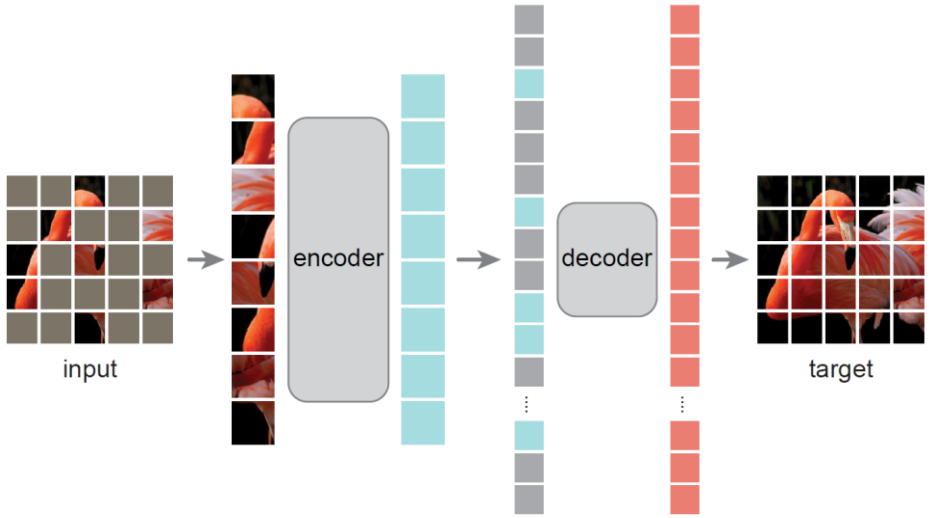
Masked Autoencoders Are Scalable Vision Learners



14th November, 2021
PR12 Paper Review
JinWon Lee

Kaiming He et al., "Masked Autoencoders Are Scalable Vision Learners"

Introduction

- Deep learning has witnessed an explosion of architectures of continuously growing capability and capacity.

- Aided by the rapid gains in hardware, models today can easily overfit one million images and begin to demand hundreds of millions of—often publicly inaccessible—labeled images.

- This appetite for data has been successfully addressed in natural language processing (NLP) by self-supervised pretraining.

Introduction

- The solutions, based on autoregressive language modeling in GPT and masked autoencoding in BERT, are conceptually simple: they remove a portion of the data and learn to predict the removed content.

- These methods now enable training of generalizable NLP models containing over one hundred billion parameters.

Introduction

- The idea of masked autoencoders, a form of more general denoising autoencoders, is natural and applicable in computer vision as well.

- However, despite significant interest in this idea following the success of BERT, progress of autoencoding methods in vision lags behind NLP.

What makes masked autoencoding different between vision and language?

- Until recently, architectures were different. In vision, convolutional networks were dominant over the last decade.

- Convolutions typically operate on regular grids and it is not straightforward to integrate 'indicators' such as mask tokens or positional embeddings into convolutional networks.

- This architectural gap, however, has been addressed with the introduction of Vision Transformers (ViT) and should no longer present an obstacle.

What makes masked autoencoding different between vision and language?

- Information density is different between language and vision.
- Languages are human-generated signals that are highly semantic and information-dense.
- When training a model to predict only a few missing words per sentence, this task appears to induce sophisticated language understanding.
- Images, on the contrary, are natural signals with heavy spatial redundancy—e.g., a missing patch can be recovered from neighboring patches with little high-level understanding of parts, objects, and scenes.

What makes masked autoencoding different between vision and language?

- The autoencoder's decoder, which maps the latent representation back to the input, plays a different role between reconstructing text and images.
- In vision, the decoder reconstructs pixels, hence its output is of a lower semantic level than common recognition tasks.
- This is in contrast to language, where the decoder predicts missing words that contain rich semantic information.
- While in BERT the decoder can be trivial (an MLP), we found that for images, the decoder design plays a key role in determining the semantic level of the learned latent representations.

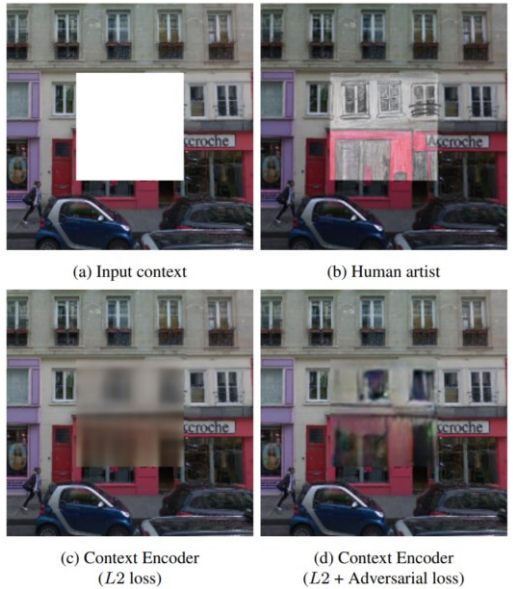
Related Work

- Masked language modeling
  - BERT and GPT, are highly successful methods for pre-training in NLP.
  - These methods hold out a portion of the input sequence and train models to predict the missing content.
  - These methods have been shown to scale excellently and a large abundance of evidence indicates that these pre-trained representations generalize well to various downstream tasks.

Related Work

- Autoencoding
  - It has an encoder that maps an input to a latent representation and a decoder that reconstructs the input.
  - Denoising autoencoders (DAE) are a class of autoencoders that corrupt an input signal and learn to reconstruct the original, uncorrupted signal.
  - A series of methods can be thought of as a generalized DAE under different corruptions, e.g., masking pixels or removing color channels.

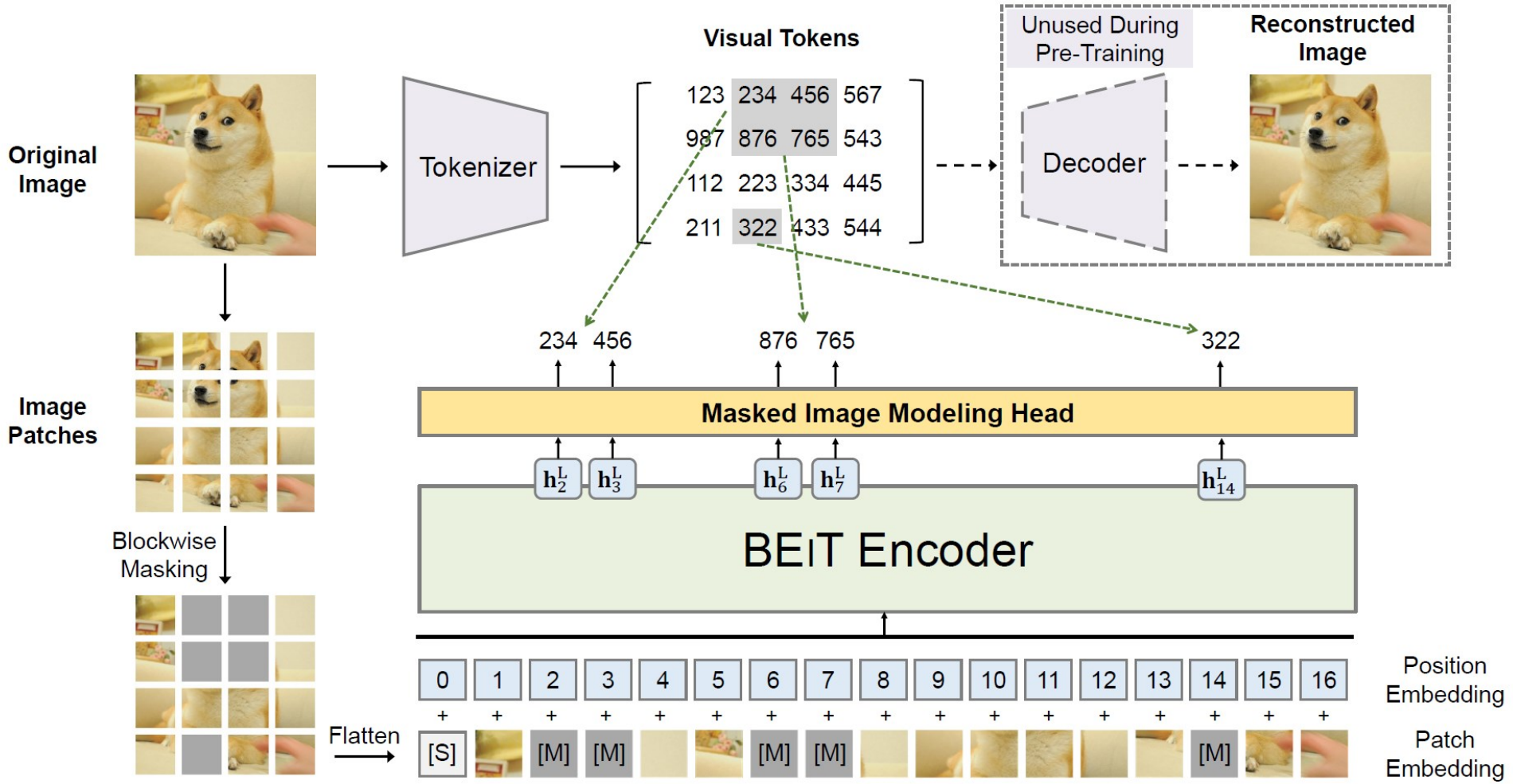
Related Work


(a) Input context (b) Human artist

(c) Context Encoder ($L2$ loss) (d) Context Encoder ($L2$ + Adversarial loss)

- Masked image encoding
  - The pioneering work of presents masking as a noise type in DAE.
  - Context Encoder inpaints large missing regions using convolutional networks.
  - Motivated by the success in NLP, related recent methods are based on Transformers. iGPT operates on sequences of pixels and predicts unknown pixels. The ViT paper studies masked patch prediction for self-supervised learning. Most recently, BEiT proposes to predict discrete tokens.

BEiT: BERT Pre-Training of Image Transformers



Related Work

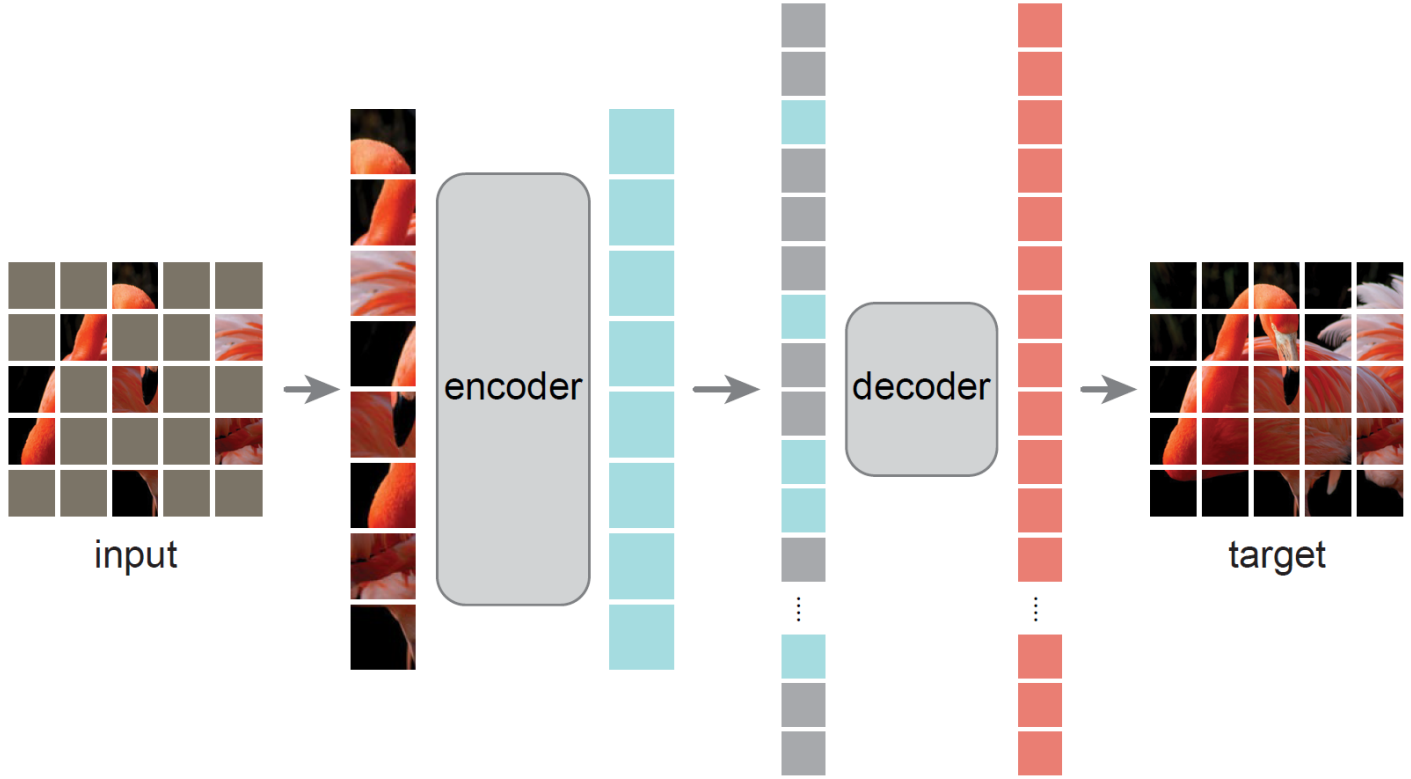
- Self-supervised learning(SSL)
  - SSL approaches have seen significant interest in computer vision, often focusing on different pretext tasks for pre-training.
  - Recently, contrastive learning has been popular, which models image similarity and dissimilarity (or only similarity) between two or more views.
  - Contrastive and related methods strongly depend on data augmentation.
  - Autoencoding pursues a conceptually different direction, and it exhibits different behaviors.

Approach

- Suggested masked autoencoder(MAE) is a simple autoencoding approach that reconstructs the original signal given its partial observation.
- Like all autoencoders, MAE has an encoder that maps the observed signal to a latent representation, and a decoder that reconstructs the original signal from the latent representation.
- Unlike classical autoencoders, The authors adopt an asymmetric design that allows the encoder to operate only on the partial, observed signal (without mask tokens) and a lightweight decoder that reconstructs the full signal from the latent representation and mask tokens.

MAE Architecture

- During pre-training, a large random subset of image patches (e.g., 75%) is masked out.
- The encoder is applied to the small subset of visible patches.
- Mask tokens are introduced after the encoder, and the full set of encoded patches and mask tokens is processed by a small decoder that reconstructs the original image in pixels.
- After pre-training, the decoder is discarded and the encoder is applied to uncorrupted images to produce representations for recognition tasks.



Results

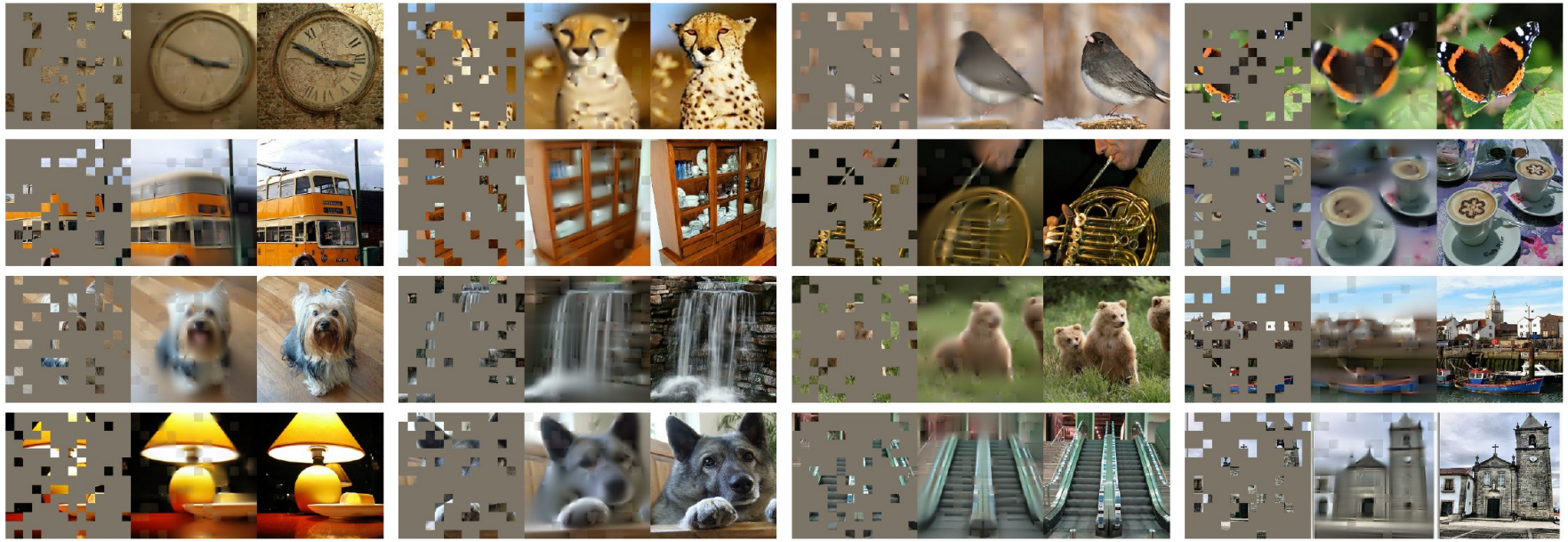


Figure 2. Example results on ImageNet *validation* images. For each triplet, we show the masked image (left), our MAE reconstruction† (middle), and the ground-truth (right). The masking ratio is 80%, leaving only 39 out of 196 patches. More examples are in the appendix.
†*As no loss is computed on visible patches, the model output on visible patches is qualitatively worse. One can simply overlay the output with the visible patches to improve visual quality. We intentionally opt not to do this, so we can more comprehensively demonstrate the method's behavior.*

Figure 3. Example results on COCO validation images, using an MAE trained on ImageNet (the same model weights as in Figure 2). Observe the reconstructions on the two right-most examples, which, although different from the ground truth, are semantically plausible.

MAE Details

- Masking
  - Following ViT, an image is divided into regular non-overlapping patches. Then a subset of patches is sampled and masked.
  - Sampling strategy is straightforward: random patches without replacement, following a uniform distribution.
  - Random sampling with a high masking ratio largely eliminates redundancy, thus creating a task that cannot be easily solved by extrapolation from visible neighboring patches.

MAE Details

- MAE encoder
  - MAE encoder is a ViT but applied only on *visible, unmasked patches*.
  - Just as in a standard ViT, MAE encoder embeds patches by a linear projection with added positional embeddings.
  - However, this encoder only operates on a small subset (e.g., 25%) of the full set.
  - This can reduce overall pre-training time by 3x or more and likewise reduce memory consumption, enabling us to easily scale MAE to large models.

MAE Details

- MAE decoder
  - The input to the MAE decoder is the full set of tokens consisting of (i) encoded visible patches, and (ii) mask tokens.
  - Each mask token is a shared, learned vector that indicates the presence of a missing patch to be predicted.
  - Positional embeddings are added to all tokens in this full set.
  - The MAE decoder is only used during pre-training to perform the image reconstruction task, so the decoder architecture can be flexible designed in a manner that is independent of the encoder design.
  - MAE's default decoder has <10% computation per token vs. the encoder. With this asymmetrical design, the full set of tokens are only processed by the lightweight decoder, which significantly reduces pre-training time.

MAE Details

- Reconstruction target
  - MAE reconstructs the input by predicting the pixel values for each masked patch.
  - The loss function computes the mean squared error (MSE) between the reconstructed and original images in the pixel space. Computing the loss only on masked patches, similar to BERT.
- Simple implementation
  - First, generate a token for every input patch (by linear projection with an added positional embedding). Next, randomly shuffle the list of tokens and remove the last portion of the list, based on the masking ratio.
  - After encoding, append a list of mask tokens to the list of encoded patches, and unshuffle this full list (inverting the random shuffle operation) to align all tokens with their targets.

ImageNet Experiments

- The authors do self-supervised pre-training on the ImageNet-1K(IN1K) training set.
- Then they do supervised training to evaluate the representations with (i) end-to-end fine-tuning or (ii) linear probing.
- Baseline: ViT-Large
  - ViT-Large(Vit-L/16) is used as backbone in ablation study.
  - ViT-L is very big and tends to overfit.
  - It is nontrivial to train supervised ViT-L from scratch and a good recipe with strong regularization is needed.

scratch, original [16]	scratch, our impl.	baseline MAE
76.5	82.5	84.9

Masking Ratio

- The optimal ratios are surprisingly high.
- 75% is good for both linear probing and fine-tuning.
- This is in contrast with BERT(15%) and also much higher than those in related works in CV(20%~50%).
- Reasoning-like behavior is linked to the learning of useful representations.
- For linear probing, the accuracy increases steadily until the sweet point, but for fine-tuning, the results are less sensitive to the ratios.
- All fine-tuning results are better that training from scratch(82.5%)

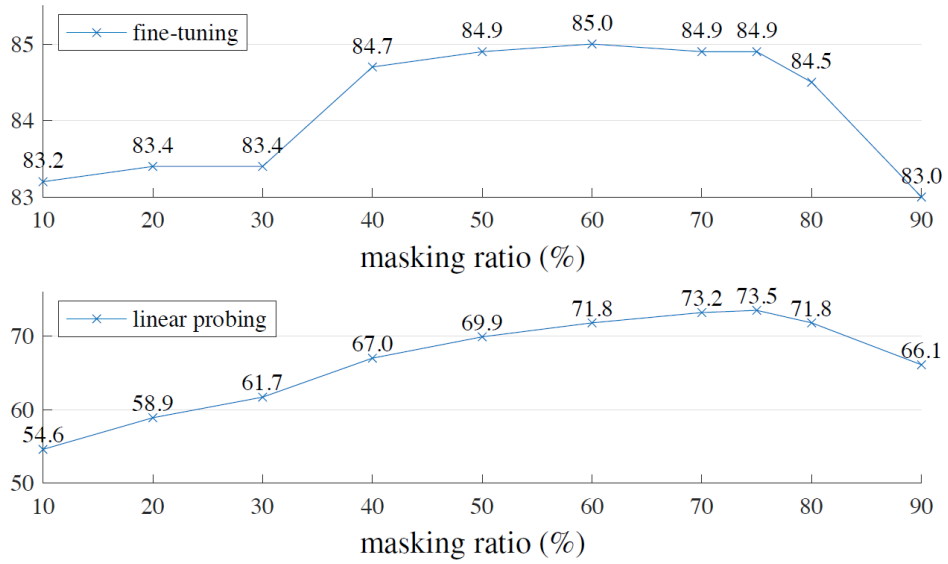


Figure 5. **Masking ratio**. A high masking ratio (75%) works well for both fine-tuning (top) and linear probing (bottom). The y-axes are ImageNet-1K validation accuracy (%) in all plots in this paper.

Decoder Design

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).

- A sufficiently deep decoder is important for linear probing. This can be explained by the gap between a pixel reconstruction task and a recognition task: the last several layers in an autoencoder are more specialized for reconstruction, but are less relevant for recognition.
- However, if fine-tuning is used, the last layers of the encoder can be tuned to adapt to the recognition task. The decoder depth is less influential.
- Interestingly, MAE with a single-block decoder can perform strongly with fine-tuning (84.8%). Note that a single Transformer block is the minimal requirement to propagate information from visible tokens to mask tokens.
- Overall, default MAE decoder is lightweight. It has 8 blocks and a width of 512-d. It only has 9% FLOPs per token vs. ViT-L (24 blocks, 1024-d).

Mask Token

- If the encoder uses mask tokens, it performs worse.

- In this case, there is a gap between pre-training and deploying: this encoder has a large portion of mask tokens in its input in pretraining, which does not exist in uncorrupted images. This gap may degrade accuracy in deployment.

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).

Mask Token

- By skipping the mask token in the encoder, training computation is greatly reduced.
- Note that the speedup can be >4x for a masking ratio of 75%, partially because the self-attention complexity is quadratic.
- In addition, memory is greatly reduced, which can enable training even larger models or speeding up more by large-batch training.
- The time and memory efficiency makes MAE favorable for training very large models.

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**×

Table 2. **Wall-clock time** of our MAE training (800 epochs), benchmarked in 128 TPU-v3 cores with TensorFlow. The speedup is relative to the entry whose encoder has mask tokens (gray). The decoder width is 512, and the mask ratio is 75%. †: This entry is estimated by training ten epochs.

Reconstruction Target

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.

- Using pixels with normalization improves accuracy. This per-patch normalization enhances the contrast locally.
- In another variant, the authors perform PCA in the patch space and use the largest PCA coefficients (96 here) as the target. Doing so degrades accuracy.
- The authors also compare an MAE variant that predicts tokens, the target used in BEiT. Specifically for this variant, the DALLE pre-trained dVAE is used as the tokenizer, following BEIT.
- This tokenization improves fine-tuning accuracy vs. unnormalized pixels, but has no advantage vs. normalized pixels.
- The dVAE tokenizer requires one more pre-training stage, which may depend on extra data (250M images). The dVAE encoder is a large convolutional network (40% FLOPs of ViT-L) and adds nontrivial overhead. Using pixels does not suffer from these problems.

Data Augmentation

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

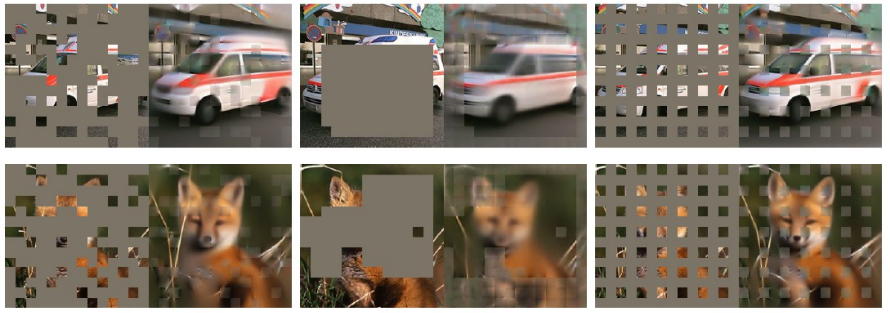
(e) **Data augmentation**. Our MAE works with minimal or no augmentation.

- MAE works well using cropping-only augmentation, either fixed-size or random-size (both having random horizontal flipping).
- Adding color jittering degrades the results.
- Surprisingly, MAE behaves decently even if using no data augmentation (only center-crop, no flipping). This property is dramatically different from contrastive learning and related methods, which heavily rely on data augmentation.
- In MAE, the role of data augmentation is mainly performed by random masking. The masks are different for each iteration and so they generate new training samples regardless of data augmentation.

Masking Sampling Strategy

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.



random 75% block 50% grid 75%

Figure 6. **Mask sampling strategies** determine the pretext task difficulty, influencing reconstruction quality and representations (Table 1f). Here each output is from an MAE trained with the specified masking strategy. Left: random sampling (our default). Middle: block-wise sampling [2] that removes large random blocks. Right: grid-wise sampling that keeps one of every four patches. Images are from the validation set.

- MAE with block-wise masking works reasonably well at a ratio of 50%, but degrades at a ratio of 75%.
- Grid-wise sampling regularly keeps one of every four patches and this is an easier task and has lower training loss. The reconstruction is sharper, but the representation quality is lower.
- Simple random sampling works the best for MAE. It allows for a higher masking ratio, which provides a greater speedup benefit.

Training Schedule

- The accuracy improves steadily with longer training. Indeed, saturation of linear probing have not been observed even at 1600 epochs.
- This behavior is unlike contrastive learning methods, e.g., MoCo v3 saturates at 300 epochs for ViT-L.
- Note that the MAE encoder only sees 25% of patches per epoch, while in contrastive learning the encoder sees 200% (two crop) or even more (multi-crop) patches per epoch.

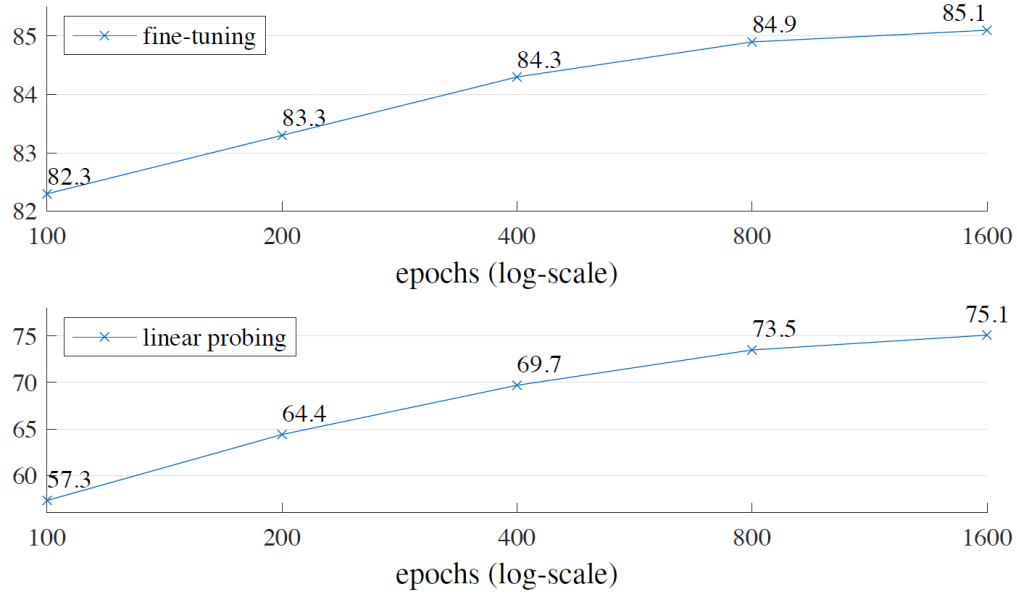


Figure 7. **Training schedules**. A longer training schedule gives a noticeable improvement. Here each point is a full training schedule. The model is ViT-L with the default setting in Table 1.

Comparisons with Self-supervised Methods

- For ViT-B, all methods perform closely.
- For Vit-L the gaps among methods are bigger, suggesting that a challenge for bigger models is to reduce overfitting.
- MAE can scale up easily and has shown steady improvement from bigger models.
- By fine-tuning with a 448 size, MAE achieve 87.8% accuracy, using only IN1K data. The previous best accuracy, among all methods using only IN1K data, is 87.1% (512 size), based on advanced networks.
- Comparing with BEiT, MAE is more accurate while being simpler and faster.

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**

Table 3. **Comparisons with previous results on ImageNet-1K**. The pre-training data is the ImageNet-1K training set (except the tokenizer in BEiT was pre-trained on 250M DALLE data [43]). All self-supervised methods are evaluated by end-to-end fine-tuning. The ViT models are B/16, L/16, H/14 [16]. The best for each column is underlined. All results are on an image size of 224, except for ViT-H with an extra result on 448. Here our MAE reconstructs normalized pixels and is pre-trained for 1600 epochs.

Comparisons with Supervised Pre-training

- In the original ViT paper, ViT-L degrades when trained in IN1K. The authors improved supervised recipe works better for training from scratch, but the accuracy is saturated.
- MAE pre-training, using only IN1K, can generalize better: the gain over training from scratch is bigger for higher-capacity models.
- It follows a trend similar to the JFT-300M supervised pre-training. This comparison shows that MAE can help scale up model sizes.

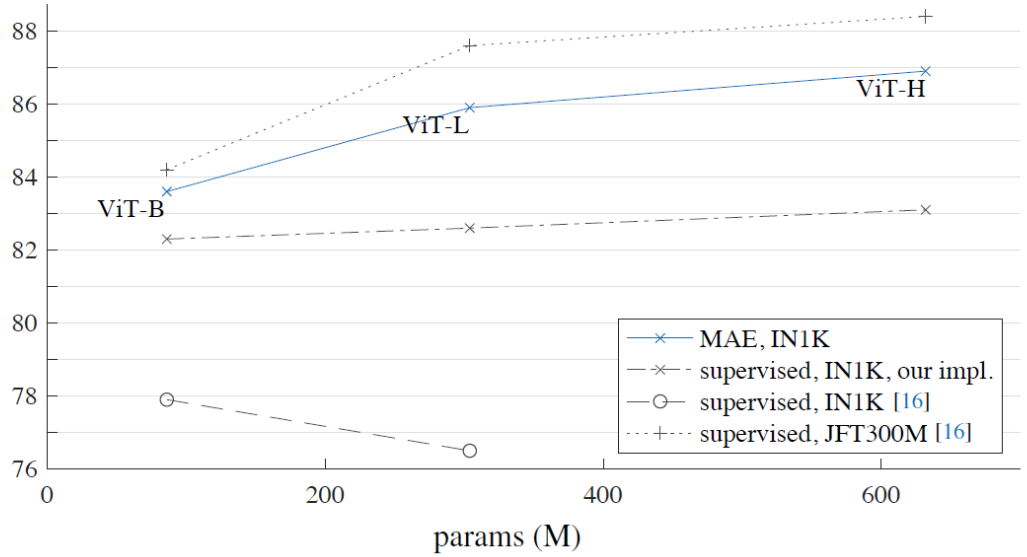


Figure 8. **MAE pre-training *vs*. supervised pre-training**, evaluated by fine-tuning in ImageNet-1K (224 size). We compare with the original ViT results [16] trained in IN1K or JFT300M.

Partial Fine-tuning

- Table 1. shows that linear probing and fine-tuning results are largely uncorrelated.
- Linear probing has been a popular protocol in the past few years; however, it misses the opportunity of pursuing strong but non-linear features—which is indeed a strength of deep learning.
- As a middle ground, we study a partial fine-tuning protocol: fine-tune the last several layers while freezing the others.

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.

Partial Fine-tuning

- Notably, fine-tuning only one Transformer block boosts the accuracy significantly from 73.5% to 81.0%. Moreover, if we fine-tune only "half" of the last block (i.e., its MLP sub-block), we can get 79.1%, much better than linear probing.

- Comparing with MoCo v3 which is a contrastive method, MOCO v3 has higher linear probing accuracy than MAE however, all of its partial fine-tuning results are worse than MAE.

- These results show that the MAE representations are less linearly separable, but they are stronger non-linear features and perform well when a non-linear head is tuned. These observations suggest that **linear separability is not the sole metric** for evaluating representation quality.

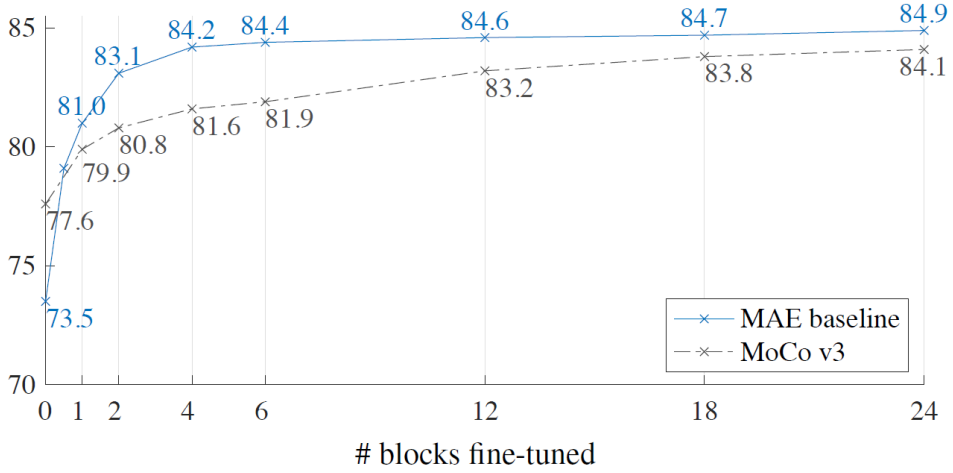


Figure 9. **Partial fine-tuning** results of ViT-L w.r.t. the number of fine-tuned Transformer blocks under the default settings from Table 1. Tuning 0 blocks is linear probing; 24 is full fine-tuning. Our MAE representations are less linearly separable, but are consistently better than MoCo v3 if one or more blocks are tuned.