Supplementary Material: Are Attention Maps Richer than we Imagined for Action Recognition?

Tanay Agrawal* Abid Ali* Antitza Dantcheva Francois Bremond
INRIA Sophia Antipolis – Méditerranée, France
{firstname.lastname}@inria.fr

A. Foundation Models and Adding Adapters

Today, most of the foundation models in computer vision are transformer-based. This work is exclusively focused on these. As stated above, we choose ViT trained using Dinov2 for this work. Adapters are added to an individual transformer encoder/block, making this method applicable to all model architectures. For completeness, we will start with a background for our choice of adapters.

As discussed in the previous section, there are many variations of adapters. Most of them use the basic block of adapters and change how it is used according to need - where they are placed and also the architecture of the block itself. This basic block is: $Adapter(x) = x + f(xW_{down})W_{up}$. f is usually $ReLU$. We choose our adapters based on the functions we want them to serve. We have two requirements from an adapter:

- Adapting to the spatial distribution of the dataset to be trained on.
- Providing a downsampled embedding to be used as input to the temporal processing module.

A.0.1 Adapting to a new spatial distribution

[6] uses the basic adapter block in two settings: serial and parallel. They show that parallel adapters work well for spatial adaptation and serial for temporal adaptation.

Following experiments, we concur that for our purpose parallel adapters work better. Specifically, we use scaled parallel adapters [4], which is a better choice over parallel adapters. They can be written as:

$$Adapter(x) = x + s.f(xW_{down})W_{up} \quad (1)$$

Here, s is a learnable scalar and f is the activation function.

A.0.2 Embedding for temporal reasoning

The basic block of adapters works well for this purpose as they inherently provide a downsampled embedding. Parallel adapters only need to learn the change in distribution to be added to the pretrained model. On the other hand, serial adapters accommodate this change along with the information that passes through the model. Therefore, for our requirements, serial adapters are traditionally believed to work better as the information in the bottleneck layer is more rich. However, the addition of AM flow forces parallel adapters to learn substantial information passing through the network too, this is discussed in section 4.3.

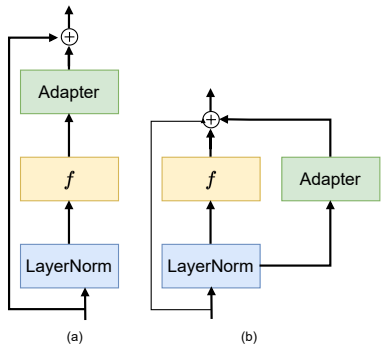

Figure S1: (a) Serial Adapters. (b) Parallel Adapters

B. Datasets

We evaluate our model on three datasets. Something-something v2 [3] is a dataset with fixed cameras having different angles and multiple objects of interest. Kinetics-400 [1] is a datasets with moving cameras, and there is a single person of interest. Toyota Smarthome [2] is a dataset with a fixed camera and one target person. The details of the datasets are as follows.

Something-Something v2 (SSv2) is a more challenging dataset, since it requires strong temporal modelling. It contains about 168.9K training videos and 24.7K validation videos for 174 classes.

Kinetics-400 (K400) is a large dataset. It contains about 240K training videos and 20K validation videos for 400 human action categories. Each video is trimmed to have a

length of 10 seconds. Although the K400 dataset provides a wide range of categories, they are known to be highly biased in spatial appearance.

Toyota Smarthome (Smarthome) is a small dataset. It contains 16.1k video clips and 31 complex daily-life activities performed naturally without strong prior instructions. For the evaluation of this dataset, we follow the cross-subject (CS) protocol proposed in [2].

C. Location of AM Flow addition.

The experiment on the Smarthome dataset is carried out to answer this, Table 3 of the main paper. Adapters with AM flow and temporal processing modules are only added to the first and last blocks of the transformer. Traditional adapters are added to the rest of the blocks. We optimised the addition of these modules only for this dataset, to demonstrate that it is possible. The method to do this was to visualise AM flow by taking PCA along the channel dimension to reduce it to 3 dimensions (corresponding to RGB). Figure S2 clarifies that the first and the last blocks produce visualisations close to what AM flow is supposed to signify, as discussed in previous sections.

For bigger datasets like K400 and SSv2, we need to add AM flow and the temporal processing module to more number of backbone blocks, but for a small dataset like smarthome, a small number of blocks are enough and thus the trainable parameters are reduced. This alleviates the challenge of having a large model and less training data.

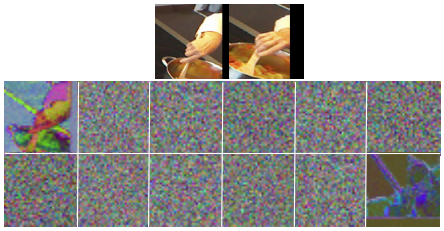

Figure S2: Computed AM flow for two frames (on top) from Smarthome. Starting from top-left, going row-wise, AM flow is visualised for each transformer block in ViT-B from beginning to the end. The figure shows that we do not need to add AM flow to each layer and here for example, only the first and last layer are important.

D. Affect of aligning encoder

We further visualised the impact of aligning encoder on computing AM flow. In Figure S3 AM flow is visualised for each transformer block in ViT-B from beginning to end. Without an aligning encoder the computed AM flow is noisy due to a change in both action and motion, as demonstrated in Figure S3(b). This is solved by incorporating an aligning encoder as illustrated in Figure S3(a).

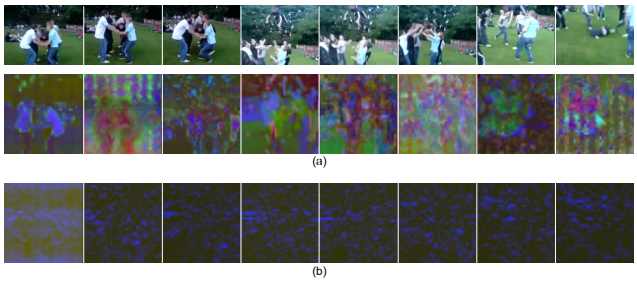

Figure S3: Computed AM flow for faceplanting action from K400 dataset. The impact of aligning encoder is visualised in (a) compared to AM flow without aligning encoder (b).

Method	Backbone	Top-1	Top-5
Ours-AM/12	ViT-B(Dinov2)	88.8	98.2
Ours-AM/16	Hiera-B	89.3	98.3

Table S1: Comparison of Dinov2 [5] backbone with Hiera [7] on K400 dataset.

E. Change of backbone

Our proposed architecture is compatible with other large-scale pretrained backbones. To demonstrate this, we use Hiera [7] as our backbone. We conduct experiments and compare this new backbone with our chosen Dinov2 [5] as shown in Table S1. This demonstrate that our method is applicable to other large-scale models.

If the model does not converge with simple training, like it did not for hiera, it is necessary to first train with the addition of AM flow in only a few initial and final blocks. Then the entire model should converge when initialised with this.

References

[1] 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] S. Das, R. Dai, M. Koperski, L. Minciullo, L. Garattoni, F. Bremond, and G. Francesca. Toyota smarthome: Real-world activities of daily living. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 833–842, 2019.

[3] 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.

[4] J. He, C. Zhou, X. Ma, T. Berg-Kirkpatrick, and G. Neubig. Towards a unified view of parameter-efficient transfer learning. In *International Conference on Learning Representations*, 2022.

[5] M. Oquab, T. Darcet, T. Moutakanni, H. V. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, R. Howes, P.-Y. Huang, H. Xu, V. Sharma, S.-W. Li, W. Galuba, M. Rabbat, M. Assran, N. Ballas, G. Synnaeve, I. Misra, H. Jegou, J. Mairal, P. Labatut, A. Joulin, and P. Bojanowski. Dinov2: Learning robust visual features without supervision, 2023.

[6] J. Park, J. Lee, and K. Sohn. Dual-path adaptation from image to video transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2203–2213, 2023.

[7] C. Ryali, Y.-T. Hu, D. Bolya, C. Wei, H. Fan, P.-Y. Huang, V. Aggarwal, A. Chowdhury, O. Poursaeed, J. Hoffman, J. Malik, Y. Li, and C. Feichtenhofer. Hiera: A hierarchical vision transformer without the bells-and-whistles. *ICML*, 2023.