Towards Sustainable Self-supervised Learning

Shanghua Gao[1,2*] **Pan Zhou**[1] **Ming-Ming Cheng**[2] **Shuicheng Yan**[1]
[1]Sea AI Lab [2]Nankai University
{gaoshanghua,zhoupan,shuicheng.yan}@sea.com, cmm@nankai.edu.cn

Abstract

Though increasingly training-expensive, most self-supervised learning (SSL) models have repeatedly been trained from scratch but not fully utilized since only a few SOTAs are adopted for downstream tasks. In this work, we explore a sustainable SSL framework with two major challenges: i) learning a stronger new SSL model based on the existing pretrained SSL model in a cost-friendly manner, ii) allowing the training of the new model to be compatible with various base models. We propose a Target-Enhanced Conditional (TEC) scheme, which introduces two components to existing mask-reconstruction based SSL. Firstly, we introduce patch-relation enhanced targets to encourage the new model to learn semantic-relation knowledge from the base model using incomplete inputs. This hardening and target-enhancing could help the new model surpass the base model, since they enforce additional patch relation modeling to handle incomplete input. Secondly, we introduce a conditional adapter that adaptively adjusts new model prediction to align with the target of each base model. Experimental results show that our TEC scheme can accelerate the learning speed and also improve SOTA SSL models, *e.g.,* MAE and iBOT, taking an explorative step towards sustainable SSL.

1 Introduction

Self-supervised learning (SSL) has achieved overwhelming success in unsupervised representation learning, with astonishing high performance in many downstream tasks like classification [50, 51], object detection and segmentation [2, 19]. In SSL, a pretext task is first built, *e.g.,* instance discrimination task [20, 8] or masked image modeling (MIM) [2, 19], and then pseudo labels are generated via the pretext task to train a network model without requiring manual labels. Though successful, SSL is developing towards a direction of causing increasingly large training costs, *e.g.,* MoCo [20] trained with 200 epochs while MAE [19] with 16,00 epochs to release its potential. However, in most cases researchers only have limited computational budgets and often cannot afford to train large SSL models. Moreover, the pretrained non-SOTA SSL models are rarely used in practice, since SOTA is updated frequently and a previous one quickly becomes useless, wasting huge training resources. Thus, a **sustainable SSL** framework is much demanded.

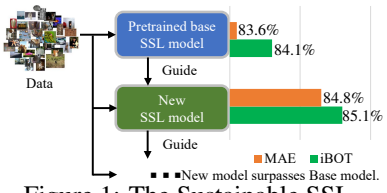

Figure 1: The Sustainable SSL.

Just like how human experience is enriched and passed from one generation to the next in human society, we try to make an SSL model absorb knowledge from the pretrained SSL model to achieve superior representation quality, so as to be reusable or "sustainable". In this way, huge training resources would be saved compared to training a new SSL model from scratch. This process is somewhat like Knowledge Distillation (KD) [21, 18]. But instead of aiming at compressing knowledge from a powerful teacher model to a compact new model in traditional KD, usually with declined performance, the sustainable SSL is targeted at a new model that learns from the base model and

*This work was done while Shanghua Gao was a research intern at Sea AI Lab.

becomes more powerful than the base model. An illustration of sustainable SSL is given in Fig. 1. For better understanding, we call the new SSL model to be trained as the new model and the pretrained SSL model as the base model. To surpass the base model, in sustainable SSL, the new model exploits not only the implicit base model knowledge but also the absent knowledge in the base model. Such a learning process follows a fully self-supervised manner, and differs from the self-training schemes [45, 47] that requires labels for supervised learning.

This work takes an explorative step towards sustainable SSL by learning from existing pretrained SSL models and surpassing them in an efficient manner. Intuitively, to achieve this challenging goal, the new model should learn not only knowledge of the base model but also more semantic-related new knowledge so as to beat the base model. We therefore choose a mask-reconstruction [19] SSL scheme for the pretraining of the new model, in which the base model generates reconstruction targets from the full input images and the new model tries to predict base model targets from random masked image input. With this pretext task, the new model is forced to learn the semantic of the full input and also its patch relations so that new model can reason the desired full information from an absent input. In Fig. 2, the attention of iBOT [50] misses many semantics, *e.g.,* ears, while TEC with iBOT as base model captures all semantics and also well distinguish all different components of an input image. Because of its almost comprehensive semantic capturing ability, TEC can flexibly select related semantics for a downstream task.

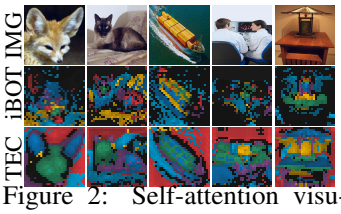

Figure 2: Self-attention visualization. Different colors denote attentions of different heads. Black means no attention.

However, different SSL base models could have various properties due to their various targets and training strategies, *e.g.,* iBOT models with more category semantics while MAE models with more image details [19]. So it is important to build high-qualified and compatible reconstruction targets from the base model so that new model learns these targets in a complementary manner. A good model target should reveal the semantic relations among patches, *e.g.,* relation between car wheels and car body, so that new model can learn this general relation patterns and adapts to downstream tasks. To this end, we propose to enhance the target quality of the base models by using two complementary reconstruction targets: a) the patch-dim normalization which normalizes base model targets along patch dimension to enhance the relations among input patches, and b) patch attention maps with rich semantics to filter out possible background noise and thus to establish the correlation between the whole input semantic and the patch semantic. For target compatibility, we introduce conditional adapters into new model so that new model prediction can be adaptable to various base models with different properties. Given a base model target, the adapters conditionally active and adjust the new model mid-level features to predict the target more effectively. These adapters are discarded after pretraining, but can serve parameter-efficient finetuning [24, 6] if kept.

We call the above method for sustainable SSL as Target-Enhanced Conditional (TEC) scheme. on ImageNet, TEC without any extra training data improves the SSL base model by a remarkable margin, *e.g.,* MAE [19] and iBOT [50]. Moreover, we also find that TEC can significantly accelerate the SSL learning process and saves training cost. We hope our initial effort towards sustainable SSL will inspire more works in the future to sustainably improve SSL in a cost-friendly manner.

2 Method

2.1 Overall framework

An overall framework of the proposed target-enhanced conditional (TEC) mask-reconstruction method is illustrated in Fig. 3. TEC follows [19, 2], and uses Vision Transformer (ViT) [14] for implementation. Under the mask-reconstruction [19] framework, TEC consists of a randomly initialized new ViT encoder to be pretrained, conditional adapters for conditional pretraining, and a multi-target decoder for reconstruction targets prediction, an SSL pretrained ViT encoder as the base model and an target-enhancing module to generate patch-relation enhanced reconstruction base model targets. Specifically, base model ViT encoder is a SSL pretrained encoder (*e.g.,* in MAE [19]), and is used to generate latent representation of a full image. Then target-enhancing module enhances the latent representation to generate two complementary reconstruction targets for new model. The new ViT encoder together with adapters take in masked image and generate adapted latent representation which is then fed into the multi-target decoder to predict the base model targets. After pretraining, the new ViT encoder is kept for downstream tasks while other parts are removed. At below, we explain the conditional pretraining aided by adapters in Section 2.2 to help new model effectively predict

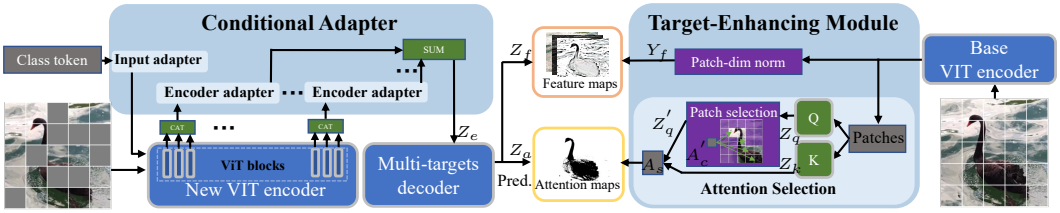

Figure 3: The overall framework of the proposed TEC.

base model targets, and then introduce the target-enhancing module to generate high-qualified base model targets in Section 2.3.

2.2 Conditional Pretraining

As aforementioned, different base models often have different properties, *e.g.,* more category semantic in iBOT while more local semantic in MAE. So the prediction of the new model should be compatible to any given base model. To resolve a similar issue on RGB image reconstruction, the works [37, 13, 16] manually select certain features from the mid-level layers of the encoder to better align with the RGB image target. However, it is almost impossible to manually select certain encoder features which are compatible to different base models with various properties. Therefore, to improve the learning performance, the new model should have the conditional adaptation ability regarding a given SSL base model.

Given a fixed pretrained model, the parameter-efficient fine-tuning scheme introduces trainable extra modules with a small number of parameters into this pretrained model for adapting it to downstream tasks in both vision [24, 6] and NLP [22, 27, 33] domains. For example, the prompting scheme [27, 33, 24] concatenates learnable input tokens, *e.g.,* class token, with patch tokens to activate certain semantic features of a fixed ViT model that are suitable for specific downstream task. Also, inserting lightweight adapter modules (*e.g.,* MLP [22, 6] and residual blocks [28]) into a fixed model can activate and modulate mid-level features of the model to predict features required by the downstream task. Inspired by these parameter-efficient fine-tuning schemes, we apply the adaptation scheme in the pretraining stage by introducing conditional adapters into the new model to handle the diversities of base models. Our adapters are only used for pretraining and will be removed during finetuning. So they do not increase extra inference cost. Actually, Tab. 8 shows that keeping these adapters in the inference phase would enhance the parameter-efficient finetuning ability of the model. We now introduce how to apply adapters, *i.e.,* input and encoder adapters, into the new model.

Input adapter. For ViT networks, one often concatenate a class token with the input patch tokens to learn the global semantic of the whole input. Since the prompting scheme shows the adaption ability of the class token, we propose to further enhance the feature adaption ability of class token by adding an input adapter. The input adapter, composed of a small two-layer MLP layer, as shown in Fig. 4, enhances the representation ability the class token so that the class token can better activate features in the new model according to the base model targets supervision. For inference, since input adapter is shared by all input samples, one can compute it in advance to save inference cost. The implementation details of input adapter is shown in Appendix.

Encoder adapter. To modulate mid-level features in the new model so that it can adapt to the base model targets, we apply a simple MLP with residual connection [6] as our encoder adapter in the pretraining phase. As we want to remove adapters after pretraining to save inference cost, we need to keep the encoder network topology unchanged after removing adapters. So we put the input of adapters in the middle of the encoder and merge all adapter outputs at the end of the encoder. The adapted features are then sent to the multi-target decoder to predict base model targets, which will be introduced in Section 2.3. The implementation details of encoder adapter is shown in Appendix.

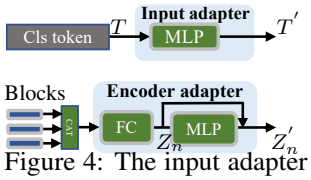

Figure 4: The input adapter and encoder adapter.

2.3 Patch-relation enhanced Reconstruction Targets

To better exploit the knowledge of base models in sustainable SSL, in the target-enhancing module, we construct two complementary targets with enhanced patch-relations: 1) feature-level targets with patch dimension normalization to strengthen the relations between patches; 2) semantic-related attention maps to learn relations between semantic-related patches and other patches. The feature targets reveal semantics of certain patches, while attention maps focus more on feature relations. The multi-target deocder for predicting these targets is introduced in Appendix.

Table 1: Semantic segmentation on ADE20k using Upernet and ViT-B.

Method	Epoch	mIoU
BEiT [2]	800	47.1
PeCo [12]	800	48.5
GE2-AE [31]	800	48.9
CAE [7]	1600	50.2
CMAE [23]	1600	50.1
MAE [19]	1600	48.1
TEC$_{\mathrm{MAE}}$	800	**49.9**
iBOT	1600	50.0
TEC$_{\mathrm{iBOT}}$	800	**51.0**

Table 2: Comparison with existing SSL methods under ImageNet-1k fully finetuning using ViT.

Model	Method	Epoch	Guidance	Top1 acc.
ViT-Base	MAE [19]	1600	RGB	83.6
	TEC	300	MAE	**84.7**$_{+1.1}$
	TEC	800	MAE	**84.8**$_{+1.2}$
	iBOT-ImageNet-22K	-	Momentum	84.4
	iBOT [50]	1600	Momentum	84.1
	TEC	300	iBOT	**84.8**$_{+0.7}$
	TEC	800	iBOT	**85.1**$_{+1.0}$
ViT-Large	MAE [19]	1600	RGB	85.9
	TEC	300	MAE	**86.5**$_{+0.6}$

Table 3: Parameter-efficient finetuning on ImageNet-1k.

Method	Epoch	Settings	Top 1 acc.
MAE	1600	Linear probing	68.0
TEC$_{\mathrm{MAE}}$	800	Linear probing	69.8
		+Input adapter FT	72.6
		+Encoder adapter FT	79.9

Table 4: TEC accelerates MAE training.

	Guide	Epoch	Top1 acc.
MAE300ep		300	82.9
MAE1600ep		1600	83.6
TEC	MAE300ep	**100**	**83.9**$_{+1.0}$
TEC	MAE300ep	300	**84.3**$_{+1.4}$
TEC	MAE1600ep	300	**84.7**$_{+1.1}$

Patch-dim normalized feature-level targets. Given a base model, we propose to normalize its target along the patch dimension to enhance the spatial patch-relations. The implementation is shown in Appendix. For MIM, this patch-dim normalization can better enhance the spatial relations among tokens than the widely used feature normalization [41, 39, 1] along channel dimension. This is because the base model features might have large variances, *e.g.,* a few values are much larger than other values. Normalization along channel dimension would lead to a degenerated case that only a few feature values are large while other features have very small values. Thus, patches within one channel might have very similar values, impair their spatial relations. In contrast, patch-dim normalization would relieve this issue, and still enhance the inherent spatial relations among patches.

Semantic attention-level targets. Self-attention in pretrained ViT models has a powerful capability of capturing semantic relations among patch tokens [4, 29]. We then propose to utilize the self-attention maps as a type of reconstruction targets for MIM to further enhance the semantic relation modeling capability of the new model. According to previous investigations on effects of self-attention maps in KD [42, 38], not all attention maps contain useful semantic relations, and severe noisy attentions even hinder student learning. Accordingly, it is necessary to select parts of attention maps for reducing the possible severe noise and also helping reduce training cost. Here we utilize the base model class token which contains sufficient global semantics to select the attention maps of top similar patch tokens which filters out the possible noises. The implementation details is shown in Appendix.

3 Experiments

We evaluate our TEC on ImageNet-1k [10] with ViT-B/L-16×16 [14] and 224×224 image resolution.

Fully finetuning on ImageNet-1k. We compare the fully finetuning performance of ViT-B on ImageNet-1k in Tab. 2. It is observed that our TEC outperforms the iBOT/MAE base models with clear gaps when trained from random initialization. To the best of our knowledge, TEC achieves new SOTA on ViT-B when solely using ImageNet-1k training data, showing the potential of sustainable SSL learning. We also verify the scaling ability of our TEC by using the ViT-L model, and find that TEC surpasses the MAE base model by 0.6% with 300 epochs pretraining from random initialization.

Parameter-efficient finetuning on ImageNet-1k. We test effectiveness of our TEC with parameter-efficient finetuning for classification on ImageNet-1k using ViT-B, with results shown in Tab. 3. Under the linear probing setting, TEC outperforms the MAE base model with 1.8%.Finetuning with input and encoder adapters for pretraining brings considerable gains of 4.6%/11.9% over MAE. Therefore, adapters used for pretraining can greatly benefit parameter-efficient finetuning.

Accelerating the training of base models. We use a 300-epoch unconverged MAE pretrained ViT-B model as the base model and train TEC with 100/300 epochs from random initialization. Tab. 4 shows TEC achieves 84.3%/83.9% with 300/100 epochs surpassing the 300-epoch MAE base models with 1.4%/1.0%. Notably, TEC even outperforms the 1600-epoch pretrained MAE by 0.3% with only 100 epoch training, showing TEC can significantly accelerate the training of the base model.

Transfer learning on semantic segmentation. We show the ADE20k [49] results using Upernet [43] with ViT-B on Tab. 1. TEC surpasses the iBOT/MAE base models with clear gain, showing greater transfer learning abilities on semantic segmentation compared to their base models. TEC shows clear advantages over strong competitors with fewer pretraining epochs, achieving new SOTA.

More results. We show more results and analysis in the Appendix.

References

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

[2] Hangbo Bao, Li Dong, and Furu Wei. Beit: Bert pre-training of image transformers. *arXiv preprint arXiv:2106.08254*, 2021.

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

[4] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *IEEE International Conference on Computer Vision (ICCV)*, 2021.

[5] Jun Chen, Ming Hu, Boyang Li, and Mohamed Elhoseiny. Efficient self-supervised vision pretraining with local masked reconstruction. *arXiv preprint arXiv:2206.00790*, 2022.

[6] Shoufa Chen, Chongjian Ge, Zhan Tong, Jiangliu Wang, Yibing Song, Jue Wang, and Ping Luo. Adaptformer: Adapting vision transformers for scalable visual recognition. *arXiv preprint arXiv:2205.13535*, 2022.

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

[8] Xinlei Chen*, Saining Xie*, and Kaiming He. An empirical study of training self-supervised vision transformers. *arXiv preprint arXiv:2104.02057*, 2021.

[9] MMSegmentation Contributors. MMSegmentation: Openmmlab semantic segmentation toolbox and benchmark. `https://github.com/open-mmlab/mmsegmentation`, 2020.

[10] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 248–255. Ieee, 2009.

[11] Xiaohan Ding, Xiangyu Zhang, Ningning Ma, Jungong Han, Guiguang Ding, and Jian Sun. Repvgg: Making vgg-style convnets great again. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13733–13742, 2021.

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

[13] Xiaoyi Dong, Jianmin Bao, Ting Zhang, Dongdong Chen, Weiming Zhang, Lu Yuan, Dong Chen, Fang Wen, and Nenghai Yu. Bootstrapped masked autoencoders for vision bert pretraining. In *European Conference on Computer Vision (ECCV)*, 2022.

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

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

[16] Peng Gao, Teli Ma, Hongsheng Li, Jifeng Dai, and Yu Qiao. Convmae: Masked convolution meets masked autoencoders. *arXiv preprint arXiv:2205.03892*, 2022.

[17] Shanghua Gao, Zhong-Yu Li, Ming-Hsuan Yang, Ming-Ming Cheng, Junwei Han, and Philip Torr. Large-scale unsupervised semantic segmentation. *arXiv preprint arXiv:2106.03149*, 2021.

[18] Jianping Gou, Baosheng Yu, Stephen J Maybank, and Dacheng Tao. Knowledge distillation: A survey. *International Journal of Computer Vision (IJCV)*, 129(6):1789–1819, 2021.

[19] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16000–16009, 2022.

[20] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[21] Geoffrey Hinton, Oriol Vinyals, Jeff Dean, et al. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2(7), 2015.

[22] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *International Conference on Machine Learning (ICML)*, pages 2790–2799. PMLR, 2019.

[23] Zhicheng Huang, Xiaojie Jin, Chengze Lu, Qibin Hou, Ming-Ming Cheng, Dongmei Fu, Xiaohui Shen, and Jiashi Feng. Contrastive masked autoencoders are stronger vision learners. *arXiv preprint arXiv:2207.13532*, 2022.

[24] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. *arXiv preprint arXiv:2203.12119*, 2022.

[25] Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *International Conference on Machine Learning (ICML)*, pages 3519–3529. PMLR, 2019.

[26] Gang Li, Heliang Zheng, Daqing Liu, Bing Su, and Changwen Zheng. Semmae: Semantic-guided masking for learning masked autoencoders. *arXiv preprint arXiv:2206.10207*, 2022.

[27] Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*, 2021.

[28] Yanghao Li, Hanzi Mao, Ross Girshick, and Kaiming He. Exploring plain vision transformer backbones for object detection. *arXiv preprint arXiv:2203.16527*, 2022.

[29] Zhong-Yu Li, Shanghua Gao, and Ming-Ming Cheng. Exploring feature self-relation for self-supervised transformer. *arXiv preprint arXiv:2206.05184*, 2022.

[30] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision (ECCV)*, pages 740–755, 2014.

[31] Hao Liu, Xinghua Jiang, Xin Li, Antai Guo, Deqiang Jiang, and Bo Ren. The devil is in the frequency: Geminated gestalt autoencoder for self-supervised visual pre-training. *arXiv preprint arXiv:2204.08227*, 2022.

[32] Jihao Liu, Xin Huang, Yu Liu, and Hongsheng Li. Mixmim: Mixed and masked image modeling for efficient visual representation learning. *arXiv preprint arXiv:2205.13137*, 2022.

[33] Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. Gpt understands, too. *arXiv preprint arXiv:2103.10385*, 2021.

[34] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[35] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International Conference on Machine Learning (ICML)*, pages 8821–8831. PMLR, 2021.

[36] Chenxin Tao, Xizhou Zhu, Gao Huang, Yu Qiao, Xiaogang Wang, and Jifeng Dai. Siamese image modeling for self-supervised vision representation learning. *arXiv preprint arXiv:2206.01204*, 2022.

[37] Luya Wang, Feng Liang, Yangguang Li, Wanli Ouyang, Honggang Zhang, and Jing Shao. Repre: Improving self-supervised vision transformer with reconstructive pre-training. *arXiv preprint arXiv:2201.06857*, 2022.

[38] Shaoru Wang, Jin Gao, Zeming Li, Jian Sun, and Weiming Hu. A closer look at self-supervised lightweight vision transformers. *arXiv preprint arXiv:2205.14443*, 2022.

[39] Chen Wei, Haoqi Fan, Saining Xie, Chao-Yuan Wu, Alan Yuille, and Christoph Feichtenhofer. Masked feature prediction for self-supervised visual pre-training. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14668–14678, 2022.

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

[41] Yixuan Wei, Han Hu, Zhenda Xie, Zheng Zhang, Yue Cao, Jianmin Bao, Dong Chen, and Baining Guo. Contrastive learning rivals masked image modeling in fine-tuning via feature distillation. *arXiv preprint arXiv:2205.14141*, 2022.

[42] Haiyan Wu, Yuting Gao, Yinqi Zhang, Shaohui Lin, Yuan Xie, Xing Sun, and Ke Li. Self-supervised models are good teaching assistants for vision transformers. In *International Conference on Machine Learning (ICML)*, pages 24031–24042. PMLR, 2022.

[43] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *European Conference on Computer Vision (ECCV)*, pages 418–434, 2018.

[44] Jiahao Xie, Wei Li, Xiaohang Zhan, Ziwei Liu, Yew Soon Ong, and Chen Change Loy. Masked frequency modeling for self-supervised visual pre-training. *arXiv preprint arXiv:2206.07706*, 2022.

[45] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. Self-training with noisy student improves imagenet classification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10687–10698, 2020.

[46] Zhenda Xie, Zheng Zhang, Yue Cao, Yutong Lin, Jianmin Bao, Zhuliang Yao, Qi Dai, and Han Hu. Simmim: A simple framework for masked image modeling. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9653–9663, 2022.

[47] I Zeki Yalniz, Hervé Jégou, Kan Chen, Manohar Paluri, and Dhruv Mahajan. Billion-scale semi-supervised learning for image classification. *arXiv preprint arXiv:1905.00546*, 2019.

[48] Kun Yi, Yixiao Ge, Xiaotong Li, Shusheng Yang, Dian Li, Jianping Wu, Ying Shan, and Xiaohu Qie. Masked image modeling with denoising contrast. *arXiv preprint arXiv:2205.09616*, 2022.

[49] Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ade20k dataset. *International Journal of Computer Vision (IJCV)*, 2018.

[50] Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. ibot: Image bert pre-training with online tokenizer. *International Conference on Learning Representations (ICLR)*, 2022.

[51] Pan Zhou, Yichen Zhou, Chenyang Si, Weihao Yu, Teck Khim Ng, and Shuicheng Yan. Mugs: A multi-granular self-supervised learning framework. In *arXiv preprint arXiv:2203.14415*, 2022.

A Method Implementation Details

Implementation of input encoder. The class token $T \in \mathbb{R}^C$ of the ViT is processed by the MLP layer to obtain an enhanced class token $T^{'} \in \mathbb{R}^C$:

$$T^{'} = \mathbf{MLP}(T),$$

where C is the embedding dimension. During pretraining, $\mathbf{MLP}(T)$ is appended to the patch tokens. $\mathbf{MLP}$ enhances the representation ability of T and enables the new model to better predict base model targets with stronger adaptability. $\mathbf{MLP}(T)$ is merged into a class token $T^{'}$ by applying the reparameterization trick [11] for inference without extra inference cost.

Implementation of encoder adapter. Given features $X = \{X_i, i = 1, \ldots, D\}$ from each block of the new model encoder where D is the number of encoder blocks, we first uniformly divide them into N groups, in which each group contains 3 blocks by default. Within the n_{th} group, we merge features from adjacent blocks to obtain feature Z_n which can save computational cost:

$$Z_n = \mathbf{FC}(\mathbf{Concat}(X_i, ..., X_j)). \tag{1}$$

Then we further feed the feature Z_n into an adapter and obtain feature Z_e:

$$Z_n^{'} = Z_n + \mathbf{MLP}(Z_n), \qquad Z_e = \sum\nolimits_{n=1}^{N} Z_n^{'}, \tag{2}$$

where is $\mathbf{MLP}$ is small MLP with 2 layers.

Implementation of patch-dim normalized feature-level targets. Specifically, for an input, assume its base model target is $Y \in \mathbb{R}^{L \times C}$ where L and C respectively denote patch number and channel dimension C. Then we normalize Y along patch dimension:

$$Y_f = (Y - \mu_L)/\sigma_L, \tag{3}$$

where μ_L and σ_L are mean and variance along the patch dimension. Indeed, Tab. 11(a) shows that such simple modification significantly improves the new model performance. After normalization, following [19], new model uses an fully-connected layer followed by the decoder to generate Z_f for predicting the base model target Y_f on masked patches:

$$L_{\text{fea}} = \|M \circ (Y_f - Z_f)\|_2^2, \tag{4}$$

where M is the mask matrix and $\circ$ denotes the element-wise product.

Implementation of semantic attention-level targets. Given the attention maps $A_c \in \mathbb{R}^{H \times L}$ between class token and patch tokens from the last ViT block in base model where L and H respectively denote patch number and head number, we average the attention map A_c along head dimension to obtain $A_c^{'} \in \mathbb{R}^{1 \times L}$. Then, we select top-$k$ patches with the largest values in $A_c^{'}$, and further compute the attention map $A_p \in \mathbb{R}^{H \times k \times L}$ among the top-k patches and all patch tokens. Consider the importance of class token, we further concatenate attention maps between itself and selected A_p to obtain our final reconstruction targets, *i.e.,* $A_s \in \mathbb{R}^{H \times (k+1) \times L}$. Note, when we compute A_s, a temperature τ is added before the Softmax operation to adjust the attention sharpness. For the new model, we respectively use two fully-connected layers to project its decoder output into two predictions $Z_q \in \mathbb{R}^{L \times C}$ and $Z_k \in \mathbb{R}^{L \times C}$. We select the same patches as in A_s from Z_q to form $Z_q^{'} \in \mathbb{R}^{(k+1) \times C}$. Then we concatenate the class token cls in new model with $Z_q^{'}$ and compute the KQ attention map $Z_a = \text{Softmax}([Z_q^{'}, \mathsf{cls}]^{\top} Z_k) \in \mathbb{R}^{H \times (k+1) \times L}$. Finally, we compute the prediction loss between the prediction Z_a and the teacher target A_s via the entropy loss:

$$L_{\text{att}} = -A_s \log Z_a. \tag{5}$$

Implementation of multi-target decoder. Due to the different property of the two reconstruction targets, namely feature target and attention target, one decoder in new model for prediction is not sufficient to handle them at the same time and often tends to results in prediction conflict. But using separate decoders for each target would increase the trainable parameters and thus training cost. To solve this problem, we use a simple decoder adaptation scheme which constructs target-specific inputs and then feed them into a shared decoder. Specifically, we feed the output feature Z_e (see Eqn. 2) of new model encoder into a fully-connected layer and then fill the masked tokens by a learnable mask token to obtain $Z_f^{'}$. Then similarly, given Z_e, we can also use a fully-connected layer and a learnable mask token to obtain $Z_m^{'}$. Next, we respectively feed $Z_f^{'}$ and $Z_m^{'}$ into a shared transformer-based decoder for predicting the feature and attention map targets of base model. Unlike the large semantic gap between encoder output and vanilla image in MAE, the base model target have similar semantics to the new model predictions. So a shallow 2-layer decoder is enough and is indeed better than the 8-layer decoder used in MAE. This also greatly reduces the training cost.

Table 5: Comparison with existing SSL methods under ImageNet-1k fully finetuning using ViT. †and gray color mean trained with implicit /explicit extra data.

Model	Method	Epoch	Guidance	Top1 acc.
ViT-Base	DINO [4]	300	NA	82.8
	MoCov3 [8]	300	NA	83.2
	MixMIM [32]	300	RGB	83.2
	MFM [44]	300	Frequency	83.1
	BEiT [2]	800	DALLE†	83.2
	SplitMask [15]	300	NA	83.6
	ConMIM [48]	800	Momentum	83.7
	SimMIM [46]	800	RGB	83.8
	SIM [36]	1600	Momentum	83.8
	CAE [7]	1600	DALLE†	83.9
	MaskFeat [39]	1600	HOG	84.0
	LoMaR [5]	1600	RGB	84.1
	BootMAE [13]	800	RGB+Momentum	84.2
	data2vec [1]	800	Momentum	84.2
	Mugs [51]	1600	NA	84.3
	MVP [40]	300	CLIP†	84.4
	PeCo [12]	800	Perceptual codebook	84.5
	CMAE [23]	1600	RGB	84.7
	Ge2-AE [31]	800	RGB+Frequency	84.8
	FD-CLIP [41]	300	CLIP†	84.9
	MAE [19]	1600	RGB	83.6
	FD-MAE [41]	300	MAE	$83.8_{+0.2}$
	TEC	300	MAE	$\mathbf{84.7}_{+1.1}$
	TEC	800	MAE	$\mathbf{84.8}_{+1.2}$
	iBOT-ImageNet-22K	-	Momentum	84.4
	iBOT [50]	1600	Momentum	84.1
	SemMAE [26]	800	iBOT	$84.5_{+0.4}$
	TEC	300	iBOT	$\mathbf{84.8}_{+0.7}$
	TEC	800	iBOT	$\mathbf{85.1}_{+1.0}$
ViT-Large	MAE [19]	1600	RGB	85.9
	TEC	300	MAE	$\mathbf{86.5}_{+0.6}$

Table 6: Semantic segmentation on ADE20k using Upernet and ViT-B.

Method	Epoch	mIoU
BEiT	800	47.1
PeCo	800	48.5
GE2-AE	800	48.9
CAE	1600	50.2
CMAE	1600	50.1
MAE	1600	48.1
$\mathbf{TEC}_{\mathrm{MAE}}$	800	**49.9**
iBOT	1600	50.0
$\mathbf{TEC}_{\mathrm{iBOT}}$	800	**51.0**

Table 7: Instance segmentation on COCO using Cascade MaskRCNN and ViT-B.

Method	$\mathrm{AP}_{\mathrm{bbox}}$	$\mathrm{AP}_{\mathrm{mask}}$
Implementation from [50]		
iBOT	51.2	44.2
$\mathbf{TEC}_{\mathrm{iBOT}}$	**52.7**	**45.4**
Implementation from [28]		
MAE	54.0	46.7
$\mathbf{TEC}_{\mathrm{MAE}}$	**54.4**	**47.1**

B More Experiments

We evaluate our TEC on ImageNet-1k [10] and use it to train ViT [14] with a 16×16 patch size and 224×224 image resolution. For all experiments, we randomly initialize the models and pretrain for 300/800 epochs via AdamW [34] of 4,096 batchsize [19]. To make sure the gain is from our sustainable SSL method instead of extra data or stronger base model , explicit/implicit extra training data or stronger base models than new models [35] are not used. So we use iBOT [50] and MAE [19] pretrained ViT models on ImageNet-1k as our base models. We apply the same masking strategy as in MAE, *e.g.,* 75% masked ratio. See more training details in Section C.

B.1 Performance comparison

B.1.1 Comparison on the ImageNet dataset

Fully finetuning on ImageNet-1k. We compare the fully finetuning performance of ViT-B on ImageNet-1k in Tab. 5. It is observed that our TEC outperforms the iBOT base model with 0.7%

Table 8: Top1 accuracy on the ImageNet-1k dataset under parameter-efficient finetuning.

Method	Epoch	Settings	Top 1 acc.
MAE	1600	Linear probing	68.0
TEC_{MAE}	800	Linear probing	69.8
		+Input adapter FT	72.6
		+Encoder adapter FT	79.9

Table 9: Semi-supervised semantic segmentation on the ImageNet-S dataset.

Pretrain	Method	Epoch	mIoU_{val}
SSL	MAE	1600	38.3
	TEC_{MAE}	800	**42.9**
SSL+FT	MAE	1600+100	61.0
	TEC_{MAE}	800+100	**62.0**

when trained with 300 epochs from random initialization. When trained with 800 epochs, the gain is enlarged to 1.0%, showing TEC benefits from longer training. Similarly, TEC brings gains of 1.1% and 1.2% over the MAE base model with 300/800 epochs pretraining. Though MAE and iBOT are both strong MIM-based methods, TEC can still further improve them with the proposed target-enhanced conditional MIM scheme, verifying its effectiveness. With the help of SSL base models, TEC outperforms other SSL methods with similar or less training cost, including methods trained with implicitly extra data, *e.g.,* MVP [40] and FD-CLIP [41]. Compared to iBOT trained with large-scale ImageNet-22k dataset, TEC has the gain of 0.7% with only ImageNet-1k data, indicating TEC is more effective than enlarging the training cost of the base model. To the best of our knowledge, TEC achieves new SOTA on ViT-B when solely using ImageNet-1k training data, showing the potential of sustainable SSL learning. We also verify the scaling ability of our TEC by using the ViT-L model, and find that TEC surpasses the MAE base model by 0.6% with 300 epochs pretraining from random initialization.

Parameter-efficient finetuning on ImageNet-1k. Parameter-efficient finetuning methods, *e.g.,* linear probing, enable finetuning a small number of parameters on the downstream task. Here we test effectiveness of our TEC with parameter-efficient finetuning for classification on ImageNet-1k using ViT-B, with results shown in Tab. 8. Under the linear probing setting, TEC outperforms the MAE base model with 1.8%, indicating the learnt new model contains more category-related semantic information. The input and encoder adapters for pretraining can also be used for parameter-efficient finetuning. Finetuning with the input adapter, *i.e.,* prompting, brings a considerable gain of 4.6%. When finetuned with input and encoder adapters, the performance gain with MAE is enlarged to 11.9%. Therefore, adapters used for pretraining can greatly benefit parameter-efficient finetuning.

Semantic segmentation on ImageNet-S. To test the pixel-level representation ability of TEC pretrained models, we conduct semantic segmentation finetuning on ImageNet-S [17] that has pixel-level training labels. We use ViT-B as the segmentation model without extra segmentation head, since the pretraining and finetuning data have no domain shift. Tab. 9 shows that when finetuning with SSL pretrained models, TEC_{MAE} improves MAE base model by 4.6% mIoU. When using supervised ImageNet fully-finetuned pretraining, TEC_{MAE} achieves a gain of 1.0% over MAE.

B.1.2 Transfer learning on downstream tasks

Here we investigate the transfer learning ability of TEC models on downstream tasks.

Semantic segmentation. For semantic segmentation on the ADE20k [49] dataset, we use Upernet [43] with ViT-B as the segmentation model. Tab. 6 shows TEC_{iBOT} surpasses the iBOT base model by 1.0% mIoU, and TEC_{MAE} achieves a 1.8% gain over its MAE base model. It can be seen that TEC pretrained models show greater transfer learning abilities on semantic segmentation compared to their base models. Also, TEC shows clear advantages over strong competitors with fewer pretraining epochs. For example, it outperforms MAE, CAE [7], and CMAE [23] by 2.9%, 0.8% and 0.9%, achieving new SOTA.

Instance segmentation. For instance segmentation on the COCO [30] dataset, to make fair comparisons and save reproduce costs, we apply the Cascade MaskRCNN [3] implemented by iBOT [50] and ViTDet [28] for TEC with iBOT/MAE base models. Tab. 7 shows that TEC surpasses the iBOT base model by 1.5% on box AP and 1.2% on mask AP. Using the Cascade MaskRCNN implementation from ViTDet, TEC still achieves a gain of 0.4% on box AP and 0.4% on mask AP, indicating stable improvements of TEC.

B.2 Ablation and Analysis

We give the ablation study and analysis of our proposed method. By default, models are pretrained with 300 epochs and evaluated with the fully finetuning on ImageNet-1k.

Conditional pretraining. The conditional modules, *i.e.,* input/encoder adapters, aid the SSL pretraining under different base models. Tab. 12 shows adapters stably improve the performance by 0.4% and 0.2% when using MAE and iBOT as base models. In Tab. 11(b), with MAE base model, input adapters improve 0.1% over baseline and encoder adapters further brings a gain of 0.3%. To observe the adaptation difference to base models, we show the average proportion of encoder adapters contributing to the encoder output Z_e in Fig. 5. iBOT base model requires adapters to provide more features from deeper layers, while MAE base model makes adapters focus more on shallow layers, which is constant with their properties, *i.e.,* iBOT base model has more high-level category semantics while MAE model has more low-level image details.

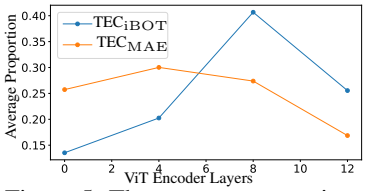

Figure 5: The average proportion of encoder adapters contributing to the encoder output Z_e.

Feature normalization on different dimensions. We normalize target features on the patch dimension to stress the relative relations among patches, which differs from existing methods that normalize on the channel dimension. In Tab. 11(a), normalizing on patch dimension achieves 0.3% gain than channel-dim normalization. While the channel-dim normalization has no effect compared to the unnormalized version. Channel-dim normalization emphasizes the feature difference in channels. Instead, our patch-dim normalization stresses the relations among patches, which is compatible to the patch-prediction in the MIM scheme. Tab. 12 shows training with patch-dim normalized feature has the 0.6%/0.4% gain over MAE/iBOT base models, showing its robustness over base models.

Semantic-related attention. The KQ attention maps naturally contain the semantic relations among patches, thus are used as the targets to enhance the patch-relation property of base model targets. Tab. 12 shows that using attention maps further improves the models trained with patch-dim normalization. Tab. 11(e) compares the effects of different types of attention maps. Only using the attention maps of the class token has no improvement, while the attention of semantic-related patches brings 0.2% gain over baseline. Therefore, it is the relation among patches that helps the MIM training. Compared to using all attention maps, using the selected semantic-related attention maps brings larger gain by reducing the noise.

Accelerating the training process of base models. By default, we use the fully pretrained SSL models as base models. To verify if the proposed TEC can improve an unconverged SSL model, we use a 300-epoch MAE pretrained ViT-B model as the base model and train TEC with 100/300 epoch from random initialization. As shown in Tab. 11(d), the 300-epoch pretrained MAE gives the performance of 82.9%. In comparison, TEC$_{\text{MAE300ep}}$ achieves 84.3%/83.9% with 300/100 epochs, surpassing the 300-epoch MAE base models with 1.4%/1.0%. Notably, TEC$_{\text{MAE300ep}}$ even outperforms the 1600-epoch pretrained MAE by 0.3% with only 100 epoch training, showing TEC can significantly accelerate the training process of the base model. Still, the TEC$_{\text{MAE1600ep}}$ taught with 1600-epoch MAE base model further improves the TEC$_{\text{MAE300ep}}$ by 0.4%, indicating our sustainable learning scheme relies on good base models to achieve better performance.

Towards general sustainable SSL. To make more steps towards sustainable SSL, we use the TEC pretrained models as the base model for a new round of TEC pretraining. Tab. 10 shows that the second-round TEC trained with the first-round TEC base model achieves 85.2%. The possible reason for the smaller improvement in the second round is caused by the limited network capacity or two-rounds of TEC pretraining learns similar knowledge.

Table 10: Towards general sustainable SSL by using the TEC as new base model.

Model	Base	Epoch	Top1 acc.
iBOT	-	1600	84.1
TEC$_{\text{iBOT}}$	iBOT	800	85.1
TEC	TEC$_{\text{iBOT}}$	800	85.2

Initializing new model with base model weights or not. The new models in the TEC framework are trained from random initialization. Tab. 11(c) compares the new model performance with/without loading the pretrained weights of base models. Randomly initialized new model outperforms the model loaded with pretrained base model weights by 0.4%. We assume that randomly initialized new models avoid the local minima of the base model, and the new model learns a different weight distribution compared to the base model.

Table 11: Ablation study on ImageNet-1K fully finetuning setting using ViT-B.

(a) Patch-norm features.

	Top1 acc.
MAE base	83.6
NA	83.9
Feature dim.	83.9
Patch dim.	**84.2**

(b) Effect of adapters.

	Top1 acc.
MAE base	83.6
No adapter	84.2
+ input adapter	84.3
+ encoder adapter	84.6

(c) Init. with base model pretrain.

	Top1 acc.
iBOT base	84.1
Load	84.4
Not load	**84.8**

(d) TEC accelerates MAE training.

	Epoch	Top1 acc.
MAE	1600	83.6
$\textbf{TEC}_{\text{MAE1600ep}}$	300	$\textbf{84.7}_{+1.1}$
MAE	300	82.9
$\textbf{TEC}_{\text{MAE300ep}}$	**100**	$\textbf{83.9}_{+1.0}$
$\textbf{TEC}_{\text{MAE300ep}}$	300	$\textbf{84.3}_{+1.4}$

(e) Effect of semantic-related patch attention.

	Top1 acc.
iBOT base	84.1
No attention	84.5
Cls token only	84.5
All attention	84.6
Attention select	**84.7**

Table 12: Ablation study on ImageNet-1K fully finetuning setting using ViT-B.

Patch-norm. feature	Semantic attention	Adapters	MAE base	iBOT base
Base model performance			83.6%	84.1%
✓			84.2%	84.5%
✓	✓		84.3%	84.7%
✓		✓	84.6%	84.7%
✓	✓	✓	84.7%	84.8%

C Experiment Implementation Details

Pretraining settings on ImageNet-1k. We follow the standard ViT network implemented in MAE. We give the pretraining settings in Tab. 13, which follows the training settings in MAE. Due to different properties of SSL teacher models, we set different parameters of semantic-related attention for MAE and iBOT teacher models as shown in Tab. 14.

Inspired by [1], we utilize the average of the last 2 blocks of the teacher model as the target. We measure the CKA similarity [25] of each mid-layer block to the output of the last block, and we observe the high feature similarity of last two blocks. Therefore, average features of last 2 layers are used.

Fully finetuning settings on ImageNet-1k. We give the fully finetuning settings on ImageNet-1k in Tab. 15. We observe that TECs trained with different teachers may have different properties. Since their teachers have different layer decay values, we set different layer decay values for TECs trained with different teachers.

Parameter-efficient finetuning settings on ImageNet-1k. Following MAE, we set linear probing settings are shown in Tab. 16. For parameter-efficient finetuning with the input adapter, we use the same training settings as used by the liner probing in Tab. 16. When finetuning with the encoder adapters, we use the same training settings as used by the fully finetuning in Tab. 15 due to more parameters are contained in encoder adapters.

Semi-supervised semantic segmentation finetuning on ImageNet-S. We give the training settings of semi-supervised semantic segmentation finetuning on ImageNet-S in Tab. 17. We set different learning rates and layer decay weights for initializing with pretrained weights with/without fully finetuning.

Downstream task settings. For semantic segmentation on ADE20K, we use the MMSegmentation [9] implementation of Upernet. The training configurations are following the MAE training configuration in MMSegmentation. For instance segmentation on COCO, we follow the training configurations of iBOT and ViTDet for TEC_{iBOT} and TEC_{MAE} with no change.

Table 13: Pretraining settings.

Configuration	Value
Optimizer	AdamW
Base learning rate	1.5e-4
Weight decay	0.05
Optimizer momentum	β_1, β_2=0.9, 0.95
Batch size	4096
Learning rate schedule	Cosine decay
Warmup epochs	40
Augmentation	RandomResizedCrop

Table 14: Parameters of semantic-related attention.

Teacher	τ	k
MAE (ViT-base)	1.8	15
MAE (ViT-large)	1.4	15
iBOT (ViT-base)	1.0	9

Table 15: Settings of fully finetuning and parameter-efficient finetuning with encoder adapters.

Configuration	Value
Optimizer	AdamW
Base learning rate	1e-3
Min learning rate	1e-6 (B), 1e-5(L)
Weight decay	0.05
Optimizer momentum	β_1, β_2=0.9, 0.999
Layer-wise lr decay	0.55 (MAE-B), 0.65 (iBOT-B), 0.65 (MAE-L)
Batch size	1024
Learning rate schedule	Cosine decay
Warmup epochs	20 (B), 5 (L)
Training epochs	100 (B), 50 (L)
Augmentation	RandAug (9, 0.5)
Label smoothing	0.1
Mixup	0.8
Cutmix	1.0
Drop path	0.1

Table 16: Settings of linear probing and parameter-efficient finetuning with the input adapter.

Configuration	Value
Optimizer	LARS
Base learning rate	0.1
Weight decay	0
Optimizer momentum	0.9
Batch size	16384
Learning rate schedule	Cosine decay
Warmup epochs	10
Training epochs	90
Augmentation	RandomResizedCrop

Table 17: Settings of semantic segmentation finetuning on ImageNet-S.

Configuration	Value
Optimizer	AdamW
Base learning rate	5e-4 (SSL), 1e-4 (SSL+FT)
Weight decay	0.05
Optimizer momentum	β_1, β_2=0.9, 0.999
Layer-wise lr decay	0.60 (SSL), 0.45 (SSL+FT)
Batch size	256
Learning rate schedule	Cosine decay
Warmup epochs	5
Training epochs	100
Augmentation	RandomResizedCrop
Drop path	0.1