Supplementary Material for Motion-aware Contrastive Video Representation Learning via Foreground-background Merging

Shuangrui Ding[1*] Maomao Li[2] Tianyu Yang[2] Rui Qian[3]
Haohang Xu[1] Qingyi Chen[4] Jue Wang[2] Hongkai Xiong[1†]
[1]Shanghai Jiao Tong University [2]Tencent AI Lab
[3]The Chinese University of Hong Kong [4]University of Michigan
{dsr1212, xuhaohang, xionghongkai}@sjtu.edu.cn tianyu-yang@outlook.com
{limaomao07, arphid}@gmail.com qr021@ie.cuhk.edu.hk chenqy@umich.edu

1. More Implementation Details

1.1. Self-supervised Pretraining Details.

In the pretraining stage, we adopt the SGD optimizer with the initial learning rate of 0.01 and weight decay of 10^{-4}, and we decay the learning rate by 0.1 at epoch 120 and 180. For the implementation of MoCo, we closely follow the parameter setting in [2]. The number of the negative queue is set to 65536 for Kinetics-400, and 2048 for UCF101, respectively. We also swap the key/queue samples so that each sample can generate the gradient for optimization. The momentum of updating the key encoder is 0.999, and the temperature hyper-parameter τ is 0.1. We use a 2-layer MLP projection head.

1.2. Augmentation Details.

We perform data augmentation using Kornia package [4]. In the pretraining and finetune phase, we crop 224×224 or 112×112 pixels from a video with RandomResizedCrop, which randomly resizes the input area between a lower bound and upper bound. We set the bound as [0.2, 1]. Then, the basic augmentation set consists of RandomGrayscale (probability 0.2), ColorJitter (probability 0.8, {brightness, contrast, saturation, hue} = {0.4, 0.4, 0.4, 0.1}), RandomHorizontalFlip (probability 0.5) and RandomGaussianBlur (probability 0.5, the kernel with radius 23 and standard deviation $\in$ [0.1, 2.0]). In the linear probe stage, we take a simpler augmentation setting instead. We only apply RandomResizedCrop with the bound [0.2, 1] and RandomHorizontalFlip (probability 0.5).

1.3. More Details on Action Recognition.

In the finetune stage, the SGD optimizer is adopted with the initial learning rate of 0.025 and weight decay of 10^{-4}. We finetune the model for 150 epochs with a batch size of 128 on 4 Tesla V100 GPUs. We decay the learning rate by 0.1 at epoch 60 and 120. Besides, we add the dropout layer before the last fully connected layer. We set dropout rate 0.7 for UCF101 and 0.5 for HMDB51, respectively.

We train the last fully connected layer in the linear probe with the initial learning rate of 5 and weight decay of 0. We finetune the model for 100 epochs with a batch size of 128 on 4 Tesla V100 GPUs. We decay the learning rate by 0.1 at epoch 60 and 80. Besides, We $\mathcal{L}_2$ normalize the embeddings before the last fully connected layer.

Figure 1. Illustration of FAME visualization. The first row is the video frame while the second row is the foreground mask FAME generates.

2. More Visualization of FAME

In Figure 1, we show more foreground masks obtained from FAME. We show that FAME can discover most regions of the foreground objects and remove the monotonous backgrounds.

3. CAM Visualization

Besides CAAM visualization, we provide the CAM [5] visualization in Figure 2. With that, we can spot the contri-

*Work done during an internship at Tencent AI Lab.
†Corresponding author. Email: xionghongkai@sjtu.edu.cn.

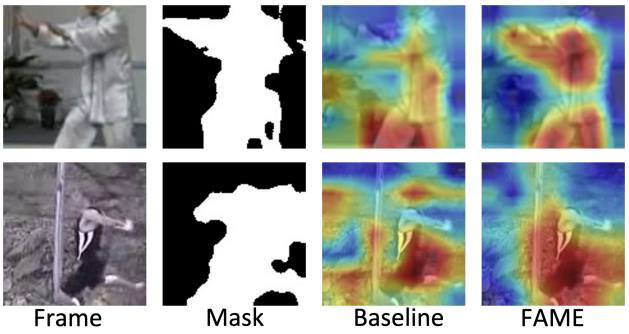


Figure 2. Class activation maps (CAM) visualization. Red areas indicate the important areas for the model to predict the action class. Comparing to baseline, FAME resists the impact of background and highlights the motion areas.

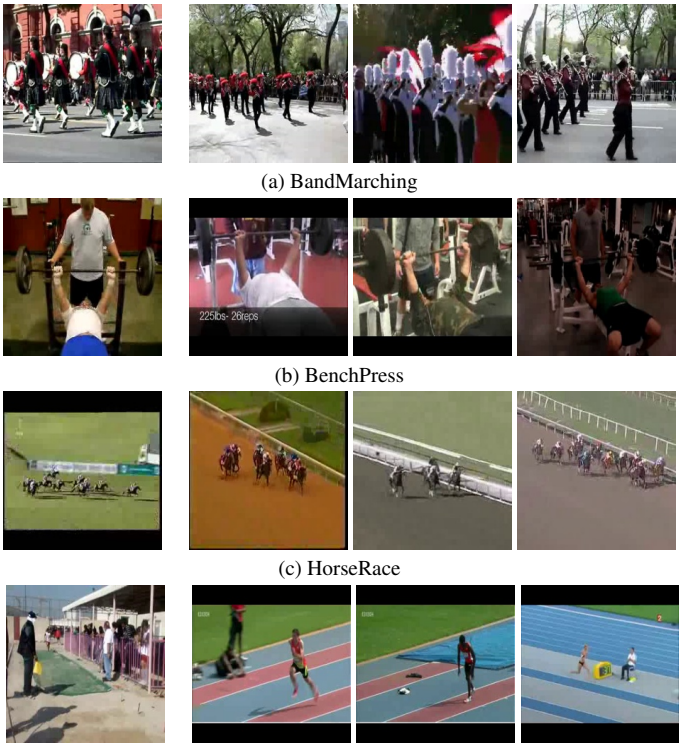
bution of each area and find crucial regions for discriminating the specific action class. We find that when integrated with FAME, the model can focus on moving foreground area rather than background context. For example, in the first row of Figure 2, FAME precisely captures the moving upper and lower body when the man is practicing TaiChi, while the baseline displays a dispersed highlight map and fails to attend to the motion area. In addition, we illustrate that the CAM activation map can almost overlap with the foreground mask generated by FAME. It testifies that our strong motion inductive augmentation guides the model to perceive the motion patterns and hinder the background bias.

4. Visualization of Video Retrieval

In Figure 3, we demonstrate the results of video retrieval. After pretraining the model on Kinetics-400, we conduct the video retrieval experiment on UCF101. The results show that our model can retrieve diverse video samples that share the same action semantics with the query, regardless of the background context. For example, in Fig 3d, the query sample contains the action in the sandpit, and our model could retrieve the long jump samples in the standard stadium. Though the backgrounds in the query and retrieved videos are quite different, our model achieves accurate retrieval by attending to the dynamic motions and understanding the true action semantics.

5. More Results on Something-something V2

We finetune our pretained model on Something-Something V2 [3]. We obtain 53.3% Top-1 accuracy with R(2+1)D, which beats RSPNet [1] under same resolution.



(a) BandMarching

(b) BenchPress

(c) HorseRace

(d) LongJump

Figure 3. Visualization results of video retrieval.The first column is the video frame of query instances. The rightmost three columns are Top-3 nearest retrieval results.

References

[1] Peihao Chen et al. Rspnet: Relative speed perception for unsupervised video representation learning. In *AAAI*, 2021. 2

[2] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020. 1

[3] Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, et al. The" something something" video database for learning and evaluating visual common sense. In *Proceedings of the IEEE international conference on computer vision*, pages 5842–5850, 2017. 2

[4] Edgar Riba, Dmytro Mishkin, Daniel Ponsa, Ethan Rublee, and Gary Bradski. Kornia: an open source differentiable computer vision library for pytorch. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3674–3683, 2020. 1

[5] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE international conference on computer vision*, pages 2921–2929, 2016. 1