Is Space-Time Attention All You Need for Video Understanding?

Supplementary Materials

Gedas Bertasius [1] **Heng Wang** [1] **Lorenzo Torresani** [1 2]

Our supplementary materials consist of:

1. Implementation Details.
2. Additional Ablations.
3. Additional Qualitative Results.

1. Implementation Details

Our TimeSformer implementation is built using PySlowFast (Fan et al., 2020) and pytorch-image-models (Wightman, 2019) packages. Below, we describe specific implementation details regarding the training and inference procedures of our model.

Training. We train our model for 15 epochs with an initial learning rate of 0.005, which is divided by 10 at epochs 11, and 14. During training, we first resize the shorter side of the video to a random value in $[256, 320]$. We then randomly sample a 224×224 crop from the resized video. For our high-resolution model, TimeSformer-HR, we resize the shorter side of the video to a random value in $[448, 512]$, and then randomly sample a 448×448 crop. We randomly sample clips from the full-length videos with a frame rate of $1/32$. The batch size is set to 16. We train all our models using synchronized SGD across 32 GPUs. The momentum is set to 0.9, while the weight decay is set to 0.0001.

Unless otherwise noted, in our experiments we use the "Base" ViT model (Dosovitskiy et al., 2020). Temporal and spatial attention layers in each block are initialized with the same weights, which are obtained from the corresponding attention layer in ViT.

Inference. As discussed in the main draft, during inference we sample a single temporal clip in the middle of the video. We scale the shorter spatial side of a video to 224 pixels (or 448 for TimeSformer-HR) and take 3 crops of size 224×224 (448×448 for TimeSformer-HR) to cover a larger spatial extent within the clip. The final prediction is obtained by averaging the softmax scores of these 3 predictions.

Other models in our comparison. To train I3D (Carreira & Zisserman, 2017), and SlowFast (Feichtenhofer et al., 2019), we use the training protocols that were used in the original papers. For I3D, we initialize it with a 2D ImageNet CNN, and then train it for 118 epochs with a base learning rate of 0.01, which is divided by 10 at epochs 44 and 88. We use synchronized SGD across 32 GPUs following the linear scaling recipe of Goyal et al. (2017a). We set the momentum to 0.9, and weight decay to 0.0001. The batch size is set to 64. For the SlowFast model, when initialized from ImageNet weights, we use this same exact training protocol. When training SlowFast from scratch, we use the training protocol described by the authors (Feichtenhofer et al., 2019). More specifically, in that case, the training is done for 196 epochs with a cosine learning rate schedule, and the initial learning rate is set to 0.1. We use a linear warm-up for the first 34 epochs starting with a learning rate of 0.01. A dropout of 0.5 is used before the final classification layer. The momentum is set to 0.9, the weight decay is 0.0001, and the batch size is set to 64. Just as before, we adopt the linear scaling recipe (Goyal et al., 2017a).

Datasets. Kinetics-400 (Carreira & Zisserman, 2017) consists of $240K$ training videos and $20K$ validation videos that span 400 human action categories. Kinetics-600 (Carreira et al., 2018) has $392K$ training videos and $30K$ validation videos spanning 600 action categories. Something-Something-V2 (Goyal et al., 2017b) contains $170K$ training videos and $25K$ validation videos that span 174 action categories. Lastly, Diving-48 (Li et al., 2018) has $16K$ training videos and $3K$ testing videos spanning 48 fine-grained diving categories. For all of these datasets, we use standard classification accuracy as our main performance metric.

2. Additional Ablations

Smaller & Larger Transformers. In addition to the "Base" ViT model (Dosovitskiy et al., 2020), we also experimented with the "Large" ViT. We report that this yielded results 1%

[1]Facebook AI [2]Dartmouth College. Correspondence to: Gedas Bertasius <gberta@seas.upenn.edu>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021.

worse on both Kinetics-400, and Something-Something-V2. Given that our "Base" model already has $121M$ parameters, we suspect that the current datasets are not big enough to justify a further increase in model capacity. We also tried the "Small" ViT variant, which produced accuracies about 5% worse than our default "Base" ViT model.

Larger Patch Size. We also experimented with a different patch size, i.e., $P = 32$. We report that this variant of our model produced results about 3% worse than our default variant using $P = 16$. We conjecture that the performance decrease with $P = 32$ is due to the reduced spatial granularity. We did not train any models with P values lower than 16 as those models have a much higher computational cost.

The Order of Space and Time Self-Attention. Our proposed "Divided Space-Time Attention" scheme applies temporal attention and spatial attention one after the other. Here, we investigate whether reversing the order of time-space attention (i.e., applying spatial attention first, then temporal) has an impact on our results. We report that applying spatial attention first, followed by temporal attention leads to a 0.5% drop in accuracy on both Kinetics-400, and Something-Something-V2. We also tried a parallel space-time self-attention. We report that it produces 0.4% lower accuracy compared to our adopted "Divided Space-Time Attention" scheme.

3. Additional Qualitative Results

In Figure 1, we present space-time attention visualizations obtained by applying TimeSformer on Something-Something-V2 videos. To visualize the learned attention, we use the Attention Rollout scheme presented in (Abnar & Zuidema, 2020). Our results suggest that TimeSformer learns to attend to the relevant regions in the video in order to perform complex spatiotemporal reasoning. For example, we can observe that the model focuses on the configuration of the hand when visible and the object-only when not visible.

References

Abnar, S. and Zuidema, W. Quantifying attention flow in transformers, 2020.

Carreira, J. and Zisserman, A. Quo vadis, action recognition? A new model and the kinetics dataset. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, 2017.

Carreira, J., Noland, E., Banki-Horvath, A., Hillier, C., and Zisserman, A. A short note about kinetics-600. *CoRR*, 2018.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. An image is worth 16x16 words: Transformers for image recognition at scale. *CoRR*, 2020.

Fan, H., Li, Y., Xiong, B., Lo, W.-Y., and Feichtenhofer, C. Pyslowfast. https://github.com/facebookresearch/slowfast, 2020.

Feichtenhofer, C., Fan, H., Malik, J., and He, K. Slowfast networks for video recognition. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV*, 2019.

Goyal, P., Dollár, P., Girshick, R., Noordhuis, P., Wesolowski, L., Kyrola, A., Tulloch, A., Jia, Y., and He, K. Accurate, large minibatch sgd: Training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*, 2017a.

Goyal, R., Kahou, S. E., Michalski, V., Materzynska, J., Westphal, S., Kim, H., Haenel, V., Fründ, I., Yianilos, P., Mueller-Freitag, M., Hoppe, F., Thurau, C., Bax, I., and Memisevic, R. The "something something" video database for learning and evaluating visual common sense. *CoRR*, 2017b.

Li, Y., Li, Y., and Vasconcelos, N. Resound: Towards action recognition without representation bias. In *The European Conference on Computer Vision (ECCV)*, September 2018.

Wightman, R. Pytorch image models. https://github.com/rwightman/pytorch-image-models, 2019.

Figure 1. Visualization of space-time attention from the output token to the input space on Something-Something-V2. Our model learns to focus on the relevant parts in the video in order to perform spatiotemporal reasoning.