DDAE: Towards Deep Dynamic Vision BERT Pretraining

Honghao Chen[1,2*], **Xiangwen Kong**[3], **Xiangyu Zhang**[3], **Xin Zhao**[1,2], **Kaiqi Huang**[1,2,4†]

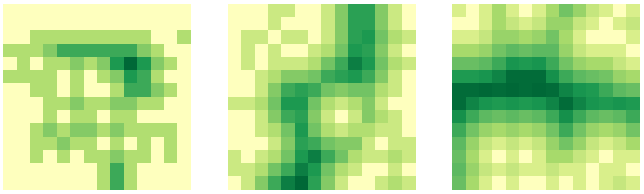
[1]CRISE, Institute of Automation, Chinese Academy of Sciences
[2]School of Artificial Intelligence, University of Chinese Academy of Sciences
[3]MEGVII Technology
[4]CAS Center for Excellence in Brain Science and Intelligence Technology
chenhonghao2021@ia.ac.cn, {kongxiangwen, zhangxiangyu}@megvii.com, {xzhao, kaiqi.huang}@nlpr.ac.cn

Abstract

Recently, masked image modeling (MIM) has demonstrated promising prospects in self-supervised representation learning. However, existing MIM frameworks recover all masked patches equivalently, ignoring that the reconstruction difficulty of different patches can vary sharply due to their diverse distance from visible patches. In this paper, we propose a novel deep dynamic supervision to enable MIM methods to dynamically reconstruct patches with different degrees of difficulty at different pretraining phases and depths of the model. Our deep dynamic supervision helps to provide more locality inductive bias for ViTs especially in deep layers, which inherently makes up for the absence of local prior for self-attention mechanism. Built upon the deep dynamic supervision, we propose Deep Dynamic AutoEncoder (DDAE), a simple yet effective MIM framework that utilizes dynamic mechanisms for pixel regression and feature self-distillation simultaneously. Extensive experiments across a variety of vision tasks including ImageNet classification, semantic segmentation on ADE20K and object detection on COCO demonstrate the effectiveness of our approach.

Introduction

Aided by the rapid gains in hardware, deep learning has ushered in the era of big models and big data. Along with the ever-growing model capacity, the demand for data can easily reach hundreds of millions (Dosovitskiy et al. 2020), which is not publicly accessible for labeled data. Self-Supervised Learning(SSL) frameworks, such as DINO (Caron et al. 2021), MoCo(Chen, Xie, and He 2021), BEiT (Bao, Dong, and Wei 2021), etc., have grown in concern in vision model pretraining without the need for labels. In particular, the recently proposed Masked Image Modeling (MIM) methods (He et al. 2022; Xie et al. 2022; Dong et al. 2021; Chen et al. 2022a,b) have shown remarkably impressive performance. Inspired by BERT (Devlin et al. 2018) in NLP, MIM pretrains the encoder by reconstructing the masked image patches from visible patches. MIM methods enable Vision Transformers (ViTs) to learn rich visual representations and exhibit great potential in various downstream tasks.



(a) 50% (b) 75% (c) 90%

Figure 1: Distance map under different mask ratios. We normalize the Euclidean distance from masked patch to the nearest visible patch. Darker color means larger distance from visible patches. High mask ratio leads to diverse distance for masked patches.

However, success comes with remaining obstacles. At present, all MIM frameworks recover all patches equivalently, ignoring the fact that the reconstruction difficulty of different patches can vary sharply, and the semantic reasoning and local perception ability required for patches are not the same. Generally, recovering patches with more visible patches around will be simpler, as long as the model has sufficient local perception ability. In contrast, reconstruction with few visible patches around requires the model to have strong semantic reasoning ability, given that there is little access to neighboring information. Treating all pixels equally will neglect this demand for different properties of the model, inevitably limiting its representation ability. Therefore, we ask if there is a way to focus on objectives with diverse characteristics as training progresses so that better representations can be learned overall.

Motivated by this observation and answering the question above, we propose a novel deep dynamic supervision to re-weight patches conditioned on their difficulty. Our deep dynamic supervision consists of two designs: **i) Dynamic.** We first define the reconstruction difficulty according to the distance between masked patches and visible ones, generating a distance map as shown in Fig 1. Then, we calculate loss mask conditioned on the distance map through learnable parameters β. With the update of β, the model dynamically focuses on different regions as the training progresses; **ii) Deep.** Since different layers naturally learn features with

[*]Work done partly during internship at MEGVII Technology.
[†]Corresponding author.

distinct levels of semantic, our dynamic loss enables us to exert different supervision signals for intermediate layers. Specifically, we set the learnable parameters β for intermediate layers to be independent of each other, guiding the diversification of the model. We show in Section 5 that these two designs both can bring improvement independently and more importantly, they can boost each other further when coupled, making a "$1 + 1 > 2$" effect.

Built upon the deep dynamic supervision, we propose a concise and effective SSL framework termed **D**eep **D**ynamic **A**uto**E**ncoder (DDAE). Our DDAE performs raw pixel regression and feature self-distillation simultaneously, taking into account both low-level information and high-level semantics. For the feature self-distillation, we directly align the encoder's intermediate features with the corresponding features of the momentum encoder, where the momentum encoder is updated by Exponential Moving Average (EMA) (Grill et al. 2020; He et al. 2020). The deep dynamic supervision is applied for both feature self-distillation and pixel regression. Note that our DDAE does not require any extra tokenizer or pre-trained teacher for distillation, it is a tokenizer-free framework and trains from scratch.

Our approach demonstrates strong performance in both short and longer shedules. For instance, the base-size DDAE can achieve 83.6% top-1 accuracy with only 100 epochs pretraining, surpassing MAE (He et al. 2022) and BEiT (Bao, Dong, and Wei 2021) pretrained for 800 epochs. For a longer pretraining schedule, we outperform MAE by +0.8% top-1 accuracy gains on ImageNet, +1.5% mIoU gains on ADE20K, and +1.7% AP^{box} gains on COCO with an even shorter pretraining time.

Moreover, our deep dynamic supervision is shown to inject more local priors for ViTs especially in deep layers, which we believe inherently makes up for the absence of local prior for self-attention mechanism and thus enhances the performance. We directly migrate the core design deep dynamic supervision to the representative methods w/o and w/ extra tokenizer, MAE (He et al. 2022) and BEiT-v2 (Peng et al. 2022a) respectively, surpassing original methods with consistent improvements. Since deep dynamic supervision does not introduce any additional structure, it can also enhance other MIM frameworks seamlessly.

Related Work

Masked Image Modeling

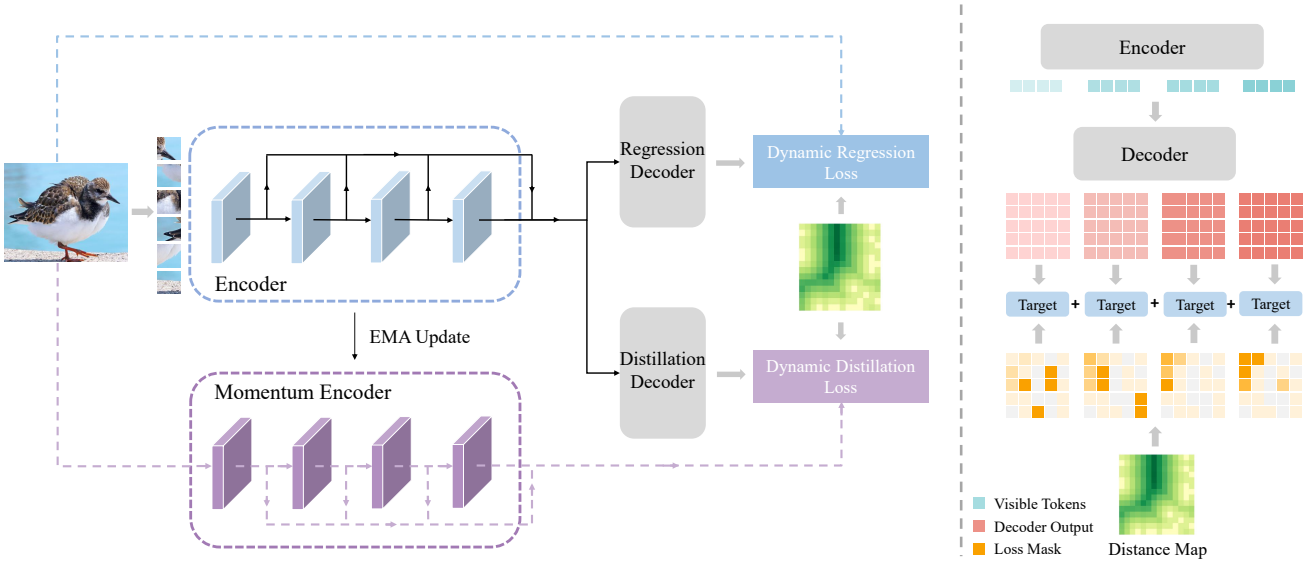
Inspired by BERT (Devlin et al. 2018) in NLP, MIM learn representation by reconstructing masked image patches from visible patches. Existing MIM methods can be divided into two categories according to the need for the additional tokenizer. **W/ extra tokenizer methods:** Represented by the pioneering work BEiT (Bao, Dong, and Wei 2021), these methods (Bao, Dong, and Wei 2021; Dong et al. 2021; Wei et al. 2022b; Chen et al. 2022a; Peng et al. 2022b) firstly transform image patches into semantic visual tokens through a pretrained discrete VAE (Ramesh et al. 2021)as visual tokenizer, then the corresponding tokens of masked image patches are reconstructed to pretrain the encoder. The tokenizer needs to be offline pretrained with fixed model architectures and extra data (Zhang et al. 2019b; Ramesh et al. 2021; Radford et al. 2021), some methods even further require an off-the-shelf DNN as teacher to distill tokenizer pre-training (Peng et al. 2022b). **W/O extra tokenizer methods:** MAE (He et al. 2022)constructs an asymmetric encoder-decoder structure, and directly performs raw pixel regression of masked image patches. SimMIM (Xie et al. 2022) allows hierarchical transformers such as Swin (Liu et al. 2021) to be directly applied to MIM. MaskFeat (Wei et al. 2022a), Data2vec (Baevski et al. 2022), BootMAE (Dong et al. 2022), SdAE (Chen et al. 2022b) explored the choice of reconstruction targets. However, existing MIM methods reconstruct all patches equivalently, neglecting the demand for different properties of the model and inevitably limiting its representation ability. Several inpainting works explored the dynamic loss design (Pathak et al. 2016; Yeh et al. 2016; Yu et al. 2018; Wang et al. 2018), but they are simply spatially discounted which is not truly dynamic during training. In contrast, our dynamic mechanism is controlled by learnable parameters and updated end to end through gradient back-propogation. Moreover, inpainting focuses on the quality of the generated image while MIM focuses on the representation of the encoder. Notably, we do not use any advanced tokenizer and train from scratch since we sheerly seek to delve into MIM's properties only.

Deep Supervision

Deep supervision methods are proposed to accelerate model's convergence and alleviate the problem of gradient vanishment. Through applying aux layers to transmit the supervision signal to the shallow layers, deep supervision has been used in early classification models (Szegedy et al. 2015; Lee et al. 2015) and extended to other visual recognition tasks (Xie and Tu 2015; Newell, Yang, and Deng 2016; Zhang et al. 2018b; Mosinska et al. 2018; Zhang et al. 2018a). Despite these advances, modern CNN classification models rarely use auxiliary classifiers since directly appending simple auxiliary classifiers on top of the early layers of a modern network hurts its performance (Huang et al. 2017). Several works utilize deep supervision to enhance the network through knowledge distillation (Sun et al. 2019; Zhang et al. 2019a). (Zhang et al. 2022) proposed Contrastive Deep Supervision to use contrastive learning signals for intermediate layers. A concurrent work (Ren et al. 2023) conduct an empirical study on the usage of deep supervision in MIM, but the signals of intermediate layers are still the same. Different from them, we introduce inconsistency in the intermediate signals for the first time. More importantly, deep supervision and our proposed dynamic loss fit surprisingly well, and their coupling leads to further gains. Although deep supervision has faded in modern supervised learning, DDAE makes a step towards unleashing its potential in MIM.

Method

In this section, we firstly elaborate on the basic framework of MIM. Then we introduce our proposed DDAE's two core designs termed Deep Dynamic Supervision and Deep Self-Distillation in Section 3.2 and Section 3.3 respectively. Our framework is illustrated in Fig 2.



(a) Overall Framework of DDAE (b) Deep Dynamic Loss

Figure 2: (a) Overall Framework of DDAE. Only visible patches are fed into the encoder while full patches are fed into the momentum encoder. We perform raw pixel regression and feature self-distillation both with deep dynamic loss. (b) Deep Dynamic Loss. Lighter color indicates that the signal comes from the shallower layer of the encoder. Best viewed in color.

Preliminary

Formally, MIM firstly divides the input image $X \in \mathbb{R}^{H\times W\times C}$ into non-overlapping flattened patches $\mathbf{x} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N]$, where $\mathbf{x}_i \in \mathbb{R}^{P^2C}$ according to the patch size P, and $N = (H \times W)/P^2$ is the number of patches. It then samples a random binary mask $\mathbf{m} = [m_1, m_2, ..., m_N]$, where $m_i \in \{0, 1\}$ to mask a portion of the flattened patches. The masked patches $\mathbf{x}_M \triangleq \mathbf{x} \odot \mathbf{m}$ are discarded (He et al. 2022) or substituted by learnable tokens `[MASK]` (Bao, Dong, and Wei 2021), and the rest patches $\mathbf{x}_{1-M} \triangleq \mathbf{x} \odot (1 - \mathbf{m})$ are used to reconstruct the dropped features or images to learn rich representations. The optimization target of MIM can be formulated as follow:

$$\min_{\theta,\phi} \mathop{\mathbb{E}}_{X\sim D} \mathcal{M}(d_\phi([f_\theta(\mathbf{x}_{1-M}), [\text{MASK}] \odot \mathbf{m}]), \mathbf{x}_M) \quad (1)$$

where $\odot$ denotes element-wise multiplication; $f_\theta(\cdot)$ and $d_\phi(\cdot)$ are encoder and decoder respectively; $\mathcal{M}(\cdot, \cdot)$ is the similarity measurement, which varies in different works, *e.g.*, l_2-distance in pixel space (He et al. 2022), perceptual loss in codebook space (Dong et al. 2021) or self-distillation loss in feature space (Chen et al. 2022b). In our work, we use the l_2-distance as our measurement, $\mathcal{M}(a, b) = ||a - b||^2$, for both of the pixel reconstruction and self-distillation. To simplify the formulation, we ignore the mask token term and use $d_\phi(\cdot)$ to represent $d_\phi([\cdot, [\text{MASK}] \odot \mathbf{m}])$.

Deep Dynamic Supervision

Dynamic. Firstly, we define the difficulty of reconstruction according to the distance between each masked patch and visible patches, generating a *distance map* with a distance transform $D(\cdot)$. For each masked token ($m_i = 1$), the distance transform assigns a number that is the Euclidean distance between that token and the nearest unmasked token in 2D space. Naturally, it is difficult to recover a patch that is far from visible ones and demands stronger semantic reasoning ability. On the contrary, reconstruction of a patch with visible ones nearby only requires fair local perception.

As shown in Fig 1, existing MIM methods often use a high mask ratio like 75% (which is proved to be critical to MIM's success (He et al. 2022)), values of the distance map vary diversely. Since the distance map is based on patches (14×14) rather than pixels (224×224), it only brings about 2% extra wall-clock time cost.

To guide the model to focus on different patches (corresponding to the requirement of distinctive properties) in different training phases, we propose $h_\beta(\cdot)$ to learn a dynamic coefficient β of the distance map to generate loss weight, directly applying dynamic focus to the loss weight of corresponding patches. h_β is derived as follows:

$$h(m_i|\beta) = \begin{cases} \dfrac{\exp(D(m_i)^\beta)}{\sum_{m_j=1} \exp(D(m_j)^\beta)} & , m_i = 1 \\ 0 & , m_i = 0 \end{cases} \quad (2)$$

We re-scale the loss weight map into $[0, 1]$ according to the max value of the weights. When $\beta > 0$, larger distance leads to greater loss weight, so the model pays more attention to the reconstruction of difficult patches. Increasing the value of β can exacerbate this trend. Conversely, when $\beta < 0$, larger distance leads to smaller loss weight, and decreas-

Algorithm 1: Pseudocode of Dynamic Loss

```
# inputs:
  # dis_map:distance map
  # β:learnable parameter
  # id_m:idx for masked patches
  # id_r:idx to restore order
# outputs:
  # l_mask:loss mask to reweight loss
  # l_norm:L2 norm of loss mask
  B, L = dis_map.shape

  # calculate loss mask, Eqn.(2)
  l_mask = gather(dis_map,dim=1,idx=id_m)
  l_mask = l_mask ** β
  l_mask = softmax(l_mask, dim=1)

  # calculate L2 norm of loss mask
  l_base = softmax(zeros((l_mask.shape)))
  b_norm = norm(l_base,p=2,dim=1).sum()
  l_norm = norm(l_mask,p=2,dim=1).sum()
  l_norm -= b_norm

  # normalize loss mask, restore shape
  l_mask /= max(l_mask,dim=1)
  len_keep = L - l_mask.shape[1]
  l_keep = zeros(B,len_keep)
  l_mask = cat((l_keep,l_mask),dim=1)
  l_mask = gather(l_mask,dim=1,idx=id_r)
```

ing β results in more importance attached to simple patches. Along with the changing of β, the model dynamically focuses on patches with diverse degrees of recovery difficulty. Algorithm 1 provides the pseudo-code of Dynamic Loss in a PyTorch-like style.

Deep. As commented in the introduction, the reconstruction of patches with diverse difficulties requires distinct characteristics of the model, and layers at different depths naturally learn features at different levels. For the sake of comprehensive and diverse feature modeling, we employ pixel-level dynamic supervision at varying depths of layer, further facilitating discriminative intrinsic representation learning. Taking the standard ViT with B blocks as the encoder, we divide the blocks into K groups and extract the features at the end of each group. For example, we set $K = 4$, then we extract output features of block 3, 6, 9 and 12 and feed them into the decoder respectively to recover masked patches. K is also called *supervisory number*, since K is equal to the number of supervisory signals. Then the loss function of pixel reconstruction L_{pixel} is derived as follows:

$$L_{pixel}(\theta, \phi, \beta) = \sum_{i=1}^{K} h_{\beta_i}(\mathbf{m}) \odot ||d_{\phi_p}(f_\theta^{(g_i)}(\mathbf{x}_{1-M})) - \mathbf{x}_M||^2 \tag{3}$$

where $f^{(i)}(\cdot)$ denotes to the output features of block-i in encoder, and $g_i = \frac{B}{K} i$ denotes to the group index; d_{ϕ_p} is the regression decoder; Note that β of different layers are independent of each other.

Deep dynamic supervision does not introduce any additional structure, so it can be seamlessly migrated to existing MIM structures. We discussed its universality in Table 5 where we migrated it to the representative MIM methods of one-stage and multi-stage.

Deep Self-Distillation

In addition to raw pixel regression, the layer-by-layer correspondence between intermediate features is more suitable for the design of deep dynamic supervision. Therefore, we designed Deep Self-Distillation based on BootMAE (Dong et al. 2022) to further strengthen the model through high-level semantic features. Specifically, the momentum encoder provides the features of masked patches of each layer as the target of self-distillation, so that the intermediate features of each layer corresponding to the encoder have their own targets of self-distillation. The momentum encoder is updated by the encoder using the exponential moving average method (EMA) (He et al. 2020). Formally, denoting the parameters of the encoder f_θ as θ and those of the momentum encoder $f_{\theta'}$ as $\theta^{'}$, we update $\theta^{'}$ by:

$$\theta^{'} \leftarrow m\theta^{'} + (1-m)\theta \tag{4}$$

Here $m \in [0, 1)$ is a momentum coefficient and set to 0.9999 by default. Note that $\theta^{'}$ are not updated by back-propagation but by equation 4 after each training step. The feature self-distillation also uses deep dynamic supervision as pixel regression. Note that the regression of raw pixels is one-vs-all, the features of all layers will be reconstructed by one regression decoder. The feature self-distillation is all-vs-all, which means the shallow features of the encoder are self-distilled by the corresponding shallow features of the momentum encoder, and the deep features are self-distilled by the corresponding deep features. Pixel regression and feature self-distillation use separate decoders, which both consist of two-layer transformer blocks. Since the decoders are light-weighted, the additional cost brought by deep supervision is acceptable. Now the deep self-distillation formula is:

$$L_{distill}(\theta, \beta) = \sum_{i=1}^{K} h_{\beta_i}(\mathbf{m}) \odot ||d_{\phi_d}(f_\theta^{(g_i)}(\mathbf{x}_{1-M})) - f_{\theta'}^{(g_i)}(\mathbf{x}_M)||^2 \tag{5}$$

where $f_\theta(\cdot)$ and $f_{\theta'}(\cdot)$ are encoder and momentum encoder respectively; d_{ϕ_d} is the distillation decoder; We add an L_2 regularization term to constrain the magnitude of β. Since L_2 norm reaches minimum under uniform distributions, the L_2 regularization loss here is to ensure that the loss mask does not deviate too far from a uniform mask, this can effectively prevent beta from being too extreme(in which case the model only focus on the hardest or simplest patches). The overall loss function is:

$$L = L_{pixel} + L_{distill} + \sum_i \lambda_i ||h_{\beta_i}(\mathbf{m})||_2 \tag{6}$$

where $\boldsymbol{\lambda}$ are the scale factors to tune the L_2 regularization term and is set to $0.1 \times g_i$ by default.

Methods	#Epochs	Finetune(↑)	Linear
Methods using ViT-B:			
DINO	300	82.8	78.2
MoCo-v3	300	83.2	76.7
BEiT	800	83.2	56.7
MAE	800	83.4	64.4
MAE	1600	83.6	68.0
DDAE(ours)	**100**	**83.6**	60.9
SimMIM	800	83.8	56.7
MaskFeat	1600	84.0	N/A
DeepMIM	1600	84.0	N/A
SdAE	800	84.0	N/A
DDAE(ours)	**800**	**84.4**	68.1
Methods using ViT-L:			
BEiT	800	85.2	73.5
MAE	800	85.4	73.9
MAE	1600	85.9	76.6
DDAE(ours)	**800**	**86.1**	77.0

Table 1: Image classification accuracy (%) comparison on ImageNet-1K of different methods with ViT-B/L as backbone. We report the fine-tuning and linear probing accuracy and our method DDAE consistently outperforms previous self-supervised methods with large margins.

Method	Pixel	EMA	Finetune
			<u>83.0</u>
		✓	83.4
	✓		83.3
DDAE	✓	✓	**83.6 (+0.6)**

Table 2: Ablation study on Deep Dynamic Supervision (DDS). Here EMA means using feature self-distillation and ✓means applying DDS on that part.

Pixel	EMA	DS	DL	Finetune
✓				82.6
✓			✓	82.8
✓	✓			<u>83.0</u>
✓	✓		✓	83.2 (+0.2)
✓	✓	✓		83.3 (+0.3)
✓	✓	✓	✓	**83.6 (+0.6)**

Table 3: Ablation study on Dynamic Loss. We decouple dynamic loss and deep supervision to give a clear ablation. Notably, they can boost each other further when coupled.

Experiments

Implementation

We conduct experiments on ImageNet-1K without labels as the pretraining data for self-supervised learning. The input resolution is set as 224×224 during pretraining and partitioned into 16×16 size patches. We pretrain the standard ViT small, base and large architectures, i.e., ViT-S/16, ViT-B/16 and ViT-L/16. The pixel regression decoder and feature distillation decoder both consist of 2 transformer blocks, along with an extra linear projection head for predictions. The dimension for the pixel regression decoder is 512 while for feature distillation decoder is set the same as the encoder. We use block-wise masking with a ratio of 75%. The data augmentation is only standard random cropping and horizontal flipping. All β are initialized as -0.5 by default. The distance map is generated in the dataloader according to the mask. Since the dataloader is achieved with multi-threads for parallel processing, the generation of the distance map only brings about 2% extra wall-clock time cost compared with original data-processing. See Appendix A for details of the pretraining settings.

Image Classification

We evaluate both fine-tuning accuracy and linear probing accuracy on ImageNet-1k. Table 1 presents the comparison with previous state-of-the-art MIM-based methods. Our DDAE acheives consistent advantages both in a short schedule and a longer schedule. In particular, with only 100 epochs pre-training, DDAE can achieve comparable performance with MAE using 1600 epochs pre-training and surpass 800 epochs pre-trained BEiT. Furthermore, with a longer pretraining schedule, DDAE achieves 84.4% top-1 accuracy, surpassing previous self-supervised methods by a large margin.

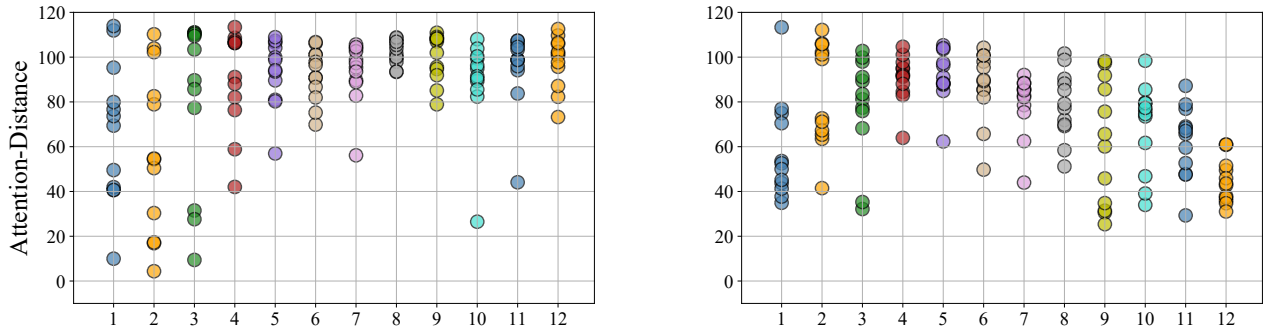
In terms of linear probing, our approach exceeds the above MIM methods with the same training epochs, but is not as good as the contrastive-based methods. Contrastive-based methods compare across images while MIM-based methods exploit the whole image structure, which may care about more than 1000 classes (Chen et al. 2022a). This phenomenon is also reported in MAE (He et al. 2022) and BEiT (Bao, Dong, and Wei 2021), so for MIM-based methods, fine-tuning measurement may be a better metric to validate their effectiveness.

Ablation Study and Analysis

Ablation on deep dynamic supervision. Then, we study the influence of Deep Dynamic Supervision (DDS) on pixel regression and feature self-distillation, we conduct ablation experiments with ViT-B as encoder, pretrain for 100 epochs and finetune 100 epochs on ImageNet-1K. Results in Table 2 suggests that either ablation downgrades the performance. Applying DDS on Pixel and Feature both can improve the feature representation learned and more importantly, the benefits brought by them can be superimposed.

Ablation on dynamic loss. We further decouple Dynamic Loss (DL) and Deep Supervision (DS) to give a clear ablation on the importance of each part. As shown in Table 3, the results demonstrate that DL itself can bring consistent improvements under different ablations. When combined with DS, the performance gains (+0.6) is even greater than the naive sum of them (0.2 + 0.3). This indicates that our introduction of DL and DS both can bring improvement independently and they can boost each other further when coupled.

Ablation on the supervisory number K. We further explored the impact of supervisory number K on performance.



(a) MAE (b) DDS-MAE

Figure 3: The averaged attention distance in different attention heads (dots) w.r.t the layer number on MAE (a) and DDS-MAE (b). The same ViT-B is used as backbone. DDS remarkably reduces the average attention distance in deep layers.

The models are finetuned on ImageNet-1K for 100 epochs. We set ViT-S as encoder and pretrain for 100 epochs. Specifically, we set K to be 1, 2, 4, and 6 respectively. The method to determine the block index is the same as that described in Section 3.2. As shown in Table 4, the additional benefits brought by more than four supervisory signals have been saturated. So we set $K = 4$ as default in our framework.

More local priors for ViTs. We conjecture that the fine-grained task built by MIM enables ViT to pay more attention to local information besides using global semantic inference, which inherently makes up for the absence of local prior in the structure of ViTs. We depict the average attention distance of vanilla MAE and DDS-MAE to demonstrate our design's effectiveness in providing more local priors in deep layers. Specifically, we compute averaged attention distance in each attention head of each layer (Dosovitskiy et al. 2020). We conduct experiments based on MAE since it is a simple enough MIM framework. As illustrated in Fig 3, DDS remarkably reduces the average attention distance of the model in deep layers. This phenomenon indicates that through deep dynamic supervision the model obtains more local priors in deep layers. Moreover, for DDS part, different heads in relatively deeper layers behave more diversely, which may also contribute to DDS's effectiveness.

Analysis on dynamic curve of β. We depict the curve of dynamic coefficients β changes during training for different layers in Fig 4. Firstly, for the comparison among layers, the ultimate β values of shallow layers are obviously smaller than that of deep layers. Note that a smaller

K	Block Index	Finetune
1	[12]	79.1
2	[6, 12]	79.5
4	[3, 6, 9, 12]	**79.9**
6	[2, 4, 6, 8, 10, 12]	**79.9**

Table 4: Ablation study on supervisory numbers K. Note all supervisions use Dynamic Loss.

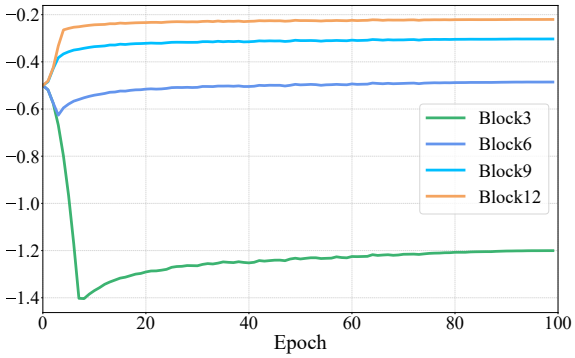


Figure 4: β dynamic curve for different layers. Smaller β means more importance is attached to simple patches.

β value means that the layer pays more attention to simple patches which are close to visible ones. Therefore, the shallow layer tends to undertake the reconstruction task of simpler patches, while the deep layer will attach more importance to the reconstruction of relatively more difficult patches. The reconstruction targets of different layers vary sharply, which guides the diversification of different components of the network through inconsistent signals. Secondly, for layer 3 and 6, β drops sharply at the beginning, then rises slowly as the training progresses. For layer 9 and 12, since the initialization value is too small, they keep rising during the whole training. Basically, they all reflect the trend of easy first and difficult later. The final values of β and relative relationship among layers are highly robust to initialization, we show that even inverse initialization leads to the same final values in appendix B.

Migrating deep dynamic supervision to existing MIM methods. Although for simplicity we build our framework as a tokenizer-free approach, our core design, Deep Dynamic Supervision(DDS), is compatible with other existing MIM methods. To demonstrate that, we conduct experiments to migrate DDS into representative w/o and w/ extra tokenizer MIM methods, MAE (He et al. 2022) and BEiT-

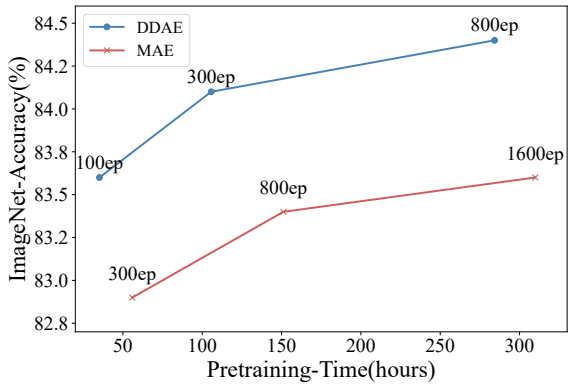


Figure 5: Pre-training efficiency comparison. We compare pre-training time cost and ImageNet top-1 accuracy on ViT-Base using NVIDIA V100 GPUs.

Methods	Extra tokenizer	Epochs	FT
MAE	w/o	300	82.9
DDS-MAE	w/o	100	**83.2**
BEiT-v2	w/	100	83.9
DDS-BEiT-v2	w/	100	**84.1**

Table 5: Plug-and-play Experiments.

v2 (Peng et al. 2022a) respectively. We extract intermediate features of the encoder, then feed them into decoder together with the encoder's final output. The reconstruction targets are all set the same as the final output. Reconstruction losses for intermediate layers are multiplied by their corresponding dynamic loss weight as mentioned in Section 3.2. The plug-and-play results reported in Table 5 demonstrates our design's generality for both those w/o and w/ extra tokenizer MIM approaches.

Pre-training efficiency comparison. We depicted the comparison of pre-training time and performance in Fig 5. Although the cost of DDAE *de facto* surpasses MAE per epoch, it can speed up convergence and outperforms MAE significantly within a similar training time. For instance, 100-epoch pretrained DDAE performs on par with 1600-epoch pretrained MAE; 800-epoch DDAE exceeds 1600-epoch MAE by a remarkablely large margin (+0.8) with an even shorter training time.

Visualization of the attention map. We further depict the shallow layer's attention map of MAE and DDAE in Fig 6. Specifically, we depict the attention map averaged over 12 attention heads between the class token and the patch tokens in the shallow layer (Block 3) of the ViT-B encoder pretrained on ImageNet-1K. Our DDAE forms target perception attention effectively in the very early stage, while MAE still behaves to be rudimentary.

Downstream Tasks

Semantic segmentation. We evaluate the learned representation of DDAE on the ADE20K benchmark (Zhou et al. 2019), which consists of 25K images and 150 semantic categories. We use the UperNet (Xiao et al. 2018) task layer for semantic segmentation. We train Upernet 160K iterations with single-scale inference. The results are reported in Table 6 with mean Intersection of Union (mIoU) as the evaluation metric. Results exhibit that our DDAE performs better than other methods.

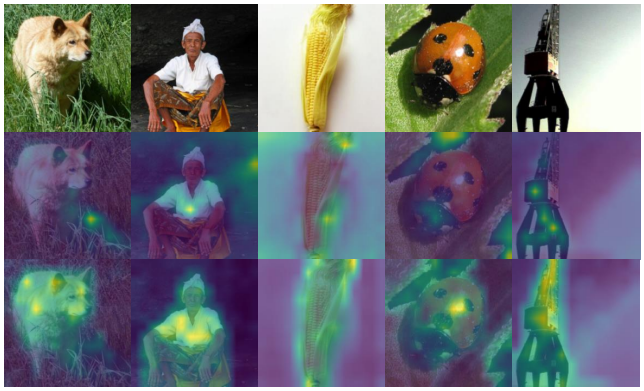


Figure 6: Visualization of attention map. The attention map averaged over 12 attention heads between the class token and the patch tokens in the shallow layer (Block 3) of the ViT-B encoder pretrained on ImageNet-1K. Top: Input image, Middle: MAE, and Bottom: our DDAE.

Methods	Epochs	mIoU	AP^b	AP^m
BEiT	800	47.1	46.3	41.1
MAE	1600	48.1	48.4	42.6
SimMIM	800	48.5	49.3	43.1
SdAE	800	49.0	49.7	43.3
DDAE(ours)	800	**49.6**	**50.1**	**43.8**

Table 6: Semantic segmentation comparison on ADE20K and object detection/instance segmentation comparison on COCO. The same ViT-B is used as backbone.

Object Detection. Following SdAE (Chen et al. 2022b), We adopt the Mask R-CNN (He et al. 2017) framework to perform fine-tuning on COCO (Lin et al. 2014) in an end-to-end manner. The fine-tuning is conducted with $1\times$ (12 training epochs) schedule and the same ViT-B is used as backbone. We report the box AP for object detection and the mask AP for instance segmentation. Our DDAE consistently outperforms other models, further demonstrating our superiority on downstream tasks.

Conclusion

In this paper, we delve into dynamic MIM. We propose a novel deep dynamic supervision to facilitate dynamic focus on different patches through pretraining, providing progressively richer representation. Built upon DDS, we propose a simple yet effective framework DDAE. Our experiments demonstrate that DDAE produces consistent improvements over strong baselines. Moreover, we empirically find our approach can inject more local priors for ViTs in deep layers, which we believe crucial for the high performance of DDAE.

Acknowledgments

This work is supported in part by the National Key R&D Program of China (Grant No.2022ZD0116403), the National Natural Science Foundation of China (Grant No. 61721004), the Strategic Priority Research Program of Chinese Academy of Sciences (Grant No. XDA27000000), and the Youth Innovation Promotion Association CAS.

References

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

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

Caron, M.; Touvron, H.; Misra, I.; Jégou, H.; Mairal, J.; Bojanowski, P.; and Joulin, A. 2021. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 9650–9660.

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

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

Chen, Y.; Liu, Y.; Jiang, D.; Zhang, X.; Dai, W.; Xiong, H.; and Tian, Q. 2022b. SdAE: Self-distillated Masked Autoencoder. *arXiv preprint arXiv:2208.00449*.

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

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

Dong, X.; Bao, J.; Zhang, T.; Chen, D.; Zhang, W.; Yuan, L.; Chen, D.; Wen, F.; and Yu, N. 2022. Bootstrapped Masked Autoencoders for Vision BERT Pretraining. *arXiv preprint arXiv:2207.07116*.

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

Grill, J.-B.; Strub, F.; Altché, F.; Tallec, C.; Richemond, P.; Buchatskaya, E.; Doersch, C.; Avila Pires, B.; Guo, Z.; Gheshlaghi Azar, M.; et al. 2020. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, 33: 21271–21284.

He, K.; Chen, X.; Xie, S.; Li, Y.; Dollár, P.; and Girshick, R. 2022. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 16000–16009.

He, K.; Fan, H.; Wu, Y.; Xie, S.; and Girshick, R. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 9729–9738.

He, K.; Gkioxari, G.; Dollár, P.; and Girshick, R. 2017. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, 2961–2969.

Huang, G.; Chen, D.; Li, T.; Wu, F.; Van Der Maaten, L.; and Weinberger, K. Q. 2017. Multi-scale dense networks for resource efficient image classification. *arXiv preprint arXiv:1703.09844*.

Lee, C.-Y.; Xie, S.; Gallagher, P.; Zhang, Z.; and Tu, Z. 2015. Deeply-supervised nets. In *Artificial intelligence and statistics*, 562–570. PMLR.

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, 740–755. Springer.

Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; and Guo, B. 2021. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 10012–10022.

Mosinska, A.; Marquez-Neila, P.; Koziński, M.; and Fua, P. 2018. Beyond the pixel-wise loss for topology-aware delineation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3136–3145.

Newell, A.; Yang, K.; and Deng, J. 2016. Stacked hourglass networks for human pose estimation. In *European conference on computer vision*, 483–499. Springer.

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

Peng, Z.; Dong, L.; Bao, H.; Ye, Q.; and Wei, F. 2022a. BEiT v2: Masked Image Modeling with Vector-Quantized Visual Tokenizers. *arXiv preprint arXiv:2208.06366*.

Peng, Z.; Dong, L.; Bao, H.; Ye, Q.; and Wei, F. 2022b. BEiT v2: Masked Image Modeling with Vector-Quantized Visual Tokenizers. *arXiv preprint arXiv:2208.06366*.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, 8748–8763. PMLR.

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

Ren, S.; Wei, F.; Albanie, S.; Zhang, Z.; and Hu, H. 2023. DeepMIM: Deep Supervision for Masked Image Modeling. *arXiv preprint arXiv:2303.08817*.

Sun, D.; Yao, A.; Zhou, A.; and Zhao, H. 2019. Deeply-supervised knowledge synergy. In *Proceedings of the*

IEEE/CVF Conference on Computer Vision and Pattern Recognition, 6997–7006.

Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; and Rabinovich, A. 2015. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1–9.

Wang, Y.; Tao, X.; Qi, X.; Shen, X.; and Jia, J. 2018. Image inpainting via generative multi-column convolutional neural networks. *Advances in neural information processing systems*, 31.

Wei, C.; Fan, H.; Xie, S.; Wu, C.-Y.; Yuille, A.; and Feichtenhofer, C. 2022a. Masked feature prediction for self-supervised visual pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14668–14678.

Wei, L.; Xie, L.; Zhou, W.; Li, H.; and Tian, Q. 2022b. MVP: Multimodality-guided Visual Pre-training. *arXiv preprint arXiv:2203.05175*.

Xiao, T.; Liu, Y.; Zhou, B.; Jiang, Y.; and Sun, J. 2018. Unified perceptual parsing for scene understanding. In *Proceedings of the European conference on computer vision (ECCV)*, 418–434.

Xie, S.; and Tu, Z. 2015. Holistically-nested edge detection. In *Proceedings of the IEEE international conference on computer vision*, 1395–1403.

Xie, Z.; Zhang, Z.; Cao, Y.; Lin, Y.; Bao, J.; Yao, Z.; Dai, Q.; and Hu, H. 2022. Simmim: A simple framework for masked image modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9653–9663.

Yeh, R.; Chen, C.; Lim, T. Y.; Hasegawa-Johnson, M.; and Do, M. N. 2016. Semantic image inpainting with perceptual and contextual losses. *arXiv preprint arXiv:1607.07539*, 2(3).

Yu, J.; Lin, Z.; Yang, J.; Shen, X.; Lu, X.; and Huang, T. S. 2018. Generative image inpainting with contextual attention. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 5505–5514.

Zhang, H.; Wu, H.; Sun, W.; and Zheng, B. 2018a. Deeptravel: a neural network based travel time estimation model with auxiliary supervision. *arXiv preprint arXiv:1802.02147*.

Zhang, L.; Chen, X.; Zhang, J.; Dong, R.; and Ma, K. 2022. Contrastive Deep Supervision. *arXiv preprint arXiv:2207.05306*.

Zhang, L.; Song, J.; Gao, A.; Chen, J.; Bao, C.; and Ma, K. 2019a. Be Your Own Teacher: Improve the Performance of Convolutional Neural Networks via Self Distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

Zhang, M.; Jiang, S.; Cui, Z.; Garnett, R.; and Chen, Y. 2019b. D-vae: A variational autoencoder for directed acyclic graphs. *Advances in Neural Information Processing Systems*, 32.

Zhang, Z.; Zhang, X.; Peng, C.; Xue, X.; and Sun, J. 2018b. Exfuse: Enhancing feature fusion for semantic segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, 269–284.

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