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

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

[1] University of Oxford
[2] Google Research

Abstract. In this paper, we address the challenges posed by the substantial training time and memory consumption associated with video transformers, focusing on the ViViT (Video Vision Transformer) model, in particular the Factorised Encoder version, as our baseline for action recognition tasks. The factorised encoder variant follows the late-fusion approach that is adopted by many state of the art approaches. Despite standing out for its favorable speed/accuracy tradeoffs among the different variants of ViViT, its considerable training time and memory requirements still pose a significant barrier to entry. Our method is designed to lower this barrier and is based on the idea of freezing the spatial transformer during training. This leads to a low accuracy model if naively done. But we show that by (1) appropriately initializing the temporal transformer (a module responsible for processing temporal information) (2) introducing a compact adapter model connecting frozen spatial representations (a module that selectively focuses on regions of the input image) to the temporal transformer, we can enjoy the benefits of freezing the spatial transformer without sacrificing accuracy. Through extensive experimentation over 6 benchmarks, we demonstrate that our proposed training strategy significantly reduces training costs (by) and memory consumption while maintaining or slightly improving performance by up to 1.79% compared to the baseline model. Our approach additionally unlocks the capability to utilize larger image transformer models as our spatial transformer and access more frames with the same memory consumption. We also show the generalization of this approach to other factorized encoder models. The advancements made in this work have the potential to advance research in the video understanding domain and provide valuable insights for researchers and practitioners with limited resources, paving the way for more efficient and scalable alternatives in the action recognition field.

1 Introduction

Action recognition focuses on understanding and identifying actions in video sequences with applications in surveillance, human-computer interaction, and video content analysis. The field has advanced significantly due to large-scale annotated datasets [5] and a shift from hand-crafted features [24, 41] to deep

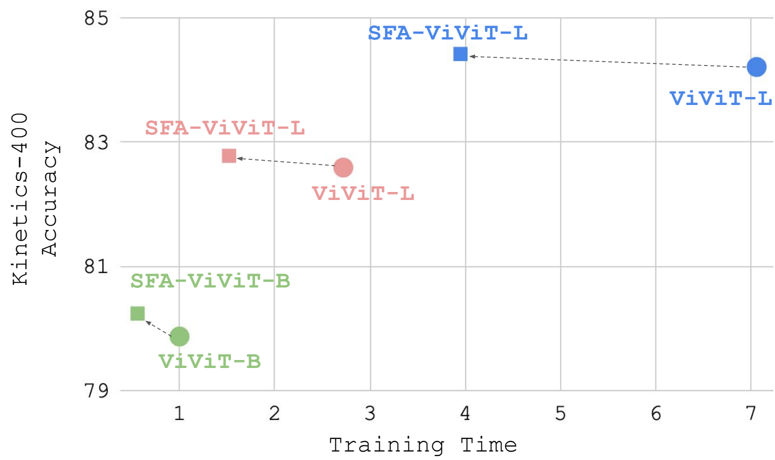

Fig. 1: Comparison of our initialization method vs conventional training of ViViT. Training time is scaled relative to setting ViViT-B training time to '1 unit' (7.93 hours). We see clear time saving using our initialization scheme and for larger models, the training time saved is much larger.

learning models like convolutional networks (CNNs) [5, 13, 22, 36]. Recently, transformers have revolutionized computer vision by offering an alternative to traditional CNNs, leading to the development of many new state-of-the-art architectures [3, 10, 28]. Moreover, the flexibility of transformers have inspired researchers to adapt these models to more complex problems, including emotion recognition [12, 14], video understanding and action recognition [1, 29].

Transformers, however, are notoriously expensive [15], and Video Transformer-based architectures [1, 2, 29], which integrate information across space and time are even more so. And memory consumption and training times become even more significant when working with large-scale video datasets with long sequences [5]. These high computational costs present a particular challenge for researchers with limited resources, especially those from universities and smaller companies. The goal of our work is therefore to cut the cost of training: we want to train transformer-based video models with fewer resources or use larger model variants and handle more frames with the same resources.

Our primary focus is on a diverse range of factorized encoder models, among which ViViT [1] stands out as a prominent example. Specifically, we focus on the "factorized encoder" variant of ViViT which has separate spatial and temporal transformer stages, where the spatial transformer is responsible for extracting features from individual frames, while the temporal transformer processes the temporal dynamics across frames. We choose this factorized encoder design because it is more efficient compared to, e.g., the variant of ViViT using all-to-all spatiotemporal attention, while still achieving high accuracies and has thus been adopted as the building block for recent state-of-the-art architectures on various tasks [6, 19, 42, 46–48].

To address the challenge of reducing training time and memory usage without compromising the sophistication and accuracy of the original model, our approach is based on the simple idea of freezing the spatial backbone. Freezing the spatial backbone has many advantages: by not backpropagating through this transformer, training is faster and requires less memory (allowing for the model to handle more frames). We also inherit the benefits of pretraining the spatial transformer on a large dataset (such as JFT [20]). Naively implemented, however, we show that this approach falls very short in accuracy. Instead, with a few simple (but important) tweaks to the above idea, we propose a method that has the same advantages of freezing the spatial transformer, but does not compromise on accuracy.

Our method proceeds in two stages. In the first stage we pretrain a cheap version of the model using fewer frames, e.g., 8 frames as opposed to, e.g., 32 frames. In the second stage we fine tune this model with more frames, which is more expensive but in this stage we freeze the spatial encoder and introduce a compact "adapter" model connecting frozen spatial representations to the temporal transformer, negating the need for end-to-end training of the spatial transformer. Critically this includes pre-training the temporal transformer (by initializing from stage 1) which is often overlooked in current video models which typically initialize this component from scratch. However our experiments show that this step is critical if we wish to not sacrifice performance.

With our training recipe, we match or slightly outperform conventional training of ViViT at roughly half the cost as seen in Figure 1. A notable benefit of our training recipe, is its ability to process up to 80 frames (ViViT-B) on typical university-grade GPUs, a significant leap from the previous capacity of 16 frames. This expansion in processing power broadens the range of video data manageable under resource-constrained settings. As we elaborate in Section 4.7, our research underscores the potential to democratize access to advanced video transformer models. Another notable benefit is the model's ability to now use even larger models as the spatial transformer, we introduce ViViT-g as seen in Section 4.6. This accessibility paves the way for future video action recognition research, irrespective of resource constraints. We also show in Section 4.9 that our method generalizes to non-ViViT models such as Dualformer [26] and VideoSwin [29]. Hereafter, we refer to our version of ViViT as SFA-ViViT, where SFA denotes '**S**patial **F**rozen and **A**dapter Initialized'.

2 Related Work

Transformers for Videos Action recognition is a key research area in computer vision, addressed by many traditional [24, 41] and CNN based approaches [5, 16, 17, 21, 27, 36, 39, 44] aided by the release of large-scale datasets [5, 23, 37]. Since we focus on transformer based architectures, a thorough review of earlier methods are out of this scope. More recently, the transformer architecture, initially developed for NLP tasks [40], has been adapted for video understanding and action recognition tasks, leading to state-of-the-art models such as TimeSformer [2], ViViT [1], VideoSwin [29], and Uniformer [25] These transformer-based models leverage self-attention mechanisms to capture complex spatiotemporal patterns in action recognition tasks. TimeSformer [2] is one of the first transformer-based models for video understanding, adapting the transformer architecture to video by treating it as a sequence of flattened image patches. ViViT [1] integrates spatial and temporal transformers to efficiently capture spatiotemporal information in video sequences. VideoSwin [29] is a hierarchical transformer that applies local windowing for efficiency, enabling the model to handle longer video sequences. Similar to VideoSwin, Dualformer [26] also adopts a hierarchical architecture but uses a local-global stratified transformer design to capture both fine-grained and global video features efficiently. More recently, Uniformer [25] integrates 3D convolution and spatiotemporal self-attention, MTV [47] proposes a multi-view transformer model using distinct encoders for each video "view", improving accuracy as the number of views increases. The Multiscale Vision Transformers (MViT) [11] model streamlines computation and memory usage by operating at different resolutions, focusing on high-level features at lower resolutions and low-level details at higher ones, effectively leveraging both spatial and temporal information in visual tasks. TubeViT [33] introduces a method of sparsely sampling different-sized 3D segments from videos, facilitating efficient joint image and video learning, and allowing the adaptation of larger models to videos with less computational resources. Typically these models have FLOPs in the range of TFLOPs and training times that last more than days on the largest of GPUs/TPUs available, making them infeasible to train or use in lower resourced settings such as academia. It is critical that we find a way to train these models with limited resources while maintaining their performance. To this end, we focus on the factorised encoder version of ViViT as the late-fusion approach followed is used as a foundation for state-of-the-art approaches of various tasks [6, 19, 42, 46–48] and hence believe that the initialization scheme proposed can be used for future methods working on similar architectures. Similar to the approach for ViViT factorized encoder, the hierarchical architectures of VideoSwin and DualFormer can also benefit from staged training. By freezing the early layers that capture local spatial-temporal features, and adding an adapter before fine-tuning the later layers focused on global context, the overall training can be made more efficient.

Efficient Transformers in Videos Efficiency is a nuanced topic [9], as there are multiple cost indicators of efficiency (for example, GFLOPs, inference time, training time, memory usage), and models which improve efficiency in one dimension, are not necessarily better in other dimensions [9]. TokenLearner [35] proposes a method that adaptively learns tokens for efficient image and video understanding tasks, enabling effective modeling of pairwise attention over longer temporal horizons or spatial content. TokenLearner reduces the GFLOPs required by ViViT by about half, but does not significantly change the training time or the inference time of ViViT. Spatial Temporal Token Selection (STTS) [43] proposes a dynamic token selection framework for spatial and temporal dimensions that ranks token importance using a lightweight scorer network, selecting top-scoring tokens for downstream evaluation in an end-to-end training process. STTS again reduces the GFLOPs, but the training time and inference time do not change significantly. TokShift [49], a zero-parameter, zero-FLOPs operator that models temporal relations in transformer encoders by temporally shifting partial token features across adjacent frames but again requires the same training time as the original model. By densely integrating TokShift into a plain 2D vision transformer, a computationally efficient, convolution-free video transformer is created for video understanding. Most similar to our work is the ST-Adapter [32], that utilizes built-in spatio-temporal reasoning in a compact design, allowing pre-trained image models to reason about dynamic video content with a small per-task parameter cost, surpassing existing methods in both parameter-efficiency and performance. However, it does not change FLOPs or inference time at all. Unlike ST-Adapter, we use a spatial only adapter which we show is enough to reproduce the performance of the baseline model at close to half the training time. In particular, our proposed method improves the training time and training memory usage, addressing the key problem of researchers and practitioners being able to train video models. It does not, however, change the inference time compared to a standard ViViT model. We consider overall train time for the same hyperparameters and use the same hardware for a direct comparison. We consider efficiency in this paper as the time saved in the overall training of the model.

3 Methodology

3.1 Revisiting ViViT

The Video Vision Transformer (ViViT) extends the Vision Transformer architecture to handle video data by incorporating spatio-temporal reasoning. The idea behind ViViT is to process video input as a sequence of image patches, combining spatial and temporal information through a series of transformer layers, which include multi-head self-attention, layer normalization, and feed-forward networks. The output is used for video classification.

In the "vanilla" variant of ViViT, one extracts spatio-temporal tokens from a video then forwards all tokens through a transformer encoder which explicitly models all pairwise interactions between all spatio-temporal tokens. We build off of the more efficient "Factorized Encoder" variant of ViViT whose architecture consists of two separate transformer encoders, a *spatial transformer* modeling interactions between tokens from the same temporal index and a *temporal transformer* modeling interactions between tokens from different temporal indices. Despite having more parameters, it requires fewer floating point operations (FLOPs) than vanilla ViViT. Because the Factorised Encoder variant strikes a good balance point between accuracy and processing speed, it has also been adopted as the foundation for other architectures [6,19,42,46–48], reinforcing its utility and robustness.

3.2 Our training strategy

We concentrate on the factorised encoder variant of ViViT as it is already the most efficient version of the baseline. Henceforth, when we talk about ViViT we refer to this variant of ViViT. Consider the ViViT model that contains a spatial transformer with parameters $\theta_{spatial}$ and a temporal transformer with parameters $\theta_{temporal}$:

$$\begin{aligned} X_{spatial} &= T_{spatial}(X_{in}; \theta_{spatial}) \\ X_{out} &= T_{temporal}(X_{spatial}; \theta_{temporal}). \end{aligned} \tag{1}$$

In conventional ViViT training, $\theta_{spatial}$ is initialized from an image pre-trained checkpoint such as ImageNet-21k [34] or JFT [20] and the $\theta_{temporal}$ is initialized from scratch. During backpropagation, the gradient flows through the entire model. This entails training two sizable transformer models end-to-end, which is a highly resource-intensive process, as the transformer architecture is inherently computationally demanding, especially with more frames and larger ViViT variants (e.g., ViViT-H).

One approach to reducing training time is to freeze the parameters of the spatial transformer $\theta_{spatial}$. By not backpropagating through $\theta_{spatial}$, gradient updates are faster and require less memory, allowing us to access more frames without encountering out-of-memory issues. But as we show in experiments, the accuracy of the resulting model with frozen $\theta_{spatial}$ is not competitive (in accuracy) with the baseline training approach.

We present a two stage approach (see Fig. 2) to training ViViT models that inherits the same benefits of freezing the spatial transformer, while not compromising on model quality.

Stage 1. In Stage 1, we pretrain our ViViT model on a reduced number of frames initializing the spatial transformer using a pre-trained image checkpoint. We do not freeze the spatial transformer during this stage, but critically, Stage 1 serves to also initialize the temporal transformer.

To set the number of frames at this stage, we must balance the goal of efficiency (using fewer frames) against our finding in experiments that pre-training on too few frames can lead to suboptimal results. In our ablations, we identify a sweet spot at 8 frames.

Stage 2. In Stage 2, we fine tune our ViViT model on the full frame count (e.g. 128 frames) initializing both spatial and temporal transformer parameters learned in Stage 1. Because this stage is significantly more expensive, in stage 2, we freeze the spatial transformer parameters $\theta_{spatial}$ and add a lightweight adapter module with parameters $\theta_{adapter}$ following the spatial transformer:

$$\begin{aligned} X_{spatial} &= T_{spatial}(X_{in}; \theta_{spatial}) \\ X_{adapter} &= A_{adapter}(X_{spatial}; \theta_{adapter}) \\ X_{out} &= T_{temporal}(X_{adapter}; \theta_{temporal}) \end{aligned} \tag{2}$$

In this setting, by backpropagating only through the temporal transformer and the lightweight adapter module (in our experiments, a two layer MLP), we effectively cut total training time by half.

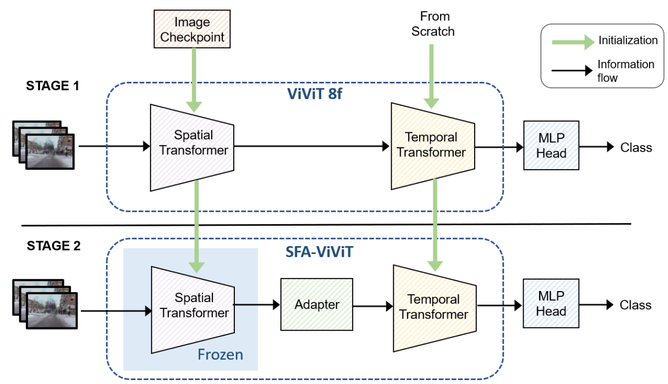

Fig. 2: STAGE 1: We first use the full ViViT-FE model on 8 frames by initializing the spatial transformer from an image checkpoint and the temporal transformer from scratch. STAGE 2: We then use this as our checkpoint to initialize the spatial and temporal transformer for models using more frames (such as 32, 64 or 128). We then freeze the spatial transformer and add an adapter model to finetune spatial transformer features. The temporal transformer is finetuned from the same checkpoint.

The crucial finding here is that the spatial transformer requires only short-term context for initialization (after which it remains frozen), whereas the temporal transformer necessitates long-term context to achieve its optimal performance. Further details and empirical analysis can be found in the next section.

4 Experimental Analysis

Through a series of comprehensive experiments which we now present, we investigate the significance of the spatial transformer, examining the impact of pre-training datasets and how larger models affect action recognition performance. We also explore the importance of initializing the temporal transformer by employing various initialization schemes and datasets, assessing whether the number of frames is critical for initializing larger models, initializing full ViViT models, and initializing models on one dataset while fine-tuning on another.

4.1 Datasets

We evaluate on all the datasets considered in [1] (specifically, Kinetics-400 [5], Kinetics-600 [4], EPIC-Kitchens [8], Something-something v2 [18] and Moments-in-time [31]) as well as the Something-Else [30] dataset. As these datasets are common in the community, we include further details in the supplementary along with the implementation details.

4.2 Ablation Study

We first address two critical aspects: the significance of fine-tuning the spatial transformer and the importance of initializing the temporal transformer. To do so, we conduct a series of experiments in various scenarios, which are detailed below. Our analysis focuses on the Something-something dataset, utilizing the large version of the ViViT model, referred to as ViViT-L.

We examine four main elements that modify the structure of conventional ViViT training and these are mentioned with indices in Table 1 namely: I. The freezing of the spatial transformer ($\theta_{spatial}$ is initialized and then frozen), II. The freezing of the temporal transformer ($\theta_{temporal}$ is frozen), III. The addition of an adapter (lightweight module with parameters $\theta_{adapter}$), IV. Next, we initialize the temporal transformer using VideoMAE [38], while keeping the spatial transformer frozen and the adapter incorporated and V. The initialization of the temporal transformer ($\theta_{spatial}$ and $\theta_{temporal}$ are initialized using the ViViT$_{8f}$).

It is important to note that the VideoMAE training is an extremely expensive process as can be seen in the table. But combined with the line below it, these two models, which significantly outperform lines I, II and III, show that properly initializing the temporal transformer is critical.

Index	Spatial Frozen	Adapter	Temporal Frozen	Temporal Init	Top-1 Acc	Top-5 Acc	Train Time
-	×	×	×	×	64.45	87.48	14.17 h
I	✓	×	×	×	27.75	56.73	0.5x()
II	×	×	✓	×	25.80	53.07	0.5x()
III	✓	✓	×	×	38.77	68.93	0.53x()
IV	✓	✓	×	$VMAE$	58.54	85.83	2.51x()
V	✓	✓	×	$ViViT-8f$	63.85	**87.62**	**0.62x()**
VI	×	✓	×	$ViViT-8f$	**64.13**	87.35	1.37x()

Table 1: Ablation study results illustrating the impact of various modifications to the ViViT-B model, including spatial and temporal transformer freezing, adapter addition, and initialization methods, on top-1 and top-5 accuracy. Dataset is Something-something v2.

Additionally, initializing the spatial transformer yields further improvement. The adapter plays a vital role in augmenting performance when the spatial transformer is frozen, and due to its lightweight nature, it will be an essential component of our training methodology.

4.3 How many frames should we use for Stage 1?

Next we experiment with various frame counts for stage 1 training, We test seven variants: JFT [20] checkpoint (image-based), 2, 4, 8, 16, 32, and 48-frame ViViT checkpoints. We then fine-tune these with a frozen spatial transformer and add

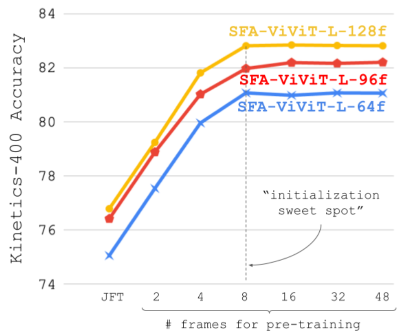

Fig. 3: The effect of initializing with different numbers of frames (JFT, 2, 4, 8, 16, 32, and 48), freezing the spatial transformer and adding an adapter model and fine-tuning using 64, 96, and 128 frames. Results on Kinetics400 dataset, 'f' refers to frames.

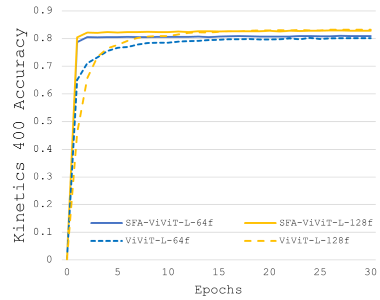

Fig. 4: Comparison of our initialization method vs conventional training of ViViT on Top-1 accuracy and loss on the Kinetics-400 dataset using 64 and 128 frames. We see that our initialization gives a significant headstart to the models.

an adapter model using 64, 96, and 128 frames (see Figure 3). Results show that using too few frames for Stage 1 training can underperform (with image-only initialization from a JFT [20] checkpoint performing the worst). Thus we deduce that short term temporal context is essential for initializing the spatial transformer. Performance also plateaus after 8 frames, and given that using more frames increases training time, we settle on using 8 frames as our "sweet spot" for Stage 1 training. We consider this an ablation, as this is done only on Kinetics and kept *fixed for all models and datasets.*

4.4 Does the proposed training increase convergence speed?

Another potential question that may arise concerns the impact of this initialization method in terms of convergence speed, if any. This specific aspect holds considerable significance owing to its potential ability to drastically curtail the duration of time required for training and the number of epochs necessary to effectively train the model. Moreover, an important element to take into account is the effect of freezing the spatial transformer. This approach decreases the memory needed to store the model but also considerably enhances the training speed. To provide a clearer picture, we have plotted the validation curves with and without initialization, which can be seen in Figure 4. Note that with the proposed initialization, we get a significant head start in overall accuracy.

4.5 What about initializing on one dataset and finetuning on other?

In our study, we find training an 8-frame version of the standard ViViT model affordable. We consider training this on a temporally dependent dataset like

Something-something, then fine-tuning on other datasets like Kinetics-400. We examine two scenarios: first, the standard ViViT model trained on Kinetics-400 using 8-frames, and second, the same model trained on Something-something. Post-training, we freeze the spatial transformer, add an adapter, and fine-tune the models on the alternate dataset with more frames. We contrast this with models fine-tuned on their original datasets (see Table 2). Results favor initializing larger frame models from the 8-frame version on the same dataset. Thus, for the final comparison in Sec 4.8, we initialize models with 8-frame versions of the baseline.

Checkpoint	SSv2	K400
K400-init	44.71/74.53	**82.81**/**93.98**
SSv2-init	**63.85**/**87.62**	76.79/92.35

Table 2: A summary of cross-dataset initialization of the proposed model and performance comparison. We use Kinetics400 and Something-something v2 as our datasets.

Backbone	Top-1	Top-5	Steps
ViViT-L	79.64	91.73	48k steps
ViViT-H	81.02	93.09	39k steps
ViViT-g	**81.81**	**94.55**	29k steps

Table 3: A comparison of top-1 and top-5 accuracies for the ViViT-g model with the proposed training strategy, which incorporates a larger spatial transformer backbone. All models use 48 frames for fair comparison. Results are on Kinetics400 dataset.

4.6 Extending the image backbone to 1.5B parameters

An intriguing consequence of our approach is the ability to incorporate larger backbones into the spatial transformer, made possible by the additional memory available to us as a result of freezing the spatial transformer during training. Consequently, we introduce ViViT-g, which integrates the ViT-g model (with 1.5B parameters) as its backbone. To ensure a fair comparison, we focus solely on training and inference using 48 frames, and abstain from employing multiview or multicrop testing. Our objective is to investigate the potential impact of a more substantial spatial transformer backbone on the overall performance and show the potential of larger spatial backbones that are possible due to our training process. It is essential to note that the full ViViT-g model could not process more than 8 frames due to memory limitations. However, our proposed strategy allows processing up to 48 frames. A comparison of accuracies is presented in Table 3 along with the number of steps needed to reach the best performance. Dataset used is Kinetics-400 and all the ViT checkpoints are JFT-pretrained [20].

4.7 Comparison of Memory Usage with Standard ViViT Training and Proposed Method

In this study, we compare the number of frames that can be accessed using the standard ViViT training scheme against our proposed scheme, employing a set of 64 v3 TPUs that have 16 GB each. We further evaluate the performance of ViViT variants, including H, and g, in comparison with the SFA-ViViT using the same variant configurations. Maintaining identical hyperparameters, we ensure a local batch size of 1. Our findings indicate that the conventional ViViT training approach restricts frame accessibility to 96 frames for the ViViT-H model, and a mere 8 frames for the ViViT-g model, before reaching memory limitations. Conversely, our proposed method enables access to 128 frames for ViViT-H, and up to 48 frames when utilizing ViViT-g with the same hardware. Furthermore, we investigate the impact of utilizing university grade GPUs by conducting ViViT experiments on an NVIDIA RTX 2080 Ti GPU farm equipped with 8 GPUs having 12 GB each. Under these circumstances, ViViT can only process 16 frames using a local batch size of 1. However, our proposed training strategy enables a notable improvement, expanding the frame capacity to 80 frames (ViViT-B) helping us reproduce ViViT results on lower end GPUs. This enhancement provides a valuable opportunity for researchers with limited resources to attain performance levels comparable to those with extensive resources. We show a comparison of number of frames accessible with and without our training recipe in Table 4.

Model	4	8	16	32	48	64	96	128
ViViT-H	✓	✓	✓	✓	✓	✓	✓	×
SFA-ViViT-H	✓	✓	✓	✓	✓	✓	✓	✓
ViViT-g	✓	✓	×	×	×	×	×	×
SFA-ViViT-g	✓	✓	✓	✓	✓	×	×	×

Table 4: Memory usage in different ViViT training schemes is compared using the Kinetics400 dataset on 64 TPUs v3 with 16GB memory each. A ✓ indicates accessible frames given hardware constraints, while a × signals an out-of-memory (OOM) error.

4.8 Comparison on all benchmarks to the baseline model

In this section, we present a comprehensive comparative analysis, focusing on the proposed approach and the baseline model. We report the Top-1 accuracy, Top-5 accuracy and the overall training time. The evaluation is conducted on the large and huge variants of ViViT across three datasets, namely Kinetics400, Kinetics600, and Moments in Time (MiT), with the summarized results tabulated in Table 5. The findings indicate a slight enhancement in accuracy for both Kinetics400 and Kinetics600 datasets, whereas a notable 1.79% increase in top-1 accuracy is observed for the MiT dataset using the proposed method. Furthermore, the proposed approach showcases a significant reduction in training time, accounting for approximately 56% of the original duration. This reduction emphasizes the advantageous nature of the proposed approach. To calculate the total training time for the SFA version, the train time of the 8 frame (Stage 1) ViViT model is combined with the train time of the (Stage 2) SFA-ViViT model. Conversely, the total training time for the standard ViViT encompasses the total train time for the same number of frames that SFA-ViViT is trained on for fair comparison.

Model	Kinetics-400		Kinetics-600		Moments in Time	
	Accuracy	Train Time	Accuracy	Train Time	Accuracy	Train Time
ViViT-L_{128f}	82.59/93.09	1x(21.57 h)	83.29/**95.82**	1x(26.14 h)	-	-
ViViT-L_{128f} + $\mathbf{SFA}_{8f}$	**82.78/94.03**	**0.56x()**	**83.47**/95.29	**0.56x()**	-	-
ViViT-H_{96f}	84.21/94.66	1x(56.71 h)	84.18/95.68	1x(60.45 h)	38.17/62.84	1x(110.79 h)
ViViT-H_{96f} + $\mathbf{SFA}_{8f}$	**84.42/94.72**	**0.57x()**	**84.39/96.20**	**0.57x()**	**39.96/64.39**	**0.59x()**

Table 5: Performance Comparison of various versions of ViViT with the proposed training strategy for Kinetics-400, Kinetics-600 and Moments in Time. Accuracies listed as Top-1/Top-5.

Model	Something-something		Something-Else		Epic-Kitchens	
	Accuracy	Train Time	Accuracy	Train Time	Accuracy	Train Time
ViViT-L_{128f}	**64.45**/87.48	1x(14.17 h)	53.14/73.98	1x(3.84 h)	43.53/56.55/**65.40**	1x(5.61 h)
ViViT-L_{128f} + $\mathbf{SFA}_{8f}$	63.85/**87.62**	**0.62x()**	**53.60/74.47**	**0.62x()**	**43.54/56.78**/65.16	**0.63x()**

Table 6: Performance Comparison of ViViT-L with the proposed training strategy for Something-something v2, Something-Else and Epic-Kitchens. Accuracies listed as Top-1/Top-5, for Epic Kitchens Top-1 noun-verb/ Top-1 noun/ Top-1 Verb.

We further examine the performance of ViViT-L incorporating our proposed training strategy in comparison to the original version on three additional datasets: Something-something, Something-Else, and Epic-Kitchens. A consistent trend is observed, with the modified approach outperforming the baseline model, at only a 62% cost of the baseline training time. In summary, our proposed training strategy demonstrates promising potential by yielding comparable or slightly improved performance across all datasets. This is obtained while maintaining a training cost ranging from 56% to 62% of the original model, thus highlighting its effectiveness. Results can be seen in Table 6.

4.9 Beyond the ViViT model

We adapted our approach to tiny versions of Dualformer [26], VideoSwin [29] and Video-FocalNets [45] on the Kinetics 400 dataset, demonstrating up to 2x faster training with minimal accuracy loss. Thus, our method exhibits consistent improvements in training efficiency across multiple video architectures. We believe these comparisons better showcase the generalized benefits of our freeze-and-finetune approach for efficient video model training. These results can be seen in Table 7. While these models do not directly fall under the factorized encoder models, we can follow a similar pipeline to train them under the SFA training regime. For the Dualformer-T and VideoSwin-T, we first train 8-frame versions of these models and then freeze the first 3 blocks of the models (Stage 1 to Stage 3) and only finetune the last block.

Model	Total Train Time	Accuracy
Dualformer-T_{32f}	1x(30.72 h)	**79.53**
Dualformer-T_{32f}-SFA$_{8f}$	**0.57x()**	79.48
Dualformer-B_{32f}	1x(68.57 h)	81.07
Dualformer-B_{32f}-SFA$_{8f}$	**0.59x()**	**81.12**
VideoSwin-T_{32f}	1x(41.45 h)	78.62
VideoSwin-T_{32f}-SFA$_{8f}$	**0.52x()**	**78.68**
VideoSwin-B_{32f}	1x(115.23 h)	80.45
VideoSwin-T_{32f}-SFA$_{8f}$	**0.53x()**	**80.52**
Video-FocalNet-T_{32f}	1x(11.28 h)	79.76
Video-FocalNet-T_{32f}-SFA$_{8f}$	**0.52x()**	**79.95**
Video-FocalNet-B_{32f}	1x(28.84 h)	83.55
Video-FocalNet-B_{32f}-SFA$_{8f}$	**0.51x()**	**84.25**

Table 7: Training efficiency improvements from SFA approach on tiny Dualformer and VideoSwin models. Our method achieves > 2x faster training with minimal accuracy drop in Kinetics-400.

4.10 Comparing with other efficient training strategies

While we have already mentioned that strategies such as ST-Adapter, STTS etc are efficient in parameters and GFLOPs and not in training time. We demonstrate the results of training these in comparison to the proposed training method in terms of training time and inference time. The results demonstrate that our proposed SFA training pipeline confers substantial training time savings compared to these methods. For example, SFA-ViViT-B attains similar accuracy as ST-Adapter at approximately 0.58 times the total training time on Kinetics-400. We would like to point out that only STTS does better than the proposed strategy in terms of throughput, but as shown earlier requires twice the amount of time to train the model. All methods use 32 frames. These results are reported in Table 8.

Model	Total Train Time	Throughput	Accuracy
ViT-B + ST-Adapter [32]	1x(34.18 hours)	90	82.0
ViViT-B [1]	1.03x()	88	82.6
STTS [43]	1.07x()	**129**	80.7
SBP [7]	0.92x()	89	82.2
SFA$_{8f}$-ViViT-B_{32f} (Ours)	**0.58x()**	88	82.8

Table 8: Training efficiency improvements from SFA approach in comparison to other efficient training strategies. We require approximately half of the training time in comparison to other efficient training strategies.

5 Evaluations Discussion

Our evaluations on ViViT was not to obtain SOTA performance using ViViT, but to show the reduced time required to train the model. In the ViViT paper [1], the reported results are different from what we report. This is due to some videos being removed from the Kinetics dataset. We re-run ViViT using the official codebase and report results on the datasets using this.

The point of using 128 frames was not to show that using more frames improves performance, but instead to show that for larger models, using our training strategy we can actually access more frames due to memory being saved. The logic behind the training strategy holds for lower number of frames such as 32 as well. Hence, using multiview inference would still save time. We would still need only $\mathbf{0.6}x$ of the overall training time. Also, as pointed out in Section 4.7, larger models such as ViViT-g can only access 8 frames using standard ViViT-FE training. However, using our approach, we can access 48 frames. A noteable benefit of accessing more frames is in long-form video understanding tasks and while such experiments have not been conducted, this training approach promises improved results in the task.

6 Conclusion

We have investigated the challenges posed by the substantial training time and memory consumption of video transformers, particularly focusing on the factorised encoder variant of the ViViT model as our baseline. To address these challenges, we proposed two effective strategies: utilizing a compact adapter model for fine-tuning image representations instead of end-to-end training of the spatial transformer, and initializing the temporal transformer using the baseline model trained with 8 frames. Our proposed training strategy has demonstrated the potential to significantly reduce training costs and memory consumption while maintaining, or even slightly improving, performance compared to the baseline model. Furthermore, we observed that with proper initialization, our baseline model can achieve near-peak performance within the first 10% of training epochs. While we focus on the factorized encoder variant of ViViT, we also show the generalization power of the training regime by comparing results of DualFormer, VideoSwin and Video-FocalNets. We also show that the proposed method requires approximately half the time required in comparison to other efficient training strategies. The advancements made in this work could help research in the video understanding domain by enabling access to more frames and the utilization of larger image models as the spatial transformer, all while maintaining the same memory consumption. Our findings provide valuable insights for researchers and practitioners with limited resources, paving the way for more efficient and scalable alternatives in action recognition. Future work may focus on further optimizing and refining these strategies, and exploring their application to other video transformer architectures.

References

1. Arnab, A., Dehghani, M., Heigold, G., Sun, C., Lučić, M., Schmid, C.: Vivit: A video vision transformer. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 6836–6846 (2021)
2. Bertasius, G., Wang, H., Torresani, L.: Is space-time attention all you need for video understanding? In: ICML. vol. 2, p. 4 (2021)
3. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16. pp. 213–229. Springer (2020)
4. Carreira, J., Noland, E., Banki-Horvath, A., Hillier, C., Zisserman, A.: A short note about kinetics-600. arXiv preprint arXiv:1808.01340 (2018)
5. Carreira, J., Zisserman, A.: Quo vadis, action recognition? a new model and the kinetics dataset. In: proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6299–6308 (2017)
6. Chen, J., Ho, C.M.: Mm-vit: Multi-modal video transformer for compressed video action recognition. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 1910–1921 (2022)
7. Cheng, F., Xu, M., Xiong, Y., Chen, H., Li, X., Li, W., Xia, W.: Stochastic backpropagation: A memory efficient strategy for training video models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8301–8310 (2022)
8. Damen, D., Doughty, H., Farinella, G.M., Fidler, S., Furnari, A., Kazakos, E., Moltisanti, D., Munro, J., Perrett, T., Price, W., et al.: Scaling egocentric vision: The epic-kitchens dataset. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 720–736 (2018)
9. Dehghani, M., Arnab, A., Beyer, L., Vaswani, A., Tay, Y.: The efficiency misnomer. arXiv preprint arXiv:2110.12894 (2021)
10. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)
11. Fan, H., Xiong, B., Mangalam, K., Li, Y., Yan, Z., Malik, J., Feichtenhofer, C.: Multiscale vision transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6824–6835 (2021)
12. Foteinopoulou, N.M., Patras, I.: Emoclip: A vision-language method for zero-shot video facial expression recognition. arXiv preprint arXiv:2310.16640 (2023)
13. Gowda, S.N.: Human activity recognition using combinatorial deep belief networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops. pp. 1–6 (2017)
14. Gowda, S.N., Gao, B., Clifton, D.: Fe-adapter: adapting image-based emotion classifiers to videos (2024)
15. Gowda, S.N., Hao, X., Li, G., Sevilla-Lara, L., Gowda, S.N.: Watt for what: Rethinking deep learning's energy-performance relationship. arXiv preprint arXiv:2310.06522 (2023)
16. Gowda, S.N., Rohrbach, M., Keller, F., Sevilla-Lara, L.: Learn2augment: Learning to composite videos for data augmentation in action recognition. In: Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXI. pp. 242–259. Springer (2022)

17. Gowda, S.N., Rohrbach, M., Sevilla-Lara, L.: Smart frame selection for action recognition. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 35, pp. 1451–1459 (2021)
18. Goyal, R., Ebrahimi Kahou, S., Michalski, V., Materzynska, J., Westphal, S., Kim, H., Haenel, V., Fruend, I., Yianilos, P., Mueller-Freitag, M., et al.: The" something something" video database for learning and evaluating visual common sense. In: Proceedings of the IEEE international conference on computer vision. pp. 5842–5850 (2017)
19. Gritsenko, A., Xiong, X., Djolonga, J., Dehghani, M., Sun, C., Lučić, M., Schmid, C., Arnab, A.: End-to-end spatio-temporal action localisation with video transformers. arXiv preprint arXiv:2304.12160 (2023)
20. Hinton, G., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531 (2015)
21. Ji, S., Xu, W., Yang, M., Yu, K.: 3d convolutional neural networks for human action recognition. IEEE transactions on pattern analysis and machine intelligence **35**(1), 221–231 (2012)
22. Kim, K., Gowda, S.N., Mac Aodha, O., Sevilla-Lara, L.: Capturing temporal information in a single frame: Channel sampling strategies for action recognition. arXiv preprint arXiv:2201.10394 (2022)
23. Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., Serre, T.: Hmdb: a large video database for human motion recognition. In: 2011 International conference on computer vision. pp. 2556–2563. IEEE (2011)
24. Laptev, I.: On space-time interest points. International journal of computer vision **64**, 107–123 (2005)
25. Li, K., Wang, Y., Gao, P., Song, G., Liu, Y., Li, H., Qiao, Y.: Uniformer: Unified transformer for efficient spatiotemporal representation learning. arXiv preprint arXiv:2201.04676 (2022)
26. Liang, Y., Zhou, P., Zimmermann, R., Yan, S.: Dualformer: Local-global stratified transformer for efficient video recognition. In: European Conference on Computer Vision. pp. 577–595. Springer (2022)
27. Lin, J., Gan, C., Han, S.: Tsm: Temporal shift module for efficient video understanding. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 7083–7093 (2019)
28. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 10012–10022 (2021)
29. Liu, Z., Ning, J., Cao, Y., Wei, Y., Zhang, Z., Lin, S., Hu, H.: Video swin transformer. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 3202–3211 (2022)
30. Materzynska, J., Xiao, T., Herzig, R., Xu, H., Wang, X., Darrell, T.: Something-else: Compositional action recognition with spatial-temporal interaction networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1049–1059 (2020)
31. Monfort, M., Andonian, A., Zhou, B., Ramakrishnan, K., Bargal, S.A., Yan, T., Brown, L., Fan, Q., Gutfreund, D., Vondrick, C., 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)
32. Pan, J., Lin, Z., Zhu, X., Shao, J., Li, H.: St-adapter: Parameter-efficient image-to-video transfer learning. Advances in Neural Information Processing Systems **35**, 26462–26477 (2022)

33. Piergiovanni, A., Kuo, W., Angelova, A.: Rethinking video vits: Sparse video tubes for joint image and video learning. arXiv preprint arXiv:2212.03229 (2022)
34. Ridnik, T., Ben-Baruch, E., Noy, A., Zelnik-Manor, L.: Imagenet-21k pretraining for the masses. arXiv preprint arXiv:2104.10972 (2021)
35. Ryoo, M., Piergiovanni, A., Arnab, A., Dehghani, M., Angelova, A.: Tokenlearner: Adaptive space-time tokenization for videos. Advances in Neural Information Processing Systems **34**, 12786–12797 (2021)
36. Simonyan, K., Zisserman, A.: Two-stream convolutional networks for action recognition in videos. Advances in neural information processing systems **27** (2014)
37. Soomro, K., Zamir, A.R., Shah, M.: Ucf101: A dataset of 101 human actions classes from videos in the wild. arXiv preprint arXiv:1212.0402 (2012)
38. Tong, Z., Song, Y., Wang, J., Wang, L.: Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. arXiv preprint arXiv:2203.12602 (2022)
39. Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M.: Learning spatiotemporal features with 3d convolutional networks. In: Proceedings of the IEEE international conference on computer vision. pp. 4489–4497 (2015)
40. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)
41. Wang, H., Kläser, A., Schmid, C., Liu, C.L.: Dense trajectories and motion boundary descriptors for action recognition. International journal of computer vision **103**, 60–79 (2013)
42. Wang, J., Yang, Z., Hu, X., Li, L., Lin, K., Gan, Z., Liu, Z., Liu, C., Wang, L.: Git: A generative image-to-text transformer for vision and language. Transactions of Machine Learning Research
43. Wang, J., Yang, X., Li, H., Liu, L., Wu, Z., Jiang, Y.G.: Efficient video transformers with spatial-temporal token selection. In: Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXV. pp. 69–86. Springer (2022)
44. Wang, L., Xiong, Y., Wang, Z., Qiao, Y., Lin, D., Tang, X., Van Gool, L.: Temporal segment networks: Towards good practices for deep action recognition. In: European conference on computer vision. pp. 20–36. Springer (2016)
45. Wasim, S.T., Khattak, M.U., Naseer, M., Khan, S., Shah, M., Khan, F.S.: Video-focalnets: Spatio-temporal focal modulation for video action recognition. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 13778–13789 (2023)
46. Xiong, X., Arnab, A., Nagrani, A., Schmid, C.: M&m mix: A multimodal multiview transformer ensemble. arXiv preprint arXiv:2206.09852 (2022)
47. Yan, S., Xiong, X., Arnab, A., Lu, Z., Zhang, M., Sun, C., Schmid, C.: Multiview transformers for video recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3333–3343 (2022)
48. Yang, A., Nagrani, A., Seo, P.H., Miech, A., Pont-Tuset, J., Laptev, I., Sivic, J., Schmid, C.: Vid2seq: Large-scale pretraining of a visual language model for dense video captioning. In: CVPR 2023-IEEE/CVF Conference on Computer Vision and Pattern Recognition (2023)
49. Zhang, H., Hao, Y., Ngo, C.W.: Token shift transformer for video classification. In: Proceedings of the 29th ACM International Conference on Multimedia. pp. 917–925 (2021)