Supplemental Material of
EVA: Exploring the Limits of Masked Visual Representation Learning at Scale

Yuxin Fang[1,2†] Wen Wang[3,2†] Binhui Xie[4,2†] Quan Sun[2] Ledell Wu[2]
Xinggang Wang[1‡] Tiejun Huang[2] Xinlong Wang[2‡] Yue Cao[2‡]

[1]Huazhong University of Science and Technology [2]Beijing Academy of Artificial Intelligence
[3]Zhejiang University [4]Beijing Institute of Technology

Code & Models: baaivision/EVA/01

config	value
peak learning rate	1e-4
optimizer	AdamW [13, 15]
optimizer hyper-parameters	$\beta_1, \beta_2, \epsilon$ = 0.9, 0.98, 1e-6
layer-wise lr decay [1, 6]	0.85
learning rate schedule	cosine decay
weight decay	0.05
input resolution	224^2
batch size	4096
warmup epochs	15
training epochs	60
drop path [11]	0.4
augmentation	RandAug (9, 0.5) [7]
label smoothing [17]	0.1
cutmix [21]	1.0
mixup [22]	✗
random erasing [23]	✗
random resized crop	(0.5, 1)
ema	✗

Table 1. Intermediate fine-tuning setting for ImageNet-21K.

config	value
peak learning rate	3e-5
optimizer	AdamW
optimizer hyper-parameters	$\beta_1, \beta_2, \epsilon$ = 0.9, 0.999, 1e-8
layer-wise lr decay	0.95
learning rate schedule	cosine decay
weight decay	0.05
input resolution	336^2 / 560^2
batch size	512
warmup epochs	2
training epochs	10 / 15
drop path	0.4
augmentation	RandAug (9, 0.5)
label smoothing	0.3
cutmix	✗
mixup	✗
random erasing	✗
random resized crop	(0.08, 1)
ema	0.9999
test crop ratio	1.0

Table 2. Fine-tuning setting for ImageNet-1K.

A. Appendix

The MIM pre-training and contrastive language-image pre-training settings are already available in our main submission. Here we summarize the detailed configurations for image classification (§A.1), video action classification (§A.2), object detection & instance segmentation (§A.3), and semantic segmentation (§A.4).

A.1. Image Classification

The fine-tuning hyper-parameters for ImageNet-21K and ImageNet-1K are shown in Table 1 and Table 2, respectively.

A.2. Video Action Classification

For video action classification tasks, a two-stage fine-tuning process is adopted. The statistics of video datasets we used are available in Table 3.

dataset & split	#clips	avg. length	#classes
Kinetics-400 train [12]	234,584	10s	400
Kinetics-400 val [12]	19,760	10s	400
Kinetics-600 train [2]	412,688	10s	600
Kinetics-600 val [2]	29,779	10s	600
Kinetics-700 train [3]	534,063	10s	700
Kinetics-700 val [3]	33,914	10s	700
Kinetics-722 (ours)	629,395	10s	722

Table 3. Video dataset statistics.

In the first stage, we conduct intermediate fine-tuning on a merged dataset coined Kinetics-722 (K-722) that integrates all valid training samples from Kinetics-400 (K-400) [12], Kinetics-600 (K-600) [2] and Kinetics-700 (K-700) [3]. The input video resolution is 224^2 with 8 frames. Notably, for a fair and legal comparison, we removed leaked videos in all validation sets and duplicated videos in all training sets based on the videos' "youtube id". Accordingly, the cleaned K-722 contains 0.63M training videos, covering 722 human

†Interns at Beijing Academy of Artificial Intelligence (BAAI). ‡Corresponding authors: Yue Cao (caoyue10@gmail.com), Xinlong Wang (xinlong.wang96@gmail.com) and Xinggang Wang (xgwang@hust.edu.cn).

config	value
optimizer	AdamW
optimizer hyper-parameters	$\beta_1, \beta_2, \epsilon$ = 0.9, 0.98, 1e-6
weight decay	0.05
peak learning rate	8e-6
learning rate schedule	cosine decay
warmup epochs	5
epochs	40
batch size	256
input resolution	224^2
random flip	0.5
multiscale crop	(1, 0.875, 0.75, 0.66)
color jitter	0.8
grayscale	0.2
cutmix	1.0
mixup	0.8
label smoothing	0.1
drop path	0.3
layer-wise lr decay	✗

Table 4. Kinetics-722 intermediate fine-tuning settings.

config	K-400 [12]	K-600 [2]	K-700 [3]
optimizer		AdamW	
optimizer hyper-parameters		$\beta_1, \beta_2, \epsilon$ = 0.9, 0.98, 1e-6	
weight decay		0.05	
peak learning rate		1e-6	
minimal learning rate		1e-6	
warmup epochs		0	
epochs	1	2	2
batch size		256	
input resolution		224^2	
random flip		0.5	
multiscale crop		(1, 0.875, 0.75, 0.66)	
color jitter		0.8	
grayscale		0.2	
mixup		✗	
cutmix		✗	
label smoothing		0.1	
drop path		0.2	
layer-wise lr decay		0.95	
multi-view inference		4 clips, 3 crops	

Table 5. Hyper-parameters used in the video action recognition.

action classes. Table 4 lists the detailed settings & hyper-parameters for fine-tuning on this dataset.

In the second stage, we further fine-tune on each dataset using more input video frames of 16 with a resolution of 224^2. For the frame sampling, we adopt the sparse sampling strategy [19]. During testing, we follow the common practice of multi-view inference [8, 14, 18, 20] with 4 temporal clips and 3 spatial crops. The final prediction is the ensemble of all trials. Table 5 lists the detailed hyper-parameters for fine-tuning on K-400, K-600 and K-700.

A.3. Object Detection & Instance Segmentation

The detailed hyper-parameters are shown in Table 6 and Table 7. For intermediate fine-tuning on Objects365 [16], the model is trained with a batch size of 128 for 380k iterations. To accelerate the training process, we use a smaller input resolution of 1024^2 for the first 320k iteration. Afterward, the input resolution is lifted to 1280^2 for a better adaptation to the fine-tuning of COCO and LVIS.

For fine-tuning COCO and LVIS, the learning rate is initialized as 2.5e-5 and step by a factor of 10 for the last 5k iterations. As shown in Table 7, we use almost identical hyper-parameters for training COCO and LVIS. Except for the commonly used repeat factor sampling [10] and federated loss [24] that are specialized for long-tailed recognition, the only difference in training is that we train the model for 45k steps on COCO, while a longer 75k step on LVIS, since the tail classes generally take a longer schedule to converge [9].

config	value
optimizer	AdamW
optimizer hyper-parameters	$\beta_1, \beta_2, \epsilon$ = 0.9, 0.999, 1e-8
learning rate	1e-4
layer-wise lr decay	0.9
training steps	380k
training input resolution	$1024^2 \rightarrow 1280^2$
batch size	128
weight decay	0.1
drop path	0.6

Table 6. Objects365 object detection intermediate fine-tuning settings.

config	COCO	LVIS
optimizer	AdamW	
optimizer hyper-parameters	$\beta_1, \beta_2, \epsilon$ = 0.9, 0.999, 1e-8	
learning rate	2.5e-5	
learning rate schedule	step decay	
training steps	45k	75k
learning decay step	40k	70k
batch size	64	
training input resolution	1280^2	
weight decay	0.1	
layer-wise lr decay	0.9	
drop path	0.6	
repeat threshold	-	0.001
frequency weight power	-	0.5
max numbers of detection	100	1000

Table 7. COCO and LVIS object detection & instance segmentation fine-tuning settings.

config	COCO-Stuff	ADE20K
optimizer	AdamW	
optimizer hyper-parameters	$\beta_1, \beta_2, \epsilon$ = 0.9, 0.999, 1e-8	
peak learning rate	1.5e-5	2.5e-5
batch size	32	64
fine-tuning steps	60000	20000
layer-wise lr decay	0.95	
weight decay	0.5	
drop path	0.5	
input resolution	896^2	
seg head #enc. & #dec.	6 & 8	
seg head dim	1024	
relative position bias	✗	

Table 8. COCO-Stuff-164K and ADE20K semantic segmentation fine-tuning settings.

A.4. Semantic Segmentation

Detailed configurations about semantic segmentation are available in Table 8. Our settings basically follow ViT-Adapter [4] with Mask2Former [5] as the segmentation head. For ADE20K, we use COCO-Stuff pre-trained weights as initialization.

References

[1] Hangbo Bao, Li Dong, and Furu Wei. Beit: Bert pre-training of image transformers. *arXiv preprint arXiv:2106.08254*, 2021. 1

[2] Joao Carreira, Eric Noland, Andras Banki-Horvath, Chloe Hillier, and Andrew Zisserman. A short note about kinetics-600. *arXiv preprint arXiv:1808.01340*, 2018. 1, 2

[3] Joao Carreira, Eric Noland, Chloe Hillier, and Andrew Zisserman. A short note on the kinetics-700 human action dataset. *arXiv preprint arXiv:1907.06987*, 2019. 1, 2

[4] Zhe Chen, Yuchen Duan, Wenhai Wang, Junjun He, Tong Lu, Jifeng Dai, and Yu Qiao. Vision transformer adapter for dense predictions. *arXiv preprint arXiv:2205.08534*, 2022. 3

[5] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. *arXiv preprint arXiv:2112.01527*, 2021. 3

[6] Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. ELECTRA: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555*, 2020. 1

[7] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *CVPRW*, 2020. 1

[8] Christoph Feichtenhofer, Haoqi Fan, Yanghao Li, and Kaiming He. Masked autoencoders as spatiotemporal learners. *arXiv preprint arXiv:2205.09113*, 2022. 2

[9] WeiFu Fu, CongChong Nie, Ting Sun, Jun Liu, TianLiang Zhang, and Yong Liu. Lvis challenge track technical report 1st place solution: Distribution balanced and boundary refinement for large vocabulary instance segmentation. *arXiv preprint arXiv:2111.02668*, 2021. 2

[10] Agrim Gupta, Piotr Dollar, and Ross Girshick. Lvis: A dataset for large vocabulary instance segmentation. In *CVPR*, 2019. 2

[11] Gao Huang, Yu Sun, Zhuang Liu, Daniel Sedra, and Kilian Q Weinberger. Deep networks with stochastic depth. In *ECCV*, 2016. 1

[12] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017. 1, 2

[13] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 1

[14] Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer. In *CVPR*, 2022. 2

[15] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 1

[16] Shuai Shao, Zeming Li, Tianyuan Zhang, Chao Peng, Gang Yu, Xiangyu Zhang, Jing Li, and Jian Sun. Objects365: A large-scale, high-quality dataset for object detection. In *ICCV*, 2019. 2

[17] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *CVPR*, 2016. 1

[18] Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. *arXiv preprint arXiv:2203.12602*, 2022. 2

[19] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *ECCV*, 2016. 2

[20] Chen Wei, Haoqi Fan, Saining Xie, Chao-Yuan Wu, Alan Yuille, and Christoph Feichtenhofer. Masked feature prediction for self-supervised visual pre-training. In *CVPR*, 2022. 2

[21] 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. 1

[22] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017. 1

[23] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random erasing data augmentation. In *AAAI*, 2020. 1

[24] Xingyi Zhou, Vladlen Koltun, and Philipp Krähenbühl. Probabilistic two-stage detection. *arXiv preprint arXiv:2103.07461*, 2021. 2