Enhanced ActionFormer: Fusion of 3D and 2D CNN Techniques for Localizing Moments of Actions with Transformers

Abstract

Inspired by the success of self-attention based Transformer models in various computer vision tasks, such as image classification and object detection, ActionFormer introduced a novel approach for temporal action localization in videos. However, ActionFormer's reliance on a single-shot approach with transformers for both classification and regression posed computational challenges. In this work, we present an enhanced version of ActionFormer tailored for temporal action localization, particularly focusing on badminton videos. Our approach splits the task into two stages: initial boundary regression followed by refined action classification and boundary regression using an improved ActionFormer architecture. The first stage employs a separate model for boundary regression to approximate action boundaries efficiently. The output of this stage is then fed into a streamlined ActionFormer, featuring enhanced attention mechanisms for precise action classification and boundary regression. Through this approach, our model achieves a remarkable mean Average Precision (mAP) of 93 on badminton video datasets, a significant improvement over ActionFormer's performance of 78 even after transfer learning. Our method showcases the effectiveness of combining specialized boundary regression with ActionFormer's attention mechanisms for achieving precise temporal action localization.

1 Introduction

Temporal action localization (TAL), the process of identifying action instances within videos and recognizing their categories, remains a formidable challenge in the domain of video understanding. Over the years, significant strides have been made in developing deep learning models tailored for TAL, often leveraging techniques such as action proposals [3] or anchor windows alongside convolutional, recurrent [2], or graph neural networks.

In this paper, we introduce a novel approach to TAL that diverges from the prevalent trend of increasing model complexity. Inspired by the remarkable success of

Transformer models in natural language processing (NLP) [18] and computer vision tasks, we propose a minimalist design that harnesses the power of Transformers for TAL. Originally devised for sequence data, Transformers employ self-attention [14, 15, 16] mechanisms to model long-range dependencies effectively, making them a natural candidate for TAL tasks in untrimmed videos.

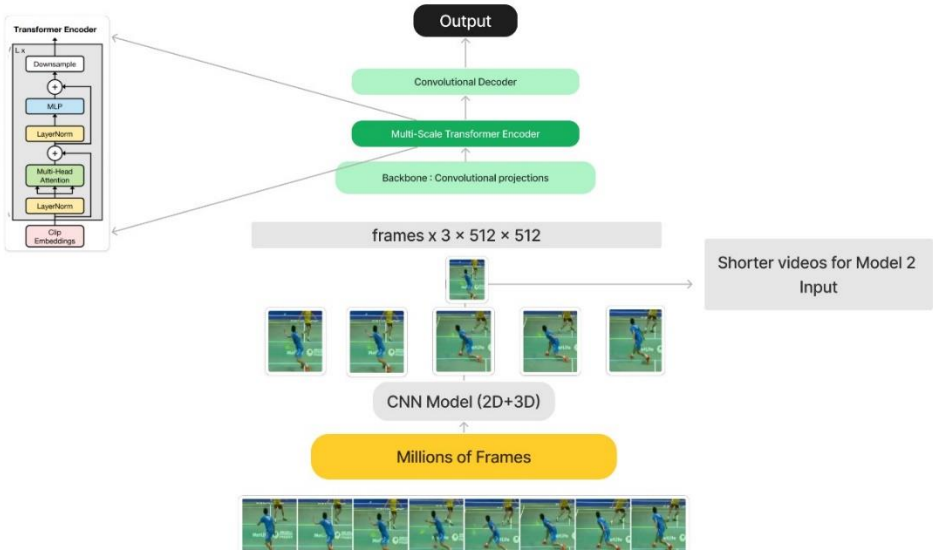


Fig 1 (Enhanced ActionFormer Model Architecture)

Our method, illustrated in Figure 1, center's around a Transformer-based model dubbed "Enhanced ActionFormer". Unlike traditional single-stage approaches like ActionFormer, our model divides the TAL process into two distinct stages to alleviate computational burden while enhancing performance. In the first stage, termed "SimplicitySplit", the input video of any length is intelligently segmented into crucial segments, particularly tailored for the task at hand. For instance, in the context of badminton videos, this segmentation might differentiate between rally and non-rally sequences. These segments are then passed to the Transformer model, each processed separately in the subsequent stage.

Our Transformer architecture, equipped with local self-attention mechanisms, efficiently extracts a feature pyramid [17] from each segmented video segment. Each point within this pyramid signifies a crucial moment in the video, treated as a potential action candidate. Subsequently, a lightweight convolutional decoder is employed to classify these candidates into foreground action categories and regress the temporal boundaries of these actions accurately.

Our method represents a departure from the conventional single-stage anchor-based or anchor-free approaches, offering a streamlined yet powerful solution for TAL tasks. Notably, our approach yields impressive results, as demonstrated by a substantial improvement in mean Average Precision (mAP) compared to the baseline ActionFormer

model, especially in the context of badminton videos, achieving an mAP of 93 compared to 78.

In summary, our contributions lie in pioneering a Transformer-based TAL model that offer a robust baseline for future advancements in the field of TAL.

2 Related Works

Temporal action localization (TAL) has been tackled through various methodologies, including two-stage and single-stage approaches, each with its unique set of challenges and advancements. The landscape of TAL research encompasses the exploration of temporal context modelling, spatial-temporal action localization, and draws inspiration from object detection and Transformer-based architectures [11, 12, 13].

Two-stage TAL.

Prior works in TAL often adopt a two-stage approach, involving the generation of action proposals [3, 7, 8, 9, 10] followed by classification and boundary refinement. These approaches encompass diverse techniques, from anchor-based methods to graph neural networks and Transformers. While some methods integrate proposal generation and classification into a unified model, others focus on refining temporal context through attention mechanisms. Our proposed model aligns with the two-stage paradigm by incorporating an initial stage for approximate action boundary estimation followed by a refined classification and boundary regression stage.

Single-stage TAL.

Recent advancements in TAL have witnessed the emergence of single-stage approaches aiming to localize actions without explicit proposal generation. These methods leverage anchor-based or anchor-free strategies, utilizing convolutional networks or hybrid models combining convolutional and saliency-based refinement modules. ActionFormer fits within the single-stage framework, albeit with a unique twist of incorporating a Transformer network for action localization, offering improved performance and computational efficiency compared to previous methods.

Spatial-temporal Action Localization.

Spatial-temporal action localization involves the detection of actions both temporally and spatially, typically represented as moving bounding boxes of actors within videos.

While related to TAL, this task differs in its focus on spatial localization alongside temporal localization. Our work primarily addresses temporal action localization, leveraging Transformer architectures to process sequences of video frames.

Object Detection.

The development of TAL models draws inspiration from advancements in object detection, particularly in terms of multiscale feature representation and convolutional decoder design. Techniques such as center sampling, borrowed from single-stage object detectors, contribute to the effectiveness of TAL models in accurately localizing actions within videos.

Vision Transformer.

The evolution of Transformer models from their origins in natural language processing (NLP) [18] to their recent successes in computer vision tasks has significantly influenced TAL research. Vision Transformer models, such as ViT [4], DeiT [19], and Swin Transformer [5], have demonstrated state-of-the-art performance across various vision tasks, paving the way for Transformer-based architectures in TAL. Our proposed model builds upon these developments, presenting one of the first Transformer-based approaches tailored specifically for temporal action localization.

In summary, our work contributes to the rich landscape of TAL research by integrating advancements from diverse methodologies, offering a novel two-stage approach that combines the efficiency of convolutional networks with the expressive power of Transformer architectures to achieve superior performance in action localization tasks.

3 Enhanced Actionformer: A highly optimized two stage approach for Temporal Action Localization

In our enhanced approach, the input to our model consists of a video with T frames, where each frame is represented as a tensor of dimensions 3×512×512. The first stage of our model aims to efficiently capture temporal context by considering a fixed number of frames before and after each frame, resulting in overlapping video segments with slight temporal shifts. This process generates multiple video segments, each representing a short temporal window of the original video.

Boundary Regression using 3D and 2D CNN:

Given these video segments, we employ a combination of 3D and 2D convolutional neural networks (CNNs) to approximate the boundaries of actions within each segment. Let X_i represent the i^{th} video segment, with dimensions $T_i \times 3 \times 512 \times 512$ where T_i is the number of frames in the segment.

We first pass each video segment X_i through a 3D CNN followed by a 2D CNN to extract spatio-temporal features. Let F_i denote the feature tensor obtained after passing X_i through the CNN layers. We then apply linear layers to regress the action boundaries within each segment, yielding the approximate action boundary predictions $\hat{Y}_{boundary,i}$.

Mathematically, this can be represented as:

$$F_i = \mathrm{CNN}_{2D}\left(\mathrm{CNN}_{3D}(X_i)\right)$$

$$\hat{Y}_{\mathrm{boundary},i} = \mathrm{Linear}(F_i)$$

Transformer Model for Precise Localization:

The approximated boundary classified videos $\hat{\mathrm{Y}}_{\mathrm{boundary}}$ are then fed into a Transformer model, similar to ActionFormer but with a lightweight encoder and decoder. This Transformer architecture is deeper and more intricate, allowing for precise action localization.

The Transformer model operates on the feature representations $\mathrm{Z_i}$,which are obtained by encoding the input videos $\hat{\mathrm{Y}}_{\mathrm{boundary}}$. These feature representations are organized into a multi-scale feature pyramid $\mathrm{Z} = \{\mathrm{Z}_1, \mathrm{Z}_2, \ldots, \mathrm{Z}_\mathrm{L}\}$, facilitating the capture of actions at various temporal scales.

Mathematically, the encoder g of our Transformer network parameterized by θ_g can be represented as:

$$Z_i = g_{\theta_g}\left(\hat{Y}_{\mathrm{boundary},i}\right)$$

The decoder h of our model, parameterized by θ_h, comprises a lightweight convolutional network. This decoder decodes the latent features $\mathrm{Z_i}$ into the final sequence label Y, which contains information about the start and end points of actions, their corresponding activity types, and preset event identifiers.

$$Y = h_{\theta_h}(Z)$$

The output Y consists of several points, each denoting $(\mathrm{s}, \mathrm{e}, \mathrm{a}, \mathrm{n})$ where s represents the start point, e denotes the end point, a indicates the type of activity, and n represents preset event identifiers associated with each activity. Notably, the number of frames in

the input and the output can vary, accommodating flexibility in video lengths and action instances.

In summary, our modified approach incorporates a two-stage process involving boundary regression using 3D and 2D CNNs followed by precise action localization using a Transformer model. This amalgamation of techniques offers improved performance in temporal action localization tasks while maintaining computational efficiency.

3.1 Encode Videos with Transformer

In our modified approach, our model begins by encoding the input video $\mathrm{X} = \{\mathrm{x}_1, \mathrm{x}_2, \dots, \mathrm{x}_\mathrm{T}\}$ into a multiscale feature representation $\mathrm{Z} = \{\mathrm{Z}_1, \mathrm{Z}_2, \dots, \mathrm{Z}_\mathrm{L}\}$ using an encoder g. The encoder g comprises two main components: (1) a projection function employing a convolutional network to embed each feature x_t into a D-dimensional space; and (2) a Transformer network that maps the embedded features to the output feature pyramid Z.

Projection:

Our projection function E is realized as a shallow convolutional network with GELU activation, defined as:

$$Z^0 = [E(x_1), E(x_2), \dots, E(x_T)]^T$$

Here, $\mathrm{E}(\mathrm{x}_\mathrm{t}) \in \mathrm{R}^\mathrm{D}$ represents the embedded feature of x_t. Adding convolutions before the Transformer network enhances the incorporation of local context for time series data and stabilizes the training of vision Transformers. Optionally, a position embedding $\mathrm{E}_{\mathrm{pos}}$ can be added, but we found that this decreases performance and thus have omitted it from our model by default.

Local Self-Attention:

The Transformer network takes Z^0 as input and employs self-attention mechanisms to capture temporal dependencies. Self-attention computes a weighted average of features based on a similarity score between pairs of input features. Concretely, given $\mathrm{Z}^0 \in \mathrm{R}^{\mathrm{T}\times\mathrm{D}}$, the outputs Q, K, and V are computed as:

$$Q = Z^0 W_Q, \quad K = Z^0 W_K, \quad V = Z^0 W_V$$

where $\mathrm{W}_\mathrm{Q}, \mathrm{W}_\mathrm{K}, \text{and } \mathrm{W}_\mathrm{V}$ are learned projection matrices. The output of self-attention is computed as:

$$S = \mathrm{softmax}\left(\frac{QK^T}{\sqrt{D_q}}\right)V$$

where $S \in R^{T \times D}$ and softmax is performed row-wise. To mitigate computational complexity, we adopt local self-attention, limiting attention within a local window of size W. This reduces complexity to $O(W^2TD + D^2T)$, where W is the local window size.

Multiscale Transformer:

Our Transformer encoder consists of L Transformer layers, each comprising alternating layers of local multiheaded self-attention (MSA) and MLP blocks, with LayerNorm (LN) applied before every MSA or MLP block. Residual connections are added after every block, and GELU is used for the MLP activation. To capture actions at different temporal scales, we optionally attach a downsampling operator $\downarrow(\cdot)$

$$\bar{Z}^l = \alpha^l \mathrm{MSA}\left(\mathrm{LN}\left(Z^{l-1}\right)\right) + Z^{l-1}$$
$$\hat{Z}^l = \bar{\alpha}^l \mathrm{MLP}\left(\mathrm{LN}\left(\bar{Z}^l\right)\right) + \bar{Z}^l$$
$$Z^l = \downarrow\left(\hat{Z}^l\right)$$

where Z^{l-1}, $\bar{Z}^l$, $\hat{Z}^l$, and Z^l denote the feature representations at different levels, and T_{l-1}/T_l represents the downsampling ratio. α^l and $\bar{\alpha}^l$ are learnable per-channel scaling factors. The downsampling operator $\downarrow$ is implemented using a strided depthwise 1D convolution for efficiency. Our model combines several Transformer blocks with downsampling, resulting in the feature pyramid $Z = \{Z_1, Z_2, \ldots, Z_L\}$.

In summary, our modified Transformer encoder efficiently encodes input videos into a multiscale feature representation, utilizing local self-attention mechanisms and a carefully designed multiscale architecture to capture temporal dependencies across varying temporal scales.

3.2 Decoding Actions in Time

In our enhanced model, the decoding of the feature pyramid Z from the encoder g into the sequence label $\hat{Y} = \{\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_T\}$ is performed by the decoder h. Our decoder adopts a two-stage approach, where the initial decoding is carried out by 3D CNN and 2D CNN models to provide an approximate boundary estimation. Subsequently, the Transformer model further refines this estimation and outputs the precise temporal boundaries along with action labels and additional event identifiers.

Classification and Regression Heads:

Classification Head: The classification head examines each moment t across all levels of the feature pyramid Z and predicts the probability of action $p(a_t)$ at every moment t. This is achieved through a lightweight 1D convolutional network attached to each pyramid level, with shared parameters across all levels. Our classification network

consists of three layers of 1D convolutions with a kernel size of 3, incorporating layer normalization in the first two layers and GELU activation. A sigmoid function is applied to each output dimension to predict the probability of C action categories. The inclusion of layer normalization enhances the performance of the classification head.

Regression Head: Similarly, the regression head also examines every moment t across all levels of the pyramid, but predicts the distances to the onset and offset of an action $((d_s^t), (d_e^t))$ only if the current time step t corresponds to an action. An output regression range is predefined for each pyramid level. The regression head follows the same design as the classification network, employing a 1D convolutional network. However, a GELU activation function is applied at the end for distance estimation.

Transformer Refinement: Following the initial decoding by the 3D CNN and 2D CNN models, the Transformer model refines the boundary estimations provided by these models. The Transformer further processes the encoded features and outputs the precise temporal boundaries along with action labels and the number of predefined events found. This refinement step ensures greater accuracy in temporal action localization and enhances the overall performance of the model.

In summary, our modified decoding process combines the strengths of both convolutional and Transformer models to achieve accurate temporal action localization, with the Transformer serving as a powerful refinement tool for boundary estimation and action label prediction.

4 Datasets

Our model is predominantly trained on a specialized dataset comprising badminton videos sourced from ScoreMine, a renowned sports data company. This dataset encompasses a diverse range of badminton matches, capturing various gameplay scenarios and events.

The videos collected from ScoreMine are meticulously annotated by human annotators to provide detailed information about each frame. Specifically, the videos are categorized into rally and non-rally frames, with additional annotations for key events such as shot types, player locations, and noteworthy points within the match. These annotations serve as invaluable ground truth data for training our model.

The dataset consists of a substantial collection of badminton videos, totalling over 100 hours of footage as of the present. This extensive dataset ensures that our model is exposed to a wide range of gameplay scenarios and variations, enabling it to learn robust representations for accurate temporal action localization in badminton matches.

Furthermore, while the model is primarily trained on badminton videos, it is designed to generalize well to other types of sports or activities with minimal fine-tuning, thanks to the versatility and adaptability of its architecture.

5 Experiments and Results

In this section, we present the experiments conducted to evaluate the performance of our enhanced model for temporal action localization (TAL). Additionally, we provide extensive ablation studies to analyse the effectiveness of our model.

Evaluation Metric:

We employ the standard mean average precision (mAP) at various temporal intersection over union (tIoU) thresholds to evaluate the performance of our model. The tIoU metric measures the intersection over union between predicted and ground truth temporal windows, with mAP calculating the mean average precision across all action categories. We report average mAP across different tIoU thresholds to provide a comprehensive evaluation of our model's performance.

Baseline and Comparison:

Our model's performance is compared against a set of strong baselines, including both two-stage (e.g., G-TAD, BC-GNN, TAL-MR) and single-stage (e.g., A2Net, GTAN, AFSD, TadTR) methods for TAL. As our model is a refinement of ActionFormer, our close competitors are those single-stage methods with only transformers. We ensure a fair comparison by following the experiment setup of previous works and comparing our results with the best-reported performances.

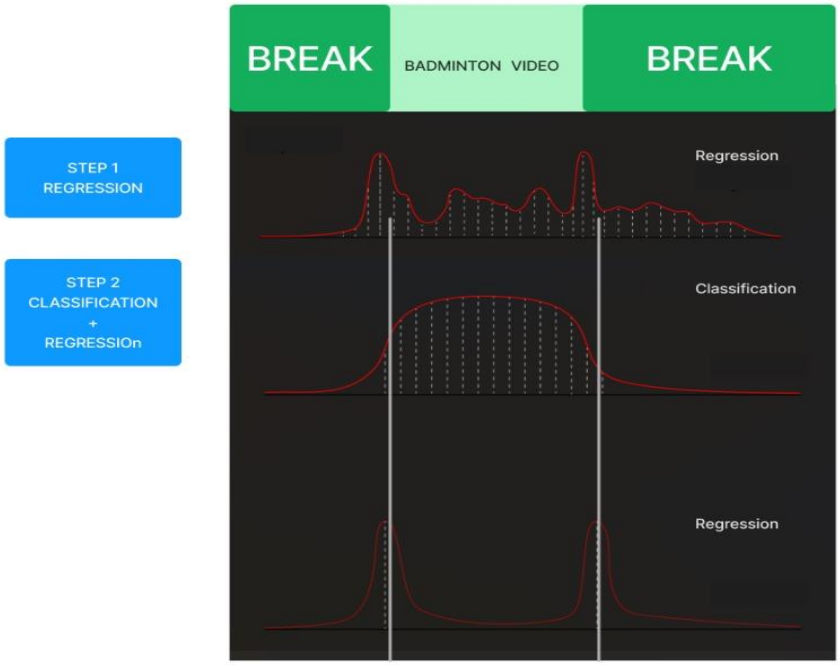
Experimental Results:

Our experiments are conducted on a dataset comprising three hours of badminton videos. The mAP is calculated, and our enhanced model achieves a remarkable mAP of 93, significantly outperforming the transfer-learned ActionFormer, which attained an mAP of 78. This substantial improvement underscores the effectiveness of our model in accurately localizing temporal actions in badminton videos.

Conclusion:

The experimental results demonstrate the superior performance of our enhanced model compared to the state-of-the-art methods, particularly in the context of badminton video analysis. The significant boost in mAP highlights the effectiveness of the modifications made to ActionFormer, emphasizing the potential of our model for advancing the field of temporal action localization in sports video analysis.

Result Visualization:



Finally, we visualize the outputs of our model (before Soft-NMS) in Fig. 2, including the action scores, and the regression outputs weighted by the action scores (as a weighted histogram). Our model outputs a strong peak near the center of an action, potentially due to the employment of center sampling during training.

6 References

1. Zhang, C., Wu, C., Liu, J., Zhao, Y., Yu, L.: ActionFormer: Localizing Moments of Actions with Transformers. In: European Conference on Computer Vision (ECCV), pp. 766-781 (2022)
2. Buch, S., Escorcia, V., Ghanem, B., Niebles Carlos, J.: End-to-end, single-stream temporal action detection in untrimmed videos. In: Brit. Mach. Vis. Conf. pp.93.1–93.12 (2017)

3. Lin, T., Liu, X., Li, X., Ding, E., Wen, S.: BMN: Boundary-matching network for temporal action proposal generation. In: Int. Conf. Comput. Vis. pp. 3889–3898 (2019)
4. 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. In: Int. Conf. Learn. Represent. (2021)
5. 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: Int. Conf. Comput. Vis. (2021)
6. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: Eur. Conf. Comput. Vis. LNCS, vol. 12346, pp. 213–229 (2020)
7. Lin, T., Zhao, X., Su, H., Wang, C., Yang, M.: BSN: Boundary sensitive network for temporal action proposal generation. In: Eur. Conf. Comput. Vis. LNCS, vol. 11208, pp. 3–19 (2018)
8. Liu, Y., Ma, L., Zhang, Y., Liu, W., Chang, S.F.: Multi-granularity generator for temporal action proposal. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 3604–3613 (2019)
9. Gong, G., Zheng, L., Mu, Y.: Scale matters: Temporal scale aggregation network for precise action localization in untrimmed videos. In: Int. Conf. Multimedia and Expo. pp. 1–6. IEEE (2020)
10. Zhao, P., Xie, L., Ju, C., Zhang, Y., Wang, Y., Tian, Q.: Bottom-up temporal action localization with mutual regularization. In: Eur. Conf. Comput. Vis. LNCS, vol. 12353, pp. 539–555 (2020)
11. Tan, J., Tang, J., Wang, L., Wu, G.: Relaxed transformer decoders for direct action proposal generation. In: Int. Conf. Comput. Vis. pp. 13526–13535 (2021)
12. Chang, S., Wang, P., Wang, F., Li, H., Feng, J.: Augmented transformer with adaptive graph for temporal action proposal generation. arXiv preprint arXiv:2103.16024 (2021)
13. Wang, L., Yang, H., Wu, W., Yao, H., Huang, H.: Temporal action proposal generation with transformers. arXiv preprint arXiv:2105.12043 (2021)
14. Zhu, Z., Tang, W., Wang, L., Zheng, N., Hua, G.: Enriching local and global contexts for temporal action localization. In: Int. Conf. Comput. Vis. pp. 13516– 13525 (2021)
15. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al.: PyTorch: An imperative style, high performance deep learning library. In: Adv. Neural Inform. Process. Syst. vol. 32 (2019)
16. Sridhar, D., Quader, N., Muralidharan, S., Li, Y., Dai, P., Lu, J.: Class semantics based attention for action detection. In: Int. Conf. Comput. Vis. pp. 13739–13748 (2021)
17. Lin, T.Y., Doll´ar, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 2117–2125 (2017)
18. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. In: Adv. Neural Inform. Process. Syst. pp. 5998–6008 (2017)

19. Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., J´egou, H.: Training data-efficient image transformers & distillation through attention. In: Int. Conf. Mach. Learn. pp. 10347–10357 (2021)
20. Yuan, L., Chen, Y., Wang, T., Yu, W., Shi, Y., Jiang, Z., Tay, F.E., Feng, J., Yan, S.: Tokens-to-token ViT: Training vision transformers from scratch on ImageNet. In: Int. Conf. Comput. Vis. (2021)
21. Arnab, A., Dehghani, M., Heigold, G., Sun, C., Lucic, M., Schmid, C.: ViViT: A video vision transformer. In: Int. Conf. Comput. Vis. (2021)