MixMAE Supplementary Material

A. Training Details

A.1. Hyperparameters of Pretraining and Finetuning

We include details about the hyperparameters for reimplementation.

Pretraining. The default setting is in Table 1. We use xavier_uniform [7] to initialize all Transformer blocks following original ViT [5]. We by default use batch size of 1024 and scale the learning rate with linear rule [8]: lr=base_lr $\times$ batch_size / 256.

config	value
optimizer	AdamW [16]
base learning rate	1.5×10^{-4}
weight decay	0.05
optimizer momentum	β_1, β_2=0.9,0.95 [2]
learning rate schedule	cosine decay [15]
warmup epochs	40
augmentation	RandomResizedCrop

Table 1. Pretraining on ImageNet-1K.

Finetuning on ImageNet-1K. The default setting is in Table 2. We use layer-wise learning rate decay following [1, 3]. The decay ratio is swept in $\{0.7, 0.75, 0.8\}$, and we find 0.7 performs best. Following pretraining, the learning rate is scaled with linear rule: lr=base_lr $\times$ batch_size / 256.

Finetuning on other classification datasets. We reuse the setting in Table 2. We adjust the drop path rate for each dataset.

Finetuning on COCO. We use the Mask RCNN [10] framework with the encoder of MixMAE as its backbone. We follow the training setting in [9, 13]. In particular, we use large-scale jitter [6] augmentation with 1024$\times$1024 resolution and [0.1, 2.0] scale range. We use step learning rate schedule with 0.25 epochs of warmup. We finetune Swin-B/-L for 55/80 epochs. We use a layer-wise learning rate and set the decay ratio to 0.85/0.9 for Swin-B/-L.

Finetuning on ADE20K. We use the UperNet [18] framework with the encoder of MixMAE as its backbone. We finetune for 16K iterations with a batch size of 16. We use the layer-wise learning rate and set the decay ratio to 0.85/0.9 for Swin-B/-L. We adopt others settings from BEiT [1].

config	value
optimizer	AdamW
base learning rate	5×10^{-4}
layer-wise lr decay [1, 3]	0.7
batch size	1024
weight decay	0.05
optimizer momentum	β_1, β_2=0.9,0.999
learning rate schedule	cosine decay
warmup epochs	5
training epochs	100 (B), 50 (L/H)
augmentation	RandAug(9, 0.5) [4]
LabelSmooth [17]	0.1
Mixup [21]	0.8
CutMix [20]	1.0
drop path [11]	0.15 (B), 0.2 (L), 0.3 (H)

Table 2. Finetuning on ImageNet-1K.

Type	AP^{box}	AP^{mask}
Mix	**51.5**	**45.9**
Zero	51.0	45.3
Learnable	50.9	45.1
Shuffle	46.5	41.6
Zoomin	47.9	42.6

Table 3. Filling content.

# Images (ratio)	AP^{box}	AP^{mask}
2 (0.5)	51.5	45.9
2 w/ [M] (0.75)	51.2	45.4
3 (0.67)	51.6	45.9
4 (0.75)	**52.3**	**46.4**
5 (0.8)	51.4	45.4

Table 4. Number of mixing images.

A.2. Additional Results of Ablation Studies

A.2.1 Ablation results on COCO

We show more results of our ablation studies on COCO benchmark in Table 3 4 5 6. We find that the performance on the COCO is similar to that on ADE20K.

A.2.2 Pretraining Time Comparison

We compare the wall-clock time of the pretrain in Table 8. The pretrain time is measured on 8 NVIDIA-A100-SXM-80GB GPUs with a total batch size of 1024.

# Epochs	AP^{box}	AP^{mask}
300	51.5	45.9
600	52.2	46.5
900	**52.4**	**46.7**

Table 5. Pretraining epochs.

Dual	AP^{box}	AP^{mask}
✓	**51.5**	**45.9**
✗	50.0	44.4

Table 6. Dual reconstruction.

Method	Backbone	Pretrain Epochs	Top-1 Acc.
Supervised	ViT-B	-	81.8
MAE	ViT-B	1600	83.6
BEiT	ViT-B	800	83.2
MixMAE	ViT-B	600	**83.8**

Table 7. Performance of MixMAE and other methods on ViT.

Method	Backbone	Pretrain epochs	Pretrain Time (GPU hours)	Top-1 Acc.
SimMIM [19]	Swin-B	800	116	84.0
MAE [9]	ViT-B	1600	123	83.6
BEiT [1]	ViT-B	800	151	83.2
MixMAE	Swin-B	600	85	84.6
MixMAE	Swin-B/W14	300	64	84.8
MixMAE	Swin-B/W14	600	127	85.1

Table 8. Wall-clock time comparison of MIM methods.

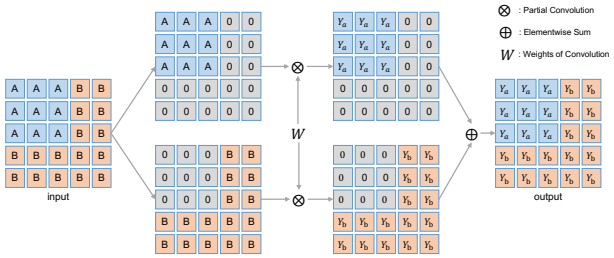

Figure 1. Mixed convolution.

Method	Backbone	Input Size	Pretrain Data	Top-1 Acc.
BiT-S [12]	Res50x3	448×448	ImageNet-1K	80.0
BiT-M [12]	Res50x3	448×448	ImageNet-21K	84.0
MixMAE	Res50x3	224×224	ImageNet-1K (w/o labels)	81.8
BiT-S [12]	Res101x3	448×448	ImageNet-1K	80.3
BiT-M [12]	Res101x3	448×448	ImageNet-21K	84.3
MixMAE	Res101x3	224×224	ImageNet-1K (w/o labels)	82.6

Table 9. **Results on ConvNets.** All results of MixMAE are obtained by pretraining for 300 epochs and finetuning for 100 epochs on ImageNet-1K. We report the top-1 accuracy on ImageNet-1K.

A.2.3 Performance on ViT.

We show the results on ViT [5] in Table 7.

A.3. Extend to ConvNets

While our MixMAE uses a hierarchical Transformer as the encoder, we also explore popular ConvNets. In particular, we use ResNet50x3 and ResNet101x3 as the encoder and compare the finetuning results on ImageNet-1K with BiT [12]. To reduce the difficulty of the pretext task, we extend the idea of partial convoluation [14] and propose a *mixed* version, as illustrated in Figure 1.

We compare the results in Table 9. In particular, our MixMAE outperforms BiT-S by a large margin with half the input size. We note that BiT-M achieves better results by pretraining with 10 $\times$ larger dataset ImageNet-21K. We believe the results of MixMAE can be further improved by using much larger datasets as shown by [1], and we leave it as future work.

References

[1] Hangbo Bao, Li Dong, and Furu Wei. Beit: Bert pre-training of image transformers. In *ICLR*, 2021.

[2] Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. Generative pretraining from pixels. In *ICML*, 2020.

[3] Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. Electra: Pre-training text encoders as discriminators rather than generators. In *ICLR*, 2020.

[4] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *CVPRW*, 2020.

[5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021.

[6] Golnaz Ghiasi, Yin Cui, Aravind Srinivas, Rui Qian, Tsung-Yi Lin, Ekin D Cubuk, Quoc V Le, and Barret Zoph. Simple copy-paste is a strong data augmentation method for instance segmentation. In *CVPR*, 2021.

[7] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *AISTATS*, 2010.

[8] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch sgd: Training imagenet in 1 hour. *arXiv:1706.02677*, 2017.

[9] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Doll'ar, and Ross B. Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, 2021.

[10] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, 2017.

[11] Gao Huang, Yu Sun, Zhuang Liu, Daniel Sedra, and Kilian Q Weinberger. Deep networks with stochastic depth. In *ECCV*, 2016.

[12] Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Joan Puigcerver, Jessica Yung, Sylvain Gelly, and Neil Houlsby. Big transfer (bit): General visual representation learning. In *ECCV*, 2020.

[13] Yanghao Li, Saining Xie, Xinlei Chen, Piotr Dollar, Kaiming He, and Ross Girshick. Benchmarking detection transfer learning with vision transformers. *arXiv:2111.11429*, 2021.

[14] Guilin Liu, Kevin J Shih, Ting-Chun Wang, Fitsum A Reda, Karan Sapra, Zhiding Yu, Andrew Tao, and Bryan Catanzaro. Partial convolution based padding. *arXiv:1811.11718*, 2018.

[15] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. In *ICLR*, 2016.

[16] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2017.

[17] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *CVPR*, 2016.

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

[19] 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 *CVPR*, 2021.

[20] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *ICCV*, 2019.

[21] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *ICLR*, 2017.