Optimizing Factorized Encoder Models: Time and Memory Reduction for Scalable and Efficient Action Recognition Supplementary

Shreyank N Gowda[1], Anurag Arnab[2], and Jonathan Huang[2]

[1] University of Oxford
[2] Google Research

1 ViViT hyperparameters

	K400	K600	MIT	Epic-Kitchens	SSv2	Selse
Optimisation						
Optimiser	Synchronous SGD					
Momentum	0.9					
Batch size	128					
Learning rate schedule	cosine with linear warmup					
Linear warmup epochs	2.5					
Base learning rate	0.1	0.1	0.25	0.5	0.5	0.5
Epochs	30	30	10	50	35	35
Data augmentation						
Random crop probability	1.0					
Random flip probability	0.5					
Scale jitter probability	1.0					
Maximum scale	1.33					
Minimum scale	0.9					
Colour jitter probability	0.8	0.8	0.8	-	-	-
Rand augment number of layers [4]	-	-	-	2	2	-
Rand augment magnitude [4]	-	-	-	15	20	-
Other regularisation						
Stochastic droplayer rate, p_{drop} [12]	-	-	-	0.2	0.3	-
Label smoothing [15]	-	-	-	0.2	0.3	-
Mixup [16]	-	-	-	0.1	0.3	-

Table 1: The hyperparameters utilized in the experiments conducted for the primary research paper are detailed here. If a regularisation method is not employed, it is represented by a "–". Constant values that are present across all columns are mentioned just once. For simplicity, abbreviations have been used to denote different datasets: Kinetics 400 is represented as K400, Kinetics 600 as K600, Moments in Time as MiT, Epic Kitchens as EK, Something-Something v2 as SSv2 and Something-Else as Selse.

In Table 1 we list the hyperparameters used for each dataset. For fair comparison we re-run SFA-ViViT using the same hyperparameters as ViViT. We

reproduce results using ViViT official codebase and the varying results are only due to changes in the video dataset (some videos on YouTube are removed over time).

2 Implementation Details

We use Scenic [6] for our implementation. Since we build on ViViT, we directly work on top of the codebase and stick to the default parameters used by ViViT in terms of hyperparameters. Full details of these can be found in supplementary.

Our adapter is a two-layer fully connected network that takes as input the output from the spatial transformer and the output from the adapter is passed as input to the temporal transformer.

The hyper-parameters of the transformer models are set to the standard: the number of heads are 12/16/16/16, number of layers are 12/24/32/40, hidden sizes are 768/1024/1280/1408 and MLP dimensions are 3072/4096/5120/6144 for the base/large/huge/giant versions respectively. The 8-frame ViViT model is trained for 30 epochs. We also experiment with initializing larger models with an 8-frame model trained for 10 epochs. Details of this can be found in the supplementary.

For our hardware, we use 64 v3 TPUs for all experiments. However, we also show results using 8 NVIDIA GeForce 2080 Ti (w/12 GB memory). This is a typical setting in a small academic lab.

3 Datasets

As Kinetics consists of YouTube videos which may be removed by their original creators, we note the exact sizes of our dataset.

Kinetics-400 [3]: Kinetics-400 is a large-scale video dataset with 400 classes introduced by Google's DeepMind. It has 235693 training samples and 53744 validation and test samples. The dataset encompasses various categories, such as object manipulation, human-object interaction, and body movements. Each class contains approximately 400 video samples, with each video lasting around 10 seconds.

Kinetics-600 [2]: Kinetics-600 is an extension of the Kinetics-400 dataset, with an increased number of classes, totaling 600 human action classes. This dataset contains approximately 380735 training samples and 56192 validation and test samples. The additional classes broaden the scope of the dataset, thereby providing more diverse training data for video recognition tasks.

EPIC Kitchens [5]: EPIC Kitchens is a large-scale dataset focusing on egocentric (first-person) videos of daily kitchen activities. It consists of 55 hours of video captured by 32 different participants in their own kitchens, with 67217 training samples and 22758 samples for validation and testing. The dataset includes 97 verb classes and 300 noun classes. Epic Kitchens is particularly useful for understanding human-object interactions and fine-grained actions in everyday settings.

Something-something v2 [11]: The Something-something v2 dataset is a collection of short video clips focused on common objects and human actions. It contains around 168913 training clips and 24777 test clips distributed across 174 action classes. This dataset aims to capture more abstract and high-level understanding of actions, as well as temporal relationships among objects.

Moments in Time [14]: The Moments in Time dataset is a large-scale video dataset containing one million short video clips, each lasting three seconds. It covers 339 classes of dynamic events and aims to provide a diverse set of visual and auditory representations of these events with 791297 training samples and 33900 test samples. This dataset is particularly useful for understanding the temporal aspects of various activities and events, as well as their associated contexts.

Something-else [13]: Something-Else utilizes the videos from SthSth-V2 as its foundation, and introduces novel training and testing partitions for two new tasks that examine the ability to generalize: compositional action recognition and few-shot action recognition. Our attention is solely on the compositional action recognition task, which aims to prevent any object category overlap between the 54919 training videos and the 57876 validation videos.

4 How important is the pre-training image dataset for action recognition performance?

While we know from the original ViViT paper [1] that using larger ViT [7] backbones result in better performances, we do a more thorough ablation here by considering variations of the ViT model such as the hybrid ViT (ResNet-ViT-L pre-trained on ImageNet21k [10]), ViT-L pre-trained on ImageNet21k, ViT-L pre-trained on JFT and ViT-H pre-trained on JFT. We report these results in Table 2, with the conclusion of larger backbones pre-trained on larger datasets yields highest accuracies. We report top-1 and top-5 accuracies on the Kinetics-400 dataset and we freeze the spatial transformer here without any fine-tuning or adapter. We also keep the temporal transformer fixed in size here for fair comparison. Essentially, the performance difference is purely from the output of the spatial transformer changing due to different backbones.

Backbone	16-frames	32-frames	48-frames
ResNet-ViT-L (ImageNet21k)	66.09/88.30	66.63/88.50	66.88/88.65
ViT-L (ImageNet21k)	65.59/85.86	68.34/87.80	70.09/88.91
ViT-L (JFT)	69.76/88.41	73.98/90.88	75.08/91.69
ViT-H (JFT)	73.68/90.23	75.85/91.53	77.90/92.72

Table 2: Comparison of impact of different backbones for the spatial transformer. We use ResNet-ViT-L pre-trained on ImageNet21k, ViT-L pre-trained on ImageNet21k, ViT-L pre-trained on JFT and ViT-H pre-trained on JFT. Listed as (Top-1 accuracy/ Top-5 accuracy).

5 Different Number of Frames Initialization Training

We consider variants of the training we talk about in the paper. There are various forms that we can consider. For instance, we can train the standard ViViT 8 frame model for just 10 epochs and use that to initialize our model. In the paper all initializations are done using 8 frame model trained for 30 epochs. Further, we could initialize smaller versions of SFA-ViViT like a 32 frame version for 10 epochs and then initialize SFA-ViViT 128 frames using this 32 frame version. We plot this in Figure 1 and see various versions and conclude that in the end, the best speed-accuracy trade-off was obtained when the standard ViViT 8 frame model was trained on 30 epochs and then the 128 frame model is initialized using this.

We define the models in the figure as follows:

- Model A: ViViT-L-8f for 10 epochs + SFA-ViViT-L-32f for 10 epochs + SFA-ViViT-L-128f for 10 epochs
- Model B: ViViT-L-8f for 10 epochs + SFA-ViViT-L-128f for 20 epochs
- Model C: ViViT-L-8f for 10 epochs + SFA-ViViT-L-128f for 30 epochs
- Model D: ViViT-L-8f for 30 epochs + SFA-ViViT-L-128f for 30 epochs
- Model E: ViViT-L-128f for 30 epochs

We see that training the ViViT-L-8f model for the full 30 epochs and then using that to initialize the SFA-ViViT-L-128f model gave us the best results. But we could potentially reduce the cost of training to 0.25x if we sacrifice 2% accuracy. All results are on the Kinetics400 dataset.

6 Improving accuracy by selecting frames

We consider frame selection approaches using attention and relational networks [9] and also using alternative approaches such as RL [8]. What we notice is that using this on the Stage I part of training did not significantly alter accuracy of Stage II, however, it adds a significant pre-training overhead. Doing it on Stage II, improves the overall accuracy marginally, but the associated cost significantly increases with the number of frames and hence this is something we decided not to pursue further.

7 How long do we need to train the model?

We showed in the paper that using SFA based initialization helps us reach "near-peak" performance really quickly. We define this near-peak performance as 1 % less than the eventual best performance of the model. Thus another natural question is: in order to save time, why not stop training the SFA version earlier? We note that although the standard ViViT model trains for 'x' epochs (see Table. 1 for exact number), it often reaches this "peak" performance much earlier and hence for fair comparison with the standard ViViT model, in the paper, we run on the same number of epochs. These results can be seen in Table 3.

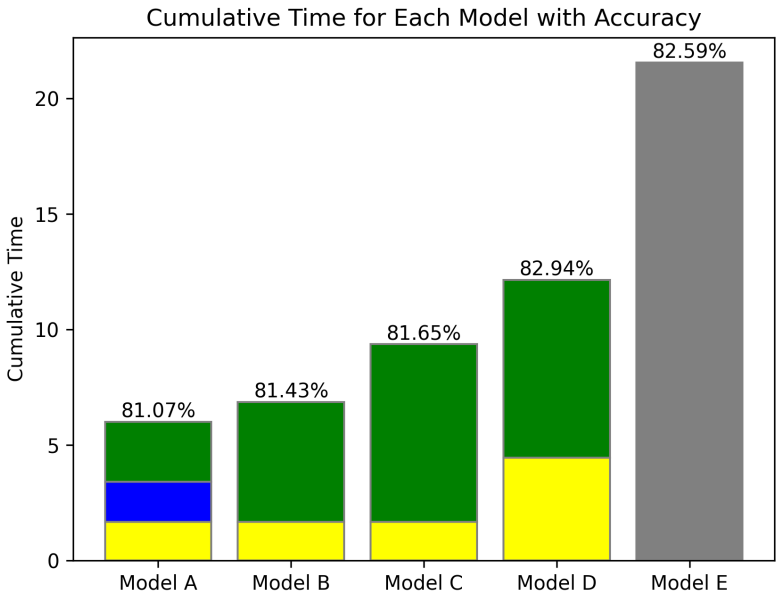

Fig. 1: Stacked bar chart representing the cumulative processing times of Models A-E. Each color within a bar corresponds to a specific sub-model ('a' in yellow, 'b' in blue, 'c' in green, 'd' in gray) contributing to the total computation time of each model. Model accuracies are indicated at the top of each respective bar. 'a' = ViViT-L-8f, 'b' = SFA-ViViT-L-32f, 'c' = SFA-ViViT-L-128f, 'd' = ViViT-L-128f. All results are using Kinetics-400 dataset and using ViViT-L variants.

Model	Dataset	NPP Epoch	Best Epoch
ViViT-L	K400	20	29
SFA-ViViT-L	K400	5	28
ViViT-L	K600	21	28
SFA-ViViT-L	K600	5	23
ViViT-L	SSv2	29	35
SFA-ViViT-L	SSv2	4	24

Table 3: Comparison of near peak performance (NPP) epoch and best performance epoch for ViViT and SFA-ViViT for different datasets and models. All results are on Kinetics400 dataset.

8 What about initializing standard ViViT models?

Since our method proposes an initialization scheme, we also test it on the standard ViViT models that do not have their spatial transformer frozen. In this particular scenario, we only want to check if the peak performance can be reached faster. However, it is important to note that with our proposed training scheme we also reduce the overall training time by close to half. This can be seen in Table 4.

Model	Dataset	NPP	Best
ViViT-L	K400	20	29
ViViT-L init with 8f ViViT-L	K400	4	25
ViViT-H	K400	22	27
ViViT-H init with 8f ViViT-H	K400	5	22

Table 4: Comparison of near peak performance (NPP) epoch and best performance epoch for initializing the full ViViT model with and without the 8f variant. We see the benefit of initialization as the “near-peak" performance is reached at a much earlier stage when initialized with the 8f variant. All results are on Kinetics400 dataset.

References

1. A. Arnab, M. Dehghani, G. Heigold, C. Sun, M. Lučić, and C. Schmid. Vivit: A video vision transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6836–6846, 2021. 3
2. J. Carreira, E. Noland, A. Banki-Horvath, C. Hillier, and A. Zisserman. A short note about kinetics-600. *arXiv preprint arXiv:1808.01340*, 2018. 2
3. J. Carreira and A. Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017. 2
4. E. D. Cubuk, B. Zoph, J. Shlens, and Q. V. Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 702–703, 2020. 1
5. D. Damen, H. Doughty, G. M. Farinella, S. Fidler, A. Furnari, E. Kazakos, D. Moltisanti, J. Munro, T. Perrett, W. Price, et al. Scaling egocentric vision: The epic-kitchens dataset. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 720–736, 2018. 2
6. M. Dehghani, A. Gritsenko, A. Arnab, M. Minderer, and Y. Tay. Scenic: A jax library for computer vision research and beyond. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21393–21398, 2022. 2
7. A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. An image is

worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 3

8. S. N. Gowda. Synthetic sample selection for generalized zero-shot learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 58–67, 2023. 4
9. S. N. Gowda, M. Rohrbach, and L. Sevilla-Lara. Smart frame selection for action recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 1451–1459, 2021. 4
10. S. N. Gowda and C. Yuan. Colornet: Investigating the importance of color spaces for image classification. In *Computer Vision–ACCV 2018: 14th Asian Conference on Computer Vision, Perth, Australia, December 2–6, 2018, Revised Selected Papers, Part IV 14*, pages 581–596. Springer, 2019. 3
11. R. Goyal, S. Ebrahimi Kahou, V. Michalski, J. Materzynska, S. Westphal, H. Kim, V. Haenel, I. Fruend, P. Yianilos, M. 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. 3
12. G. Huang, Y. Sun, Z. Liu, D. Sedra, and K. Q. Weinberger. Deep networks with stochastic depth. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14*, pages 646–661. Springer, 2016. 1
13. J. Materzynska, T. Xiao, R. Herzig, H. Xu, X. Wang, and T. Darrell. Something-else: Compositional action recognition with spatial-temporal interaction networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1049–1059, 2020. 3
14. M. Monfort, A. Andonian, B. Zhou, K. Ramakrishnan, S. A. Bargal, T. Yan, L. Brown, Q. Fan, D. Gutfreund, C. Vondrick, et al. Moments in time dataset: one million videos for event understanding. *IEEE transactions on pattern analysis and machine intelligence*, 42(2):502–508, 2019. 3
15. C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016. 1
16. H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017. 1