BEiT v2: Masked Image Modeling with Vector-Quantized Visual Tokenizers

Zhiliang Peng[1]*, **Li Dong**[2], **Hangbo Bao**[2], **Qixiang Ye**[1], **Furu Wei**[2]
University of Chinese Academy of Sciences[1]
Microsoft Research[2]
https://github.com/microsoft/unilm

Abstract

Masked image modeling (MIM; Bao et al. 2022a) has demonstrated impressive results in self-supervised representation learning by recovering corrupted image patches. However, most methods still operate on low-level image pixels, which hinders the exploitation of high-level semantics for representation models. In this study, we propose to use a semantic-rich visual tokenizer as the reconstruction target for masked prediction, providing a systematic way to promote MIM from pixel-level to semantic-level. Specifically, we introduce vector-quantized knowledge distillation to train the tokenizer, which discretizes a continuous semantic space to compact codes. We then pretrain vision Transformers by predicting the original visual tokens for the masked image patches. Moreover, we encourage the model to explicitly aggregate patch information into a global image representation, which facilities linear probing. Experiments on image classification and semantic segmentation show that our approach outperforms all compared MIM methods. On ImageNet-1K (224 size), the base-size BEiT v2 achieves 85.5% top-1 accuracy for fine-tuning and 80.1% top-1 accuracy for linear probing. The large-size BEiT v2 obtains 87.3% top-1 accuracy for ImageNet-1K (224 size) fine-tuning, and 56.7% mIoU on ADE20K for semantic segmentation. The code and pretrained models are available at https://aka.ms/beit.

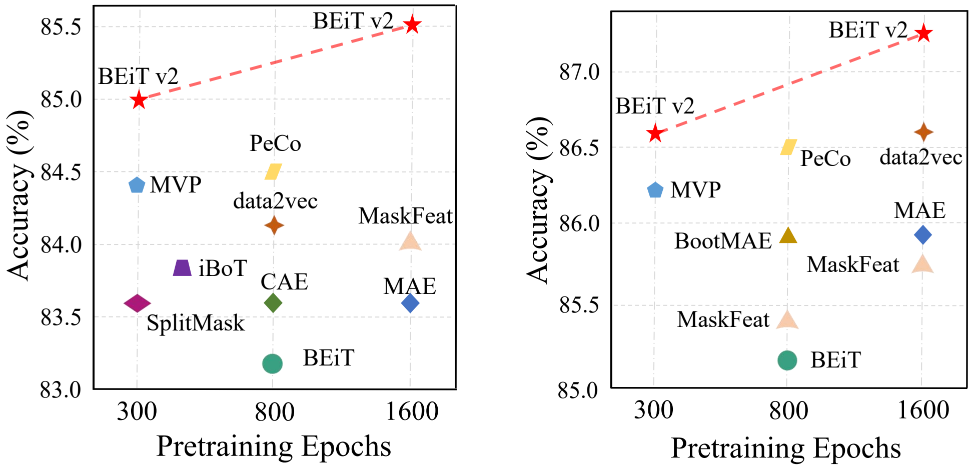


Figure 1: **Left**: ViT-B/16; **right**: ViT-L/16. Top-1 fine-tuning accuracy on ImageNet (224 size).

[*] Contribution during internship at Microsoft Research.

1 Introduction

Masked image modeling (Bao et al., 2022a) has shown impressive results in learning visual representations, which greatly relieves the annotation-hungry issue of vision Transformers. Given an image, the methods typically first corrupt it by masking some patches. The pertaining task is to recover the original image. Taking the pioneering work BEiT (Bao et al., 2022a) as an example, each image has two views during pretraining, *i.e.*, image patches, and visual tokens. The original image is first tokenized into discrete tokens. Randomly sampled image patches are then masked before being fed to vision Transformers. The pretraining objective is to recover the original visual tokens based on the corrupted image patches. After pretraining a vision encoder, we can directly finetune the model on various downstream tasks by appending lightweight task layers.

Under the mask-then-predict framework, the main difference between previous work lies in the reconstruction targets, such as visual tokens (Bao et al., 2022a;b; Dong et al., 2021; El-Nouby et al., 2021; Chen et al., 2022), raw pixels (He et al., 2022; Fang et al., 2022; Liu et al., 2022), and hand-crafted HOG features (Wei et al., 2021). However, recovering low-level supervision tends to waste modeling capacity on pretraining high-frequency details and short-range dependencies. For example, when masking a "hat" wearing on a man's head, we prefer the model to learn the high-level concept of the masked "hat" given the whole context, rather than struggling with pixel-level details. In comparison, the masked words in language modeling (Devlin et al., 2019; Dong et al., 2019) are usually supposed to have more semantics than pixels. This motivates us to tap the potential of MIM by exploiting semantic-aware supervision during pretraining.

In this work, we introduce a self-supervised vision representation model BEiT v2, which aims at improving BEiT pretraining via learning a semantic-aware visual tokenizer. Specifically, we propose the Vector-Quantized Knowledge Distillation (VQ-KD) algorithm to discretize a semantic space. The VQ-KD encoder first converts the input image to discrete tokens according to a learnable codebook. Then the decoder learns to reconstruct the semantic features encoded by a teacher model, conditioning on the discrete tokens. After training VQ-KD, its encoder is used as a visual tokenizer for BEiT pretraining, where the discrete codes serve as supervision signals.

In addition, we propose to pretrain global image representations by explicitly encouraging the CLS token to aggregate all patches (Gao and Callan, 2021). The mechanism resolves the issue that masked image modeling only pretrains patch-level representations. As a result, the performance of linear probing is improved with the help of aggregated global representations.

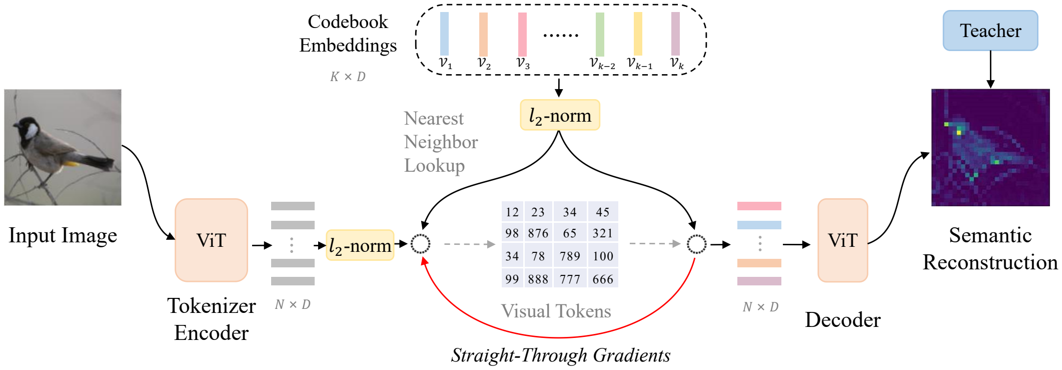
We conduct self-supervised learning on ImageNet-1k for both base- and large-size vision Transformers, which are evaluated on several downstream tasks, *e.g.*, image classification, linear probing, and semantic segmentation. As shown in Figure 1, BEiT v2 outperforms previous self-supervised learning algorithms by a large margin on ImageNet fine-tuning, *e.g.*, improving over BEiT (Bao et al., 2022a) by about two points for both ViT-B/16 and ViT-L/16. Our method outperforms all compared MIM methods on ImageNet linear probing, while achieving large performance gains on ADE20k for semantic segmentation.

The contributions of this study are summarized as follows:

- We introduce vector-quantized knowledge distillation, promoting masked image modeling from pixel-level to semantic-level for self-supervised representation learning.
- We propose a patch aggregation strategy, which enforces global representation given patch-level masked image modeling.
- We conduct extensive experiments on downstream tasks, such as ImageNet fine-tuning, linear probing, and semantic segmentation. Experimental results indicate that our method dramatically improves performance across model size, training step, and downstream tasks.

2 Methods

BEiT v2 inherits the BEiT (Bao et al., 2022a) framework for masked image modeling. Specifically, given an input image, we use a visual tokenizer to tokenize the image to discrete visual tokens. Then we mask a proportion of image patches and feed it into the vision Transformer. The pretraining task is to recover the masked visual tokens based on the corrupted image. In Section 2.2, we introduce a



12	23	34	45
98	876	65	321
34	78	789	100
99	888	777	666

Figure 2: Training process of visual tokenizer, which maps an image to discrete visual tokens.

vector-quantized knowledge distillation algorithm and use it to train a visual tokenizer. In Section 2.3, we employ the visual tokenizer for BEIT pretraining. Moreover, we propose to explicitly encourage the model to pretrain global image representations by constructing an architecture bottleneck.

2.1 Image Representations

We use vision Transformers (ViTs; Dosovitskiy et al. 2020) as the backbone networks to obtain image representations. Given the input image $\boldsymbol{x} \in \mathbb{R}^{H \times W \times C}$, we reshape the image $\boldsymbol{x}$ into $N = HW/P^2$ patches $\{\boldsymbol{x}_i^p\}_{i=1}^N$, where $\boldsymbol{x}^p \in \mathbb{R}^{N \times (P^2 C)}$ and (P, P) is the patch size. In our experiments, we split each 224×224 image into a 14×14 grid of image patches, where each patch is 16×16. Then the image patches $\{\boldsymbol{x}_i^p\}_{i=1}^N$ are flattened and linearly projected into input embeddings for Transformers. We denote the encoding vectors as $\{\boldsymbol{h}_i\}_{i=1}^N$ for N image patches.

2.2 Training Visual Tokenizer

The visual tokenizer maps an image to a sequence of discrete tokens. To be specific, the image $\boldsymbol{x}$ is tokenized to $\boldsymbol{z} = [z_1, \cdots, z_N] \in \mathcal{V}^{(H/P) \times (W/P)}$, where the vocabulary $\mathcal{V}$ (*i.e.*, visual codebook) contains $|\mathcal{V}|$ discrete codes. Notice that the number of tokens is the same as the number of image patches in our work. We propose vector-quantized knowledge distillation (VQ-KD) to train the visual tokenizer. As shown in Figure 2, VQ-KD has two modules during training, *i.e.*, visual tokenizer, and decoder.

The tokenizer is consist of a vision Transformer encoder, and a quantizer. The tokenizer first encodes the input image to vectors. Next, the vector quantizer looks up the nearest neighbor in the codebook for each patch representation $\boldsymbol{h}_i$. Let $\{\boldsymbol{e}_1, \cdots, \boldsymbol{e}_{|\mathcal{V}|}\}$ denote the codebook embeddings. For the i-th image patch, its quantized code is obtained by:

$$z_i = \arg\min_j ||\ell_2(\boldsymbol{h}_i) - \ell_2(\boldsymbol{e}_j)||_2. \tag{1}$$

where ℓ_2 normalization is used for codebook lookup (Yu et al., 2021). The above distance is equivalent to finding codes according to cosine similarity.

After quantizing the image to visual tokens, we feed the ℓ_2-normalized codebook embeddings $\{\ell_2(\boldsymbol{e}_{z_i})\}_{i=1}^N$ to the decoder. The decoder is also a multi-layer Transformer network. The output vectors $\{\boldsymbol{o}_i\}_{i=1}^N$ aim at reconstructing the semantic features of a teacher model, *e.g.*, DINO (Caron et al., 2021), and CLIP (Radford et al., 2021). Let $\boldsymbol{t}_i$ denote the teacher model's feature vector of the i-th image patch. We maximize the cosine similarity between the decoder output $\boldsymbol{o}_i$ and the teacher guidance $\boldsymbol{t}_i$.

Because the quantization process (Equation 1) is non-differentiable. As shown in Figure 2, in order to back-propagate gradients to the encoder, the gradients are directly copied from the decoder input to the encoder output (van den Oord et al., 2017). Intuitively, the quantizer looks up the nearest code for each encoder output, so the gradients of codebook embeddings indicate useful optimization directions for the encoder.

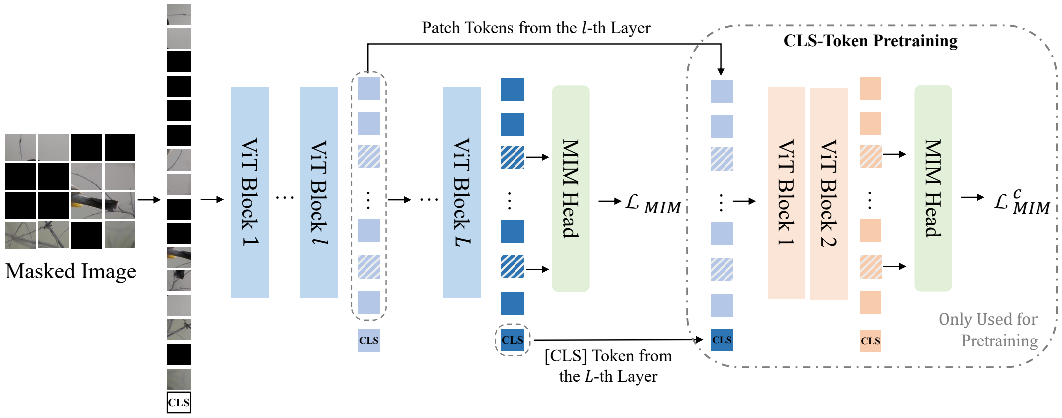


Figure 3: Overview of masked image modeling. The pretraining loss is the summation of $\mathcal{L}_{\text{MIM}}$ and $\mathcal{L}^c_{\text{MIM}}$. The loss term $\mathcal{L}^c_{\text{MIM}}$ explicitly encourages the `CLS` token to aggregate patch information to global representations.

The training objective of VQ-KD is:

$$\max \sum_{x \in \mathcal{D}} \sum_{i=1}^{N} \cos\left(\boldsymbol{o}_i, \boldsymbol{t}_i\right) - ||\text{sg}[\ell_2(\boldsymbol{h}_i)] - \ell_2(\boldsymbol{e}_{z_i})||_2^2 - ||\ell_2(\boldsymbol{h}_i) - \text{sg}[\ell_2(\boldsymbol{e}_{z_i})]||_2^2, \tag{2}$$

where $\text{sg}[\cdot]$ stands for the stop-gradient operator that is an identity at the forward pass while having zero gradients during the backward pass, and $\mathcal{D}$ represents the image data used for tokenizer training.

Improving codebook utilization. A common issue of vector quantization training is codebook collapse. In other words, only a small proportion of codes are used. We empirically find that several techniques are useful to improve the codebook utilization rate. Yu et al. (2021) propose to apply dimensional reduction and ℓ_2 normalization for codebook lookup. Equation 1 shows that we compute the ℓ_2-normalized distance to find the nearest code. Moreover, we reduce the dimension of the lookup space to 32-d. The low-dimensional codebook embeddings are mapped back to higher-dimensional space before feeding into the decoder. In addition, we adopt exponential moving averages to update the codebook embeddings (van den Oord et al., 2017), which tends to be more stable in our experiments.

2.3 Pretraining BEiT v2

We follow the masked image modeling (MIM) setup in BEiT (Bao et al., 2022a) to pretrain vision Transformers for image representations. Given an input image x, we block-wisely choose around 40% image patches to be masked. We term the masked position as $\mathcal{M}$, and then use a shared learnable embedding $\boldsymbol{e}_{[\text{M}]}$ to replace the original image patch embeddings $\boldsymbol{e}_i^p$ if $i \in \mathcal{M}$: $\boldsymbol{x}_i^{\mathcal{M}} = \delta(i \in \mathcal{M}) \odot \boldsymbol{e}_{[\text{M}]} + (1 - \delta(i \in \mathcal{M})) \odot \boldsymbol{x}_i^p$, where $\delta(\cdot)$ is the indicator function. Subsequently, we prepend a learnable `CLS` token to the input, *i.e.*, $[\boldsymbol{e}_{\texttt{CLS}}, \{\boldsymbol{x}_i^{\mathcal{M}}\}_{i=1}^N]$, and feed them to the vision Transformer. The final encoding vectors are denoted as $\{\boldsymbol{h}_i\}_{i=0}^N$, where $\boldsymbol{h}_0$ is for the `CLS` token.

Next, a masked image modeling head is used to predict the visual tokens of the masked positions based on the corrupted image $x^{\mathcal{M}}$. For each masked position $\{\boldsymbol{h}_i : i \in \mathcal{M}\}_{i=1}^N$, a softmax classifier predicts the visual tokens $p(z'|x^{\mathcal{M}}) = \text{softmax}_{z'}(\boldsymbol{W}_c \boldsymbol{h}_i + \boldsymbol{b}_c)$, where $x^{\mathcal{M}}$ is the masked image, and $\boldsymbol{W}_c, \boldsymbol{b}_c$ are the classifier weights. The visual tokens are obtained by the tokenizer trained in Section 2.2, which provides supervision for masked image modeling. Finally, the training loss of MIM can be formulated as:

$$\mathcal{L}_{\text{MIM}} = -\sum_{x \in \mathcal{D}} \sum_{i \in \mathcal{M}} \log p(z_i | x^{\mathcal{M}}) \tag{3}$$

where z_i means the visual tokens of the original image, and $\mathcal{D}$ represents the pretraining images.

Pretraining global representation. Inspired by (Gao and Callan, 2021), we explicitly pretrain the CLS token for global representation. Our goal is to mitigate the discrepancy between patch-level pretraining and image-level representation aggregation. As illustrated in Figure 3, we construct a representation bottleneck to guide the CLS token to gather information. For a L-layer Transformer, let $\{h_i^l\}_{i=1}^N$ denote the l-th layer's output vectors, where $l = 1, \cdots, L$. In order to pretrain the last layer's CLS token $h_{\texttt{CLS}}^L$, we concatenate it with the intermediate l-th layer's patch vectors $\{h_i^l\}_{i=1}^N$, *i.e.*, $\boldsymbol{S} = [h_{\texttt{CLS}}^L, h_1^l, \cdots, h_N^l]$. Then we feed $\boldsymbol{S}$ to a shallow (*e.g.*, two layers) Transformer decoder and conduct masked prediction. Notice that we also compute the MIM loss at the L-th layer as in Equation 3. So the final training loss is the summation of two terms, *i.e.*, the original loss at the L-th layer, and the shallow Transformer decoder's MIM loss. In our implementation, we also share the MIM softmax weights for both heads.

Intuitively, the model favors pushing the global information to $h_{\texttt{CLS}}^L$, because the model tends to fully utilize the parameters from $l + 1$-th layer to L-th layer, in order to decrease the additional MIM loss. The information-flow bottleneck encourages the CLS token to obtain more reliable global representations than untrained counterparts. The newly added shallow decoder is only used to pretrain the CLS token, which is discarded after pretraining.

3 Experiments

We add task layers upon the pretrained BEiT v2 model to evaluate the quality of learned visual representations. For the image classification task, we conduct experiments on ImageNet-1K dataset (Russakovsky et al., 2015) in two protocols: one is fine-tuning top-1 accuracy (fine-tuning the whole parameters). The other one is linear probing top-1 accuracy (only fine-tuning the classification head). For pixel-level tasks, we conduct experiments on ADE20K dataset for semantic segmentation tasks and report the corresponding mIoU protocol.

3.1 Pretraining Setup

Visual tokenizer training. We instantiate the visual tokenizer of VQ-KD as ViT-B/16 for both base- and large-size pretraining. The decoder network is a three-layer standard Transformer, which has the same dimension and number of attention heads as the tokenizer encoder. We use OpenAI CLIP-B/16 (Radford et al., 2021) as the teacher model and train VQ-KD on ImageNet-1k with 224×224 resolution. For the codebook, we set the code size K as 8192 and code dimension D as 32 by default. We run the training process for 100 epochs with a batch size of 512. The AdamW (Loshchilov and Hutter, 2019) optimizer is used with $\beta_1 = 0.9$, $\beta_2 = 0.99$. The weight decay is set to 0.05. Refer to Appendix A for more training details.

Masked image modeling. We follow the settings used in BEiT (Bao et al., 2022a) pretraining. We used ImageNet-1K without labels as the pretraining data for self-supervised learning. We set the input resolution as 224x224 during pretraining. We pretrain base- and large-size vision Transformers (Dosovitskiy et al., 2020) with a 16×16 patch size, *i.e.*, ViT-B/16 and ViT-L/16, respectively. For the CLS token pretraining, we set $l = 9$ for ViT-B/16, $l = 21$ for ViT-L/16, and the depth as 2 by default. We use block-wise masking with the ratio of 40% (*i.e.*, about 75 image patches). We use AdamW to train the model for 300 or 1600 epochs with a 2048 batch size. We use cosine decay for the learning rate schedule with a ten-epoch warmup. The peak learning rate is 1e-5 for both base and large sizes. We apply the initialization algorithm as in (Bao et al., 2022a) to stabilize Transformer training. More pretraining details can be found in Appendix B.

3.2 Image Classification

We evaluate both fine-tuning accuracy and linear probing accuracy on ImageNet-1k. We also evaluate the robustness on several ImageNet variants to demonstrate the favorable generalization ability.

Fine-tuning. We follow the protocol proposed in BEiT (Bao et al., 2022a) to finetune the pretrained BEiT v2 model (see Appendix C for more details).

Table 1 reports the top-1 fine-tuning accuracy results. We compare BEiT v2 with recent MIM methods, such as BEiT (Bao et al., 2022a), MAE (He et al., 2022), CAE (Chen et al., 2022),

Methods	Pretraining Epochs	ImageNet	ADE20k
Base-size models (ViT-B/16)			
BEiT (Bao et al., 2022a)	300	82.9	44.7
CAE (Chen et al., 2022)	300	83.3	47.7
SplitMask (El-Nouby et al., 2021)	300	83.6	45.7
MaskFeat (Wei et al., 2021)	300	83.6	N/A
PeCo (Dong et al., 2021)	300	84.1	46.7
MVP (Wei et al., 2022)	300	84.4	52.4
iBoT (Zhou et al., 2022)	400	83.8	50.0
BEiT v2 (ours)	300	**85.0**	**52.7**
Base-size models (ViT-B/16) + pretrain longer			
BEiT (Bao et al., 2022a)	800	83.2	45.6
CAE (Chen et al., 2022)	800	83.6	48.8
PeCo (Dong et al., 2021)	800	84.5	48.5
data2vec (Baevski et al., 2022)	800	84.2	N/A
MAE (He et al., 2022)	1600	83.6	48.1
BEiT v2 (ours)	1600	**85.5**	**53.1**
Large-size models (ViT-L/16)			
iBoT (Zhou et al., 2022)	250	84.8	N/A
MaskFeat (Wei et al., 2021)	300	84.4	N/A
MVP (Wei et al., 2022)	300	86.3	54.3
BEiT v2 (ours)	300	**86.6**	**55.0**
Large-size models (ViT-L/16) + pretrain longer			
BEiT (Bao et al., 2022a)	800	85.2	53.3
MaskFeat (Wei et al., 2021)	1600	85.7	N/A
MAE (He et al., 2022)	1600	85.9	53.6
data2vec (Baevski et al., 2022)	1600	86.6	N/A
BEiT v2 (ours)	1600	**87.3**	**56.7**

Table 1: Fine-tuning results of image classification and semantic segmentation on ImageNet-1K and ADE20k. We report top-1 accuracy (%) and mIoU (%), respectively. We use the UperNet task layer for semantic segmentation with single-scale inference.

Methods	Linear Probe
BEiT (Bao et al., 2022a)	56.7
MAE (He et al., 2022)	67.8
CAE (Chen et al., 2022)	68.3
MVP (Wei et al., 2022)	75.4
MoCo v3 (Chen et al., 2021)	76.7
BEiT v2 (ours)	**80.1**

Table 2: Top-1 accuracy of linear probing on ImageNet-1k. All methods are based on ViT-B/16 pretrained for 300 epochs except MAE for 1600 epochs.

Methods	ImageNet Adversarial	ImageNet Rendition	ImageNet Sketch
ViT-B/16			
MAE	35.9	48.3	34.5
BEiT v2	**54.4**	**61.0**	**45.6**
ViT-L/16			
MAE	57.1	59.9	45.3
BEiT v2	**69.0**	**69.9**	**53.5**

Table 3: Robustness evaluation on three ImageNet variants (Hendrycks et al., 2021b;a; Wang et al., 2019).

SplitMask (El-Nouby et al., 2021), iBoT (Zhou et al., 2022), MaskFeat (Wei et al., 2021), PeCo (Dong et al., 2021), data2vec (Baevski et al., 2022), and MVP (Wei et al., 2022).

From Table 1, base-size BEiT v2 with 300 epochs pretraining schedule reaches 85.0% top-1 accuracy, which suppresses BEiT, CAE, SplitMask and PeCo by 2.1%, 1.7%, 1.4% and 0.9% respectively. Moreover, BEiT v2 outperforms iBoT by 1.2%, and data2vec by 0.8%. Compared with masked distillation methods, like MVP, BEiT v2 also shows superiority. Furthermore, with a longer pretraining schedule, BEiT v2 achieves 85.5% top-1 accuracy, developing a new state of the art on ImageNet-1K among self-supervised methods.

VQ-KD Architecture	Codebook	Reconst. Loss	Codebook Usage	ImageNet Fine-tuning	ImageNet Linear Probe	ADE20k
Small & 1x384x6	8192×32	0.183	100%	84.3	76.0	51.0
Base & 1x768x12		0.164	100%	84.7	78.5	51.8
Base & 3x768x12		0.145	95%	84.7	77.9	51.9
Base & 6x768x12		0.136	77%	84.6	63.0	50.1
Base & 3x768x12	8192×16	0.145	100%	84.7	76.7	51.7
	8192×64	0.148	67%	84.7	77.6	51.6

Table 4: Ablation studies for different VQ-KD settings. "Base&1x768x12" means the encoder network is ViT-Base while the decoder is a Transformer with depth 1, dimensions 768, and head 12. "Reconst. Loss" is the reconstruction loss of VQ-KD. Reconstruction loss and codebook usage are measured on the validation set. After 300 epochs of pretraining, we report the top-1 fine-tuning accuracy and linear probing accuracy on ImageNet-1k, and mIoU on ADE20k. The default setting is highlighted in gray .

Meanwhile, BEIT V2 using ViT-L/16 with 300 epochs reaches 86.6% top-1 accuracy, which is comparable to data2vec with 1600 epochs. A longer pretraining schedule further boosts the performance to 87.3%.

Linear probing. Linear probing is widely considered a measure for self-supervised learning. It keeps the backbone model frozen, and trains a linear classification head based on the image-level representations. We average the patch tokens as the global representation for the models without CLS token pretraining. Otherwise, we consider the CLS token as the global representation.

Table 2 demonstrates the top-1 accuracy for linear probing. We compare BEIT V2 with MIM methods BEIT, MAE, CAE, MVP and contrastive method MoCo v3. All methods are based on ViT-B/16 and pretrained for 300 epochs except MAE for 1600 epochs. Notably, the visual tokenizer of BEIT and CAE are DALL-E (Ramesh et al., 2021). BEIT V2 can surpass BEIT, CAE and MVP by 23.4%, 11.8% and 4.7% respectively. Additionally, BEIT V2 can also outperform MoCo v3, whose pretraining gets a global representation in a contrastive learning fashion. The results indicate that BEIT V2 produces decent image-level representations.

Robustness evaluation. We evaluate the robustness of BEIT V2 on various ImageNet validation sets, *i.e.*, ImageNet-Adversarial (Hendrycks et al., 2021b), ImageNet-Rendition (Hendrycks et al., 2021a) and ImageNet-Sketch (Wang et al., 2019). We report the results in Table 3. Compared with MAE (He et al., 2022), BEIT V2 achieves dramatic gains across datasets, demonstrating the superiority of the proposed method in terms of generalization.

3.3 Semantic Segmentation

Semantic segmentation is a dense prediction task, which generates class label for each pixel of the input image. Following the setting proposed in BEIT (Bao et al., 2022a), we conduct experiments on ADE20K benchmark (Zhou et al., 2019), which includes 23K mages and 150 semantic categories. We use UperNet (Xiao et al., 2018) task layer and finetune the model for 160K iterations with the input resolution 512 × 512. Refer to Appendix D for details. Table 1 shows that BEIT V2 significantly outperforms previous self-supervised methods. Moreover, using the ViT-L/16 model, the performance can reach 56.7, which builds a new state-of-the-art for masked image modeling on ADE20k.

3.4 Ablation Studies

Visual tokenizer training. We investigate the impact of VQ-KD on BEIT V2 in terms of the model architecture and codebook size. We report the results in Table 4. We adopt ViT-B/16 without the CLS token pretraining as the baseline model, and pretrain it with 300 epochs. From Table 4, deeper decoder of VQ-KD obtains better reconstruction, but lower codebook usage and downstream task performance. Moreover, we show that reducing dimension for codebook lookup improves codebook utilization (Yu et al., 2021).

l-th Layer	Head Depth	Shared MIM Head	ImageNet Fine-tuning	ImageNet Linear Probe	ADE20k
Without CLS token pretraining					
-	-	-	84.7	77.9	51.9
With CLS token pretraining					
9	2	✓	**85.0**	**80.1**	52.7
9	2	✗	84.8	79.5	51.9
9	1	✓	84.8	78.9	51.7
9	3	✓	84.7	78.1	52.0
6	2	✓	84.9	77.5	**53.1**
11	2	✓	84.5	69.4	51.8

Table 5: Ablation studies for CLS token pretraining. **l-th Layer** means the path tokens from the intermediate l-th layer of the backbone. **Head Depth** means the CLS token pretraining head depth. **Shared MIM Head** means whether share the MIM head parameters or not. The default setting is highlighted in gray .

VQ-KD Targets	ImageNet	ADE20k
Pretraining 300 epochs		
DINO	84.4	49.2
CLIP	85.0	52.7
Pretraining 1600 epochs		
CLIP	**85.5**	**53.1**
Performance of teacher models		
DINO	83.6†	46.8†
CLIP	84.9†	-

Table 6: Ablation studies on VQ-KD targets. † indicates our fine-tuning results.

CLS token pretraining. Table 5 presents the ablation studies on CLS token pretraining. The shallower head (1/2-layer) performs better than the deeper head (3-layer), suggesting the shallower head (lower model capacity) pays more attention to the input CLS token than the deeper head (higher model capacity). Moreover, the proposed method outperforms the variant without CLS token pretraining. The improvement of linear probe also indicates better image-level representation.

VQ-KD targets. In Table 6, we train the VQ-KD under the supervision of DINO (Caron et al., 2021) and CLIP (Radford et al., 2021). DINO is pretrained solely on ImageNet-1k, and CLIP is pretrained on 400M image-text pairs datasets. We use the official base-size checkpoints to train VQ-KD, and directly finetune them following the BEiT recipe. One can see that when the teacher is DINO, BEiT v2 respectively reaches 84.4% and 49.2% on ImageNet and ADE20k, respectively. The results also surpass DINO itself by a large margin. When the teacher model is CLIP, BEiT v2 can get consistent improvement, demonstrating the scalability of the proposed VQ-KD. More importantly, the results show that our method can outperform the teacher models with masked image modeling.

4 Related Work

Visual tokenizer. VQ-VAE (van den Oord et al., 2017) converts an image into a sequence of discrete codes and then reconstructs the input image based on discrete codes. DALL-E (Ramesh et al., 2021) uses the Gumbel-softmax relaxation for quantization instead of the nearest neighbor lookup in VQ-VAE. VQGAN (Esser et al., 2021) and ViT-VQGAN (Yu et al., 2021) introduce Transformer block to train a better autoencoder to maintain fine details with adversarial and perceptual loss. Moreover, ViT-VQGAN proposes factorized and ℓ_2-normalized code for codebook learning. In comparison, the proposed VQ-KD aims at reconstructing semantic knowledge from the teacher rather than original pixels. So we can construct a highly compact semantic codebook for masked image modeling.

Masked image modeling. Masked language modeling task achieves great success in language task (Devlin et al., 2019; Dong et al., 2019; Bao et al., 2020). Motivated by it, BEIT (Bao et al., 2022a) proposes the masked image modeling (MIM) task by recovering discrete visual tokens (Ramesh et al., 2021). After that, the prediction target for MIM is explored by many recent works. MAE (He et al., 2022) considers MIM as a denoising pixel-level reconstruction task. Knowledge distillation (Wei et al., 2021; 2022) and self-distillation (Zhou et al., 2022; Baevski et al., 2022) propose to mimic the features provided by the teacher at the masked positions. PeCo (Dong et al., 2021) regards MoCo v3 (Chen et al., 2021) as the perceptual model in VQGAN training (Esser et al., 2021), to get a better tokenizer for BEIT pretraining. By training a semantic visual tokenizer with VQ-KD, we boost the performance of BEIT pretraining to a new height.

5 Conclusion

In this paper, we propose vector-quantized knowledge distillation (VQ-KD) to train a visual tokenizer for vision Transformer pretraining. VQ-KD discretizes a continuous semantic space that provides supervision for masked image modeling rather than relying on image pixels. The semantic visual tokenizer greatly improves the BEIT pretraining and significantly boosts the transfer performance on downstream tasks. In addition, a CLS token pretraining mechanism is introduced to explicitly encourage the model to produce global image representations, narrowing the gap between the patch-level pretraining and image-level representation aggregation.

Acknowledgement

We would like to acknowledge Yuxin Fang and Jinghao Zhou for the helpful discussions.

References

Alexei Baevski, Wei-Ning Hsu, Qiantong Xu, Arun Babu, Jiatao Gu, and Michael Auli. Data2vec: A general framework for self-supervised learning in speech, vision and language. *arXiv preprint arXiv:2202.03555*, 2022.

Hangbo Bao, Li Dong, Furu Wei, Wenhui Wang, Nan Yang, Xiaodong Liu, Yu Wang, Jianfeng Gao, Songhao Piao, Ming Zhou, and Hsiao-Wuen Hon. UniLMv2: Pseudo-masked language models for unified language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020*, volume 119 of *Proceedings of Machine Learning Research*, pages 642–652. PMLR, 2020. URL http://proceedings.mlr.press/v119/bao20a.html.

Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. BEiT: BERT pre-training of image transformers. In *International Conference on Learning Representations*, 2022a. URL https://openreview.net/forum?id=p-BhZSz59o4.

Hangbo Bao, Wenhui Wang, Li Dong, and Furu Wei. VL-BEiT: Generative vision-language pretraining. *ArXiv*, abs/2206.01127, 2022b.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. *arXiv preprint arXiv:2104.14294*, 2021.

Xiaokang Chen, Mingyu Ding, Xiaodi Wang, Ying Xin, Shentong Mo, Yunhao Wang, Shumin Han, Ping Luo, Gang Zeng, and Jingdong Wang. Context autoencoder for self-supervised representation learning. *arXiv preprint arXiv:2202.03026*, 2022.

Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. *ArXiv*, abs/2104.02057, 2021.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4171–4186. Association for Computational Linguistics, 2019.

Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. Unified language model pre-training for natural language understanding and generation. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 13042–13054, 2019.

Xiaoyi Dong, Jianmin Bao, Ting Zhang, Dongdong Chen, Weiming Zhang, Lu Yuan, Dong Chen, Fang Wen, and Nenghai Yu. PeCo: Perceptual codebook for bert pre-training of vision transformers. *arXiv preprint arXiv:2111.12710*, 2021.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *preprint arXiv:2010.11929*, 2020.

Alaaeldin El-Nouby, Gautier Izacard, Hugo Touvron, Ivan Laptev, Hervé Jegou, and Edouard Grave. Are large-scale datasets necessary for self-supervised pre-training? *arXiv preprint arXiv:2112.10740*, 2021.

Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *CVPR*, pages 12873–12883, 2021.

Yuxin Fang, Li Dong, Hangbo Bao, Xinggang Wang, and Furu Wei. Corrupted image modeling for self-supervised visual pre-training. *ArXiv*, abs/2202.03382, 2022.

Luyu Gao and Jamie Callan. Condenser: a pre-training architecture for dense retrieval. *arXiv preprint arXiv:2104.08253*, 2021.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, 2022.

Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, Dawn Song, Jacob Steinhardt, and Justin Gilmer. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *IEEE ICCV*, 2021a.

Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. In *IEEE CVPR*, 2021b.

Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, Furu Wei, and Baining Guo. Swin transformer v2: Scaling up capacity and resolution. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, pages 8748–8763. PMLR, 2021.

A. Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. *ArXiv*, abs/2102.12092, 2021.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *IJCV*, 2015.

Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. In *NeurIPS*, NIPS'17, page 6309–6318, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. Learning robust global representations by penalizing local predictive power. In *Advances in Neural Information Processing Systems*, pages 10506–10518, 2019.

Chen Wei, Haoqi Fan, Saining Xie, Chao-Yuan Wu, Alan Yuille, and Christoph Feichtenhofer. Masked feature prediction for self-supervised visual pre-training. *arXiv preprint arXiv:2112.09133*, 2021.

Longhui Wei, Lingxi Xie, Wengang Zhou, Houqiang Li, and Qi Tian. MVP: Multimodality-guided visual pre-training. *arXiv preprint arXiv:2203.05175*, 2022.

Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *ECCV*, 2018.

Jiahui Yu, Xin Li, Jing Yu Koh, Han Zhang, Ruoming Pang, James Qin, Alexander Ku, Yuanzhong Xu, Jason Baldridge, and Yonghui Wu. Vector-quantized image modeling with improved vqgan. *arXiv preprint arXiv:2110.04627*, 2021.

Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ADE20K dataset. *Int. J. Comput. Vis.*, 127(3):302–321, 2019. doi: 10.1007/s11263-018-1140-0. URL https://doi.org/10.1007/s11263-018-1140-0.

Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. Image BERT pre-training with online tokenizer. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=ydopy-e6Dg.

A Hyperparameters for VQ-KD Training

Hyperparameters	Values
Encoder layers	12
Decoder layers	{1, 3}
Hidden size	768
FFN inner hidden size	3072
Attention heads	12
Attention head size	64
Patch size	16×16
Codebook size	8192×32
Training epochs	100
Batch size	512
Adam β	(0.9, 0.99)
Peak learning rate	2e-4
Minimal learning rate	1e-5
Learning rate schedule	Cosine
Warmup epochs	5
Gradient clipping	✗
Dropout	✗
Stoch. depth	✗
Weight decay	1e-4
Data Augment	RandomResizeAndCrop
Input resolution	224×224

Table 7: Hyperparameters for training VQ-KD on ImageNet-1K.

B Hyperparameters for BEIT V2 Pretraining

Hyperparameters	Base Size	Large Size
Layers	12	24
Hidden size	768	1024
FFN inner hidden size	3072	4096
Attention heads	12	16
Layer scale	0.1	1e-5
Patch size	16×16	
Relative positional embeddings	✓	
Shared relative positional embeddings	✓	
Training epochs	300*/1600	
Batch size	2048	
Adam β	(0.9, 0.98*/0.999)	
Peak learning rate	1.5e-3	
Minimal learning rate	1e-5	
Learning rate schedule	Cosine	
Warmup epochs	10	
Gradient clipping	3.0	
Dropout	✗	
Drop path	0*/0.1	
Stoch. depth	✗	
Weight decay	0.05	
Data Augment	RandomResizeAndCrop	
Input resolution	224×224	
Color jitter	0.4	

Table 8: Hyperparameters for BEIT V2 pretraining on ImageNet-1K. * means that the hyperparameters are adopted when the pretraining schedule is 300 epochs.

C Hyperparameters for Image Classification Fine-tuning

Hyperparameters	**ViT-B/16**	**ViT-L/16**
Peak learning rate	5e-4	5e-4
Fine-tuning epochs	100	50
Warmup epochs	20	5
Layer-wise learning rate decay	0.65	0.8
Batch size	1024	
Adam ϵ	1e-8	
Adam β	(0.9, 0.999)	
Minimal learning rate	1e-6	
Learning rate schedule	Cosine	
Repeated Aug	✗	
Weight decay	0.05	
Label smoothing ε	0.1	
Stoch. depth	0.1	0.2
Dropout	✗	
Gradient clipping	✗	
Erasing prob.	0.25	
Input resolution	224×224	
Rand Augment	9/0.5	
Mixup prob.	0.8	
Cutmix prob.	1.0	
Relative positional embeddings	✓	
Shared relative positional embeddings	✗	

Table 9: Hyperparameters for fine-tuning BEiT v2 on ImageNet-1K.

D Hyperparameters for ADE20K Semantic Segmentation Fine-tuning

Hyperparameters	**ViT-B/16**	**ViT-L/16**
Input resolution	512×512	
Peak learning rate	{0.5, 0.8, 1.0}e-4	
Fine-tuning steps	160K	
Batch size	16	
Adam ϵ	1e-8	
Adam β	(0.9, 0.999)	
Layer-wise learning rate decay	{0.75, 0.8, 0.85}	
Minimal learning rate	0	
Learning rate schedule	Linear	
Warmup steps	1500	
Dropout	✗	
Stoch. depth	0.1	0.2
Weight decay	0.05	
Relative positional embeddings	✓	
Shared relative positional embeddings	✗	

Table 10: Hyperparameters for fine-tuning BEiT v2 on ADE20K.