Action Recognition using Vision Transformers

Elizaveta Andrushkevich, Kavitha Appulingam, Yiwen Chan, and Artem Karamyshev
University of Surrey

***Abstract*—Abstract. A summary of the paper that includes the most interesting results.**

I. Introduction

Your introduction goes here! Some examples of commonly used commands and features are listed below, to help you get started.akdfjlkadgnakldfnalkdjnflandljnlakfjndljanlldkjfnlkdan

II. Literature Review

III. Methodology

A. Data

The HMDB dataset was used. Having noticed several duplicate frames, the first stage was to remove this and ensure the data was clean.

B. Sampling Methods

C. Model

IV. Experiments

Exp.	Length	Sampling	Acc.	Top-5	F1
1.a-fixed	8	Fixed (8)	0.852	0.968	0.719
1.a-eq	8	Equidistant	0.820	0.949	**0.818**
1.b-fixed	16	Fixed (8)	0.840	0.955	0.730
1.b-eq	16	Equidistant	0.820	0.964	**0.818**
1.c-eq	32	Equidistant	0.835	0.976	0.803
full-aug	8	Fixed (8)	0.846	0.946	0.736
b/c-aug	8	Fixed (8)	0.840	0.974	0.725
n/q-aug	8	Fixed (8)	0.821	0.966	0.709
filtered	8	Fixed (8)	0.852	0.979	0.770
r3d	10	Fixed (8)	0.697	0.921	0.574
hpo_best	8	Fixed (8)	**0.903**	**0.984**	**0.816**

TABLE I
Performance metrics for different experimental configurations.

A. Experimental Setup

We evaluate different approaches to the given problem through three key aspects: clip sampling strategies, data augmentation techniques, and hyperparameter optimisation. All experiments are based on the TimeSformer [**?**] architecture.

B. Clip Sampling Strategies

We implement two sampling strategies:

- **Fixed sampling**: With step k, we sample every k-th frame from the video.
- **Equidistant sampling**: Distributes frames evenly across the video sequence, providing uniform temporal coverage.

We experiment with different combinations of clip length, sampling strategy, and corresponding parameters. We can observe that the fixed sampling strategy is more effective for the given problem, with the best accuracy achieved when the clip length is 8. This can be explained by the model's architecture, which employs cross-frame attention and is sensitive to the temporal resolution of the input video. At the same time, the equidistant sampling tends to result in better F1-score, indicating more balanced performance across different action classes, that can vary significantly in duration and complexity.

C. Data Augmentation Techniques

We investigate the impact of different augmentation strategies on model performance, building upon the best-performing clip sampling configuration (8 frames with fixed sampling). Our augmentation pipeline consists of two main components:

- **Brightness and Contrast**: Random adjustments to brightness, contrast, hue, saturation, and RGB channel shifts. These transformations help the model become invariant to lighting conditions and colour variations.
- **Noise and Quality**: Gaussian noise and image compression artefacts. These augmentations improve robustness to real-world video quality variations and compression.

As shown in Table I, the full augmentation pipeline (full-aug) achieves comparable accuracy to the baseline while slightly improving the F1-score. This suggests that augmentations could help maintain performance on common cases while improving recognition of more challenging actions. The brightness/contrast augmentations alone show similar performance, indicating that color-space transformations can be effective for this task. The noise/quality augmentations result in slightly lower metrics, suggesting that these transformations might be too aggressive for the current dataset and model architecture. This could be due to the HMDB dataset already containing videos with varying quality levels.

D. Hyperparameter Optimisation

We conduct hyperparameter optimisation to fine-tune the model's performance. The optimisation focuses on three key parameters:

- **Learning Rate**
- **Weight Decay**
- **Batch Size**

The optimisation process is performed on the best-performing configuration from previous experiments. As shown in Table I, the hyperparameter tuning results in improved model performance, with the best configuration achieving higher accuracy and F1-score compared to the baseline.

The optimisation process reveals that the model is particularly sensitive to learning rate variations, with smaller learning rates generally leading to more stable training. The best performing configuration uses a smaller batch size, suggesting that more frequent parameter updates might be beneficial for this task.

E. Additional Experiments

We conduct two additional experiments to further understand the model's behaviour:

- **R3D [?] Model**: We evaluate the performance of a 3D ResNet architecture (R3D) on the same task. As shown in Table I, the R3D model achieves significantly lower performance compared to TimeSformer. This demonstrates the need for a more complex architecture for video action recognition.
- **Filtered Dataset**: We investigate the impact of removing similar consecutive frames using optical flow and SSIM-based filtering. While the filtered dataset shows promising results with improved F1-score, proper analysis and balancing of such filtering techniques would require more extensive investigation and is better suited for a separate future work.

V. Conclusion and Future Work

We have successfully addressed the video action recognition task using the TimeSformer architecture, achieving strong performance on the HMDB dataset. Our best model achieves 90.3% accuracy, 98.4% top-5 accuracy and 81.6% F1-score, significantly outperforming the R3D baseline (69.7% accuracy) and the simple tuning of the TimeSformer model. Through systematic experimentation, we have demonstrated how TimeSformer model can be effectively adapted to the HMDB dataset.

Our experiments reveal several key insights. The TimeSformer architecture proves particularly effective for this task due to its ability to capture both spatial and temporal relationships through self-attention. The choice of sampling strategy significantly impacts model performance, with fixed sampling providing better accuracy while equidistant sampling offering more balanced performance across classes. Data augmentation techniques help improve model robustness, though their effectiveness varies depending on the transformation type.

Future work could explore several promising directions. First, more thorough data exploration and preprocessing could be beneficial, particularly in terms of frame filtering and class balancing. The promising results from our initial frame filtering experiment suggest that a more comprehensive analysis of temporal redundancy could lead to further improvements. Second, the varying complexity of different action classes suggests that adaptive methods, which can adjust their processing based on action characteristics, might be more effective than uniform approaches. Finally, exploring architectures that can better handle the temporal dynamics of different action types could lead to more robust performance across all classes.

VI. References

VII. Appendix