Supplementary Material for
Turbo Training with Token Dropout

Tengda Han[1]
htd@robots.ox.ac.uk
Weidi Xie[1,2]
weidi@robots.ox.ac.uk
Andrew Zisserman[1]
az@robots.ox.ac.uk

[1] Visual Geometry Group
Department of Engineering Science
University of Oxford

[2] Coop. Medianet Innovation Center
Shanghai Jiao Tong University
Shanghai, China

We provide the implementation details in Sect. 1. For the code and models of this paper, please refer to our project page: https://www.robots.ox.ac.uk/~vgg/research/turbo/.

1 Implementation Details

Architectural Details. In our implementation, we adopt the standard ViT-B architectures as [4, 10]. Specifically, the encoder is a 12-layer transformer with 768 feature dimension and the light-weight decoder is a 8-layer transformer with 512 feature dimension. The input spatial-temporal patch has a size of $t \times h \times w = 2 \times 16 \times 16$. We use sinusoidal positional embeddings [11]. For both the action classification and long-video activity classification tasks, we pass the encoder's final-layer 'CLS' token into a linear layer for classification. For learning video-language representation, we project both the video feature and language feature with a 2-layer MLP, then compute the InfoNCE loss $\mathcal{L}_{\texttt{NCE}}$ as introduced in the main paper Page 5.

Config	Act. Classification	V-L Training	Long-video Activity Classification
ViT-B encoder depth	12 layers	12 layers	12 layers
ViT-B encoder dimension	768	768	768
decoder depth	8 layers	8 layers	8 layers
decoder dimension	512	512	512
optimizer	AdamW [7]	AdamW	AdamW
base learning rate	1e-3	1e-4	3e-4
weight decay	0.05	0.05	0.05
learning rate schedule	cosine-decay [8]	cosine-decay	cosine-decay
warm-up epochs	10	0.5	10(BF), 5(COIN)
training epochs	100	5	100(BF), 50(COIN)
repeated sampling [2, 5]	1	4	4
augmentation	RandAug(9,0.5) [1]	MultiScaleCrop	RandAug(9,0.5)
label smoothing [9]	0.1	-	0.1
mixup [13]	0.8	-	0.8
cutmix [12]	1.0	-	1.0
drop path [6]	0.1	0.0	0.1

Table 1. Implementation details of action classification, video-language training and long-video activity classification tasks.

Training Details. The details of training action classification, video-language training and long-video activity classification tasks are listed in Table 1. Note that, for action classification and long-video activity classification tasks, we use the same data augmentation as in [4, 10]; for video-language training, we only use basic cropping augmentation due to the adequate amount of training data from the HTM-AA [3] dataset (3.3M clip-sentence pairs).

References

[1] Ekin Dogus Cubuk, Barret Zoph, Jon Shlens, and Quoc Le. Randaugment: Practical automated data augmentation with a reduced search space. In *NeurIPS*, 2020.

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

[3] Tengda Han, Weidi Xie, and Andrew Zisserman. Temporal alignment networks for long-term video. In *Proc. CVPR*, 2022.

[4] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proc. CVPR*, 2022.

[5] Elad Hoffer, Tal Ben-Nun, Itay Hubara, Niv Giladi, Torsten Hoefler, and Daniel Soudry. Augment your batch: better training with larger batches. In *Proc. CVPR*, 2020.

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

[7] Frank Hutter Ilya Loshchilov. Decoupled weight decay regularization. In *Proc. ICLR*, 2019.

[8] Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with warm restarts. In *Proc. ICLR*, 2017.

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

[10] 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.

[11] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017.

[12] 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 *Proc. ICCV*, 2019.

[13] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *Proc. ICLR*, 2018.