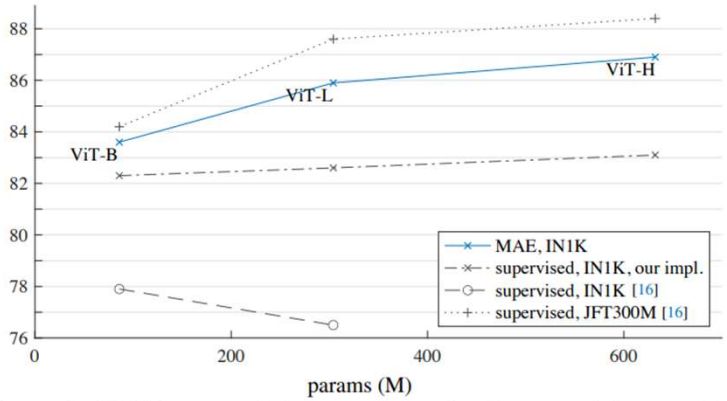
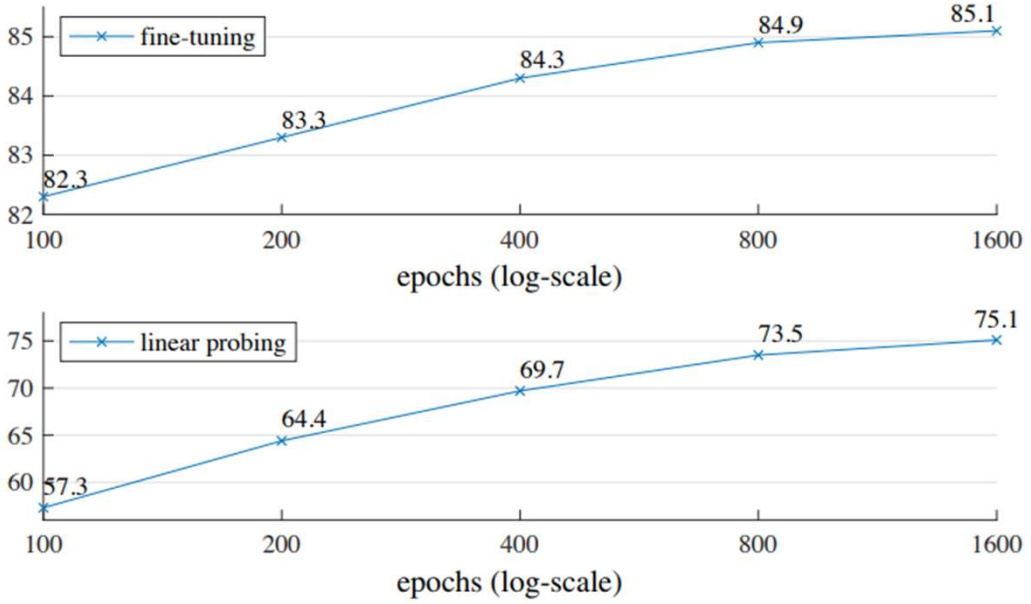
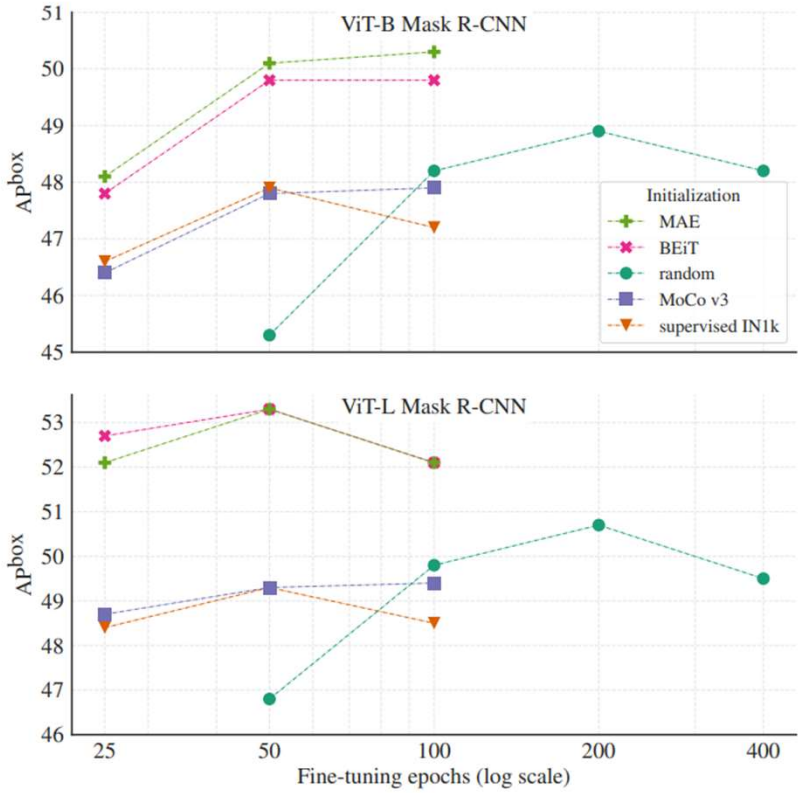


Figure 8. **MAE pre-training *vs*. supervised pre-training**, evaluated by fine-tuning in ImageNet-1K (224 size). We compare with the original ViT results [16] trained in IN1K or JFT300M.



Transfer Learning for objection detection



Discussion

- MIM is a harder task than contrastive learning
- Decoder in MAE
- A better “image tokenizer” or a better “decoder”?
- A simpler task layer for downstream task in CV?
- A faster pretraining-framework?
- Multi-modality

Thanks!