Mamba®: Vision Mamba Also Needs Registers

Supplementary Material

Appendix

A. More Technical Details

Table 7. Pre-training configurations

Configuration	Small	Base	Large
input size		128	
epochs		300	
optimizer		AdamW	
weight decay		0.05	
base learning rate	5e-4	2e-4	2e-4
batch size	1024	2048	2048
drop path		0.1	
label smoothing		✗	
random erasing		✗	
Rand Augmentation		✗	
repeated augmentation		✓	
ThreeAugmentation		✓	

Table 8. Intermediate training configurations

Configuration	Small	Base	Large
input size		224	
epochs		100	
optimizer		AdamW	
weight decay		0.05	
base learning rate		2e-4	
batch size		1024	
drop path	0.2	0.4	0.4
label smoothing		✗	
random erasing		✗	
Rand Augmentation		✗	
repeated augmentation		✓	
ThreeAugmentation		✓	

We train Mamba®-Tiny by the configurations of DeiT-Tiny [38] but follow a weaker data augmentation strategy used in [39]. For bigger sizes of Mamba®models, we use a three-stage training approach to prevent over-fitting and reduce effective training epochs. We summarize the recipes of pre-training, intermediate training, and fine-tuning in Table 7, Table 8, and Table 9, respectively. For all stages, the learning rate is calculated by base lr $\times$ batch size / 512, following a cosine decay scheduling with 5 epochs warmup. We use color jitter with a factor of 0.3, mixup and cutmix with alpha setting to 0.8 and 1.0, respectively.

Table 9. Fine-tuning configurations

Configuration	Small	Base	Large
input size		224	
epochs		20	
optimizer		AdamW	
weight decay		0.1	
base learning rate		1e-5	
batch size		512	
drop path	0.2	0.4	0.6
label smoothing		0.1	
random erasing		✗	
Rand Augmentation		rand-m9-mstd0.5-inc1	
repeated augmentation		✗	
ThreeAugmentation		✗	