Contrastive Masked Autoencoders are Stronger Vision Learners

Zhicheng Huang   Xiaojie Jin   Chengze Lu   Qibin Hou
Ming-Ming Cheng   Dongmei Fu   Xiaohui Shen   Jiashi Feng

Abstract—Masked image modeling (MIM) has achieved promising results on various vision tasks. However, the limited discriminability of learned representation manifests there is still plenty to go for making a stronger vision learner. Towards this goal, we propose Contrastive Masked Autoencoders (CMAE), a new self-supervised pre-training method for learning more comprehensive and capable vision representations. By elaboratively unifying contrastive learning (CL) and masked image model (MIM) through novel designs, CMAE leverages their respective advantages and learns representations with both strong instance discriminability and local perceptibility. Specifically, CMAE consists of two branches where the online branch is an asymmetric encoder-decoder and the momentum branch is a momentum updated encoder. During training, the online encoder reconstructs original images from latent representations of masked images to learn holistic features. The momentum encoder, fed with the full images, enhances the feature discriminability via contrastive learning with its online counterpart. To make CL compatible with MIM, CMAE introduces two new components, i.e. pixel shifting for generating plausible positive views and feature decoder for complementing features of contrastive pairs. Thanks to these novel designs, CMAE effectively improves the representation quality and transfer performance over its MIM counterpart. CMAE achieves the state-of-the-art performance on highly competitive benchmarks of image classification, semantic segmentation and object detection. Notably, CMAE-Base achieves 85.3% top-1 accuracy on ImageNet and 52.5% mIoU on ADE20k, surpassing previous best results by 0.7% and 1.8% respectively. The source code is publicly accessible at https://github.com/ZhichengHuang/CMAE.

Index Terms—Masked image modeling, constrastive learning, self-supervised learning.

◆

1 INTRODUCTION

MASKED image modeling (MIM) [23, 30, 56] has been attracting increasing attention recently in the self-supervised learning field, due to its method simplicity and capability of learning rich and holistic representations. Following the idea of masked language modeling in NLP [19], they randomly mask a large portion of the training image patches and use an auto-encoder [31] to reconstruct the original signals (e.g., raw pixels, offline extracted features) of the masked patches. It has been shown in [23, 30, 56] that such a simple framework outperforms previous self-supervised learning methods in both ImageNet classification [18] and some downstream tasks, like object detection and semantic segmentation.

When we reflect on the success of MIM, it is inevitable to compare it with another well-proven and prevailing SSL method, i.e. contrastive learning (CL) [2, 42]. By adopting a simple discriminative idea that pulling closer representations from the same image and pushing away different images, CL methods naturally endow the pretained model with strong instance discriminability. In contrast to CL, MIM focuses more on learning local relations in input image for fulfilling the reconstruction task, instead of modeling the relation among

different images [35]. Therefore, it is suspected that MIM is less efficient in learning discriminative representations. This issue has been manifested by experimental results in [30, 56]. Based on above analysis, it is thus natural to ask such a question: *can we leverage contrastive learning to further strengthen the representation learned by MIM methods?* or, in other words, *would MIM methods benefit from contrastive learning?* Along this direction, a few contemporary works attempt to train vision representation models [46, 65] by simply combining contrastive learning and MIM learning objectives. But they only show marginal performance gain compared to MIM methods. These results signify that it is non-trivial to fully leverage the advantages of both frameworks. The challenges are ascribed to various distinctions between them, including input augmentations, training objectives, model architectures, etc.

To overcome the challenges and learn better image representations for downstream tasks, we aim to explore a possible way to boost the MIM with contrastive learning in a unified framework. With a series of careful studies, we find that input view augmentation and latent feature alignment play important roles in harmonizing MIM and contrastive learning. We thus put dedicated efforts to these components to develop our method.

An overview of the proposed method is shown in Figure 1. More specifically, our method introduces a contrastive MAE (CMAE) framework for representation learning. It adopts a siamese architecture [5]. One branch is an online updated asymmetric encoder-decoder that learns latent representations to reconstruct masked images from a few visible patches, similar to MAE. The other branch is a momentum

---

• Z. Huang and D. Fu are with the School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing, China. (zhicheng.huang@xs.ustb.edu.cn, fdm_ustb@ustb.edu.cn)
• X. Jin, X. Shen and J. Feng are with Bytedance Inc., China. ({jinxiaojie, shenxiaohui.kevin, jshfeng}@bytedance.com)
• C. Lu, Q. Hou and M.M. Cheng are with School of Computer Science, Nankai University, Tianjin, China. (czlu919@outlook.com, andrewhoux@gmail.com, cmm@nankai.edu.cn)
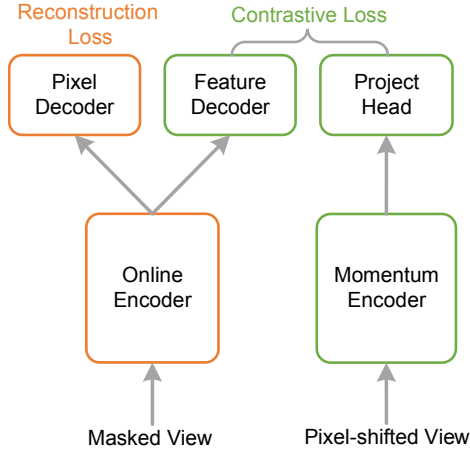• Corresponding author: **Xiaojie Jin** and **Jiashi Feng**

Fig. 1: Overview of CMAE. CMAE improves over its MIM counterpart by leveraging contrastive learning through novel designs. To make contrastive learning compatible with MIM, we propose a feature decoder to complement the masked features and a weakly spatial shifting augmentation method for generating plausible contrastive views.

Fig. 2: Comparisons with previous state-of-the-art MIM methods on ImageNet-1K in terms of top-1 accuracy at different pre-training epochs.

encoder that provides contrastive learning supervision. To leverage the contrastive learning to improve the feature quality of encoder output, we introduce an auxiliary feature decoder into the online branch, whose output features are used for contrastive learning with the momentum encoder outputs.

We carefully design each CMAE component to enable contrastive learning to benefit the MIM. Different from online encoder whose inputs only contain the visible patches, the CMAE momentum encoder is fed with the full set of image patches. This design ensures semantic integrity of its output features to guide the online encoder. Another notable design choice is: our method uses two decoders, one is to predict the image pixel and perform the MIM task; and another is to recover the features of masked tokens. Since the semantics of each patch are incomplete and ambiguous, it is problematic to use the features of patches directly for contrastive learning. Using an auxiliary feature decoder can address this issue and thus benefit the latent representation learning within the online branch. Moreover, different from existing methods that use strong spatial data augmentations for inputs, we propose a pixel shifting augmentation method for generating more plausible positive views in contrastive learning. Such a simple augmentation is proven effective for improving MIM with contrastive learning. With the above novel designs, the online encoder of our CMAE method can learn more discriminative features of holistic information and achieve state-of-the-art performance on various pre-training and transfer learning vision tasks.

Our contributions are summarized as follows. 1) We propose a new CMAE method to explore how to improve the representation of MIM by using contrastive learning. Its learned representations not only preserve the local context sensitive features but also model the instance discriminativeness among different images. 2) To impose contrastive learning upon MIM, we propose a feature decoder to complement the masked features and a weakly spatial shifting augmentation method for generating plausible contrastive views, both
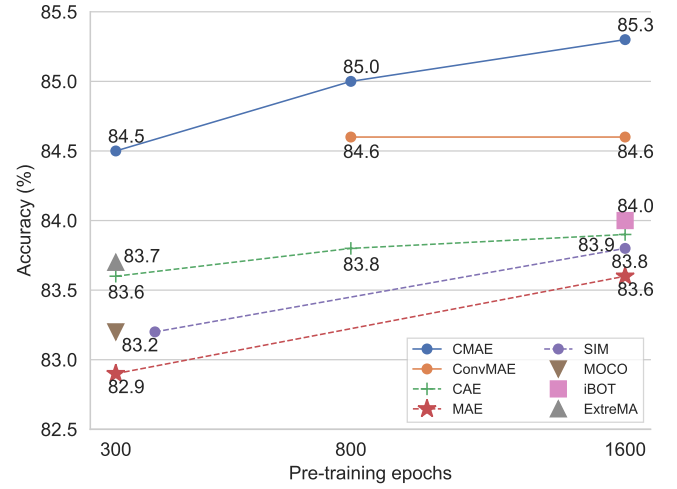
of which are effective in improving the encoder feature quality. 3) As shown in Figure 2, our method significantly improves the learned representation of MIM and sets new state-of-the-art performance. Notably, compared with prior arts, CMAE achieves absolute gains of 0.7% on ImageNet-1k classification validation split, 1.8% mIoU on ADE20K semantic segmentation validation dataset and 0.4% AP^b and 0.5% AP^m on CoCo validation split.

2 RELATED WORK

Self-supervised learning is attracting increasing attention in computer vision. A bunch of methods have been proposed to advance this technique from different perspectives [4, 12, 22, 26, 30, 42, 60, 65]. Broadly speaking, these methods can be categorized into two groups depending on their employed pretext tasks, i.e., contrastive learning [27, 29, 32] and mask image modeling [3, 30, 56].

Contrastive learning aims to learn instance discriminative representations to distinguish an image from the others. This is achieved by pulling together the representation of different views of an individual image and pushing away the other images. Thus most contrastive methods adopt siamese network [13, 26, 29]. To create different view for the same image, a plentiful of methods have been deployed [12, 42, 47]. Among these methods, data augmentations are most commonly in contrastive learning and investigated in Sim-CLR. [12]. In practice, contrastive methods rely heavily on a large number of negative samples [12, 29, 47, 53]. To better utilize negative samples, MoCo [29] uses a large queue to cache negative examples in memory such that it can take in more negative examples for contrastive learning. BYOL [26] uses an online encoder to predict the output of a momentum encoder, where the momentum encoder is key to avoiding training collapse. To simplify BYOL, SimSiam [13] proposes the stop-gradient technique to replace the momentum updating. Besides, there are some methods to solve this issue from different views [7, 8, 9, 60]. SwAV [9] proposed an online clustering method and adopted the Sinkhorn-Knopp [16] transform to assist the clustering for each batch. Barlow Twins [60] adopts the cross-correlation matrix to constrain
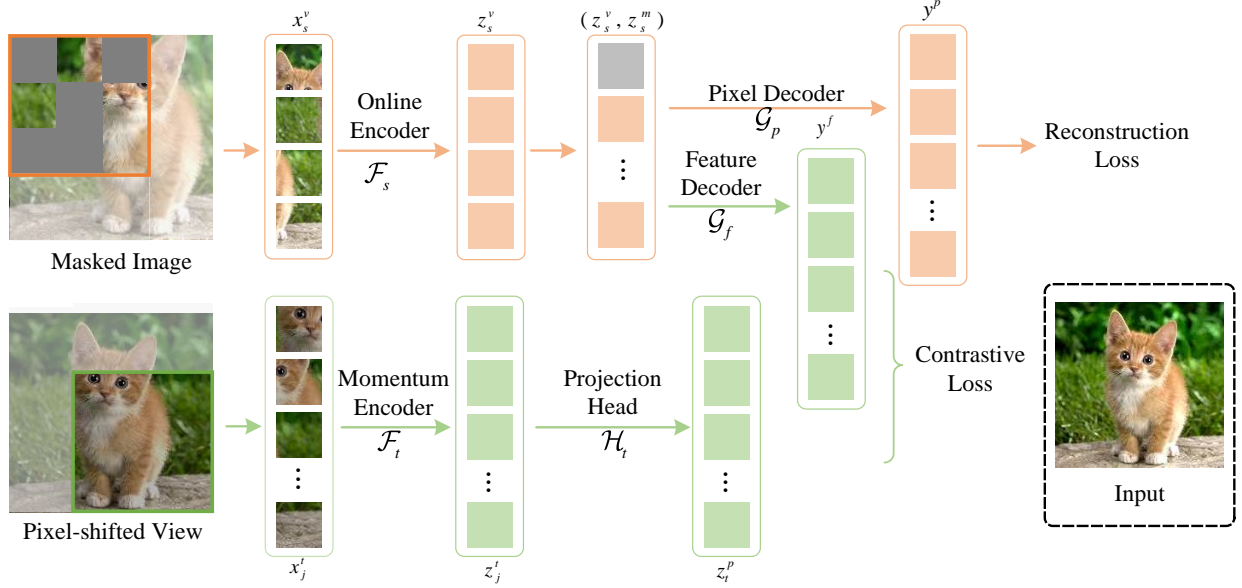
Fig. 3: Overall pipeline. Our method contains three components: the online encoder, momentum encoder, and online decoder. Given a training image, it applies *pixel shifting* to generate different views, which are then fed into the online and momentum encoders respectively. The online encoder randomly masks a fraction of the image patches and operates on the visible ones. The momentum encoder operates on the whole view after pixel shifting. The pixel decoder learns to reconstruct the input image from the image tokens (along with MASK tokens) provided by the online encoder, while the *feature decoder* learns to predict the features of the input image for contrastive learning with the momentum encoder output features. During pre-training, the parameters of the momentum encoder and projection head are updated using an exponential moving average algorithm. After the pre-training, only the online encoder is kept for downstream applications.

the network to void training collapse. Recently, MoCo-v3 [14] and DINO [10] are proposed to extend MoCo [29] and BYOL [26] respectively by using Vision Transformer (ViT) as their model backbones. Although contrastive learning methods provide discriminative vision representations, most of them focus on learning global representations while lacking spatial-sensitive representation.

Mask image modeling [3, 11, 21] is inspired by the success of Masked Language Modeling in NLP [19, 43] and learns vision representation by constructing the original signal from partial observations. Based on the reconstruction target, these methods can be devided into: pixel-domain reconstruction [22, 30, 38, 50, 56] and auxiliary features/tokens prediction [3, 15, 20]. SimMIM [56] and MAE [30] propose to reconstruct the raw pixel values from either the full set of image patches (SimMIM) or partially observed patches (MAE) to reconstruct the raw image. Compared with SimMIM, MAE is more pre-training efficient because of masking out a large portion of input patches. To learn more semantic features, MaskFeat [50] introduces the low-level local features (HOG [17]) as the reconstruction target, and Ge^2-AE adopts the MIM task in frequency domain [38], while CIM [22] opts for more complex input.

Several methods adopt an extra model to generate the target to pre-train the encoder. For instance, BEiT [3] uses the discretized tokens from an offline tokenizer [44] to train the encoder. PeCo [20] instead uses an offline visual vocabulary to guide the encoder. Differently, CAE [15] uses both the online target and offline network to guide the training of encoder. Furthermore, iBOT [65] introduces an online tokenizer to produce the target to distill the encoder. MVP [51] adopts a structure identical to BEiT [3] and replaces

the tokenizer (e.g. d-VAE in BEiT [3]) with the vision branch of a multimodal model like CLIP that is pre-trained on image-text pairs. Similarly, SIM adopts the siamese network to reconstruct the representations of tokens, based on another masked view [46]. MSN [1] matches the representation of masked image to that of original image using a set of learnable prototypes.

Despite MIM models exhibiting favorable optimization properties [52] and delivering promising performance, their focus is on learning relationships among the tokens in the input image, rather than modeling the relation among different images as in contrastive learning, which results in less discriminative learned representations. Our method diverges from existing works by proposing innovative designs to fully leverage the advantages of MIM and contrastive learning, thereby providing local context-sensitive representations with the desired discriminativeness for input images.

3 METHOD

3.1 Framework

The overall framework of our method is illustrated in Figure 3 that consists of three components. The *online encoder and decoder* learn to reconstruct input images from masked observations. Different from existing MIM methods (e.g., MAE [30] and SimMIM [56]), our method further processes the input image via a spatially shifted cropping operation. More importantly, our decoder incorporates an additional feature decoder for predicting the input image features. The *momentum encoder* transforms the augmented view of the input image into a feature embedding for contrastive learning with the predicted one from the online feature decoder. In

this way, the learned representations by the online encoder capture not only holistic features of the input images but also discriminative features, thus achieving better generalization performance. We now elaborate on these components in detail.

Let us denote the input image I_s to the online encoder as being tokenized into a sequence of N image patch tokens $\{x_i^s\}_{i=1}^N$, where N is the total number of image patches. For the masked version of I_s, the set of visible tokens is represented as $\{x^v\}$. Similarly, the input image I_t is tokenized into a sequence of image patch tokens $\{x_j^t\}_{j=1}^N$ to serve as the input to the momentum encoder.

Online encoder. The online encoder $\mathcal{F}_s$ maps the visible tokens x_s^v to embedding features z_s^v. Given the token sequence $\{x_i^s\}_{i=1}^N$, we mask out a large ratio of patches and feed the visited patches to the online encoder. The online encoder adopts the Vision Transformer (ViT) architecture [21], following MAE [30]. It first embeds the visible tokens x_s^v by linear projection as token embeddings, and adds the positional embeddings [49] p_s^v. We feed the fused embedding to a sequence of transformer blocks, and get the embedding features z_s^v.

$$z_s^v = \mathcal{F}_s(x_s^v + p_s^v) \tag{1}$$

After pre-training, the online encoder $\mathcal{F}_s$ is used for extracting image representations in downstream tasks.

Momentum encoder. The momentum encoder is introduced for providing contrastive supervision for the online encoder to learn discriminative representations. Different from existing siamese-based methods [10, 65], our momentum encoder $\mathcal{F}_t$ only serves for contrastive learning, as well as guiding the online encoder to learn more discriminative features. It shares the same architecture as the online encoder $\mathcal{F}_s$, but takes the whole image as input, in order to reserve the semantic integrity and the discriminativeness of the learned representations. Using the whole image as input to the momentum encoder is important for the method performance, which is experimentally verified in Section 4.4. Unlike tokens in NLP, whose semantic are almost certain, image token is ambiguous in its semantic meaning [65]. To avoid ambiguity, we adopt global representations for contrastive learning. The mean-pooled feature of momentum encoder is used for its simplicity, i.e.

$$z^t = \frac{1}{N} \sum_{j=1}^N \mathcal{F}_t(x_j^t), \tag{2}$$

where x_j^t is the input token for momentum encoder, z_j^t is the output sequence of momentum encoder and z^t is the feature obtained after performing mean pooling operation on the output sequence z_j^t, which is used to represent the input image.

Different from the online encoder, we update parameters of the momentum encoder by exponential moving average (EMA). That is, denoting the parameters of $\mathcal{F}_s$ and $\mathcal{F}_t$ as θ_s and θ_t respectively, the parameters are updated by $\theta_t \leftarrow \mu\theta_t + (1-\mu)\theta_s$. Here μ is fixed as 0.996 across the experiments. Momentum update is used since it stabilizes the training by fostering smooth feature changes, as found in MoCo [29] and BYOL [26].

Online decoder. The decoder aims to map the latent features z_s^v and MASK token features to the feature space of the

momentum encoder and the original images. Specifically, the decoder receives both the encoded visible tokens z_s^v and MASK tokens z_s^m.

Similar to MAE [30], position embeddings are added to input tokens. Due to different mapping targets, our online decoder has two branches of decoder structure, one is a pixel decoder, and another is a feature decoder. The pixel decoder $\mathcal{G}_p$ learns to reconstruct the pixel of the masked patches. We use the full set of tokens, which contains both z_s^v and z_s^m, to predict the pixel of patches y^m. This module can promote the model to learn holistic representation for each patch in an image. We set the pixel decoder to be stacked transformer blocks:

$$y'_m = \mathbb{I} \cdot \mathcal{G}_p(z_s^v, z_s^m), \tag{3}$$

where $\mathbb{I}$ is an indicator to only select the prediction corresponding to masked tokens from output sequence y^p, and y'_m is the output prediction for the masked patches.

To align with the output of the momentum encoder, feature decoder $\mathcal{G}_f$ is applied to recover the feature of masked tokens. The feature decoder has the same structure as the pixel decoder but non-shared parameters for serving a different learning target. The prominence of such design choices will be discussed in the architecture part in Section 3.4. Given the encoded visible tokens z_s^v, we add the masked tokens z_s^m and use this full set to predict the feature of masked tokens. Similar as done in momentum encoder, we apply the mean pooling operation on the output of feature decoder y^f as the whole image representation y_s, and then use this feature for contrastive learning.

$$y_s = \frac{1}{N} \sum \mathcal{G}_f(z_s^v, z_s^m), \tag{4}$$

where N is the total number of tokens in the full set.

3.2 View augmentations

Typically, masking image modeling pre-training tasks only utilizes a single view of the input image, which only contains visited patches. But contrastive learning often adopts two different augmented views. To make MIM and contrastive learning be compatible with each other, our method also generates two different views and feeds them to its online and momentum branches, respectively.

In contrastive learning, the most commonly used view augmentations can be divided into two types: spatial transfer (e.g., random resized cropping, flipping) and color transfer (e.g., color jittering and random grayscaling). For MIM tasks, color enhancements degrade the results [30], so we do not apply them to the input of the online branch. Spatial and color data augmentations are applied to the momentum branch input to avoid a trivial solution.

We first consider two branches using two different random crops, following the common practice in contrastive learning. However, we observe this recipe has an adverse effect on model performance (refer to Section 4.4). We conjecture that this issue is related with the large disparity between the inputs of online/momentum encoders when randomly cropped regions are far apart or scarcely semantic-relevant. Different with using intact paired views in usual contrastive methods, the operation of masking out a large portion of input in MIM may amplify such disparity and

	Training objective			Input	Architecture		Accuracy
	reconstruct loss	intra-view match	inter-image contrast	positive view alignment	feature complement	separate encoder/decoder	
MSN [1]	✗	✓	✗	✗	✗	✗	83.4
ExtreMA [54]	✗	✓	✗	✓	✗	✗	83.7
MAE [30]	✓	✗	✗	N.A.	N.A.	✓	83.6
CAE [15]	✓	✗	✗	✗	✗	✓	83.9
iBot [65]	✓	✓	✗	✗	✗	✗	84.0
SIM [46]	✓	✓	✓	✗	✗	✓	83.8
CMAE	✓	✓	✓	✓	✓	✓	84.7

TABLE 1: Comparison of CMAE with previous methods on training objective, input generation and architecture. The top-1 accuracy on ImageNet is also presented. Please refer to Section 3.4 for more detailed explanations.

therefore creates false positive views. Consequently, performing contrastive learning on these misaligned positive pairs actually incurs noise and hampers the learning of discriminative and meaningful representations.

To address the above issue, we propose a weakly augmentation method named **pixel shifting** for generating the inputs of online/momentum encoders. The core idea is first to obtain a master image by a resized random cropping from the original image. Then two branches share the same master image and generate respective views by slightly shifting cropping locations over the master image. In more details, we denote the master image as I. The shape of I is $(w + p, h + p, 3)$, where w, h is the width and height of target input size for our model and p is the longest shifting range allowed. For online branch, we use the region of $[0 : w, 0 : h, :]$ as our input image I_s. For momentum branch, we use the region of $[r_w : r_w + w, r_h : r_h + h, :]$ as our input image I_t. r_w and r_h are independent random values in the range of $[0, p)$. Subsequently, we only apply masking operation without color augmentation for the input of online encoder I_t, which is consistent with MAE. For the momentum encoder's input image, we utilize color augmentation but do not apply masking operation. The distinct augmentations for each encoder input generate different views of the same image to facilitate contrastive learning between the online and momentum encoders.

3.3 Training Objective

Reconstruction loss. Following [30], we use the normalized pixel as target in the reconstruction task. We adopt the Mean Squared Error (MSE) as loss function and compute the loss only on masked patches between the pixel decoder prediction and the original image. The math formulation is

$$L_r = \frac{1}{N_m} \sum (y'_m - y_m)^2, \qquad (5)$$

where N_m is the number of masked patches in an image, and L_r is the loss value.

Contrastive loss. For clarity, we describe the contrastive loss design of our method from two aspects: loss function and head structure. Two widely used loss functions are taken into consideration, i.e. InfoNCE [12, 29] loss and BYOL-style [10, 26] loss. The former seeks to simultaneously pull close positive views from the same sample and push away negative samples while the latter only maximizes the similarity between positive views. Although some recent

works find they may be inherently unified [45], we still analyze them separately due to their diverse effects on representation learning. In our method, we observe better performance using InfoNCE [42] so we use it defaultly. Details are referred to Section 4.4. For the head structure, we adopt the widely used "projection-prediction" structure following [14, 26]. Specifically, we append the "projection-prediction" and "projection" head to feature decoder and momentum encoder respectively. The projection head $\mathcal{H}_t$ with momentum encoder is also updated by exponential moving average. Due to the large differences on generating inputs for online/momentum encoder (refer to Section 3.2), we use asymmetric contrastive loss, which is distinguished from previous methods [14, 26]. The representation from feature decoder y_s is transformed by the "projection-prediction" structure to get y_s^p. Similarly for the representation from momentum encoder z^t, we apply the projection head and get z_t^p. We then compute the cosine similarity ρ between them:

$$\rho = \frac{y_s^p \cdot z_t^p}{\|y_s^p\|_2 \|z_t^p\|_2}. \qquad (6)$$

We denote ρ^+ as the positive pairs cosine similarity, which is constructed by y_s^p and z_t^p from the same image. ρ_j^- indicates the cosine similarity for the j-th negative pair. We use the z_t^p from different images in a batch to construct negative pairs. The loss function of InfoNCE loss is

$$L_c = -\log \frac{\exp(\rho^+/\tau)}{\exp(\rho^+/\tau) + \sum_{j=1}^{K-1}(\exp(\rho_j^-/\tau))}, \qquad (7)$$

where τ is the temperature constant, which is set to 0.07. K is the batch size.

The overall learning target is a weighted combination of reconstruction loss L_r and contrastive loss L_c defined as:

$$L = L_r + \lambda_c L_c. \qquad (8)$$

3.4 Connections and analysis

To elucidate the correlations and distinctions between the CMAE and preceding methodologies, we undertake comparative assessments from various perspectives such as training objective, input, and architecture. The outcomes are demonstrated in Table 1. Our primary focus is on methodologies that utilize either contrastive information in MIM or masked image input. Approaches that solely employ MIM or contrastive learning are not within the purview

Method	Pre-training epochs	Params.(M)	Supervision	Accuracy
MoCo-v3 [14]	300	86	RGB	83.2
DINO [10]	300	86	RGB	82.8
CIM [22]	300	86	RGB	83.3
BEiT [3]	800	86	DALLE	83.2
SimMIM [56]	800	86	RGB	83.8
PeCo [20]	800	86	Perceptual Codebook	84.5
MaskFeat [50]	1600	86	HOG	84.0
CAE [15]	1600	86	DALLE+RGB	83.9
iBOT [65]	1600	86	RGB	84.0
SIM [46]	1600	86	RGB	83.8
MAE [30]	1600	86	RGB	83.6
CMAE (ours)	800	86	RGB	84.4
CMAE (ours)	1600	86	RGB	**84.7**
ConvMAE* [23]	800	86	RGB	84.6
ConvMAE* [23]	1600	86	RGB	84.6
CMAE* (ours)	800	86	RGB	85.0
CMAE* (ours)	1600	86	RGB	**85.3**

TABLE 2: Comparison of our model with existing methods on ViT-B. We evaluate them with the top-1 accuracy on ImageNet. The symbol of * throughout experiments denotes using convolutions instead of linear transformation as the tokenizer for visual patches.

of this discussion, as they are evidently distinct from our method.

Training objective. The CMAE leverages both the reconstruction loss and contrastive loss during optimization. As inferred from Eq. (7), the contrastive loss in CMAE encompasses both intra-view matching and inter-image contrast. Consequently, the generated representations are encouraged to exhibit desirable characteristics of instance discrimination and spatial sensitivity. In contrast, methodologies such as MSN [1] and ExtreMA [54], which have divergent motivations from ours, disregard reconstruction loss and employ masked input for regularization or data augmentation purposes. iBot [65] exclusively adopts a distillation loss between positive views by maximizing intra-view matching scores, overlooking contrastive learning with negative samples. Additionally, CMAE only adopts an asymmetric loss for contrastive learning, which is less computationally expensive than iBot. Although SIM [46] also utilizes both losses, it differs from CMAE in terms of the reconstruction target. While CMAE restores the masked content of the same view, SIM reconstructs the features of another view. Our empirical results demonstrate that CMAE is not just simpler but also more effective in representation learning, as evidenced by superior performance.

Input. The majority of prior contrastive learning methods [12, 29] implement robust augmentation techniques (e.g., random crop, random scale) to generate positive views from the same image. These operations are also commonly used in contrastive learning models under the masked image modeling scenario, such as the iBot. However, considering that the masking operation, which utilizes a large masking ratio (e.g. 75% in [30]), already significantly degrades the input, applying these augmentations further could generate invalid positive views, thereby hindering contrastive learning. In contrast, we propose a novel, moderate data augmentation operation called pixel shifting for achieving better alignment between positive views. Compared to ExtreMA [54], which employs the exact same view in two siamese branches, pixel shifting introduces a moderate input variance, which proves

beneficial for contrastive learning (refer to Table 4a).

Architecture. In CMAE, a lightweight feature decoder is appended after the online encoder to supplement the masked features. This is a notable distinction from other methods, such as SIM and iBot, which directly utilize the representations of the visible patches to match that of the unmasked view. We contend that conducting contrastive learning between the features of the masked parts and the input images is impractical, considering they exhibit different levels of abstraction and semantic coverage. The feature decoder is anticipated to facilitate optimization by diminishing the distribution gap between contrastive features. The efficacy of the feature decoder is empirically validated, as shown in Table 4c. Notably, the design of CMAE is non-intrusive, allowing for its straightforward application to existing MIM models, such as MAE and ConvMAE, without necessitating significant modifications to the MIM model.

4 EXPERIMENTS

4.1 Implementation Details

Pre-training. We follow the settings of MAE [30] to pre-train our model. We adopt AdamW [40] optimizer as default, and the momentum is set to $\beta_1 = 0.9$, $\beta_2 = 0.95$. Besides, the weight decay is set to 0.05. We use the linear scaling rule [25]: $lr = base\_lr \times batch\_size/256$ to set the learning rate. The base learning rate is 1.5×10^{-4} with a batch size of 4096. Cosine learning rate schedule [39] with a warmup of 40 epochs is adopted. All pre-training experiments are conducted on 32 NVIDIA A100 GPUs.

Encoder Structure. We use the ViT [21] base model as our default setting. To further validate the extensibility of our proposed model, we replace the ViT with a hybrid convolutional ViT which is also used by ConvMAE [23]. In the hybrid ViT, a multi-layer convolutional network [34] is used as token projection. Note the hybrid ViT is made to have the same model size as the ViT counterpart for fair comparison. We also experiment with scaled up encoders for evaluating the scalability of our method.

Method	Pre-Epochs	mIoU
MoCo-v3 [14]	300	47.3
DINO [10]	400	47.2
BEiT [3]	800	47.1
CIM [22]	300	43.5
CAE [15]	1600	50.2
iBOT [65]	1600	50.0
PeCo [20]	800	48.5
MAE [30]	1600	48.1
CMAE	1600	**51.0**
ConvMAE* [23]	1600	50.7
CMAE*	1600	**52.5**

(a) Semantic segmentation results on ADE20K. We use UperNet [55] as our default segmentation framework.

Method	Pre-Epochs	AP^{bbox}	AP^{mask}
MoCo-v3 [14]	300	47.9	42.7
BEiT [3]	800	49.8	44.4
CAE [15]	1600	50.0	44.0
iBOT‡ [65]	1600	51.2	44.2
PeCo [20]	800	44.9	40.4
SIM [46]	1600	49.1	43.8
MAE† [30]	1600	51.7	45.9
MAE [30]	1600	50.3	44.9
CMAE	1600	**52.4**	**46.5**
ConvMAE* [23]	1600	52.5	46.5
CMAE*	1600	**52.9**	**47.0**

(b) COCO object detection and segmentation. We use the Mask R-CNN model [28] as our framework. $\ddagger$ means using Cascade Mask R-CNN [6].

TABLE 3: Performance comparison on downstream tasks, including semantic segmentation and object detection. The symbol of * denotes using convolutions to embed visual patches. $\dagger$ denotes reproduced results of ours.

Setting	Accuracy
Baseline [30]	82.9
+ Contrastive learning	83.1
+ Pixel shifting aug.	83.6
+ Feature decoder	83.8

(a) Component analysis.

Rand crop	Pixel shift	Color Aug.	Accuracy
✗	✗	✗	82.9
✓	✗	✗	83.0
✗	✓	✗	83.4
✗	✓	✓	83.8

(b) Data augmentation analysis.

#Blocks	Share weight	Accuracy
0	✗	83.6
2	✗	83.8
2	✓	83.4
4	✗	83.8
4	✓	83.5

(c) Feature decoder analysis.

Loss weight	Accuracy
0.1	83.3
0.5	83.7
1.0	83.8
1.5	83.5
2.0	83.2

(d) Contrastive loss weight.

Masking ratio	Accuracy
0	83.8
0.25	83.6
0.5	83.3
0.65	83.3
0.75	83.0

(e) Momentum encoder masking ratio.

TABLE 4: Ablations. We evaluate all models on ImageNet-1k with their top-1 classification accuracies. Each model is pre-trained for 300 epochs.

4.2 Results on ImageNet

Following existing works [3, 14, 30, 56], we use ImageNet-1K [18] which consists of 1.3M images of 1k categories as the pre-training and fine-tuning dataset. The dataset contains two subsets: the training set and the validation set. We only use the training set to pre-train CMAE. After pre-training, the CMAE online encoder is used for fine-tuning on ImageNet-1k training set for 100 epochs. For the model pre-trained with 300 epochs, we adopt $5.e^{-4}$ as the base learning rate in fine-tuning. Since the longer pre-training schedule (1600 epochs) makes the model learn better initialization weights for fine-tuning [57], we set a smaller base learning rate of $2.5e^{-4}$. Besides, we follow the common fine-tuning practices to regularize the model using mixup [61], cutmix [59], drop path [33], etc.

In Table 2, we compare CMAE with competing methods on the fine-tuning classification accuracy on ImageNet. CMAE achieves a top-1 accuracy of 84.7%, which is 1.1% higher than MAE [30]. Among all models using ViT architecture, CMAE achieves the best performance. Compared with contrastive learning based methods Moco-v3 [14] and DINO [10], our model can significantly outperform them

by 1.5% and 1.9% respectively. Compared with iBOT and SIM which also use contrastive objective in MIM, our CMAE achieves higher performance with a gain of 0.7% and 0.8%, respectively. Above results strongly evidence the superiority of CMAE.

When we replace the vanilla ViT encoder with a hybrid convolutional ViT, as done in ConvMAE [23], CMAE further improves to 85.0% and 85.3% with the pre-training of 800 epochs and 1600 epochs, respectively. These results surpass those of ConvMAE under the same pre-training setting by 0.4% and 0.7% respectively, verifying the excellent extendibility of CMAE to various network structures.

Remarkably, CMAE can gain a noticeable improvement with a prolonged training schedule (from 800 epochs to 1600 epochs) while ConvMAE is observed to saturate at 800 epochs. This result suggests the stronger capability of CMAE on learning better representations.

4.3 Transfer Learning

To further validate the transferability of CMAE, we follow previous methods to evaluate pre-trained models on the

Method	iNat2017	iNat2018	iNat2019	Places365
MAE	70.5	75.4	80.5	57.9
CMAE	72.2	76.4	82.2	58.9

TABLE 5: Transfer learning accuracy on classification datasets.

semantic segmentation dataset ADE20K [64], the object detection dataset COCO2017 [37] and classification datasets. **Semantic segmentation.** ADE20K [64] has 25,562 images of 150 fine-grained categories. We adopt Upernet [55] as the default model for this task, following the settings of compared methods. The backbone ViT-B is initialized from pre-training while other modules are initialized with the Xavier [24] initialization. The model is fine-tuned on the training set of ADE20K and tested on standard validation split.

Following previous works, we report the Mean Intersection over Union (mIoU) performance of CMAE in Table 3a. We notice that CMAE significantly surpasses MAE by 2.9%, which verifies the stronger transferability of CMAE. Besides, CMAE also improve by 1.0% and 0.8% compared with iBOT [65] and CAE [15] respectively. With the same hybrid ViT backbone, CMAE significantly outperforms ConvMAE by 1.8%. Remarkably, CMAE sets a new state-of-the-art result of 52.5 by surpassing all competing methods with a large margin.

Object Detection and Segmentation. We adopt the widely used object detection and instance segmentation framework Mask-RCNN [28, 36] for benchmarking on this task. ViT-B is used as the backbone and initialized with our pre-trained model. Following MAE, we fine-tune the model on COCO train2017 split, and report box AP for object detection and mask AP for instance segmentation on val2017 split. We fine-tune the model for 100 epochs. The base learning rate is $1.e^{-4}$ with a cosine annealing schedule, and the weight decay is set to 0.1.

The comparison results with other self-supervised learning methods are shown in Table 3b. As one can see, CMAE improves over MAE from 51.7 to 52.4 on AP^b and from 45.9 to 46.5 on AP^m. With the hybrid ViT structure, CMAE consistently surpasses the competing method ConvMAE: AP^b increases from 52.5 to 52.9 and AP^m increases from 46.5 to 47.0. Above promising results again verify the effectiveness of our method.

Classification tasks. To further study transfer learning on classification tasks, we validate our model on the iNaturalists [48] and Places [63] in Table 5. Experiments across four classification tasks on these datasets showed consistent improvements of 1.0% to 1.7% in top-1 accuracy over the MAE [30]. These results provide further evidence for the efficacy of our method in enhancing the discriminative capabilities of pretaining model.

4.4 Method Analysis

To understand the effects of key components and validate design choices we adopt in CMAE, we conduct a series of ablation experiments. Unless otherwise stated, we report the performance of our model with 300 pre-training epochs in this subsection. The ablative results are listed in Table 4. In the following, we verify the effectiveness of our main design ideas, then conduct ablation experiments for each component separately.
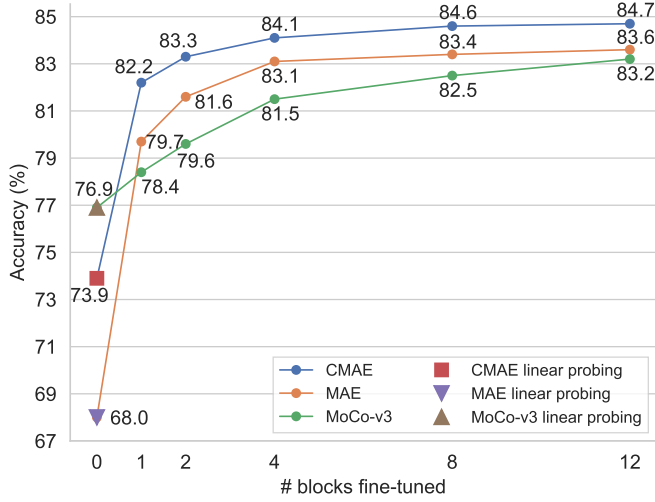
Ablation of components. In Table 4a, we show how each component, i.e. contrastive learning, pixel shifting data augmentation and feature decoder affects model's performance. We start with a vanilla implementation of contrastive learning on MAE. Specifically, following the input generation approach in contrastive methods, random cropped regions with masking are fed into online/momentum encoder. The same contrastive objective as introduced in Section 3.3 are optimized between the output of online encoder and momentum encoder. As can be seen from Table 4a, such an intuitive approach only leads to marginal improvement (0.2%). Apparently, the power of contrastive learning is not fully unleashed due to ignoring its compatibility with MIM. By using the proposed moderate data augmentation, i.e. pixel shifting, the result can increase from 83.1% to 83.6%, which evidences the advantage of pixel shifting. Moreover, applying feature decoder further boosts the model's learning capability by improving the performance to 83.8%, demonstrating its effectiveness in our method.

Contrastive loss. To explore the effect of contrastive loss in CMAE, we experiment with various loss weights, i.e. λ_c in Eq. (8). The results are shown in Table 4d. Note CMAE degenerates to the baseline MAE when loss weight is 0. When increasing the weights from 0 to 1, the model's performance increases accordingly, which verifies the importance of contrastive learning on enhancing the learned representations. When the weight of contrast learning is greater than that of MIM, we observe the phenomenon of imbalanced training occurs which adversely affects the final performance. This experiment demonstrates that both contrastive loss and reconstructive loss are critical for learning capable representations. We therefore set $\lambda_c = 1$ throughout our experiments.
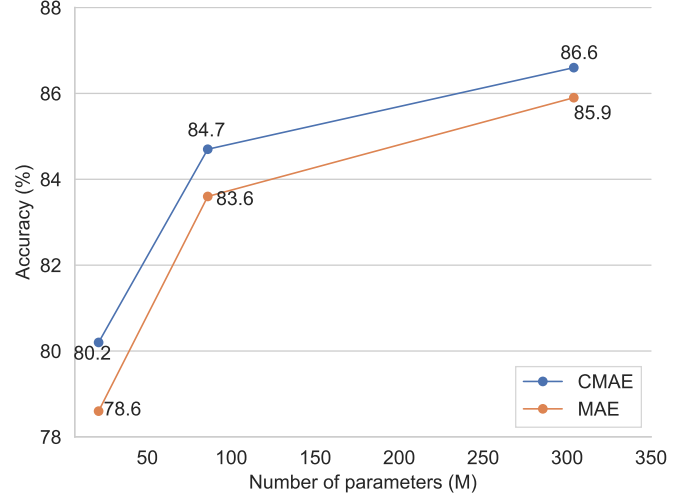
We also conduct controlled experiments with different contrastive loss forms to compare their influences on pre-training. Under the same configuration, we observe that the model trained with InfoNCE loss achieves higher performance than BYOL-style loss (83.8% vs. 83.4%). This result suggests that the way of utilizing negative samples in InfoNCE is more effective in our method.

Pixel shifting augmentation. In this section, we ablate on the importance of data augmentations. In contrast to the common practices of applying heavy data augmentation in contrastive learning, we find a moderate data augmentation is more effective in aligning contrastive learning and MIM. We divide data augmentation methods into two kinds: spatial transfer and color transfer, and evaluate their effect respectively. For spatial transfer, we compare our proposed pixel shifting with the commonly used randomly resized cropping. For color transfer, we compare two cases, i.e. with or without using color jittering for the momentum branch.

As can be seen from Table 4b, pixel shifting significantly surpasses random crop (83.4% vs. 83.0%). The superiority of pixel shifting should be attributed to its ability of generating more plausible positive views. As introduced in Section 3.2, this property helps contrastive learning better collaborate with MIM in our framework. By using color transfer, the result further improves to 83.8%, suggesting color transfer is

(a) Partial fine-tuning results using the ViT-B backbone.

(b) Scaling results with different model sizes.

Fig. 4: Performance comparison on ImageNet-1k with partial fine-tuning and model scaling. In partial fine-tuning, we adopt the model weights pre-trained with 1600 epochs. In model scaling experiments, all models are pre-trained with 1600 epochs.

Shift value	random crop	0	0-15	0-31	0-47	0-63	32-47	48-63
Accuracy	83.29	83.68	83.78	83.82	83.71	83.64	83.54	83.48

TABLE 6: Impact of varying pixel shift ranges on the ImageNet-1k classification task. "Random crop" serves as the baseline method for comparison against our proposed pixel shift approach.

complementary to our method.

We investigate the impact of different pixel shift ranges by varying the maximum allowable shift. Intuitively, larger shift ranges introduce greater misalignment between the two augmented views. As evident in Table 6, excessive shifts severely degrade model performance, conforming to our hypothesis that severely misaligned positive pairs introduce noise detrimental to contrastive learning. The results demonstrate an optimal balance exists between view diversity and alignment. Based on the observed performance across different shift ranges, we select a maximum pixel shift of 31 as the default parameter setting to maximize contrastive learning while preserving sufficient alignment.

Feature decoder. Different from existing works, we introduce a feature decoder to recover the features of masked patches when performing contrastive learning. To investigate its effectiveness, we present experiments under following two settings: sharing the weight between feature decoder and pixel decoder or not, and changing the depth of feature decoder.

In Table 4c, number "0" means not using feature decoder, i.e. the output of the online encoder which contains only the features of visible tokens is used for contrastive learning. Under this setting, our method performs worse than using a lightweight two-layer feature decoder. When increasing the depth of feature decoder, there is no significant impact on performance. However, when the depth increases to 8, we obtain a trivial solution, possibly due to the optimization difficulty caused by deeper structure. To strike a balance between efficiency and effectiveness, we set the depth to be 2. Besides, when the feature decoder shares the weight with the
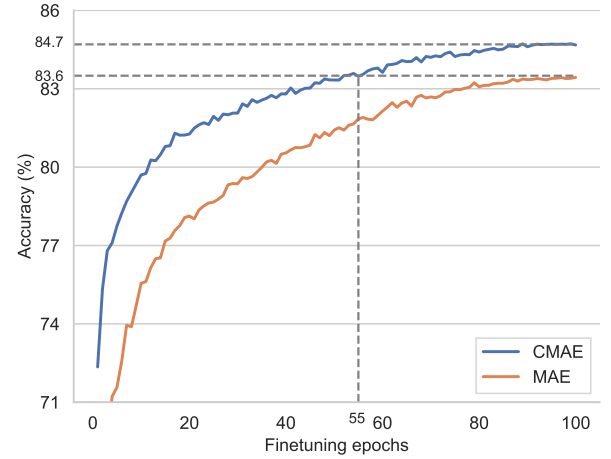

Fig. 5: Convergence speed compared with MAE. All models are pre-trained with 1600 epochs.

pixel decoder, the method performs the worst. A plausible explanation is that the two branches have different targets, thus should adopt independent weights.

Masking ratio for the momentum branch. In this experiment, we investigate whether masking a portion of image patches for the momentum branch affects the model performance. Following previous works, we select a set of masking ratios, including $\{0, 0.25, 0.5, 0.65, 0.75\}$ for the momentum branch and see how the performance changes. As shown in Table 4e, one can observe that using the complete set of the image tokens yields the best results. A possible reason is that: since the aim of adding the momentum branch is to provide our model with the contrastive supervision, incorporating the full semantics of an image is preferred. Otherwise, the masked input with degenerated semantic information may lead to a sub-optimal solution in contrastive learning. Based on this observation, the momentum branch in our model uses the whole image as input throughout our experiments.

Convergence speed. To further show our method's effectiveness, we compare the convergence behavior of CMAE and

(a) Standard variance of intra-class distance.   (b) Average distance of inter-class center   (c) Standard variance of inter-class center distance.
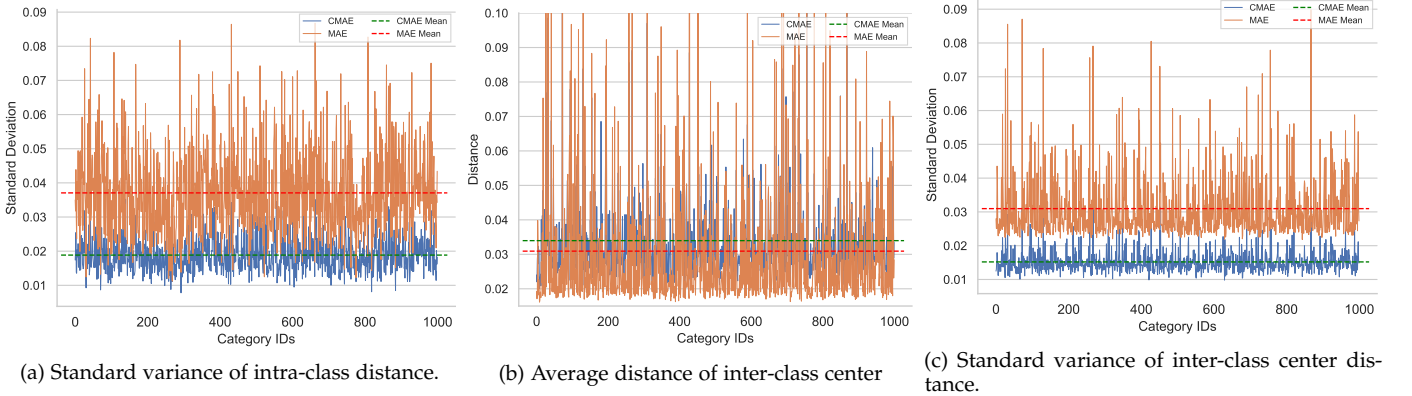
Fig. 6: Analysis of features of pre-training models. In terms of intra-class distance, our model achieves a lower standard deviation relative to MAE on the ImageNet-1k validation set. With respect to inter-class center distance, our model displays greater inter-class distances than MAE for 811 out of the 1000 categories, and manifests a reduced standard deviation of distances among category centers.

MAE when fine-tuning on ImageNet-1k. The pre-trained weights with 1600 epochs are used as initialization. As shown in Figure 5, we observe that CMAE converges much faster compared with MAE: with only 55 fine-tuning epochs, CMAE already surpasses the final performance of MAE. This result demonstrates that the representations learned by CMAE can be more easily adapted for specific tasks, an appealing property which is in line with the purposes of self-supervised pre-training.

4.5 Partial Fine-tuning and Linear Probing

In the context of task-specific training, both partial fine-tuning [30, 41, 58, 62] and linear probing methodologies retain the majority of the model components in a frozen state. However, a key distinction lies in the nature of the head being tuned: partial fine-tuning employs a non-linear head, whereas linear probing utilizes a linear one. As underscored by [30], given that linear probing exhibits minimal correlation with transfer learning performance, partial fine-tuning emerges as a superior protocol for the evaluation of non-linear, yet more potent, representations. In light of these observations, our study also places emphasis on the partial fine-tuning metric.

Specifically, we follow the experimental settings of [30] to ablate the CMAE base model with 1600 epoch pre-training. As shown in Figure 4a, the performances of our model are consistently better than MAE in all tested settings, e.g. when fine-tuning one block, we get a 2.5% gain over MAE. Above results demonstrate that our model can effectively improve the representation quality of baseline method. Note when the number of fine-tuned blocks is "0", it degenerates to linear probing. In this case, our model achieves significant improvement (5.9%) over MAE. These results indicate that our method is able to improve the representation quality under both evaluation metrics. Furthermore, in comparison to the typical contrastive model MoCo-V3 [14], MoCo-V3 exhibits superior performance in the linear probing setting. However, under the partial fine-tuning setting, CMAE surpasses MoCo-V3 in all aspects, particularly when fine-tuning only one block, where CMAE yields a 3.8% enhancement. This also serves as evidence that the features learned by our model are of higher quality.

4.6 Model Scaling

To study the scalability of our method for models of different sizes, we adopt ViT-small, ViT-base, and ViT-large as encoders and report their performance on ImageNet-1k fine-tuning. As shown in Figure 4b, CMAE can consistently boost the performance of MAE at all scales. These results clearly demonstrate the excellent scalability of CMAE.

4.7 Feature Analysis

In order to more effectively scrutinize the features obtained by our model, we utilize the ViT-base model for our investigation. Following the completion of pre-training, we extract features from the ImageNet-1k validation set and compute the following metrics:

- Average intra-class distance: This metric measures the mean distance between all pairs of images within the same class.
- Standard deviation of intra-class distances: This metric measures the variation in distances between images in the same class.
- Average inter-class distance: This metric measures the mean distance between all pairs of images from different class centers.
- Standard deviation of inter-class distances: This metric measures the variation in distances between images from different class centers.

When we compute the average intra-class distance, CMAE achieves a lower average intra-class distance of 0.0377 than MAE's 0.0380. Furthermore, as shown in Figure 6a, CMAE has a smaller standard deviation of intra-class distances than MAE (0.0189 vs. 0.0371). These results suggest that the features extracted by CMAE are more compactly clustered in the latent space. Regarding inter-class distances, we compute the average distance of each class center to other class centers and the standard deviation of distances to other class centers. As shown in Figures 6b and 6c, CMAE demonstrates larger average inter-class distances (0.0340 vs. 0.0309) and a smaller standard deviation for inter-class distances (0.0152 vs. 0.0310). This indicates that the features extracted by CMAE have a more uniform distribution for

each category in the latent space and larger inter-class distances.

In summary, the aforementioned results provide evidence that our model is capable of learning superior visual representations with enhanced discriminability compared to MAE.

5 CONCLUSION

This paper introduces a novel self-supervised learning framework named contrastive masked autoencoder (CMAE) which aims to improve the representation quality of MIM by leveraging contrastive learning. In CMAE, we propose two novel designs from the perspective of input generation and architectures respectively to harmonize MIM and contrastive learning. Through extensive experiments, it is demonstrated that CMAE can significantly improve the quality of learned representation in pre-training. Notably, on three well-established downstream tasks, i.e. image classification/segmentation/detection, CMAE achieves state-of-the-art performance. In the future, we will investigate the scaling up of CMAE to larger datasets and incorporate image-dense caption as another view for contrastive learning training based on CMAE.

REFERENCES

[1] M. Assran, M. Caron, I. Misra, P. Bojanowski, F. Bordes, P. Vincent, A. Joulin, M. Rabbat, and N. Ballas. Masked siamese networks for label-efficient learning. *arXiv preprint arXiv:2204.07141*, 2022.

[2] P. Bachman, R. D. Hjelm, and W. Buchwalter. Learning representations by maximizing mutual information across views. *Advances in Neural Information Processing Systems*, 32, 2019.

[3] H. Bao, L. Dong, and F. Wei. Beit: Bert pre-training of image transformers. *arXiv preprint arXiv:2106.08254*, 2021.

[4] A. Bardes, J. Ponce, and Y. LeCun. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. *arXiv preprint arXiv:2105.04906*, 2021.

[5] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah. Signature verification using a" siamese" time delay neural network. *Advances in Neural Information Processing Systems*, 6, 1993.

[6] Z. Cai and N. Vasconcelos. Cascade r-cnn: high quality object detection and instance segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(5):1483–1498, 2019.

[7] M. Caron, P. Bojanowski, A. Joulin, and M. Douze. Deep clustering for unsupervised learning of visual features. In *European Conference on Computer Vision*, pages 132–149. Springer, 2018.

[8] M. Caron, P. Bojanowski, J. Mairal, and A. Joulin. Unsupervised pre-training of image features on non-curated data. In *International Conference on Computer Vision*, pages 2959–2968. IEEE, 2019.

[9] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in Neural Information Processing Systems*, 33:9912–9924, 2020.

[10] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin. Emerging properties in self-supervised vision transformers. In *International Conference on Computer Vision*, pages 9650–9660. IEEE, 2021.

[11] M. Chen, A. Radford, R. Child, J. Wu, H. Jun, D. Luan, and I. Sutskever. Generative pretraining from pixels. In *International Conference on Machine Learning*, pages 1691–1703. PMLR, 2020.

[12] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning*, pages 1597–1607. PMLR, 2020.

[13] X. Chen and K. He. Exploring simple siamese representation learning. In *Conference on Computer Vision and Pattern Recognition*, pages 15750–15758. IEEE, 2021.

[14] X. Chen, S. Xie, and K. He. An empirical study of training self-supervised vision transformers. In *International Conference on Computer Vision*, pages 9640–9649. IEEE, 2021.

[15] X. Chen, M. Ding, X. Wang, Y. Xin, S. Mo, Y. Wang, S. Han, P. Luo, G. Zeng, and J. Wang. Context autoencoder for self-supervised representation learning. *arXiv preprint arXiv:2202.03026*, 2022.

[16] M. Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in Neural Information Processing Systems*, 26, 2013.

[17] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Conference on Computer Vision and Pattern Recognition*, volume 1, pages 886–893. IEEE, 2005.

[18] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Conference on Computer Vision and Pattern Recognition*, pages 248–255. IEEE, 2009.

[19] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[20] X. Dong, J. Bao, T. Zhang, D. Chen, W. Zhang, L. Yuan, D. Chen, F. Wen, and N. Yu. Peco: Perceptual codebook for bert pre-training of vision transformers. *arXiv preprint arXiv:2111.12710*, 2021.

[21] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[22] Y. Fang, L. Dong, H. Bao, X. Wang, and F. Wei. Corrupted image modeling for self-supervised visual pre-training. *arXiv preprint arXiv:2202.03382*, 2022.

[23] P. Gao, T. Ma, H. Li, J. Dai, and Y. Qiao. Convmae: Masked convolution meets masked autoencoders. *arXiv preprint arXiv:2205.03892*, 2022.

[24] X. Glorot and Y. Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256. JMLR Workshop and Conference Proceedings, 2010.

[25] P. Goyal, P. Dollár, R. Girshick, P. Noordhuis, L. Wesolowski, A. Kyrola, A. Tulloch, Y. Jia, and K. He.

Accurate, large minibatch sgd: Training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*, 2017.

[26] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. Richemond, E. Buchatskaya, C. Doersch, B. Avila Pires, Z. Guo, M. Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. In *Advances in Neural Information Processing Systems*, volume 33, pages 21271–21284, 2020.

[27] R. Hadsell, S. Chopra, and Y. LeCun. Dimensionality reduction by learning an invariant mapping. In *Conference on Computer Vision and Pattern Recognition*, volume 2, pages 1735–1742. IEEE, 2006.

[28] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *International Conference on Computer Vision*, pages 2961–2969. IEEE, 2017.

[29] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick. Momentum contrast for unsupervised visual representation learning. In *Conference on Computer Vision and Pattern Recognition*, pages 9729–9738. IEEE, 2020.

[30] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick. Masked autoencoders are scalable vision learners. In *Conference on Computer Vision and Pattern Recognition*, pages 16000–16009. IEEE, 2022.

[31] G. E. Hinton and R. Zemel. Autoencoders, minimum description length and helmholtz free energy. *Advances in Neural Information Processing Systems*, 6, 1993.

[32] R. D. Hjelm, A. Fedorov, S. Lavoie-Marchildon, K. Grewal, P. Bachman, A. Trischler, and Y. Bengio. Learning deep representations by mutual information estimation and maximization. In *International Conference on Learning Representations*, 2018.

[33] G. Huang, Y. Sun, Z. Liu, D. Sedra, and K. Q. Weinberger. Deep networks with stochastic depth. In *European Conference on Computer Vision*, pages 646–661. Springer, 2016.

[34] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4):541–551, 1989.

[35] S. Li, D. Wu, F. Wu, Z. Zang, K. Wang, L. Shang, B. Sun, H. Li, S. Li, et al. Architecture-agnostic masked image modeling–from vit back to cnn. *arXiv preprint arXiv:2205.13943*, 2022.

[36] Y. Li, S. Xie, X. Chen, P. Dollar, K. He, and R. Girshick. Benchmarking detection transfer learning with vision transformers. *arXiv preprint arXiv:2111.11429*, 2021.

[37] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, pages 740–755. Springer, 2014.

[38] H. Liu, X. Jiang, X. Li, A. Guo, D. Jiang, and B. Ren. The devil is in the frequency: Geminated gestalt autoencoder for self-supervised visual pre-training. *arXiv preprint arXiv:2204.08227*, 2022.

[39] I. Loshchilov and F. Hutter. Sgdr: Stochastic gradient descent with warm restarts. *International Conference on Learning Representations*, 2017.

[40] I. Loshchilov and F. Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[41] M. Noroozi and P. Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *European Conference on Computer Vision*, pages 69–84. Springer, 2016.

[42] A. v. d. Oord, Y. Li, and O. Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[43] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[44] A. Ramesh, M. Pavlov, G. Goh, S. Gray, C. Voss, A. Radford, M. Chen, and I. Sutskever. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pages 8821–8831. PMLR, 2021.

[45] C. Tao, H. Wang, X. Zhu, J. Dong, S. Song, G. Huang, and J. Dai. Exploring the equivalence of siamese self-supervised learning via a unified gradient framework. In *Conference on Computer Vision and Pattern Recognition*, pages 14431–14440. IEEE, 2022.

[46] C. Tao, X. Zhu, G. Huang, Y. Qiao, X. Wang, and J. Dai. Siamese image modeling for self-supervised vision representation learning. *arXiv preprint arXiv:2206.01204*, 2022.

[47] Y. Tian, D. Krishnan, and P. Isola. Contrastive multiview coding. In *European Conference on Computer Vision*, pages 776–794. Springer, 2020.

[48] G. Van Horn, O. Mac Aodha, Y. Song, Y. Cui, C. Sun, A. Shepard, H. Adam, P. Perona, and S. Belongie. The inaturalist species classification and detection dataset. In *Conference on Computer Vision and Pattern Recognition*, pages 8769–8778, 2018.

[49] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, 30, 2017.

[50] C. Wei, H. Fan, S. Xie, C.-Y. Wu, A. Yuille, and C. Feichtenhofer. Masked feature prediction for self-supervised visual pre-training. In *Conference on Computer Vision and Pattern Recognition*, pages 14668–14678. IEEE, 2022.

[51] L. Wei, L. Xie, W. Zhou, H. Li, and Q. Tian. Mvp: Multimodality-guided visual pre-training. In *European Conference on Computer Vision*, pages 337–353. Springer, 2022.

[52] Y. Wei, H. Hu, Z. Xie, Z. Zhang, Y. Cao, J. Bao, D. Chen, and B. Guo. Contrastive learning rivals masked image modeling in fine-tuning via feature distillation. *arXiv preprint arXiv:2205.14141*, 2022.

[53] Z. Wu, Y. Xiong, S. X. Yu, and D. Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Conference on Computer Vision and Pattern Recognition*, pages 3733–3742. IEEE, 2018.

[54] Z. Wu, Z. Lai, X. Sun, and S. Lin. Extreme masking for learning instance and distributed visual representations. *arXiv preprint arXiv:2206.04667*, 2022.

[55] T. Xiao, Y. Liu, B. Zhou, Y. Jiang, and J. Sun. Unified perceptual parsing for scene understanding. In *European Conference on Computer Vision*, pages 418–434. Springer, 2018.

[56] Z. Xie, Z. Zhang, Y. Cao, Y. Lin, J. Bao, Z. Yao, Q. Dai, and H. Hu. Simmim: A simple framework for masked image modeling. In *Conference on Computer Vision and Pattern Recognition*, pages 9653–9663. IEEE, 2022.

[57] Z. Xie, Z. Zhang, Y. Cao, Y. Lin, Y. Wei, Q. Dai, and

H. Hu. On data scaling in masked image modeling. *arXiv preprint arXiv:2206.04664*, 2022.

[58] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson. How transferable are features in deep neural networks? *Advances in Neural Information Processing Systems*, 27, 2014.

[59] S. Yun, D. Han, S. J. Oh, S. Chun, J. Choe, and Y. Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *International Conference on Computer Vision*, pages 6023–6032. IEEE, 2019.

[60] J. Zbontar, L. Jing, I. Misra, Y. LeCun, and S. Deny. Barlow twins: Self-supervised learning via redundancy reduction. In *International Conference on Machine Learning*, pages 12310–12320. PMLR, 2021.

[61] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*, 2018.

[62] R. Zhang, P. Isola, and A. A. Efros. Colorful image colorization. In *European Conference on Computer Vision*, pages 649–666. Springer, 2016.

[63] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. Learning deep features for scene recognition using places database. *Advances in Neural Information Processing Systems*, 27, 2014.

[64] B. Zhou, H. Zhao, X. Puig, T. Xiao, S. Fidler, A. Barriuso, and A. Torralba. Semantic understanding of scenes through the ade20k dataset. *International Journal of Computer Vision*, 127(3):302–321, 2019.

[65] J. Zhou, C. Wei, H. Wang, W. Shen, C. Xie, A. Yuille, and T. Kong. ibot: Image bert pre-training with online tokenizer. *International Conference on Learning Representations*, 2022.

Qibin Hou received his Ph.D. degree from the School of Computer Science, Nankai University. Then, he worked at the National University of Singapore as a research fellow. Now, he is an associate professor at School of Computer Science, Nankai University. He has published more than 20 papers on top conferences/journals, including T-PAMI, CVPR, ICCV, NeurIPS, etc. His research interests include deep learning, image processing, and computer vision.

Ming-Ming Cheng received his PhD degree from Tsinghua University in 2012. Then he did 2 years research fellow, with Prof. Philip Torr in Oxford. He is now a professor at Nankai University, leading the Media Computing Lab. His research interests include computer graphics, computer vision, and image processing. He received research awards including ACM China Rising Star Award, IBM Global SUR Award, CCF-Intel Young Faculty Researcher Program.

Dongmei Fu received the M.S. degree from Northwestern Polytechnical University, in 1984, and the Ph.D. degree in automation science from the University of Science and Technology Beijing (USTB), China, in 2006, where she is currently a Professor and a Doctoral Supervisor. She has taken charge of several national projects about corrosion data mining and infrared image processing. Her current research interests include automation control theory, image processing, and data mining.

Xiaohui Shen Xiaohui Shen is a Research Manager at ByteDance Inc.. Before that, he was a Senior Research Scientist at Adobe Research. He obtained his PhD degree from the Department of EECS at Northwestern University, and received the MS and BS degrees from the Department of Automation at Tsinghua University, China. His research interests include computer vision and deep learning.

Zhicheng Huang received the Master degree in artificial intelligence from University of Science and Technology Beijing, China, in 2019. He is pursuing Ph. D. at the School of Automation and Electrical Engineering. His research interests include self-supervised learning, visual-language pre-training and computer vision.

Xiaojie Jin is working as a Research Scientist in Bytedance Inc. in the USA. He received his Ph.D degree from National University of Singapore in 2018. His current research interests include self-supervised learning, multi-modal understanding, intelligent video editing and efficient deep model. His research works have been published in top-tier conferences/journals, including ICCV, ECCV, CVPR, ICML, NeurIPS, ICLR, TPAMI, TNNLS, etc.

Jiashi Feng is currently a research scientist at ByteDance. Before joining ByteDance, he was an assistant professor with the Department of Electrical and Computer Engineering at National University of Singapore and a postdoc researcher in the EECS department and ICSI at the University of California, Berkeley. He received his Ph.D. degree from NUS in 2014. His research areas include deep learning and their applications in computer vision. His recent research interest focuses on deep learning models, representation learning, and 3D vision. He received the best technical demo award from ACM MM 2012, best paper award from TASK-CV ICCV 2015, best student paper award from ACM MM 2018. He is also the recipient of Innovators Under 35 Asia, MIT Technology Review 2018. He served as the area chairs for NeurIPS, ICML, CVPR, ICLR, WACV, and program chair for ICMR 2017.

Cheng-Ze Lu is currently a master student from the College of Computer Science at Nankai University, under the supervision of Prof. Ming-Ming Cheng. Before that, he received his B.E. degree from Xidian University in 2020. His research interests include deep learning and computer vision.