Supplementary material:
SCSampler: Sampling Salient Clips from Video for Efficient Action Recognition

1. Action classification networks

In the main paper, we provide an overview of the gains in accuracy and speedup enabled by SCSampler for several video-classification models. In this section, we provide the details of the action classifier architectures used in our experiments and discuss the training procedure used to train these models.

1.1. Architecture details

3D-ResNets (R3D) are residual networks where every convolution is 3D. Mixed-convolution models (MCx) are 3D CNNs leveraging residual blocks, where the first $x-1$ convolutional groups use 3D convolutions and the subsequent ones use 2d convolutions. In our experiments we use an MC3 model. R(2+1)D are models that decompose each 3D convolution in a 2D convolution (spatial), followed by 1D convolution (temporal). For further details, please refer to the paper that introduced and compared these models [8] or the repository [1] where pretrained models can be found.

1.2. Training procedure

Sports-1M. For the Sports1M dataset, we use the training procedure described in [8] for all models except ir-CSN-152. Frames are first re-scaled to have resolution 342×256, and then each clip is generated by randomly cropping a window of size 224×224 at the same location from 16 adjacent frames. We use batch normalization after all convolutional layers, with a batch size of 8 clips per GPU. The models are trained for 100 epochs, with the first 15 epochs used for warm-up during distributed training. Learning rate is set to 0.005 and divided by 10 every 20 epochs. The ir-CSN-152 model is trained according to the training procedure described in [7].

Kinetics. On Kinetics, the clip classifiers are trained with mini-batches formed by sampling five 16-frame clips with temporal jittering, except for the ir-CSN-152 which uses 32 frame clips. Frames are first resized to resolution 342×256, and then each clip is generated by randomly cropping a window of size 224×224 at the same location from 16 adjacent frames. The models are trained for 45 epochs, with 10 warm-up epochs. The learning rate is set to 0.01 and divided by 10 every 10 epochs as in [8]. ir-CSN-152 [7] and R(2+1)D [8] are finetuned from Sports1M for 14 epochs with the procedure described in [7].

1.3. Datasets

As mentioned in the paper, we evaluate the effectiveness of SCSampler on Sports1M [3] and Kinetics-400 [4] datasets. For Sports-1M, we use the publicly defined train and test splits provided by the dataset creators, while for Kinetics-400, we use official train split for training and validation set for testing.

2. Implementation details for SCSampler

In this section, we give the implementation details of the architectures and describe the training/finetuning procedures of our sampler networks.

2.1. Visual-based sampler

Following Wu et al. [9], all of our visual samplers are pre-trained on the ILSVRC dataset [5]. The learning rate is set to 0.001 for both Sports1M and Kinetics. As in [9], the learning rate is reduced when accuracy plateaus and pre-trained layers use $100\times$ smaller learning rates. The ShuffleNet0.5 [10] (26 layers) model is pretrained on ImageNet. We use three groups of group convolutions as this choice is shown to give the best accuracy in [10]. The initial learning rate and the learning rate schedule are the same as those used for ResNet-18.

2.2. Audio-based sampler

We use a VGG model [6] pretrained on AudioSet [2] as our backbone network, with MEL spectrograms of size 40×200 as input. When fine-tuning the network with *SAL-RANK*, we use an initial learning rate of 0.01 for Sports1M and 0.03 for Kinetics for the first 5 epochs and then divide the learning rate by 10 every 5 epochs. The learning rate of the pretrained layers is multiplied by a factor of $5 * 10^{-2}$. When finetuning with the *SAL-CL* loss, we set the learning rate to 0.001 for 10 epochs, and divide it by 10 for 6 additional epochs. When finetuning with *AC* loss, we start with learning rate 0.001, and divide it by 10 every 5 epochs.

3. Additional evaluations of design choices for SCSampler

Here we present additional analyses of the design choices and hyperparameter values of SCSampler.

Audio SCSampler	accuracy (%)	runtime (min)
finetuned VGG	67.8	22.0
FC trained on VGG-conv4_2	67.0	21.6
FC trained on VGG-pool4	67.0	21.4
FC trained on VGG-fc1	59.8	21.4

Table 1: Varying the audio sampler architecture. Performance is measured as MC3-18 video accuracy (%) on the test set of miniSports with $K = 10$ sampled clips.

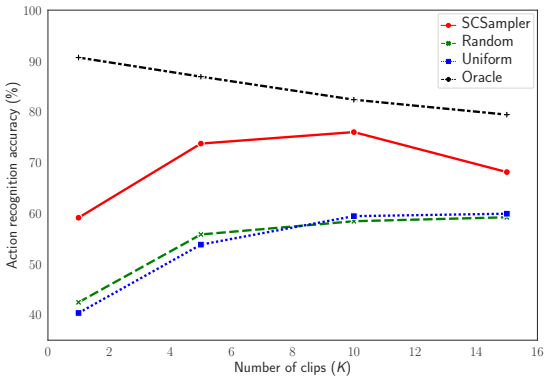

Figure 1: Video classification accuracy (%) of MC3-18 on the miniSports test set vs the number of sampled clips (K).

3.1. Varying the audio sampler architecture.

Table 1 shows video classification accuracy using different variants of our audio sampler. Given our VGG audio network pretrained for classification on AudioSet, we train it on miniSport using the following two options: finetuning the entire VGG model vs training a single FC layer on VGG activations from one layer (conv4_2, pool4, or fc1). All audio samplers are trained with the SAL-RANK loss. We can see that finetuning the audio sampler gives the best classification accuracy.

3.2. Varying the number of sampled clips (K)

Figure 1 shows how video-level classification accuracy changes as we vary the number of sampled clips (K). The sampler here is *AV-union-list*. $K = 10$ provides the best accuracy for our sampler. For the Oracle, $K = 1$ gives the top result as this method can conveniently select the clip that elicits the highest score for the correct label on each test video.

3.3. Selecting hyperparameter K' for *AV-union-list*

The *AV-union-list* method (described in section 3.3.3 of our paper) combines the audio-based and the video-based samplers, by selecting K' top-clips according to the visual sampler (with hyper-parameter K' s.t. $K' < K$) and adds a set of $K - K'$ different clips from the ranked list of the audio sampler to form a sample set of size K ($K = 10$ is used in this experiment). In Figure 2 we analyze the impact of K' on action classification. The fact that the best value is achieved at $K' = 8$ suggests that the signals from the two samplers are somewhat complementary, but the visual sampler provides a more accurate measure of clip saliency.

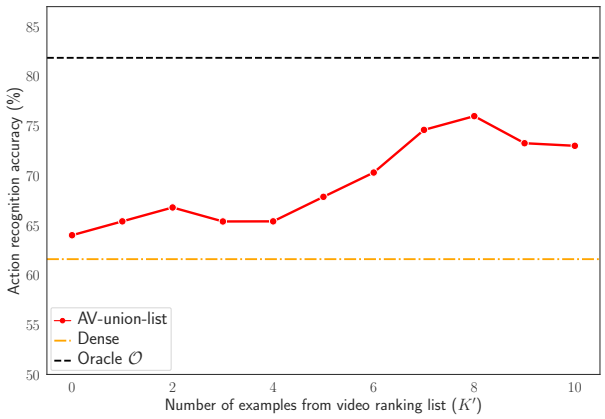

Figure 2: Varying the number of clips K' sampled by the visual sampler, when combining video-based and and audio-based sampler according to the *AV-union-list* strategy. The best action recognition accuracy is achieved when sampling $K' = 8$ clips with the video-based sampled and $K - K' = 2$ clips with the audio-based sampler. Evaluation is done on the miniSports dataset, with the MC3-18 clip classifier.

4. Comparison to Random/Uniform under the same runtime.

Fig. 3 shows runtime (per video) vs video-level classification accuracy on miniSports, obtained by varying the number of sampled clips per video (K). For this test we use MC3-18, which is the fastest clip-classifier in our comparison. The overhead of running SCSampler on each video is roughly equivalent to 3 clip-evaluations of MC3-18. Even after adding clip evaluations to Random/Uniform to obtain a comparison under the same runtime, SCSampler significantly outperforms these baselines. Note that for costlier clip-classifiers the SCSampler overhead would amount to less than one clip evaluation (e.g., 0.972 for R(2+1)D-50), making the option of Random/Uniform even less appealing for the same runtime.

5. Applying SCSampler every N clips

While our sampler is quite efficient, further reductions in computational cost can be obtained by running SCSampler every N clips in the video. This implies that the final top-K clips used by the action classifier will be selected from a subset of clips obtained by applying SCSampler

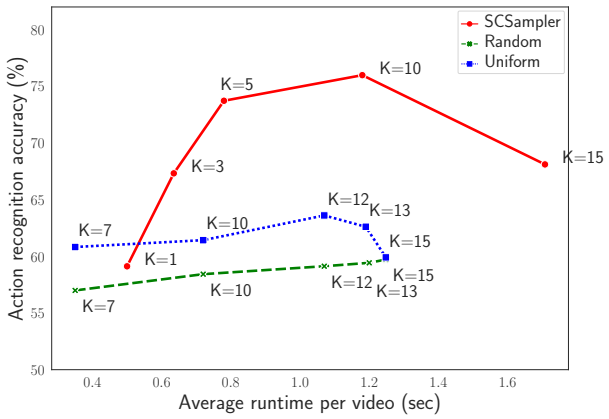

Figure 3: Video-level classification accuracy on the test of mini-iSports vs runtime per video using different numbers of sampled clips (K). The clip classifier is MC3-18.

with a stride of N clips. As usual, we fix the value of K to 10 for SCSampler. Figure 4 shows the results obtained with the best configuration of our SCSampler (see details in 4.1.1) and the ir-CSN-152 [7] action classifier on the full Sports1M dataset. We see that we can apply SCSampler with clip-strides of up to $N = 7$ before the action recognition accuracy degrades to the level of costly dense predictions. This results in further reduction of computational complexity and runtime, as we only need to apply the sampler to $\lceil L/N \rceil$ clips.

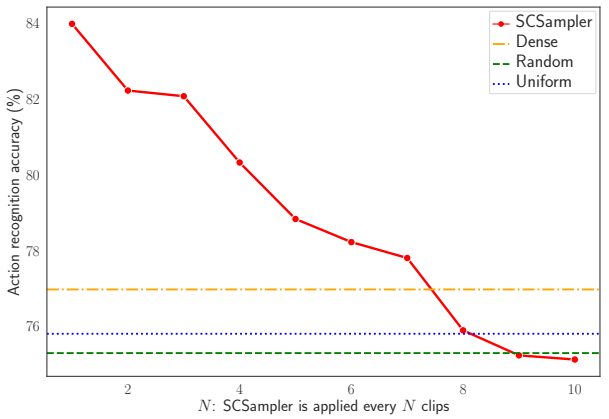

Figure 4: Applying SCSampler every N clips reduces the computational cost. Here we study how applying SCSampler with a clip-stride of N affects the action classification accuracy on Sports1M using ir-CSN-152 as clip classifier.

References

[1] Facebook. Video model zoo. https://github.com/facebookresearch/VMZ, 2018. 1

[2] J. F. Gemmeke, D. P. W. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *Proc. IEEE ICASSP 2017*, New Orleans, LA, 2017. 1

[3] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and F. Li. Large-scale video classification with convolutional neural networks. In *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014*, pages 1725–1732, 2014. 1

[4] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev, M. Suleyman, and A. Zisserman. The kinetics human action video dataset. *CoRR*, abs/1705.06950, 2017. 1

[5] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. 1

[6] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014. 1

[7] D. Tran, H. Wang, L. Torresani, and M. Feiszli. Classification with channel-separated convolutional networks. In *IEEE International Conference on Computer Vision, ICCV 2019, Seoul, South Korea, October 26-November 2, 2019*, 2019. 1, 3

[8] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri. A closer look at spatiotemporal convolutions for action recognition. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 6450–6459, 2018. 1

[9] C. Wu, M. Zaheer, H. Hu, R. Manmatha, A. J. Smola, and P. Krähenbühl. Compressed video action recognition. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 6026–6035, 2018. 1

[10] X. Zhang, X. Zhou, M. Lin, and J. Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 6848–6856, 2018. 1