Architecture-Agnostic Masked Image Modeling – From ViT back to CNN

Siyuan Li [*1,2] **Di Wu** [*1,2] **Fang Wu** [1,3] **Zelin Zang** [1,2] **Stan Z. Li** [1]

Abstract

Masked image modeling, an emerging self-supervised pre-training method, has shown impressive success across numerous downstream vision tasks with Vision transformers. Its underlying idea is simple: a portion of the input image is masked out and then reconstructed via a pre-text task. However, the working principle behind MIM is not well explained, and previous studies insist that MIM primarily works for the Transformer family but is incompatible with CNNs. In this work, we observe that MIM essentially teaches the model to learn better middle-order interactions among patches for more generalized feature extraction. We then propose an Architecture-Agnostic Masked Image Modeling framework (A^{2}MIM), which is compatible with both Transformers and CNNs in a unified way. Extensive experiments on popular benchmarks show that A^{2}MIM learns better representations without explicit design and endows the backbone model with the stronger capability to transfer to various downstream tasks.

1. Introduction

Supervised deep learning with large-scale annotated data has witnessed an explosion of success in computer vision (CV) (Krizhevsky et al., 2012a; He et al., 2016) and natural language processing (NLP) (Vaswani et al., 2017). However, a large number of high-quality annotations are not always available in real-world applications. Learning representations without supervision by leveraging pre-text tasks has become increasingly popular.

In CV, early self-supervised learning approaches (Zhang et al., 2016; Doersch et al., 2015; Gidaris et al., 2018) aim to capture invariant features through predicting transformations applied to the same image. However, these methods rely on vision ad-hoc heuristics, and the learned representations are less generic. Recently, contrastive learning approaches (Tian et al., 2020; Chen et al., 2020b; He et al., 2020) have witnessed significant progress, even outperforming supervised methods on several downstream tasks (Chen et al., 2020c; Grill et al., 2020; Zbontar et al., 2021). More recently, inspired by masked autoencoding methods (Devlin et al., 2018; Radford et al., 2018) in NLP, Masked Image Modeling (MIM) methods (Bao et al., 2022; He et al., 2022; Wei et al., 2021; Xie et al., 2021b) have brought about new advances for self-supervised pre-training on CV tasks. The transition from human language understanding to NLP masked autoencoding is quite natural because the filling of missing words in a sentence requires comprehensive semantic understanding. In analogy, humans can understand and imagine masked content by visually filling the missing structures in an image containing occluded parts.

Different from contrastive learning, which yields a clustering effect by pulling similar samples and pushing away dissimilar samples, MIM pre-training methods have not been extensively explored in the context of the expected knowledge learned. Existing works mainly focus on improving downstream tasks performance via explicit design such as trying different prediction targets (Wei et al., 2021), adopting pre-trained tokenizer (Zhou et al., 2021), utilizing complex Transformer decoder (He et al., 2022) or combining with contrastive learning (El-Nouby et al., 2021). Moreover, the success of existing MIM methods is largely confined to Vision Transformer (ViT) structures (Dosovitskiy et al., 2021) since it leads to under-performing performance to directly apply mask token (Devlin et al., 2018) and positional embedding to CNNs.

In this work, we carry out systematic experiments and show that MIM as a pre-training task essentially teaches the model to learn better middle-order interactions between patches for more generalized feature extraction regardless of the underlying network structure. Compared to the local texture features learned by low-order interactions between patches, more complex features such as shape and edge could be extracted via middle-order interactions among patches. The interaction of patches could be considered as information

[*]Equal contribution [1]AI Lab, Research Center for Industries of the Future, Westlake University, Hangzhou, 310000, China [2]College of Computer Science and Technology, Zhejiang University, Hangzhou, 310000, China [3]Institute of AI Industry Research, Tsinghua University, Beijing, 100084, China. Correspondence to: Stan Z. Li <stan.z.li@westlake.edu.cn>.

Proceedings of the 40th International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

fusion via both the convolution operation of a CNN and the self-attention mechanism of a Transformer. That is to say, CNN and Transformer should both bene t from better middle-order interactions with MIM as the pre-text task.

To bridge the gap of MIM in terms of network architectures based on our extensive experimental analysis, we propose an Architecture-Agnostic Masked Image Modeling framework (A^2MIM) that focuses on enhancing the middle-order interaction capabilities of the network. Speci cally, we mask the input image with the mean RGB value and place the mask token at intermediate feature maps of the network. In addition, we propose a loss in the Fourier domain to further enhance the middle-order interaction capability of the network. Our contributions are summarized as follows:

- We conducted systematic experiments and showed the essence of MIM is to better learn middle-order interactions between patches but not reconstruction quality.
- We proposed a novel MIM-based framework dubbed A^2MIM that bridges the gap between CNNs and Transformers. We are also the rst to perform MIM on CNNs without adopting designs native to ViTs that outperform contrastive learning counterparts.
- Extensive experiments with both Transformers and CNNs on ImageNet-1K and public benchmarks for various downstream tasks show that our method improves performances on pre-trained representations.

2. Related Work

Contrastive Learning. Contrastive learning (CL) learns instance-level discriminative representations by extracting invariant features over distorted views of the same data. MoCo (He et al., 2020) and SimCLR (Chen et al., 2020b) adopted different mechanisms to introduce numerous negative samples for contrast with the positive. BYOL (Grill et al., 2020) and its variants (Chen & He, 2020; Ge et al., 2021) further eliminate the requirement of negative samples to avoid representation collapse. Besides pairwise contrasting, SwAV (Caron et al., 2020) clusters the data while enforcing consistency between multi-augmented views of the same image. Barlow Twins (Zbontar et al., 2021) and its variants (Ermolov et al., 2021; Bardes et al., 2022) proposed to measure the cross-correlation matrix of distorted views of the same image to avoid representation collapsing. Meanwhile, some efforts have been made on top of contrastive methods to improve pre-training quality for speci c downstream tasks (Xie et al., 2021a; Xiao et al., 2021; Selvaraju et al., 2021). MoCo.V3 (Chen et al., 2021) and DINO (Caron et al., 2021) adopted ViT (Dosovitskiy et al., 2021) in CL pre-training to replace CNN backbones.

Autoregressive Modeling. Autoencoders (AE) is a typical type of architecture that allows representation learning with no annotation requirement (Hinton & Zemel, 1993). By forcing denoising property onto the learned representations, denoising autoencoders (Vincent et al., 2008; 2010) are a family of AEs that reconstruct the uncorrected input signal with a corrupted version of the signal as input. Generalizing the notion of denoising autoregressive modeling, masked predictions attracted the attention of both the NLP and CV communities. BERT (Devlin et al., 2018) performs masked language modeling (MLM) where the task is to classify the randomly masked input tokens. Representations learned by BERT as pre-training generalize well to various downstream tasks. For CV, inpainting tasks (Pathak et al., 2016) to predict large missing regions using CNN encoders and colorization tasks (Zhang et al., 2016) to reconstruct the original color of images with removed color channels are proposed to learn representation without supervision. With the introduction of Vision Transformers (ViTs) (Dosovitskiy et al., 2021; Liu et al., 2021), iGPT (Chen et al., 2020a) predicts succeeding pixels given a sequence of pixels as input. MAE (He et al., 2022) and BEiT (Bao et al., 2022) randomly mask out input image patches and reconstruct the missing patches with ViTs. Compared to MAE, MaskFeat (Wei et al., 2021) and SimMIM (Xie et al., 2021b) adopt linear layers as the decoder instead of another Transformer as in MAE. MaskFeat applied HOG as the prediction target instead of the RGB value. Other research endeavors (El-Nouby et al., 2021; Zhou et al., 2021; Assran et al., 2022; Akbari et al., 2021; Sameni et al., 2022) combine the idea of CL with MIM. Moreover, Data2Vec (Baevski et al., 2022) proposed a framework that applies the masked prediction idea for either speech, NLP, or CV. However, most MIM works are con ned to ViTs, recently proposed CIM (Fang et al., 2022) uses the output of a pre-trained tokenizer as the target and takes the output of a frozen BEiT as the encoder's input as a workaround to enable MIM on CNNs, and the concurrent work SparK (Tian et al., 2023) employs the sparse convolution operators to tackle the irregular masked input for CNNs.

3. Midst of Masked Image Modeling

3.1. Is MIM Better Image Augmentation?

Compared to CNN, Transformers gain tremendous performance improvement with carefully designed image augmentation techniques(Cubuk et al., 2020; Yun et al., 2019; Zhong et al., 2020). For instance, Random erasing and Cutmix randomly remove part of the image and replace the corresponding region with either Gaussian noise or a patch from another image. Similarly, as in most MIM pre-training tasks, some image patches are masked out and replaced with a learnable mask token. Noticing the resemblance of the masking operations, we hypothesize that MIM as a pre-training task and masking-based data augmentations en-

(a) (b) (c) (d)

Figure 1.(a)(b): Robustness against different occlusion ratios of images is studied for both ViT-S and ResNet-50 under different experimental settings (see Section 3.1). (c)(d): Distributions of the interaction strength $J^{(m)}$ are explored for both ViT-S and ResNet-50 under different experimental settings. The label indicates the pre-training method + fine-tuning augmentation used, random stands for random weight initialization. Appendix B provides more results and implement details.

hance the network's robustness towards occlusion, enabling the network with a more generalized feature extraction ability. To verify our hypothesis, we design an occlusion robustness test. Let $x \in \mathbb{R}^{3\times H\times W}$ be an input image and $y \in \mathbb{R}^C$ be its corresponding label, where C is the class number. Considering a classification task $y = f(x)$ where f denotes a neural network, the network is considered robust if the network outputs the correct label given an occluded version of the image x', namely $y = f(x')$. For occlusion, we consider the patch-based random masking as adopted in most MIM works (He et al., 2022; Xie et al., 2021b; Wei et al., 2021). In particular, we split the image of size 224×224 into patch size 16×16 and randomly mask M patches out of the total number of N patches. The occlusion ratio could then be defined as $\frac{M}{N}$. We conduct experiments on ImageNet-100 (IN-100) (Krizhevsky et al., 2012b) for both Transformer and CNN with different settings. We choose ViT-S (Dosovitskiy et al., 2021) and ResNet-50(He et al., 2016) as the network architecture. Robustness is compared under the following settings: (i) random weight initialization with no image augmentation applied; (ii) random weight initialization with different image augmentations applied; (iii) MIM pre-training as weight initialization with and without image augmentations applied. In Fig. 1, we report the average top-1 accuracy across five runs trained with different settings under various occlusion ratios. Fig. 1(a) and 1(b) show that both MIM and patch-removing alike augmentations significantly improve model occlusion robustness for both ViT-S and ResNet-50. Nevertheless, MIM yields more robust feature extraction than adopting augmentations. Although MIM and patch-removing alike augmentations share similar masking mechanisms, MIM explicitly forces the model to learn the interactions between patches in order to reconstruct missing patches enabling more robust feature extraction. Comparing Fig. 1(a) and 1(b), the convex trend of accuracy from ViT-S indicates better robustness than the concave trend from ResNet-50. This can be attributed to the higher degrees of freedom of the self-attention mechanism compared to convolution priors. We claim that the success of MIM on ViTs can be seen as resonance in terms of better patch interactions imposed by MIM while supported by the self-attention mechanism of ViTs.

3.2. Middle-order Interactions for Generalized Feature Extraction

Next, we show that MIM essentially enables better middle-order interactions between patches. Note that existing MIM works adopt a medium or high masking ratio (Xie et al., 2021b; He et al., 2022) (e.g., 60% or 70%, see Fig. 2) during pre-training, and in these settings, the pairwise interactions between patches are under a middle-size context measured by the order m. Early inpainting work based on CNN (Pathak et al., 2016) resembles MIM but attracts little attention due to limited performance. The inpainting task adopts the masking strategy as illustrated in Fig. 1(c), which masks a full large region instead of random small patches. Such masking mechanisms ignore patch interaction and focus only on reconstruction leading to poor representation quality. To investigate whether MIM makes the model more sensitive to patch interactions of some particular orders, we resort to the tool of multi-order interactions introduced by (Deng et al., 2022; Zhang et al., 2020). Intuitively, m^{th}-order interactions of patches refer to inference patterns (deep features) induced from m number of patches of the original image in the input space. With a small value of m (low-order interactions), the model simply learns local features such as texture. Formally, the multi-order interaction $I^{(m)}(i;j)$ is to measure the order of interactions between patches i and j. We define $I^{(m)}(i;j)$ to be the average interaction utility between patches i and j on all contexts consisting of m patches, where m denotes the order of contextual complexity of the interaction. Mathematically, given an input image x with a set of n patches $N = \{1,\dots,n\}$ (e.g., n pixels), the multi-order interaction $I^{(m)}(i;j)$ is defined as:

$$I^{(m)}(i;j) = \mathbb{E}_{S\subseteq N\setminus\{i,j\},|S|=m}[\Delta f(i;j;S)], \qquad (1)$$

where $\Delta f(i;j;S) = f(S\cup\{i;j\}) - f(S\cup\{i\}) - f(S\cup\{j\}) + f(S)$. $f(S)$ indicates the score of output with patches in $N\setminus S$ kept unchanged but replaced with the baseline

Figure 2. (a) Four patches (i, j, k, l) interact with each other and forms a contour or edge pattern of the fox for image categorization. (b) Image with 30% masking ratio. Masked patches i and k interact with neighboring patches j and l to predict the missing patches. (c) Image with 50% masking ratio. Masked patches force the model to extract information from unmasked patches and learn middle-order interactions for the MIM task. (d) Image with 70% masking ratio. Masked Patch i interacts with longer-range patches j and k, forming an edge pattern. (e) A typical masking pattern for existing inpainting tasks.

value (Ancona et al., 2019), where the context $S \subseteq N$. See Appendix B.2 for details. To measure the interaction complexity of the neural network, we measured the relative interaction strength $J^{(m)}$ of the encoded m-th order interaction as:

$$J^{(m)} = \frac{\mathbb{E}_{x \in \Omega} \mathbb{E}_{i,j} |I^{(m)}(i,j|x)|}{\mathbb{E}_{m'} \mathbb{E}_{x \in \Omega} \mathbb{E}_{i,j} |I^{(m')}(i,j|x)|}, \quad (2)$$

where Ω is the set of all samples and $0 \le m \le n-2$. $J^{(m)}$ is the average value over all possible pairs of patches of input samples. $J^{(m)}$ is normalized by the average value of all interaction strengths. $J^{(m)}$ then indicates the distribution (area under curve sums up to one) of the order of interactions of the network. We use $J^{(m)}$ as the metric to evaluate and analyze interaction orders of the network with MIM pre-training. We conduct experiments on IN-100 with image size 224×224 and use ViT-S (Dosovitskiy et al., 2021) and ResNet-50 (He et al., 2016) as the network architecture. We consider a patch of size 16×16 as input. For the computation of $J^{(m)}$, we adopt the sampling solution following previous works (Deng et al., 2022; Zhang et al., 2020). As can be seen from Fig. 1(c), ViT-S with random weight initialization tends to learn simple interactions with few patches (e.g., less than $0.05n$ patches) while MIM pre-trained models show a stronger interaction for relative middle-order (from $0.05n$ to $0.5n$). Similarly, as observed from 1(d), MIM pre-trained ResNet-50 enhances the middle-order interactions from $0.1n$ to $0.55n$ compared to random initialized models. Stronger middle-order interactions form more complex features such as shape and edge compared to local texture features learned from low-order interactions (Naseer et al., 2021).

4. Approach

We propose a generic MIM framework following two design rules: (a) Better middle-order interactions between patches for more generalized feature extraction. (b) No complex or non-generic designs are adopted to ensure compatibility with all network architectures. Figure 3 highlights the difference between A^{2}MIM and existing MIM frameworks in terms of three key components: masking strategy, encoder/decoder network architecture design and prediction targets.

4.1. Architecture Agnostic Framework

Mask Where Middle-order Interactions Occur. Existing works (El-Nouby et al., 2021; He et al., 2022; Wei et al., 2021) adopt the masking strategy where the input image is divided into non-overlapping patches, and a random subset of patches is masked. MAE utilizes a Transformer as a decoder and takes only the visible patches into the encoder. Masked tokens are appended to the decoder to reconstruct the masked patches. SimMIM and MaskFeat (Wei et al., 2021) utilize a fully connected layer as the decoder and feed the mask token into the encoder together with the visible patches. The mask token (Devlin et al., 2018) is a token-shared learnable parameter that indicates the presence of missing patches to be predicted. Despite different choices of decoder structures, the mask token is either placed at the input to the encoder or the decoder. Mathematically, the masking process of MIM is defined as $x_{mask} = x \odot (1 - M) + T \odot M$, where M is the random occlusion mask, and T represents the learnable mask token. Such masking at the patch embedding layer aligns with the attention mechanism of Transformers, which is robust against occlusion. However, masking at the stem layer undermines the context extraction capability of CNN, which relies on local inductive biases. Moreover, masking at input stages of the network leads to low-order interactions. Thus, we propose to mask intermediate features where the output feature contains both semantic and spatial information, and the mask token can encode interactions with a medium number of tokens (e.g., the last embedded stage). Concretely, our masking operation is defined as $z^l_{mask} = z^l + T \odot D(M)$, where z^l is the intermediate feature map at stage-l in CNN encoders (or layer-l in Transformers) and $D(\cdot)$ is the corresponding down-sampling function of the occlusion mask.

Filling Masked Tokens with RGB Mean. Existing works directly replace the occluded patches with the mask token

Figure 3. An illustration comparison between the existing MIM framework and our proposed framework. For the existing MIM framework, the input image is patched into a sequence of patches without overlapping with masked patches that are replaced with learnable mask tokens. The sequence is then input to the Transformer encoder. The $\mathcal{L}_{spa}$ is applied between the ground truth patches and the reconstructed patches from the decoder in the spatiotemporal domain. Our proposed framework uses the mean RGB value of the image instead of the mask token in the input space. We then add a learnable mask token onto the intermediate feature map l of stages of the encoder instead of replacement in the input space. The encoder could either be of the Transformer or the CNN family. In addition to the $\mathcal{L}_{spa}$, we adopt a $\mathcal{L}_{freq}$ in the Fourier domain to enhance the encoder to learn more middle-order interactions. Specifically, we apply DFT on both the ground truth image and the predicted image and then use Mean square error (MSE) to measure the difference.

in the input space or after the patch embedding (Bao et al., 2022; Xie et al., 2021b). In contrast, we use the average RGB value to fill the occluded patches as the input to the encoder and add the mask token onto the intermediate feature maps of the encoder. The masking mechanism originates from NLP where languages are of high-level semantics and do not require low-level feature extraction as image processing. Masking at the early stages of the network where low-level feature extraction happens is harmful in terms of feature extraction. The RGB mean is the DC component of images. Filling with RGB mean alleviates local statistics distortion caused by the masking operation and forces the network to model more informative medium frequencies instead of filling the masked patches with blurry color blocks (low frequencies). The proposed masking strategy is generic to both convolution and self-attention in that it accommodates low-level to semantic-level feature extraction.

4.2. Middle-order Interactions from Fourier Perspective

Current works (El-Nouby et al., 2021; He et al., 2022; Xie et al., 2021b) adopt raw RGB values as the prediction target. However, raw pixels in the spatial domain are heavily redundant and often contain low-order statistics (Bao et al., 2022; Wei et al., 2021; Zhou et al., 2021). MaskFeat (Wei et al., 2021) adopts the Histogram of Oriented Gradients (HOG) as the prediction target outperforming MAE and SimMIM. HOG is a discrete descriptor of medium or high-frequency features that captures shape patterns based on middle-order interactions. ViTs and CNNs have low-pass and high-pass filtering properties, respectively (Park & Kim, 2022; 2021). ViTs and CNNs have certain frequency bands that they each cannot model well, and both cannot model middle-order interactions well (detailed in Appendix B.3). The observation of the medium frequency descriptor HOG improves middle-order interactions and leads to the hypothesis that learning medium frequencies would help the model learn more middle-order interactions. Given a RGB image $x \in \mathbb{R}^{3\times H\times W}$, the discrete Fourier transform (DFT) of each channel is defined as:

$$F_{(u,v)} = \sum_{h=1}^{h=H}\sum_{w=1}^{w=W} x(h,w)\, e^{-2\pi j\left(\frac{uh}{H}+\frac{vw}{W}\right)}. \tag{3}$$

In addition to the common MIM loss in the spatial domain $\mathcal{L}_{spa}$, we propose $\mathcal{L}_{freq}$ in Fourier domain:

$$\mathcal{L}_{freq} = \sum_{c=1}^{c=3}\sum_{u=1}^{u=H}\sum_{w=1}^{w=W} \omega(u,v)\,\Big|\mathrm{DFT}\big(x_c^{pred}\odot M + de(x_c^{pred})\odot(1-M)\big) - \mathrm{DFT}(x_c)\Big|, \tag{4}$$

where x^{pred} is the predicted image, $de(\cdot)$ is detach gradient operation, and $\omega(u,v)$ is a dynamic frequency weighting matrix. Inspired by Focal Frequency loss (Jiang et al., 2021), we define adaptive $\omega(u,v)$ as follows:

$$\omega(u,v) = \Big|\mathrm{DFT}\big(x_c^{pred}\odot M + det(x_c^{pred})\odot(1-M)\big) - \mathrm{DFT}(x_c)\Big|, \tag{5}$$

$\omega(u,v)$ enables both ViTs and CNNs to model features of medium frequencies rather than local textures and noise corresponding to high frequencies. Since filling masked tokens with RGB mean is filling with DC components, combining our proposed masking strategy with the weighting effect of the $\mathcal{L}_{freq}$ leads to the better modeling of medium frequency features (middle-order interactions). Fig. B.3 verifies that Eq. (5) allows the model to learn previously ignored frequencies (mostly the medium frequencies). Note that $\mathcal{L}_{freq}$ introduces negligible overhead by using Fast Fourier Transform (FFT) algorithms with $O(n \log n)$ complexity to achieve DFT. The overall loss of A^{2}MIM is defined as:

$$\mathcal{L} = \mathcal{L}_{spa} + \lambda \mathcal{L}_{freq}, \tag{6}$$

where $\mathcal{L}_{spa} = \| x^{pred} - x \| \odot M$ and λ is a loss weighting hyper-parameter. We set λ to 0.1 by default.

5. Experiments

5.1. Pre-training Setup

We adopt Vision Transformer (Dosovitskiy et al., 2021) (ViT/16), ResNet (He et al., 2016), and ConvNeXt (Liu et al., 2022b) as the backbone encoder. Models are pre-trained on ImageNet-1K (IN-1K) training set with AdamW (Loshchilov & Hutter, 2019) optimizer, a batch size of 2048, and a basic learning rate of 1.2×10^{-3} adjusted by a cosine learning rate scheduler. The input image size is 224×224 with a masked patch size of 32×32, and the random masking ratio is 60%. By default, the learnable mask tokens are placed at stage-3 and layer-0 in ResNet/ConvNeXt and ViT architectures, respectively. We adopt a linear prediction head as the MIM decoder (Xie et al., 2021b). A^{2}MIM+ indicates adopting HOG as the MIM target and using the MLP decoder with depth-wise (DW) convolutions. Our experiments are implemented on OpenMixup (Li et al., 2022) by Pytorch and conducted on workstations with NVIDIA A100 GPUs. **Bold** and underline indicate the best and the second-best performance, and gray denotes the uncomparable results (e.g., not in the same technical scope). See Appendix A for pre-training details.

5.2. Image Classification on ImageNet-1K

Evaluation Protocols. We evaluate the learned representation by end-to-end fine-tuning (FT) and linear probing (Lin.) protocols on IN-1K. For FT evaluations of ViTs, we employ the fine-tuning as MAE (He et al., 2022), which applies DeiT (Touvron et al., 2021) augmentations, AdamW optimizer with a batch size of 1024 for 200 epochs, and adopt a layer-wise learning rate decay of 0.65 as BEiT (Bao et al., 2022). For FT evaluations of CNNs, ResNet variants are fine-tuned with RSB A2/A3 (Wightman et al., 2021) training settings, which employ LAMB (You et al., 2020) optimizer with a batch size 2048 for 300/100 epochs, and ConvNeXt models are fine-tuned 300-epoch with its original supervised learning settings. For the linear evaluations, ResNet-50 settings follow MoCo (He et al., 2020), which trains a linear classifier by SGD with a batch size of 256, and ViTs follow MAE, which tunes the linear layer with BN by AdamW. See Appendix A for detailed configurations.

Table 1. ImageNet-1K fine-tuning (FT) top-1 accuracy (%) of ViT-S and ViT-B models. [z] denotes our finetuned results.

Method	Date	Target	PT Epochs	ViT-S FT	ViT-B FT	ViT-L FT
Rand init.	-	Label	300	79.9	81.8	82.6
SimCLR	ICML'2020	CL	300	80.2	82.3	-
BYOL	NIPS'2020	CL	300	80.9	82.8	-
MoCoV3	ICCV'2021	CL	300	81.4	83.2	84.1
DINO	ICCV'2021	CL	300	81.5	83.6	-
BEiT	ICLR'2022	DALLE	800	81.3	83.2	85.2
SplitMask	arXiv'2022	DALLE	300	81.5	83.6	-
iBOT	ICLR'2022	EMA	800	82.3	84.0	85.2
MAE	CVPR'2022	RGB	1600	81.6	83.6	85.9
MaskFeat	CVPR'2022	HOG	800	-	84.0	85.7
Data2Vec	ICML'2022	EMA	800	-	84.2	86.2
SimMIM	CVPR'2022	RGB	800	81.7	83.8	85.6
CAE	arXiv'2022	DALLE	1600	81.8	83.6	86.3
mc-BEiT	ECCV'2022	VQGAN	800	-	84.1	85.6
BootMAE	ECCV'2022	EMA	800	-	84.2	85.9
PeCo	AAAI'2023	VQVAE	800	-	84.5	86.5
CIM	ICLR'2023	BEiT	300	81.6	83.3	-
MC-MAE	ICLR'2023	EMA	1600	82.0	83.6	86.1
MAGE-C	CVPR'2023	VQGAN	1600	-	82.9	84.3
LocalMIM	CVPR'2023	HOG	1600	-	84.0	85.8
A^{2}MIM	Ours	RGB	800	82.1	84.2	86.1
A^{2}MIM+	Ours	HOG	800	82.3	84.4	86.3

Table 2. ImageNet-1K linear probing (Lin.) and fine-tuning (FT) top-1 accuracy (%) of ResNet-50. [y] Multi-crop augmentation. [z] Our modified MIM methods for CNN.

Method	Fast Pre-training Epochs	Fast Pre-training Lin.	Fast Pre-training FT (A3)	Longer Pre-training Epochs	Longer Pre-training FT (A3)	Longer Pre-training FT (A2)
Rand init.	-	4.4	78.1	-	78.1	79.8
PyTorch (Sup.)	90	76.2	78.8	300	78.9	79.9
Inpainting	70	40.1	78.4	300	78.0	-
Relative-Loc	70	38.8	77.8	300	77.9	-
Rotation	70	48.1	77.7	300	78.2	-
SimCLR	100	64.4	78.5	800	78.8	79.9
MoCoV2	100	66.8	78.5	800	78.8	79.8
BYOL	100	68.4	78.7	400	78.9	80.1
SwAV[y]	100	71.9	78.9	400	79.0	80.2
Barlow Twins	100	67.2	78.5	300	78.8	79.9
MoCoV3	100	68.9	78.7	300	79.0	80.1
BEiT[z]	100	47.1	78.1	-	-	-
Data2Vec[z]	100	43.2	78.0	-	-	-
MAE[z]	100	37.8	77.1	300	77.2	79.0
SimMIM[z]	100	47.5	78.2	300	78.3	79.9
CIM	-	-	-	300	78.6	80.4
A^{2}MIM	100	48.1	78.8	300	78.9	80.4
A^{2}MIM+	100	50.3	78.9	300	79.0	80.5

ViTs. We first evaluate A^{2}MIM variants with ViT-S/B/L on IN-1K. We list the supervision target used by various pre-training algorithms in the third column of Tab. 1. VQVAE/DALL-E (Ramesh et al., 2021) and VQGAN (Esser

Table 3. ImageNet-1K fine-tuning (FT) top-1 accuracy (%) with ResNet and ConvNeXt of various model scales. We adopt the 300-epoch fine-tuning protocols for both architectures. ^{z} denotes our reproduced results.

Methods	#Para.	Sup.	MoCoV3^{z}	SimMIM^{z}	SparK	A^{2}MIM
Target	(M)	Label	CL	RGB	RGB	RGB
ResNet-50	25.6	79.8	80.1	79.9	80.6	80.4
ResNet-101	44.5	81.3	81.6	81.3	82.2	81.9
ResNet-152	60.2	81.8	82.0	81.9	82.7	82.5
ResNet-200	64.7	82.1	82.5	82.2	83.1	83.0
ConvNeXt-T	28.6	82.1	82.3	82.1	82.7	82.5
ConvNeXt-S	50.2	83.1	83.3	83.2	84.1	83.7
ConvNeXt-B	88.6	83.5	83.7	83.6	84.8	84.1

et al., 2021) are pre-trained image tokenizers, while EMA refers to the momentum encoder. Our A^{2}MIM outperforms CL and MIM baselines, and A^{2}MIM+ achieves competitive results as current state-of-the-art methods with complex supervision, e.g., SplitMask (MIM with CL combined), iBOT (complex teacher-student architecture), and CIM (pre-trained BEiT as supervision). Based on ViT-S/B/L, A^{2}MIM significantly improves the baseline SimMIM by 0.5%/0.4%/0.5% with the RGB target and 0.7%/0.7%/0.6% with the HOG feature as supervision.

CNNs. We then compare A^{2}MIM with classical self-supervised learning methods (Inpainting (Pathak et al., 2016), Relative-Loc (Doersch et al., 2015), and Rotation (Gidaris et al., 2018)), CL, and MIM methods with 100/300 pre-training epochs. We modified MIM methods to run them on ResNet-50: the learnable mask token is employed to the encoder for BEiT (Bao et al., 2022), Data2Vec (Baevski et al., 2022), and SimMIM (Xie et al., 2021b) after the stem (the output feature of 56×56 resolutions); the encoder of MAE randomly selects 25% from 56×56 output features of the stem as unmasked patches and takes the reorganized 28×28 patches as the input of four stages. In Tab. 2, our approach achieves competitive performance with state-of-the-art contrastive-based methods under 100-epoch FT evaluation. Note that MIM methods see fewer training samples per epoch than CL methods (e.g., 40% vs. 200% of patches) and usually require longer pre-training epochs. Based on a longer FT evaluation, A^{2}MIM (300-epoch) outperforms contrastive-based methods with even fewer training epochs. Meanwhile, A^{2}MIM also improves the baseline SimMIM^{z} (+0.8%) and the concurrent work CIM (+0.4%) in terms of 100-epoch FT for the longer pre-training. Besides, we also report the linear probing (Lin.) results of the fast pre-training for reference, although we focus on learning representations with better fine-tuning performances. Although A^{2}MIM achieves lower Lin. results than popular CL methods, A^{2}MIM still improves the baseline by 0.6%. Moreover, we further conduct scaling-up experiments of A^{2}MIM and pre-training methods based on ResNet and ConvNeXt models. Notice that two concurrent works proposed after our A^{2}MIM (SparK (Tian et al., 2023) and ConvNeXtV2 (Woo et al., 2023)) are specially designed MIM approaches for CNNs, which employ the sparse convolution to handle the irregular masked input. As shown in Table 3, we compare A^{2}MIM with DeiT (as the supervised baseline), MoCoV3, SimMIM, and SparK, where A^{2}MIM noticeably surpasses the two popular self-supervised methods (MoCoV3 and SimMIM). Despite the proposed A^{2}MIM yields inferior performances than SparK, A^{2}MIM can also work for Transformer architectures.

Table 4. Performance of object detection and semantic segmentation tasks based on ViT-B on COCO and ADE-20K.

Method	Target	Epochs	IN-1K	COCO		ADE-20K
		PT	FT	AP^{box}	AP^{mask}	mIoU
DeiT (Sup.)	Label	300	81.8	47.9	42.9	47.0
MoCoV3	CL	300	83.2	47.9	42.7	47.3
DINO	CL	400	83.6	46.8	41.5	47.2
BEiT	DALLE	300	83.2	43.1	38.2	47.1
iBOT	EMA	400	84.0	48.4	42.7	48.0
PeCo	VQ-VAE	300	84.5	43.9	39.8	46.7
MAE	RGB	1600	83.6	48.5	42.8	48.1
MaskFeat	HOG	800	84.0	<u>49.2</u>	43.2	<u>48.8</u>
SimMIM	RGB	800	83.8	48.9	43.0	48.4
CAE	DALLE	800	83.6	<u>49.2</u>	<u>43.3</u>	<u>48.8</u>
A^{2}MIM	RGB	800	<u>84.2</u>	49.4	43.5	49.0

5.3. Transfer Learning Experiments

Object detection and segmentation on COCO. To verify the transferring abilities, we benchmark CL and MIM methods on object detection and segmentation with COCO (Lin et al., 2014). For evaluation on CNN, we follow the setup in MoCo, which fine-tunes Mask R-CNN (He et al., 2017) with ResNet-50-C4 backbone using 2× schedule on the COCO train2017 and evaluates on the COCO val2017. Results in Tab. 5 indicate that A^{2}MIM (300-epoch) outperforms contrastive-based methods with longer pre-training (+0.7% AP^{box} and +0.6% AP^{mask}). For evaluation on Transformer, we follow MAE and CAE, which efficiently fine-tunes Mask R-CNN with ViT-B backbone using 1× schedule. In Tab. 4, A^{2}MIM (800-epoch) is superior to popular contrastive-based and MIM methods, e.g., outperforms MAE (1600-epoch) by 0.9% AP^{box} and 0.8% AP^{mask}.

Table 5. Performance of object detection and semantic segmentation tasks based on ResNet-50 on COCO and ADE20K.

Method	Target	Epochs	IN-1K	COCO		ADE-20K
		PT	FT	AP^{box}	AP^{mask}	mIoU
Sup.	Label	90	79.8	38.2	33.3	36.1
SimCLR	CL	800	79.9	37.9	33.3	37.6
MoCoV2	CL	800	79.8	<u>39.2</u>	34.3	37.5
BYOL	CL	400	80.1	38.9	34.2	37.2
SwAV	CL	800	<u>80.2</u>	38.4	33.8	37.3
SimSiam	CL	400	80.0	<u>39.2</u>	<u>34.4</u>	37.2
Balow Twins	CL	800	79.9	<u>39.2</u>	34.3	37.3
SimMIM^{z}	RGB	300	79.9	39.1	34.2	37.4
CIM	BEiT	300	80.4	-	-	<u>38.0</u>
A^{2}MIM	RGB	300	80.4	39.8	34.9	38.3

Figure 4. Visualizations of predicted results from SimMIM (middle) and our A^{2}MIM (right) based on ViT-S pre-trained 400-epochs on IN-1K. Notice that $\mathcal{T}(\cdot)$ denotes the mask token T to the optimal layer-5 in ViT-S. We ablate the proposed components by adding them to the baseline. Compared to results from SimMIM, reconstruction results of the modified baseline ($\mathcal{L}_{spa}$) with the RGB mean mask relieves grid-like artifacts; adding the mask token $\mathcal{T}(\cdot)$ further improves the smoothness; using the proposed $\mathcal{L}_{freq}$ helps the model to capture more informative details and contours.

Semantic segmentation on ADE20K. We then evaluate the transferring performances on semantic segmentation with ADE20K (Zhou et al., 2019) by fine-tuning FCN (Shelhamer et al., 2017) and UperNet (Xiao et al., 2018). Based on ResNet-50, all models are fine-tuned for 160K iterations with SGD following MoCo and CIM. Results in Tab. 5 show that our method outperforms CL methods by at least 0.9% mIoU and outperforms CIM (required extra pre-trained BEiT (Bao et al., 2022)) by 0.3% mIoU. Based on ViT-B, we fine-tune models for 160K iterations with AdamW following MAE and CAE. Tab. 4 shows that our approach consistently improves MIM methods (*e.g.*, outperforms MAE and SimMIM by 0.9% and 0.6% mIoU).

Table 6. Ablation of A^{2}MIM on IN-100 and IN-1K. w/o ω denotes removing the re-weighting term ω in $\mathcal{L}_{freq}$ and $\mathcal{T}(\cdot)$ denotes adding the mask token T to the optimal layer-5.

Backbones	ResNet-50		ViT-S	ViT-B
Datasets	IN-100	IN-1K	IN-100	IN-1K
Pre-training Epochs	400 ep	100 ep	400 ep	400 ep
SimMIM	87.75	78.2	85.10	83.1
$\mathcal{L}_{spa}$	88.19	78.4	85.27	83.2
$+\mathcal{L}_{freq}$ w/o ω	88.47	78.4	86.05	83.3
$+\mathcal{L}_{freq}$	88.73	78.6	86.41	83.4
$+\mathcal{L}_{freq}+\mathcal{T}(\cdot)$	88.86	78.8	86.62	83.5

5.4. Ablation Study

We next verify the effectiveness of the proposed components. Ablation studies are conducted with ResNet-50 and ViTs on IN-100 and IN-1K using the fine-tuning protocol. Based on the modified baseline SimMIM ($\mathcal{L}_{spa}$), we first compare different mask token mechanisms: *Replacing* denotes the original way in most MIM methods, and *Addition* denotes our proposed way that adds the mask token to intermediate feature maps of the backbone. Replacing masked patches in input images by RGB mean value slightly improves the baseline SimMIM, especially for ResNet-50 (88.19 *vs.* 87.75 on IN-100). Then, we verify the proposed $\mathcal{L}_{freq}$ in Tab. 6. We find that simply using $\mathcal{L}_{freq}$ without the adaptive re-weighting ω (Eqn. 5) brings limited improvements as the frequency constraint to $\mathcal{L}_{spa}$, while employing ω further enhances performances by helping the model to learn more informative frequency components. Additionally, we visualize reconstruction results in Fig. 4 to show the improvements brought by our proposed components. Refer to Appendix C and D for more ablations and visualization results.

5.5. Verification of A^{2}MIM Design Rules

We verify whether A^{2}MIM meets the intended design rules using the same experiment settings as Sec. 5.4 from two aspects. (i) A^{2}MIM is generic to incorporate advanced components proposed in previous works (*e.g.*, complex decoders, advanced prediction targets). As for the decoder structure, we replace the original linear decoder with 2-layer MLP or Transformer decoders, but find limited improvements or degenerated performances (similar to SimMIM) in Tab. 7. Inspired by PVT.V2 (Wang et al., 2022), we introduce a depth-wise (DW) convolution layer (w/ DW) to the MLP decoder (adding a 5×5 DW layer in between) and the Transformer decoder (adding a 3×3 DW layer in each FFN (Wang et al., 2022)), which brings improvements compared to the linear decoder. As for the prediction target, we follow MaskFeat to change the RGB target to the HoG feature or the output feature from ViT-B/16 pre-trained by DINO (Caron et al., 2021). Tab. 7 shows that using advanced targets significantly improves the performance of A^{2}MIM for both ResNet-50 and ViT-B. Therefore, we can conclude A^{2}MIM is a generally applicable framework. (ii) A^{2}MIM enhances occlusion robustness and middle-order interaction among patches from experiments on IN-1K in Fig. 5. We analyze occlusion robustness and interac-

(a) (b) (c) (d)

Figure 5. Robustness and interaction of A^2MIM with ViT-S and ResNet-50 on ImageNet-1K. (a)(b): Robustness against different occlusion ratios of images is studied for A^2MIM and various methods. (c)(d): Distributions of the interaction strength $J^{(m)}$ are explored.

(a) (b) (c)

Figure 6. Analysis of A^2MIM pre-training epochs and fine-tune performances with ResNet, ConvNeXt, and ViT models on ImageNet-1K. (a)(b) show CNN architectures obtain less performance gains and benefit less from longer pre-training with A^2MIM than ViTs in (c).

tion strength of A^2MIM with ViT-S (pre-training 400-epoch) and ResNet-50 (pre-training 100-epoch) on ImageNet-1K, as shown in Fig. 5. Fig. 5(a) and 5(b) shows that A^2MIM is more robust to occlusion than the baseline SimMIM and contrastive learning methods with both Transformers and CNNs. Meanwhile, we find that MIM methods learn more balanced interaction strength than both supervised and contrastive learning methods in Fig. 5(c) and 5(d). A^2MIM further improves SimMIM by capturing more middle-order interactions $(0.2n$ to $0.6n)$ with Transformers and CNNs. Therefore, we can conclude that A^2MIM helps the model to learn better middle-order interactions between patches for more generalized visual representation.

Table 7. Analysis of the scalability for advanced components.

	Module	ResNet-50	ViT-B
Decoder	Linear	78.8	82.4
	2-layer MLP	78.8	82.4
	2-layer MLP (w/ DW)	78.9	82.5
	2-layer Transformer	78.6	82.3
	2-layer Transformer (w/ DW)	78.8	82.6
Target	RGB	78.8	82.4
	HoG Feature	78.9	82.6
	DINO Feature	79.0	82.7

6. Conclusion and Limitation

In this paper, we attempted to answer the question of what is learned during MIM pre-training. We adopted multi-order interactions to study the interaction order among image patches. We discovered that MIM essentially teaches the network to learn middle-order interactions among image patches for more complex feature extraction regardless of the network architecture. Based on our findings, we further proposed a general MIM framework A^2MIM that is compatible with both Transformers and CNNs. Besides a proposed novel masking mechanism, we also proposed a loss in the Fourier domain to enhance the middle-order interaction among patches. Experimental results showed that our proposed framework improves the representations learned for CNNs and Transformers, yielding superior performance than prior methods on various downstream tasks.

Meanwhile, we list two limitations of A^2MIM, as shown in Figure 6. (i) CNNs architectures benefit less from A^2MIM pre-training compared to ViTs, e.g., ResNet and ConvNeXt gain around 1% Acc while ViTs obtain more than 2% gains. We hypothesize that the inductive bias of CNNs limits the learning of middle-order interactions induced by MIM. (ii) ViTs benefit more with longer pre-training, while no significant gain is observed for CNNs after 300 epochs pre-training. Figure 6(a) shows that ResNet-50/152 obtains limited or negative performance gains for pre-training of 800 epochs or more. We hope our work could inspire the community to further promote self-supervised pre-training.

Acknowledgement

This work was supported by National Key R&D Program of China (No. 2022ZD0115100), National Natural Science Foundation of China Project (No. U21A20427), and Project (No. WU2022A009) from the Center of Synthetic Biology and Integrated Bioengineering of Westlake University.

References

Akbari, H., Yuan, L., Qian, R., Chuang, W.-H., Chang, S.-F., Cui, Y., and Gong, B. Vatt: Transformers for multimodal self-supervised learning from raw video, audio and text. In Advances in Neural Information Processing Systems (NeurIPS), 2021.

Ancona, M., Oztireli, C., and Gross, M. Explaining deep neural networks with a polynomial time algorithm for shapley value approximation. In International Conference on Machine Learning (ICML), pp. 272–281. PMLR, 2019.

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

Baevski, A., Hsu, W.-N., Xu, Q., Babu, A., Gu, J., and Auli, M. Data2vec: A general framework for self-supervised learning in speech, vision and language. arXiv preprint arXiv:2202.03555, 2022.

Bao, H., Dong, L., and Wei, F. Beit: Bert pre-training of image transformers. In International Conference on Learning Representations (ICLR), 2022.

Bardes, A., Ponce, J., and LeCun, Y. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. In International Conference on Learning Representations (ICLR), 2022.

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

Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., and Joulin, A. Emerging properties in self-supervised vision transformers. In Proceedings of the International Conference on Computer Vision (ICCV), 2021.

Chen, M., Radford, A., Child, R., Wu, J., Jun, H., Luan, D., and Sutskever, I. Generative pretraining from pixels. In International Conference on Machine Learning (ICML), pp. 1691–1703. PMLR, 2020a.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. arXiv preprint arXiv:2002.05709, 2020b.

Chen, X. and He, K. Exploring simple siamese representation learning. arXiv preprint arXiv:2011.10566, 2020.

Chen, X., Fan, H., Girshick, R., and He, K. Improved baselines with momentum contrastive learning. arXiv preprint arXiv:2003.04297, 2020c.

Chen, X., Xie, S., and He, K. An empirical study of training self-supervised vision transformers. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 9640–9649, 2021.

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

Cubuk, E. D., Zoph, B., Shlens, J., and Le, Q. V. Randaugment: Practical automated data augmentation with a reduced search space. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 702–703, 2020.

Deng, H., Ren, Q., Chen, X., Zhang, H., Ren, J., and Zhang, Q. Discovering and explaining the representation bottleneck of dnns. In International Conference on Learning Representations (ICLR), 2022.

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

Doersch, C., Gupta, A., and Efros, A. A. Unsupervised visual representation learning by context prediction. In Proceedings of the International Conference on Computer Vision (ICCV), 2015.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. An image is worth 16x16 words: Transformers for image recognition at scale. In International Conference on Learning Representations (ICLR), 2021.

El-Nouby, A., Izacard, G., Touvron, H., Laptev, I., Jegou, H., and Grave, E. Are large-scale datasets necessary for self-supervised pre-training? arXiv preprint arXiv:2112.10740, 2021.

Ermolov, A., Siarohin, A., Sangineto, E., and Sebe, N. Whitening for self-supervised representation learning. In International Conference on Machine Learning (ICML), 2021.

Esser, P., Rombach, R., and Ommer, B. Taming transformers for high-resolution image synthesis. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 12873–12883, June 2021.

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

Ge, C., Liang, Y., Song, Y., Jiao, J., Wang, J., and Luo, P. Revitalizing cnn attentions via transformers in self-supervised visual representation learning. Advances in Neural Information Processing Systems (NeurIPS), 2021.

Gidaris, S., Singh, P., and Komodakis, N. Unsupervised representation learning by predicting image rotations. In International Conference on Learning Representations (ICLR), 2018.

Goyal, P., Dollár, P., Girshick, R., Noordhuis, P., Wesolowski, L., Kyrola, A., Tulloch, A., Jia, Y., and He, K. Accurate, large minibatch sgd: Training imagenet in 1 hour. arXiv preprint arXiv:1706.02677, 2020.

Grill, J.-B., Strub, F., Altché, F., Tallec, C., Richemond, P. H., Buchatskaya, E., Doersch, C., Pires, B. A., Guo, Z. D., Azar, M. G., et al. Bootstrap your own latent: A new approach to self-supervised learning. Advances in Neural Information Processing Systems (NeurIPS), 2020.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR), pp. 770–778, 2016.

He, K., Gkioxari, G., Dollár, P., and Girshick, R. Mask r-cnn. In Proceedings of the International Conference on Computer Vision (ICCV), 2017.

He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. Momentum contrast for unsupervised visual representation learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 9729–9738, 2020.

He, K., Chen, X., Xie, S., Li, Y., Dollár, P., and Girshick, R. Masked autoencoders are scalable vision learners. In Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR), 2022.

Hinton, G. E. and Zemel, R. Autoencoders, minimum description length and helmholtz free energy. Advances in neural information processing systems (NeurIPS), 6, 1993.

Huang, G., Sun, Y., Liu, Z., Sedra, D., and Weinberger, K. Q. Deep networks with stochastic depth. In Proceedings of the European Conference on Computer Vision (ECCV), 2016.

Jiang, L., Dai, B., Wu, W., and Loy, C. C. Focal frequency loss for image reconstruction and synthesis. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 13919–13929, 2021.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. Imagenet classification with deep convolutional neural networks. In Advances in Neural Information Processing Systems (NeurIPS), volume 25. Curran Associates, Inc., 2012a.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems (NeurIPS), pp. 1097–1105, 2012b.

Li, S., Liu, Z., Wang, Z., Wu, D., Liu, Z., and Li, S. Z. Boosting discriminative visual representation learning with scenario-agnostic mixup. ArXiv, abs/2111.15454, 2021.

Li, S., Wang, Z., Liu, Z., Wu, D., and Li, S. Z. Openmixup: Open mixup toolbox and benchmark for visual representation learning. https://github.com/Westlake-AI/openmixup, 2022.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. Microsoft coco: Common objects in context. In Proceedings of the European Conference on Computer Vision (ECCV), 2014.

Lin, T.-Y., Dollar, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 2017.

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., and Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2021.

Liu, Z., Li, S., Wu, D., Chen, Z., Wu, L., Guo, J., and Li, S. Z. Automix: Unveiling the power of mixup for stronger classifiers. In Proceedings of the European Conference on Computer Vision (ECCV), 2022a.

Liu, Z., Mao, H., Wu, C.-Y., Feichtenhofer, C., Darrell, T., and Xie, S. A convnet for the 2020s. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022b.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. In International Conference on Learning Representations (ICLR), 2019.

Naseer, M. M., Ranasinghe, K., Khan, S. H., Hayat, M., Shahbaz Khan, F., and Yang, M.-H. Intriguing properties of vision transformers. In Advances in Neural Information Processing Systems (NeurIPS), volume 34, 2021.

Park, N. and Kim, S. Blurs behave like ensembles: Spatial smoothings to improve accuracy, uncertainty, and robustness. arXiv preprint arXiv:2105.12639, 2021.

Park, N. and Kim, S. How do vision transformers work? In International Conference on Learning Representations (ICLR), 2022.

Pathak, D., Krahenbuhl, P., Donahue, J., Darrell, T., and Efros, A. A. Context encoders: Feature learning by inpainting. In Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR), pp. 2536–2544, 2016.

Radford, A., Narasimhan, K., Salimans, T., and Sutskever, I. Improving language understanding by generative pre-training, 2018.

Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., and Sutskever, I. Zero-shot text-to-image generation. ArXiv, abs/2102.12092, 2021.

Sameni, S., Jenni, S., and Favaro, P. Dilemma: Self-supervised shape and texture learning with transformers. arXiv preprint arXiv:2204.04788, 2022.

Selvaraju, R. R., Desai, K., Johnson, J., and Naik, N. Casting your model: Learning to localize improves self-supervised representations. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 11058–11067, 2021.

Shelhamer, E., Long, J., and Darrell, T. Fully convolutional networks for semantic segmentation. IEEE transactions on pattern analysis and machine intelligence, 39(4):640–651, 2017.

Song, Z., Xiao, G., Hu, G., and Zhao, C. Deep perturbation learning: Improve the network performance via image perturbations. In Proceedings of the 40th international conference on Machine learning (ICML), 2023.

Tian, K., Jiang, Y., Diao, Q., Lin, C., Wang, L., and Yuan, Z. Designing bert for convolutional networks: Sparse and hierarchical masked modeling. In International Conference on Learning Representations (ICLR), 2023.

Tian, Y., Krishnan, D., and Isola, P. Contrastive multiview coding. In Proceedings of the European Conference on Computer Vision (ECCV), 2020.

Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., and Jégou, H. Training data-efficient image transformers & distillation through attention. In International Conference on Machine Learning (ICML), pp. 10347–10357. PMLR, 2021.

van den Oord, A., Li, Y., and Vinyals, O. Representation learning with contrastive predictive coding. ArXiv, abs/1807.03748, 2018.

Van den Oord, A., Li, Y., and Vinyals, O. Representation learning with contrastive predictive coding. arXiv e-prints, pp. arXiv–1807, 2018.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, ., and Polosukhin, I. Attention is all you need. Advances in neural information processing systems (NeurIPS), 30, 2017.

Vincent, P., Larochelle, H., Bengio, Y., and Manzagol, P.-A. Extracting and composing robust features with denoising autoencoders. In Proceedings of the 25th international conference on Machine learning (ICML), pp. 1096–1103, 2008.

Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., Manzagol, P.-A., and Bottou, L. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. Journal of machine learning research, 11(12), 2010.

Wang, W., Xie, E., Li, X., Fan, D.-P., Song, K., Liang, D., Lu, T., Luo, P., and Shao, L. Pvtv2: Improved baselines with pyramid vision transformer. arXiv preprint arXiv:2106.13797, 2022.

Wei, C., Fan, H., Xie, S., Wu, C.-Y., Yuille, A., and Feichtenhofer, C. Masked feature prediction for self-supervised visual pre-training. arXiv preprint arXiv:2112.09133, 2021.

Wightman, R., Touvron, H., and Jégou, H. Resnet strikes back: An improved training procedure in timm, 2021.

Woo, S., Debnath, S., Hu, R., Chen, X., Liu, Z., Kweon, I.-S., and Xie, S. Convnext v2: Co-designing and scaling convnets with masked autoencoders. ArXiv, abs/2301.00808, 2023.

Wu, D., Li, S., Zang, Z., and Li, S. Z. Exploring localization for self-supervised fine-grained contrastive learning. In Proceedings of the British Machine Vision Conference (BMVC), 2022.

Wu, F., Li, S., Jin, X., Jiang, Y., Radev, D., Niu, Z., and Li, S. Z. Explaining graph neural networks via non-parametric subgraph matching. In Proceedings of the 40th international conference on Machine learning (ICML), 2023.

Wu, Z., Xiong, Y., Stella, X. Y., and Lin, D. Unsupervised feature learning via non-parametric instance discrimination. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018.

Xiao, T., Liu, Y., Zhou, B., Jiang, Y., and Sun, J. Unified perceptual parsing for scene understanding. In European Conference on Computer Vision (ECCV). Springer, 2018.

Xiao, T., Reed, C. J., Wang, X., Keutzer, K., and Darrell, T. Region similarity representation learning. arXiv preprint arXiv:2103.12902, 2021.

Xie, E., Ding, J., Wang, W., Zhan, X., Xu, H., Li, Z., and Luo, P. Detco: Unsupervised contrastive learning for object detection. In Proceedings of the International Conference on Computer Vision (ICCV), 2021a.

Xie, Z., Zhang, Z., Cao, Y., Lin, Y., Bao, J., Yao, Z., Dai, Q., and Hu, H. Simmim: A simple framework for masked image modeling. arXiv preprint arXiv:2111.09886, 2021b.

Xu, K., Hu, W., Leskovec, J., and Jegelka, S. How powerful are graph neural networks? In International Conference on Learning Representations (ICLR), 2019.

You, Y., Li, J., Reddi, S., Hseu, J., Kumar, S., Bhojanapalli, S., Song, X., Demmel, J., Keutzer, K., and Hsieh, C.-J. Large batch optimization for deep learning: Training BERT in 76 minutes. In International Conference on Learning Representations (ICLR), 2020.

Yun, S., Han, D., Oh, S. J., Chun, S., Choe, J., and Yoo, Y. Cutmix: Regularization strategy to train strong classifiers with localizable features. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 6023–6032, 2019.

Zbontar, J., Jing, L., Misra, I., LeCun, Y., and Deny, S. Barlow twins: Self-supervised learning via redundancy reduction. In International Conference on Machine Learning (ICML), pp. 12310–12320. PMLR, 2021.

Zhang, H., Cisse, M., Dauphin, Y. N., and Lopez-Paz, D. mixup: Beyond empirical risk minimization. In International Conference on Learning Representations (ICLR), 2018.

Zhang, H., Li, S., Ma, Y., Li, M., Xie, Y., and Zhang, Q. Interpreting and boosting dropout from a game-theoretic view. arXiv preprint arXiv:2009.11729, 2020.

Zhang, R., Isola, P., and Efros, A. A. Colorful image colorization. In Proceedings of the European Conference on Computer Vision (ECCV), 2016.

Zhong, Z., Zheng, L., Kang, G., Li, S., and Yang, Y. Random erasing data augmentation. In Proceedings of the AAAI conference on artificial intelligence (AAAI), pp. 13001–13008, 2020.

Zhou, B., Zhao, H., Puig, X., Xiao, T., Fidler, S., Barriuso, A., and Torralba, A. Semantic understanding of scenes through the ade20k dataset. International Journal of Computer Vision (IJCV), 2019.

Zhou, J., Wei, C., Wang, H., Shen, W., Xie, C., Yuille, A., and Kong, T. ibot: Image bert pre-training with online tokenizer. arXiv preprint arXiv:2111.07832, 2021.

A. Details of Comparison Experiments

This section provides experimental details for Sec. 5, e.g., pre-training and evaluation on ImageNet-1K and transferring learning settings on downstream tasks. Experiment results and models are available at https://github.com/Westlake-AI/A2MIM.

A.1. ImageNet-1K Experiments

Pre-training. The default settings of A²MIM for CNNs and ViTs are provided in Tab. A1, following SimMIM (Xie et al., 2021b). We use AdamW (Loshchilov & Hutter, 2019) optimizer with the cosine scheduler and the linear learning rate scaling rule (Goyal et al., 2020): lr = base_lr × batchsize / 2048. Similar to current MIM methods, we only employ RandomResizedCrop with the scale of (0.67, 1.0) or (0.8, 1.0) and RandomFlip while do not require other complex augmentations (e.g., Rand Augment (Cubuk et al., 2020), mixups (Zhang et al., 2018; Yun et al., 2019; Liu et al., 2022a; Li et al., 2021), or stochastic depth (Huang et al., 2016)) during pre-training. As for ResNet and ConvNeXt models, we adopt Cosine learning rate decay for 100/300 and 800 epochs pre-training. As for ViTs, we use a similar Cosine decay when pre-training epochs less than 400 while using Step decay (the learning rate multiplied by 0.1 at 700-epoch) for 800-epoch pre-training.

End-to-end fine-tuning. As shown in Tab. A2, our fine-tuning settings follow common practices of supervised image classification on ImageNet-1K. For ViT architectures, the pre-trained model is fine-tuned for 200 epochs using the BEiT (Bao et al., 2022) version of DeiT (Touvron et al., 2021) training recipe to fully explore the performance, which employs AdamW (Loshchilov & Hutter, 2019) optimizer with the cross-entropy (CE) loss and layer-wise learning rate decay. For CNNs, we adopt RSB A3 (Wightman et al., 2021) setting for 100-epoch fine-tuning, which employs LAMB (You et al., 2020) optimizer with the binary cross-entropy (BCE) loss and smaller training resolutions. To fully explore the PT performances of CNNs, we also apply 300-epoch fine-tuning with RSB A2 (Wightman et al., 2021) and ConvNeXt (Liu et al., 2022b) training settings for ResNet and ConvNeXt models. Notice that the default drop depth rates of ResNet-50/101/152/200 and ConvNeXt-T/S/B are 0.05/0.1/0.15/0.2 and 0.1/0.3/0.4 in 300-epoch fine-tuning. The learning rates and drop depth can also be tuned for different PT methods.

A.2. Object Detection and Segmentation on COCO

We adopt Mask-RCNN (He et al., 2017) to perform transfer learning to object detection and semantic segmentation on COCO (Lin et al., 2014) using MMDetection[1] and Detectron2[2] code bases. For evaluation on ResNet-50, we follow MoCo (He et al., 2020) and fine-tune Mask R-CNN with the pre-trained ResNet-50-C4 backbone with SGD optimizer using 2× schedule (24 epochs). For evaluation of ViTs, we follow MAE (He et al., 2022) and CAE (Chen et al., 2022), which apply the pre-trained ViT backbone and an FPN neck (Lin et al., 2017) in Mask R-CNN. The model is fine-tuned by AdamW optimizer with 1× schedule (12 epochs). For a fair comparison, we follow (Bao et al., 2022; Xie et al., 2021b) to turn on relative position bias in ViT (Dosovitskiy et al., 2021) during both pre-training and transfer learning, initialized as zero, and the learning rate can be tuned for different PT methods.

Table A1. ImageNet-1K pre-training settings of A²MIM for ResNet/ConvNeXt and ViT/Swin models.

Configuration	ResNet / ConvNeXt	ViT / Swin
Pre-training resolution	224 × 224	224 × 224
Mask patch size	32 × 32	32 × 32
Mask ratio	60%	60%
Optimizer	AdamW	AdamW
Base learning rate	1.2×10^{-3}	4×10^{-4}
Weight decay	0.05	0.05
Optimizer momentum	$\beta_1, \beta_2 = 0.9, 0.999$	$\beta_1, \beta_2 = 0.9, 0.999$
Batch size	2048	2048
Learning rate schedule	Cosine	Step / Cosine
Warmup epochs	10	10
RandomResizedCrop	[0.8, 1]	[0.67, 1]
Rand Augment	✗	✗
Stochastic Depth	✗	✗
Gradient Clipping	✗	max norm= 5
PT epochs	100 / 300 / 800	300 / 800

Table A2. ImageNet-1K fine-tuning recipes of ViT, RSB A2/A3, and ConvNeXt architectures. Here we take ViT-B, ResNet-50, and ConvNeXt-T as examples.

Configuration	ViT	RSB A2	RSB A3	ConvNeXt
FT epochs	200	300	100	300
Training resolution	224	224	160	224
Testing resolution	224	224	224	224
Testing crop ratio	0.875	0.95	0.95	0.875
Optimizer	AdamW	LAMB	LAMB	AdamW
Base learning rate	1×10^{-2}	5×10^{-3}	8×10^{-3}	4×10^{-3}
Layer-wise decay	0.65	✗	✗	✗
Weight decay	0.05	0.02	0.02	0.05
Batch size	1024	2048	2048	4096
Learning rate schedule	Cosine	Cosine	Cosine	Cosine
Warmup epochs	20	5	5	20
Label smoothing	0.1	✗	✗	0.1
Stochastic depth	0.1	0.05	✗	0.1
Gradient clipping	5.0	✗	✗	✗
Rand Augment	(9, 0.5)	(7, 0.5)	(6, 0.5)	(9, 0.5)
Mixup alpha	0.8	0.1	0.1	0.8
CutMix alpha	1.0	1.0	1.0	1.0
EMA decay	0.99996	✗	✗	0.9999
Loss function	CE loss	BCE loss	BCE loss	CE loss

[1] https://github.com/open-mmlab/mmdetection

[2] https://github.com/facebookresearch/detectron2

(a) (b) (c) (d)

Figure A1. (a)(b): Occlusion robustness against different occlusion ratios of images (CL vs. MIM) is studied for both ViT-S and ResNet-50 on ImageNet-100. (c)(d): Distributions of the interaction strength $J^{(m)}$ (CL vs. MIM) are explored for both ViT-S and ResNet-50 on ImageNet-100. The label indicates the pre-training method + fine-tuning augmentation used, random stands for random weight initialization.

(a) (b) (c) (d)

Figure A2. Occlusion robustness against various random or salient occlusion ratios of images is studied in (a)(b) for ViT-S, and (c)(d) for ResNet-50 using various experimental settings on ImageNet-100. The label indicates the pre-training method + fine-tuning setting used, random stands for random weight initialization.

A.3. Semantic Segmentation on ADE-20K

We adopt UperNet (Xiao et al., 2018) to perform transfer learning to semantic segmentation on ADE-20K and use the semantic segmentation implementation in MMSegmentation[3]. We initialize the FCN (Shelhamer et al., 2017) or UperNet (Xiao et al., 2018) using the pre-trained backbones (ResNet-50 or ViTs) on ImageNet-1K. For ViTs, we fine-tune end-to-end for 160K iterations with AdamW and a batch size of 16. We search a optimal layer-wise decay from $\{0.8, 0.9\}$ and search optimal a learning rate from $\{1\times10^{-4}; 2\times10^{-4}; 3\times10^{-4}\}$ for all competitors. Similar to fine-tuning settings on COCO, we use relative position bias in ViT (Dosovitskiy et al., 2021) during both pre-training and transfer learning as (Bao et al., 2022; Xie et al., 2021b). For ResNet-50, we follow MoCo (He et al., 2020), i.e., all CNN models are fine-tuned for 160K iterations by SGD optimizer with the momentum of 0.9 and a batch size of 16.

B. Empirical Experiments

This section provides background information and experimental details for Sec. 3, and additional results of occlusion robustness evaluation and multi-order interaction strength.

B.1. Occlusion Robustness

In Sec. 3.1, we analyze robustness against occlusion for models pre-trained and fine-tuned on ImageNet-100 (a subset on ImageNet-1K divided by (Tian et al., 2020)) using the official implementation[4] provided by Naseer et al. (2021). Both MIM and contrastive-based methods are pre-trained 400 epochs on ImageNet-100 using their pre-training settings on ImageNet-1K. We adopt the fine-tuning training recipe as DeiT in Tab. A2 and use the same setting training 100 epochs for both ViT-S and ResNet-50. Note that we use the modified SimMIM for ResNet-50 (replacing masked patches in the input image with the RGB mean) in all experiments.

As shown in Fig. 1 and A1, we compared MIM pre-trained models supervised methods with various augmentations and contrastive learning pre-trained methods in terms of the top-1 accuracy under various occlusion ratios. We find that MIM methods show better occlusion robustness on both Transformers and CNNs. In addition to Sec. 3.1, we also provide results of salient occlusion (i.e., dropping patches according to salient maps) for ViT-S and ResNet-50 on ImageNet-100 in Fig. A2. Note that the occlusion ratio means the ratio of dropped and total patches, and we plot the mean of accuracy across 3 runs. Overall, we can conclude that MIM

[3] https://github.com/open-mmlab/mmsegmentation

[4] https://github.com/Muzammal-Naseer/Intriguing-Properties-of-Vision-Transformers

pre-trained models have stronger robustness against random and salient occlusions than supervised and contrastive-based pre-trained methods.

B.2. Multi-order Interaction

In Sec. 3.2, we interpret what is learned by MIM by multi-order interaction (Deng et al., 2022; Zhang et al., 2020). The interaction complexity can be represented by $I^{(m)}(i,j)$ (defined in Eqn. 1), which measures the average interaction utility between variables i,j on all contexts consisting of m variables. Notice that the order m reflects the contextual complexity of the interaction $I^{(m)}(i,j)$. For example, a low-order interaction (e.g., $m = 0.05n$) means the relatively simple collaboration between variables i,j, while a high-order interaction (e.g., $m = 0.05n$) corresponds to the complex collaboration. As figured out in the representation bottleneck (Deng et al., 2022), deep neural networks (DNNs) are more likely to encode both low-order interactions and high-order interactions, but often fail to learn middle-order interactions. We hypothesize that MIM helps models learn more middle-order interactions since MIM has a natural advantage in cases where some parts of the image are masked out. In Fig. 1, we calculate the interaction strength $J^{(m)}$ (defined in Eqn. 2) for fine-tuned models on ImageNet-100 using the official implementation[5] provided by Deng et al. (2022). Specially, we use the image of 224×224 resolution as the input and calculate $J^{(m)}$ on 14×14 grids, i.e., $n = 14\times14$. And we set the model output as $f(x_S) = \log \frac{P(\hat{y}=y|x_S)}{1-P(\hat{y}=y|x_S)}$ given the masked sample x_S, where y denotes the ground-truth label and $P(\hat{y}=y|x_S)$ denotes the probability of classifying the masked sample x_S to the true category.

Figure A3. Fourier transformed feature maps. The vertical axis is the relative log amplitudes of the high-frequency components, and the horizontal axis is the normalized depth of the network. The blue columns indicate the pooling layers, while the white columns indicate the convolution layers.

[5] https://github.com/Nebularaid2000/bottleneck

Figure A4. Feature maps variance. The vertical axis is the average variance value of feature maps. DeiT (Sup.) is supervised pre-training. The results of the randomly initialized network are plotted for reference.

B.3. MIM from Frequency Perspective

We first plot the log magnitude of Fourier-transformed feature maps of ResNet-50 with different pre-training methods using the tools[6] provided by Park & Kim (2022) on ImageNet-1K. Following (Park & Kim, 2022), we first convert feature maps into the frequency domain and represent them on the normalized frequency domain (the highest frequency components are at $\{-\pi, +\pi\}$). In Fig. A3, we report the amplitude ratio of high-frequency components by using Δ log amplitude. As shown in Fig. A3, inpainting and MIM show similar low-pass filtering effects at convolution layers as compared to contrastive learning. This indicates that inpainting and MIM reduce noise and uncertainty induced by high-frequency features. We argue that the reconstruction performance of MIM is mainly related to low or high-order interactions of patches (Deng et al., 2022), while reconstruction performance is not directly related to the learned representation quality. Then, we provide the standard deviation of feature maps by block depth as (Park & Kim, 2022; 2021), which first calculates the feature map variance on the last two dimensions and then averages over the channel dimension for the whole dataset. Fig. A4 shows the feature variance of each layer of ResNet-50 with different pre-training methods on IN-1K. This figure indicates that MIM tends to reduce the feature map variance, and conversely, supervised training, inpainting, and contrastive learning based on CNN tend to increase variance (i.e., high frequencies). Compared to MIM, which learns better middle-order interactions, the inpainting task fails to filter out low-order interactions and thus leads to higher variance. To conclude, MIM methods learn middle-order interactions and reduce the feature map uncertainty (high frequencies) based on the CNN encoder for a generalized and stabilized feature extraction.

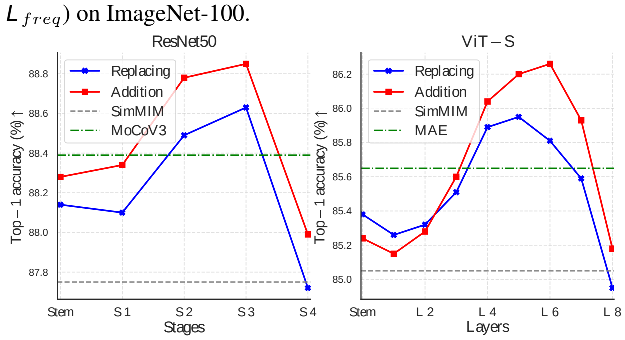
[6] https://github.com/xxxnell/how-do-vits-work

C. More Experiment Results

C.1. Ablation of Layers for Mask Token

In addition to Sec. 5.4, we analyze the optimal stage or layer for the mask token. The ablation experiments are conducted with ResNet-50 and ViTs on IN-100 and IN-1K using the fine-tuning protocol as Sec. 5.4. As shown in Fig. A5, adding the mask token to the medium stages (stage-3 of ResNet-50) or layers (layer-5 of ViT-S) yields the best performance on the pre-trained representation. Therefore, we apply the mask token to the 3-stage or the medium layer (around 3/4 of the total layers) in A^{2}MIM by default.

Figure A5. Ablation of the mask token in various stages (S) in ResNet-50 or layers (L) in ViT-S based on SimMIM (without L_{freq}) on ImageNet-100.



C.2. Ablation of the Proposed Modules

In addition to ablation studies in Sec. 5.4, we provide more ablation studies and empirical analysis on the proposed L_{freq} in the Fourier domain, as shown in Figure A6. As we discussed in Sec. 4, we hypothesize that learning medium frequencies would help better learn middle-order interactions. we thereby propose L_{freq} to tackle the dilemma of L_{spa}, which tends to learn low-frequency components (*i.e.,* contents reflected by high-order interactions). Although the reconstruction loss in the Fourier domain has a global perception, the high-frequency components are usually constructed by local details and noises (*i.e.,* low-order interactions), which might hurt the generalization abilities. Therefore, we introduce the reweight $w(u; v)$ to force the model to learn more medium-frequency components, which are identical to middle-order interactions. Then, we perform further analysis of the masked patch size for A^{2}MIM in Tab. A3. Note that we pre-train ResNet-50 for 100 epochs and ViT-B for 400 epochs on ImageNet-1K and report the fine-tuning results. As shown in Tab. A3, when the mask ratio is 60%, the optimal masked patch size is 32 32 for A^{2}MIM, which is the same as SimMIM.

D. Visualization Experimental Details

In addition to visualization results in Sec. 5.4, we visualize more reconstruction results of A^{2}MIM here. Similar to Fig. 4, we ablate the proposed components in A^{2}MIM based on ResNet-50 in Fig. A7, which demonstrates that A^{2}MIM helps ResNet-50 learn more spatial details, *i.e.,* more middle-order interactions. Moreover, we study the effects of the mask token in both ViTs and CNNs in Fig. A8.

Table A3. Ablation of masked patch size for A^{2}MIM based on ResNet-50 and ViT-B on ImageNet-1K.

Model	Masked patch size	Mask ratio	PT epoch	Top-1 Accuracy (%)
ResNet-50	8 / 16 / 32 / 64	0.6	100	78.2 / 78.6 / **78.8** / 78.7
ViT-B	8 / 16 / 32 / 64	0.6	400	82.9 / 83.4 / **83.5** / 83.3

E. Extended Related Work

In the recent decade, Deep Neural Networks (DNNs) have gained great success in various tasks with full supervision, such as computer vision (He et al., 2016; Liu et al., 2021; He et al., 2017; Song et al., 2023), natural language processing (Vaswani et al., 2017; Devlin et al., 2018; Radford et al., 2018), and graph representation learning (Xu et al., 2019; Wu et al., 2023). As DNNs scale up with more parameters, pre-training without labels by leveraging pre-text tasks has become increasingly popular. In addition to Sec. 2, we provide extended discussions of two types of popular self-supervised vision pre-training approaches.

Contrastive Learning. Contrastive learning learns instance-level discriminative representations by extracting invariant features over distorted views of the same data, which is first introduced by CPC (van den Oord et al., 2018), CMC (Tian et al., 2020), and NPID (Wu et al., 2018). MoCo (He et al., 2020) and SimCLR (Chen et al., 2020b) adopted different mechanisms to introduce negative samples for contrast with the positive. BYOL (Grill et al., 2020) and its variants (Chen & He, 2020; Ge et al., 2021) further eliminate the requirement of negative samples to avoid representation collapse. Besides pairwise contrasting, SwAV (Caron et al., 2020) clusters the data while enforcing consistency between multi-augmented views of the same image. Barlow Twins (Zbontar et al., 2021) proposed to measure the cross-correlation matrix of distorted views of the same image to avoid representation collapsing. Meanwhile, some efforts have been made on top of contrastive methods to improve pre-training quality for specific downstream tasks (Xie et al., 2021a; Xiao et al., 2021; Selvaraju et al., 2021; Wu et al., 2022), which conduct fine-grained contrastive supervisions. MoCo.V3 (Chen et al., 2021) and DINO (Caron et al., 2021) adopted ViT (Dosovitskiy et al., 2021) in self-supervised pre-training to replace CNN backbones.

Autoregressive Modeling. Autoencoders (AE) is a typical type of network architecture that allows representation learn-

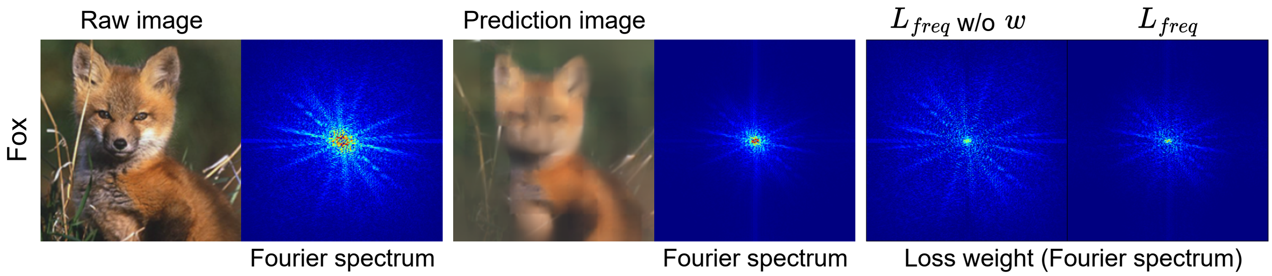


Figure A6. Visualization of predicted images and L_{freq} loss weight in Fourier domain. From the view of the Fourier spectrum, the raw image (left) contains 99% low-frequency components (usually present contents) and rich medium-frequency (structural patterns) and high-frequency components (local details and noises), while the predicted result (middle) provides fewer medium or high-frequency components. Calculated in the Fourier domain, the loss weights (right) of L_{freq} w/o w help the model to learn the full spectrum while L_{freq} focusing on the low and medium-frequency parts, which are more likely to be low-order or middle-order interactions.

ing with no annotation requirement (Hinton & Zemel, 1993). By forcing denoising property onto the learned representations, denoising autoencoders (Vincent et al., 2008; 2010) are a family of AEs that reconstruct the uncorrected input signal with a corrupted version of the signal as input. Generalizing the notion of denoising autoregressive modeling, masked predictions attracted the attention of both the NLP and CV communities. BERT (Devlin et al., 2018) performs masked language modeling (MLM), where the task is to classify the randomly masked input tokens. Representations learned by BERT as pre-training generalize well to various downstream tasks. For CV, inpainting tasks (Pathak et al., 2016) to predict large missing regions using CNN encoders and colorization tasks (Zhang et al., 2016) to reconstruct the original color of images with removed color channels are proposed to learn representation without supervision. With the introduction of Vision Transformers (ViTs) (Dosovitskiy et al., 2021; Liu et al., 2021), iGPT (Chen et al., 2020a) predicts succeeding pixels given a sequence of pixels as input. MAE (He et al., 2022) and BEiT (Bao et al., 2022) randomly mask out input image patches and reconstruct the missing patches with ViTs. Compared to MAE, MaskFeat (Wei et al., 2021) and SimMIM (Xie et al., 2021b) adopt linear layers as the decoder instead of another Transformer as in MAE. MaskFeat applied HOG as the prediction target instead of the RGB value. Other research endeavors (El-Nouby et al., 2021; Zhou et al., 2021; Assran et al., 2022; Akbari et al., 2021; Sameni et al., 2022) combine the idea of contrastive learning (CL) with MIM. SplitMask (El-Nouby et al., 2021) proposed to use half of the image pixels to predict the other half while applying InfoNCE loss (Van den Oord et al., 2018) across the corresponding latent features. MSN (Assran et al., 2022) matches the representation of an image view containing randomly masked patches and the original unmasked image. Similarly, iBOT (Zhou et al., 2021) adopts the Siamese framework to combine self-distillation with MIM. Moreover, Data2Vec (Baevski et al., 2022) proposed a framework that applies the masked prediction idea for either speech, NLP, or CV. However, most MIM works are confined to ViT architectures, recently proposed CIM (Fang et al., 2022) uses the output of a pre-trained tokenizer as the target and takes the output of a frozen BEiT as the encoder's input as a workaround to enable MIM on CNNs.

In this work, we propose A^{2}MIM with no components native to ViTs adopted to perform MIM with ViTs and CNNs. Two concurrent two after A^{2}MIM, SparK (Tian et al., 2023) and ConvNeXt.V2 (Woo et al., 2023), designed CNN-based MIM with sparse convolutions to tackle the irregular masked images. Compared to them, A^{2}MIM provides empirical explanations of why MIM works and designs an architecture-agnostic framework.

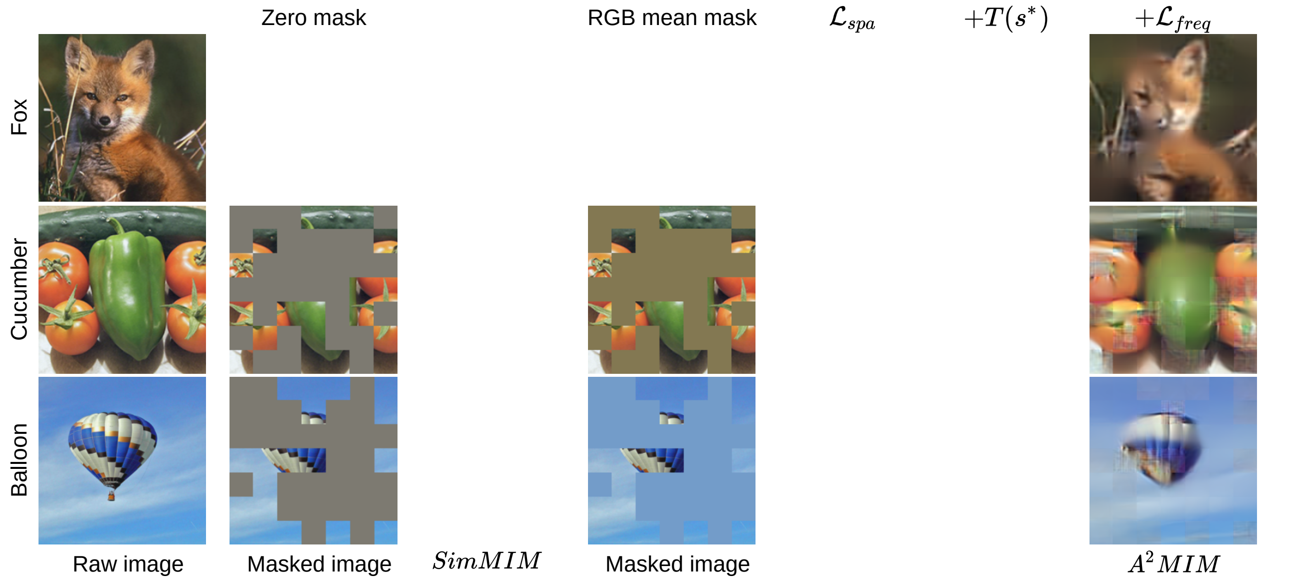


Figure A7. Visualizations of predicted results from SimMIM (middle) and our A^{2}MIM (right) based on ResNet-50 pre-trained 100-epochs on ImageNet-1K. $T(s\ )$ denotes the mask token T to the optimal stage-s in ResNet-50. We ablate the proposed components by adding them to the baseline SimMIM: replacing the zero mask with the RGB mean mask (the modified SimMIM baseline) and adding the mask token $T(s\ )$ relieve grid-like artifacts in predicted results; adding the proposed L_{freq} helps the model to capture more informative details.

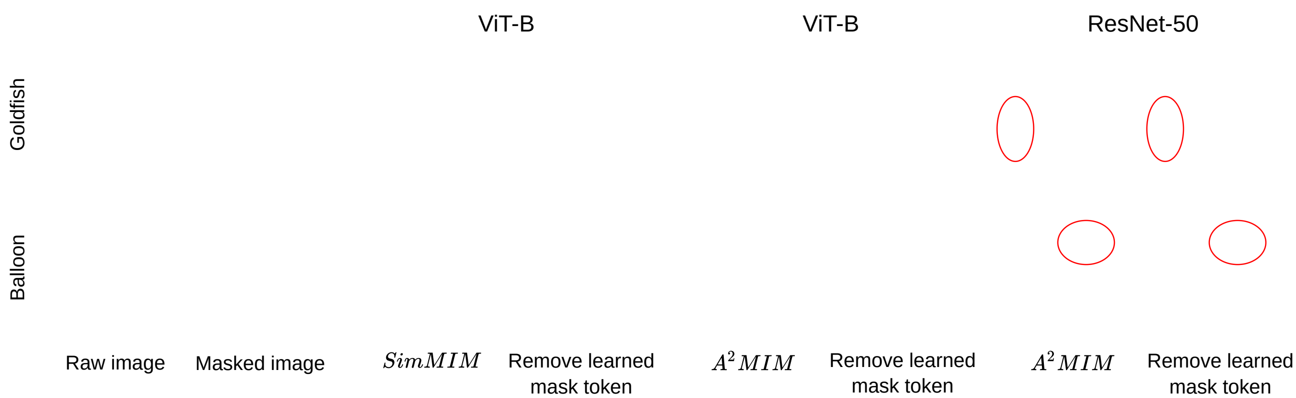


Figure A8. Visualizations of predicted results with and without the mask token on ImageNet-1K. Notice that mask tokens are adopted in the pre-trained models based on ViT-S (300-epoch) or ResNet-50 (100-epoch). Based on ViT-S, removing the mask token corrupts both contents of masked patches and overall colors in SimMIM while only corrupting the masked contents in A^{2}MIM. Based on ResNet-50, removing the mask token slightly affects spatial details in the masked patches and causes grid-like artifacts in the unmasked patches. The different effects of the mask token in ViT-S and ResNet-50 might be because the two architectures use different spatial-mixing operators and normalization layers. As for ViTs, the self-attention operation captures informative details from unmasked patches, but the non-overlap patch embedding and layer normalization mask each patch isolated. The mask token learns the mean templates (contents) of masked patches and gathers spatial details from unmasked patches by the self-attention operation. As for CNNs, each patch shares the same contents extracted by batch normalization layers, and the convolution operation extracts features from unmasked and masked patches equally. The mask token learns more high-frequency and informative details.