China Instrument and Control Society
Instrumentation
OPEN ACCESS
INSTRUMENTATION, Vol. 13, No. 1, March 2026 https://doi.org/10.15878/j.instr.202600298

Article

TransCLGA: Combining Local and Global Attention in Transformer for Temporal Action Detection

Bin Zhang[1], Yinfeng Fang[1], Xuguang Zhang[2,*], Yun Zhang[2]

[1] Department of Communication Engineering, Hangzhou Dianzi University, Hangzhou 310018, China

[2] College of Media Engineering, Communication University of Zhejiang, Hangzhou 310018, China

[*] Corresponding author email: 20250002@cuz.edu.cn

Abstract: Temporal action detection is a crucial task in computer vision, aiming to identify and locate specific actions or events in a video. How to fully extract feature information and perform efficient and accurate feature fusion has always been an important topic in temporal action detection. To solve this problem, this paper proposes a Combining Local and Global Attention in Transformer (TransCLGA) model that aims to extract rich feature information at each temporal scale and perform regulated multi-scale fusion. In the backbone network, we adopt a novel stepwise differentiated attention strategy that enhances the interaction between local and global attention, fully leveraging the advantages of the Transformer for processing global information while compensating for its deficiencies in local feature extraction. A Channel Convolutional Block is introduced to reduce interference caused by global-local information extraction and enhance feature representation. At the neck of the model, we designed a gated feature fusion pyramid that effectively integrates information across different temporal scales by selectively retaining key features, ultimately enabling accurate motion detection. Our model was evaluated on two datasets (THUMOS14 and ActivityNet1.3). The results indicate that the proposed method performs well.

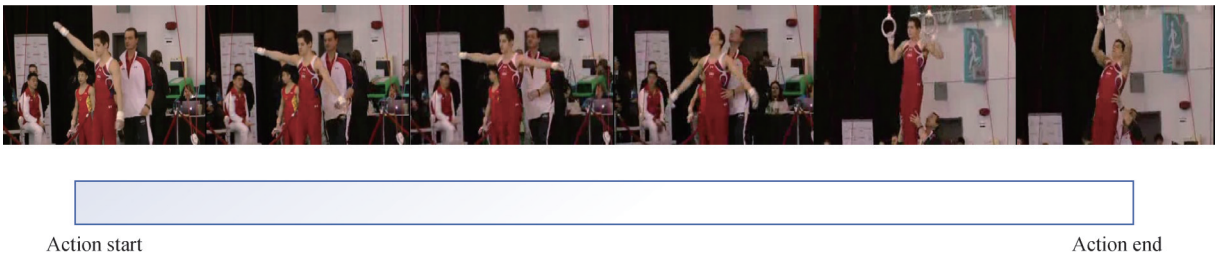
Keywords: temporal action detection; vision transformer; gated mechanism ; self-attention ; action recognition

Citation: Bin Zhang, Yinfeng Fang, Xuguang Zhang, Yun Zhang."TransCLGA: Combining Local and Global Attention in Transformer for Temporal Action Detection." Instrumentation 13, no.1 (March 2026). https://doi.org/10.15878/j.instr.202600298

1 Introduction

Temporal action detection is an important area of research in computer vision. This field encompasses both action recognition and action localization. Its primary goal is to identify and locate specific actions or events within videos. Researchers aim to determine the type of action and its temporal boundaries, including start and end times. For example, Figure 1 presents tasks related to temporal action detection. In Fig. 1 (a), the focus is on identifying the action boundaries in a video clip. In Fig. 1 (b), the task is to classify the action categories depicted in the video clip. This task has significant implications in some applications, such as video surveillance[1], sports analysis[2], and smart homes[3].

In recent years, deep learning has demonstrated exceptional results in the domain of temporal action detection. The mainstream deep learning models for this task can generally be divided into two categories: Convolutional Neural Network(CNN) -based models[46] and Transformer-based models[79]. Due to their remarkable image recognition and feature extraction abilities, Convolutional Neural Networks have found widespread application in video processing. Specifically, in the domain of action detection, CNNs analyse video content frame by frame, extracting key features while incorporating temporal information to precisely identify human actions in videos. For example, Convolutional-De-Convolutional (CDC) [5] introduces a convolutional-de-convolutional network to refine temporal boundaries,



(a) Temporal action localization aims to determine the boundaries (start and end times) of an action instance within a video

(b) Action recognition focuses on classifying the category of the action occurring in a video clip

Fig.1 Examples of temporal action detection tasks:(a)Temporal action localization aims to determine the boundaries (start and end times) of an action instance within a video.(b)Action recognition focuses on classifying the category of the action occurring in a video clip.

achieving precise localization. Temporal Action Localization Network (TAL-Net) [4] extends Faster Region-based Convolutional Neural Network (Faster R-CNN) to the temporal domain by generating action proposals with temporal region proposal networks. However, as the difficulty of datasets used for action detection increases, the limitations of solely relying on CNNs have become more apparent. Although CNNs performs well in local feature extraction, its limited receptive field makes it ineffective in capturing global information. While techniques like dilated convolutions [6] attempt to address this, they still struggle with modeling actions that span long durations or involve complex temporal dependencies. Against this backdrop, Transformer-based methods have emerged. Transformers have gradually been applied to computer vision tasks[810], and their flexibility and powerful modelling capabilities have shown distinct advantages in processing sequential data with complex and high-dimensional characteristics, such as video data. Particularly for action recognition tasks that require capturing long-range dependencies, Transformer models have become indispensable tools. Early Transformer adaptations like Temporal Action Detection with Transformer (TadTR)[8] demonstrated that pure Transformer architectures could outperform CNN-based methods by directly modeling temporal relationships across the entire video. ActionFormer[11] further improved performance by combining Transformers with multi-scale feature pyramids, achieving state-of-the-art results on multiple benchmarks. Recent advances have focused on enhancing Transformers for video understanding. Unified Video-Language Temporal Grounding (UniVTG) [12] unifies diverse video temporal grounding tasks under a single framework, leveraging pseudo-labeling strategies and multimodal pretraining to achieve state-of-the-art performance. Long-term Pre-training (LTP)[13] proposes a pre-training strategy for the DEtection TRansformer (DETR) temporal action detection model, which addresses the data scarcity issue through category-oriented feature synthesis and long-term dependency tasks, significantly enhancing the detection performance. Relaxed Transformer Decoders Network (RTD-Net) [14] proposes a direct action proposal generation method based on the Transformer architecture, by introducing a boundary attention module, relaxation matching mechanism, and a three-branch detection head design. Video Transformer[15] processes video clips as spatiotemporal tokens, while Video Vision Transformer (ViViT) [16] introduces factorized attention for efficient video modeling. These methods excel at capturing global context but often overlook fine-grained local details critical for precise boundary localization.

However, despite their superiority in handling global information, Transformers still face challenges in processing local features and fine-grained details. Thus, effectively combining both local and global information has become a crucial factor to enhancing Transformer performance in action detection tasks. To solve this problem, we tried to improve the attention mechanism. In traditional Transformer models[17] multi-head attention is a key component. Previous studies [1820] have also made improvements to the multi-head attention part in Transformer architectures. Beltagy[18] proposed Longformer, which introduces a sliding window attention pattern to handle long sequences efficiently while maintaining global attention on select positions. Wang[19] developed Linformer, which reduces the quadratic complexity of self-attention to linear through low-rank projection. Zaheer[20] presented BigBird, combining random attention, windowed local attention, and global attention for better sequence modeling. In our approach, we

introduce a Hybrid Attention Module, combining local multi-head attention and global attention in the Transformer encoder. This design aims to leverage the Transformer's strengths in global information processing while compensating for its limitations in local feature extraction. The local multi-head attention focuses on key regions in the video to extract detailed local features, while the global attention maintains sensitivity to overall information, capturing the global context of the video. By balancing local details with global context, our model demonstrates enhanced performance in handling complex actions.

When capturing global-local temporal features, interference information often arises. To mitigate this, we design a Channel Convolution Block in the backbone part, composed of Channel Recalibration and Depth-wise Convolution (DWConv), to handle interference generated during feature extraction[21] and further enhance feature representation. The Channel Recalibration mechanism emphasizes the depth wise information of features, prioritizing the most significant channels for action recognition while mitigating the influence of irrelevant or noisy channels. DWConv enhances local spatiotemporal patterns by applying lightweight convolution along feature dimensions, preserving fine-grained motion cues critical for distinguishing similar actions. The collaboration between these two mechanisms facilitates the model in capturing action features with greater precision and substantially boosts its robustness against interference in complex scenarios.

We extract global and local features at each temporal scale and employ a feature pyramid[22] at the neck of the model for feature fusion. Feature fusion technology plays an important role in enhancing the model performance, especially in complex tasks such as action detection, how to integrate features of different scales becomes a big difficulty. Unlike previous studies, we innovatively design a gated feature fusion pyramid, which cleverly incorporates a gating mechanism into the traditional pyramid structure. This mechanism selectively retains key information while effectively suppressing redundant features, thereby achieving precise fusion of multi-scale feature information. Finally, we deploy a prediction head on the pyramid structure to classify each action and make fine-grained regression predictions for the temporal boundaries.

The main contributions of this paper are as follows:

(1) We construct a Transformer-based temporal action detection model, TransCLGA, which uses an efficient and concise single-stage approach that does not rely on anchor frames. The design aims to maintain model efficiency while achieving good performance.

(2) The multi-head self-attention mechanism is improved in the backbone part and introduces a channel convolution module, allowing for the extraction of rich feature information while reducing interference.

(3) At the neck of the model, a gated feature pyramid is used to integrate the information of different time scales.

2 Related work

2.1 Temporal Action Detection

Temporal action detection is an important work in video analysis, aimed to accurately identify the categories of actions in a video and accurately determine the time boundaries of those actions, encompassing their start and end times. Two-stage methods[2325] use pre-trained CNN to extract the features of video frames, obtain a set of temporal segments with enclosed actions. Afterwards, these will be further refined in order to create more precise feature representations for correct action classification. However, the key challenge in two-stage methods is that there are necessarily two passes through video input, inherently increasing complexity and time expenditure as the clip length or resolution gets higher. Owing to the above limitations, recent studies on action detection have mainly been done along the axis toward more efficient single-stage methods [2628]. Single-stage[29] methods avoid the cumbersome candidate segment generation phase and instead adopt an end-to-end learning approach that directly extracts features from raw videos and performs action classification and temporal boundary localization in a single pass over the entire video sequence. This integrated processing workflow significantly enhances the inference speed, making the single-stage methods particularly advantageous when handling large-scale video data. Single-stage methods can be further divided into anchor-based and anchor-free methods. Anchor-based methods [27] depend on a predefined set of anchor boxes to help in temporal action detection. This helps the model in locating the action more accurately. However, the introduction of anchor boxes also brings complexity and the requirement for tuning of parameters. In contrast, anchor-free methods[26] extract information from the feature maps to perform action detection in a direct and straightforward way, without the help of predefined anchor boxes.

Inspired by the simplicity and efficiency of anchor-less methods, we design a novel anchor-free single-stage temporal action detection (TAD) method. This method not only inherits the fast inference advantages of single-stage approaches but also simplifies the model architecture by eliminating anchor boxes to reduce computational complexity. Meanwhile, we optimize our feature extraction and motion detection algorithms in tandem to continuously improve the accuracy and robustness of our models while keeping the efficiency.

2.2 Transformer in TAD

The Transformer model[17], originally introduced to address sequence-to-sequence tasks, has its core innovation point in the self-attention mechanism. This mechanism provides the model with the flexibility to focus on information from various positions within a sequence, allowing it to efficiently capture and

comprehend long-distance dependencies. This characteristic has contributed significantly to the Transformer's achievement in natural language processing (NLP), where it has exhibited outstanding performance.

The rapid development in the computer vision field, especially in video analysis tasks, requires extensive processing of temporal data. Researchers have hence introduced the Transformer smartly into the video analysis tasks to handle video frame sequences with its powerful sequence processing capabilities. Taking video frame sequences as input, with the self-attention mechanism of the Transformer, the model can capture complicated relationships between frames, aiming for an in-depth understanding of the video content. In earlier works[15,16], applications of Transformers have shown impressive performance in video analysis and motion recognition tasks. These models decompose the video data smartly into the spatial and temporal dimensions, modelling both temporal and spatial features for deep video content analysis. Ji Lin et al. [30] proposed the Temporal Shift Module, which is combined with Transformer for efficient modelling of temporal features in video recognition tasks. In the current state of research, there is an increased use of Transformers in temporal action detection, Liu[8] modelled temporal relationships by using the self-attention mechanism of Transformers and enhanced the accuracy of recognizing actions. Temporal relationships, even more complicated, can be grasped by combining Graph Neural Networks with Transformers[31,32], which enhance the performance of Transformers in detecting actions. These studies not only demonstrate the great potential of Transformers in video analysis tasks but also lay a solid foundation for their widespread application in this field. In the future, application prospects will be even broader for Transformers and their variants in video analysis, offering more possibilities for intelligent understanding and processing of video content.

2.3 Multi-Scale Feature Fusion

In early research, it was keenly recognized that features at different scales can provide rich and valuable information for visual tasks. This insight laid a solid theoretical foundation for subsequent algorithm designs. For example, the Scale-Invariant Feature Transform (SIFT) algorithm proposed by DG Lowe[33], which extracts features at different scales, exhibits good rotation and scale invariance, leading to significant success in image matching and object recognition. The importance of multi-scale features in the visual domain was further validated[34], emphasizing the key role that multi-scale features play in improving model performance. This discovery pointed the way forward for subsequent research into multi-scale feature fusion. As Convolutional Neural Networks gained prominence, the integration of multi-scale features emerged as a focal area of research within the computer vision domain. The classic Faster R-CNN model[35] in object detection cleverly combines feature maps across different scales so as to effectively blend details and semantic information, significantly enhancing object detection performance. This indeed had design value as the insights proved highly useful for successive research in multiscale feature fusion. In 2015, LIN et al. [22] proposed the Feature Pyramid Networks (FPN), an innovative architecture that develops a feature pyramid to organically fuse characteristic information at various scales, forming an effective multi-scale feature representation. This framework improves not only the model's performance in detecting objects across different scales but also introduces innovative concepts and techniques for future research on multi-scale feature integration. Besides, the self-attention mechanism in Transformers has great potential for multi-scale feature fusion. With the self-attention mechanism, the model can dynamically capture correlations and interdependencies of features at various scales. Therefore, it significantly improves the efficacy of multi-scale feature fusion. This makes Transformers more flexible and efficient in dealing with complex scenes. In multimodal learning, by fusing multi-scale features from different modalities (e.g., video and sound) [36], models can more comprehensively understand the information in complex tasks, thereby significantly improving their ability to understand and process such tasks. This research direction not only provides new ideas and methods for multimodal learning but also opens up broader development opportunities for future intelligent applications.

3 Method

3.1 Method Overview

A given input video can be represented as a set of feature vectors $X=\{x_1, x_2, \ldots, x_t\}$, where t represents the number of feature segments for each video clip. Since each video may be of a different length, the value of t also differs from one video to another. These feature vectors effectively encode the dynamic information about the video across different time periods, which forms the basis for further tasks. The goal of temporal action detection is to predict the structured label sequence $Y=\{y_1, y_2, \ldots, y_n\}$ that aligns with the input video, similar to t, here the value of n also varies with respect to the length of a video. where each label y_t is defined by an action category $p(a_t)$, start time d_t^s, and end time d_t^e. We transform the action detection problem into a sequence labeling task, independently predicting for each time point t whether it is an action boundary and its category.

Our goal is to build a single-stage anchor action detector with better performance based on ActionFormer [11]. The model structure consists of three parts: feature backbone, feature fusion pyramid, and action detection head, as shown in Figure 2.

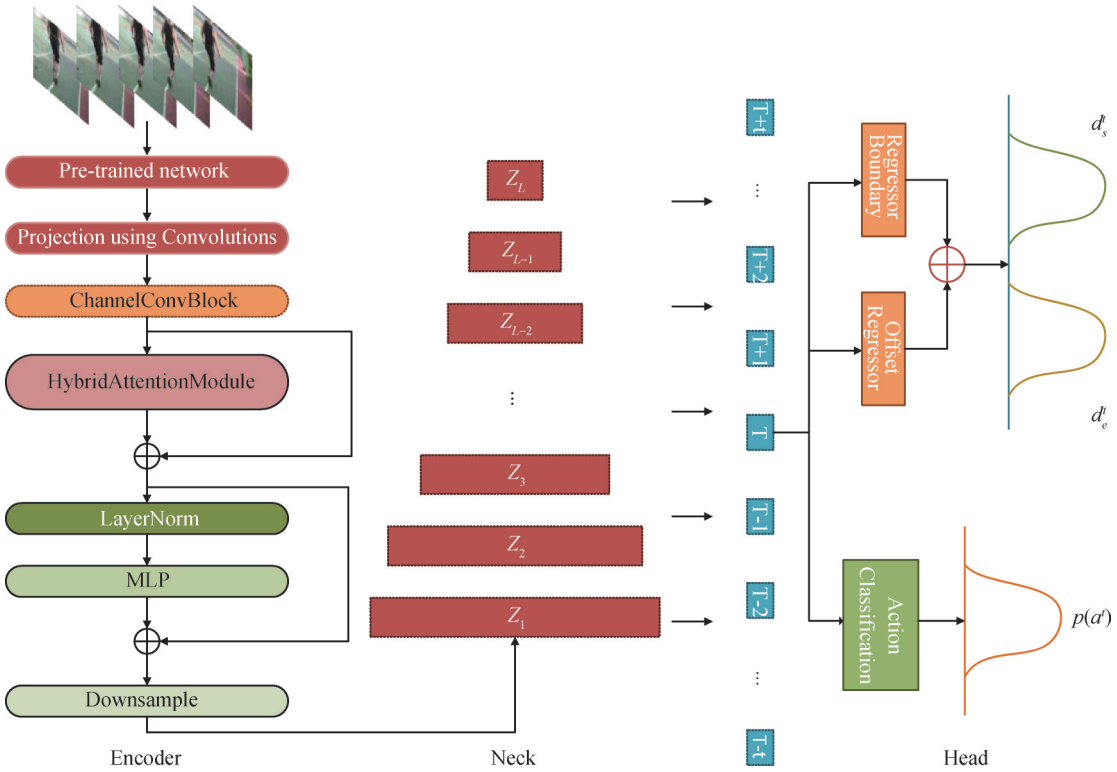


Fig.2 TransCLGA overall network structure.

Specifically, in the data preprocessing phase, we use a pre-trained action detection network (Inflated 3D ConvNet (I3D)[37], Temporal Segment Networks (TSN)[38]) to extract features from a series of video clips. A projection layer, consisting of Layer Norm and a 1D convolutional layer, is designed to embed the extracted features, as represented by the following formula:

$$L_0 = LayerNorm(Conv_2(LayerNorm(Conv_1(x)))) \quad (1)$$

These embedded representations function as input for the Transformer network within the backbone. The main structure of our Transformer encoder consists of two parts: Channel Convolutional Block and Hybrid Attention Module. Firstly, the Hybrid Attention Module, incorporating both global and local attention mechanisms enhances the model's ability to extract and utilize feature information more effectively. The local attention focuses on feature details, while the global attention supplements the capture of long-range dependencies. Additionally, the Channel Convolutional Block is introduced to reduce interference and further enhance feature representation.

In our model, after passing through five transformer layers with 2x down-sampling, multi-scale features are generated. We construct a gated feature fusion pyramid at the neck of the model to capture contextual information and perform feature fusion, dynamically adjusting the weights of different scale features.

Finally, the prediction head, consisting of a classification head and a regression head, outputs candidate actions and localizes the temporal boundaries.

3.2 Channel Convolutional Block

This module delves into the complex and subtle relationships among the channel dimensions within feature segments, aiming to enhance the effectiveness of feature representations by reasonably allocating channel weights. This is a critical issue because it directly affects the model's performance to precisely get action information from the video content[39] while minimizing interference during the feature extraction process. In addition, when extracting global and local features, two types of interference information will be generated: redundant responses in the channel dimension, and some channels may simultaneously activate irrelevant action features. The inconsistency in the spatiotemporal dimension may lead to contradictory representations of the same action area by global and local features. Inspired by Convolutional Block Attention Module (CBAM)[40], we adopt its channel recalibration module, specifically the average pooling$F^c_{avg}$and max poolingF^c_{max} operations, to extract information from the feature segments. These two pooling operations each generate a descriptor, which is then passed into a carefully designed shared network. This shared network consists of a multi-layer perceptron (MLP) that performs deep processing and fusion of the two descriptors. After processing by the shared network, the outputs of the two descriptors are combined through element-wise summation, resulting in the final feature vector that incorporates channel information. The specific process for the channel component can be summarized as:

$$CR(L) = MLP(F^c_{avg}(L)) + MLP(F^c_{maMx}(L)) \quad (2)$$

We introduce several depth wise separable

convolution layers, which enhances local spatiotemporal patterns while preserving the channel relationships. Compared with the standard convolution, this reduces computational complexity and suppresses the generation of interference information. Each depthwise separable convolution has a different kernel, allowing the model to capture multi-scale information, thus enriching the feature information. During the fusion of multi-scale information, we cleverly use 1x1 convolution operations. This not only ensures effective information fusion but also further refines feature details, making the final generated feature vector more comprehensive and expressive. The specific process can be summarized as:

$$SP(L) = Conv_{1\times1}(DwConv_5(L) + DwConv_7(L) + DwConv_{11}(L)) \quad (3)$$

The input$L_0$is processed sequentially through two components, and the overall process can be summarized as:

$$L_c = CR(L_0) \otimes L_0 \quad (4)$$

$$\tilde{F} = SP(L_c) \otimes L_c \quad (5)$$

The specific structure is shown in Figure 3. Compared to CBAM[40] which combines both channel and spatial attention, our Channel Convolutional Block (CCB) focuses solely on channel-wise recalibration followed by multi-scale depth-wise convolution. This design choice is motivated by the observation that in temporal action detection, channel relationships are more critical than spatial relationships for feature representation. The Squeeze-and-Excitation (SE)[21] only performs channel recalibration, while our CCB additionally introduces multi-scale depth-wise separable convolutions to enhance local spatiotemporal patterns while preserving channel relationships. The necessity of our CCB design stems from two key aspects of temporal action detection: (1) The channel dimension in video features often contains redundant or conflicting activations for different action classes, requiring explicit channel-wise recalibration. (2) Local temporal patterns at multiple scales are crucial for detecting actions of varying durations, which motivates our multi-scale depth-wise convolution design. With this design, our module can more accurately capture action information in the video while reducing interference during the process, providing strong support for temporal action detection tasks.

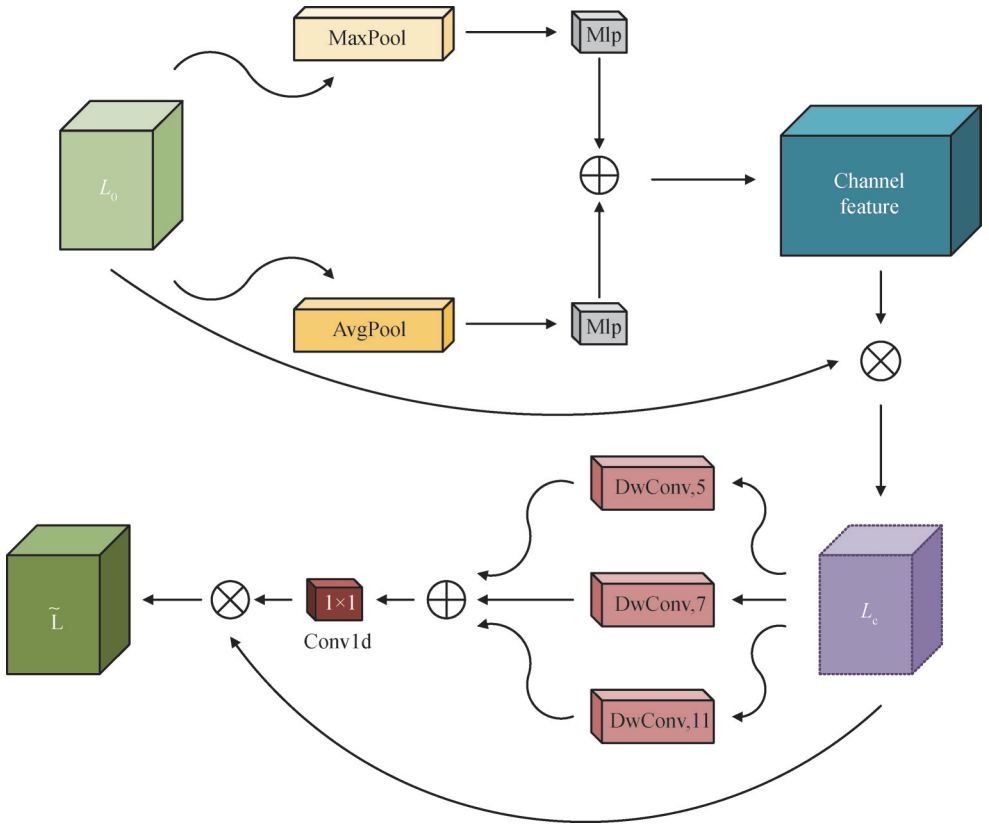


Fig.3 Channel Convolutional Block

3.3 Hybrid Attention Module

In the multi-head self-attention module in Transformer[17], input features are mapped to three different vectors: Query, Key, and Value, which are then multiplied by learned weight matrices:

$$[Q, K, V] = X[W_Q, W_K, W_V] \quad (6)$$

Where X is the input features, and W_Q, W_K, W_V are the learned weight matrices. Next, the attention weight is obtained by the dot product of Q and K, subsequently undergoing a softmax function transformation., as defined by:

$$Attention\left(Q,K,V\right)=softmax(\frac{Qk^T}{\sqrt{d_k}}) \tag{7}$$

Where d_k is the dimensionality of the K vector. The execution of multi-head self-attention entails the simultaneous computation of several attention heads. Specifically, the input features X are divided into $\{x_1,x_2,\ldots,x_i\}$ parts, where i is the number of attention heads. The outputs of all heads undergo concatenation and are then subjected to a linear transformation to yield the ultimate output:

$$MHA\left[Q,K,V\right]=Concat(H_1,H_2,\ldots,H_i)W_0 \tag{8}$$

$$H_i=Attention\left(Q_i,K_i,V_i\right) \tag{9}$$

where W_0 is the weight matrix for the output.

Pecoraro[41] proposed a local multi-head self-attention module, which adds a sliding window mechanism to the classic multi-head self-attention module. By controlling the window size, this design limits the attention scope to a pre-defined local window, successfully reducing computational complexity and making the model more efficient when processing long sequence video data. However, this approach also has some limitations, as it may overly focus on local information and neglect the global context, potentially omitting global features that are crucial for action recognition.

Indeed, feature capturing in sequence videos is a complex and delicate process that requires the model to get rich contextual information to understand the video content more accurately. This is where the global multi-head attention mechanism excels, as it attends to all positions in the video sequence, allowing for an integration of global information, thus improving the model's accuracy in action recognition.

To maximize the feature expression ability of the model, we attempt to combine global attention with local attention. Figure 4 shows the generated attention distribution map, clearly demonstrating the feature patterns of three different attention mechanisms. Local Attention presents a classic diagonal ribbon structure, where each query position only focuses on the key positions within the surrounding fixed window (w=19). This local join pattern can effectively capture short-distance dependencies in the sequence while maintaining a linear computational complexity. Global Attention forms distinct bright bars on the top and left sides of the matrix, indicating that the first few tokens, as global tokens, can establish connections with all positions in the sequence. This design enables the model to transfer long-distance global information. The third one is our Hybrid Attention, which ingeniously combines the advantages of the first two mechanisms. It not only retains the local structure of the diagonal for capturing fine-grained feature interactions at adjacent positions, but also maintains the global connection to ensure that important information can be freely obtained throughout the sequence. On this basis, we introduce a stepwise differentiation progressive framework and set different step sizes for the two types of attention. Global attention uses stride=1 to maintain the original temporal resolution and ensure the integrity of the global context. Local attention uses n_ds_strides for adaptive downsampling to reduce noise and focus on local patterns of different scales. This resolution separation design enables the two attention mechanisms to operate at different temporal granularities. Moreover, the two kinds of attention are serially connected, and the structure is shown in Figure 5. Although this module will increase the computational overhead compared with the simple local focus design, the performance improvement brought by comprehensive feature extraction makes this trade-off reasonable.

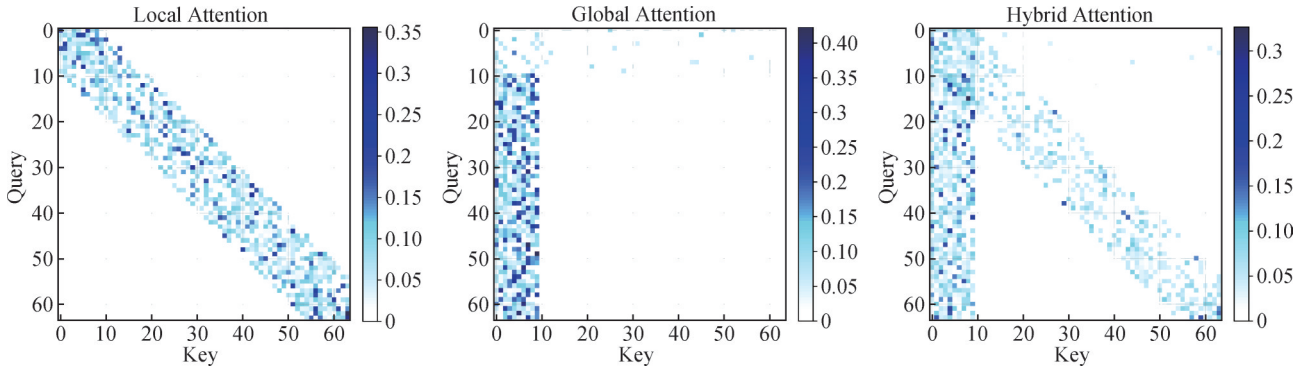


Fig.4 Attention distribution map, for ease of display, we set seq_len=64, w=19, and the depth of the color block on the right represents the attention weight

3.4 Gated Feature Fusion Pyramid

In the task of temporal action detection, each action has different durations, which is very important to construct a pyramid structure that could handle features with different temporal lengths. Although traditional Feature Pyramid Networks[22] perform well in feature fusion, in practical applications, we found low-level features always contain mixed noise information. After careful design, we come up with the Gated Feature Fusion FPN model. The core of this model lies in the introduction of a gating mechanism[42], which makes the

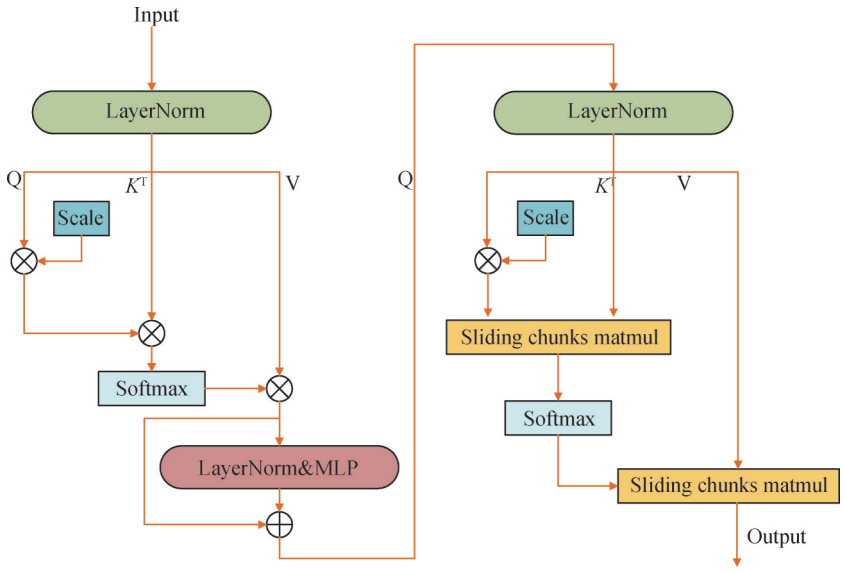


Fig.5 Hybrid Attention Module

flow of information between different levels more flexible and controllable. By dynamically adjusting the fusion ratio between high-level and low-level features, our model can adaptively handle features at different scales while effectively suppressing unnecessary noise in the low-level features. This fine control strategy over multi-scale features significantly enhances the model's performance.

We built a Gated Feature Fusion Pyramid $Z=\{Z_1, Z_2, \ldots, Z_L\}$, and its specific structure is shown in Figure 6. Under each scale, we employ a 1D convolutional network for feature processing to ensure that all features are appropriate and effective. Unlike traditional FPN structures[43,44], in the bottom-up pathway, we compute the current layer's features and, for non-bottom layers, fuse them with the features from the previous layer by element-wise multiplication. This approach puts more emphasis on the features from the bottom level. Optimizes gradient flow through element-wise multiplication (as opposed to FPN's addition), creating residual-like pathways that reduce gradient conflict during multiscale fusion. This proves particularly valuable for long videos where error accumulation would otherwise destabilize training.

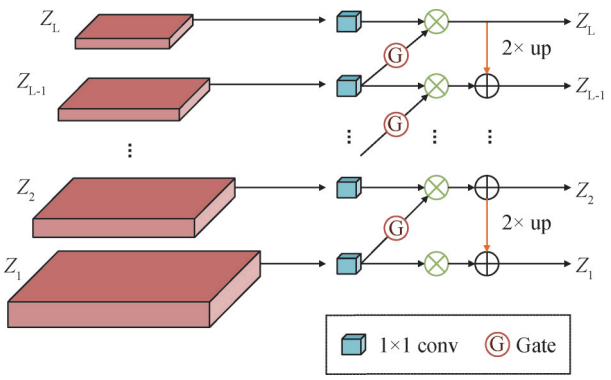


Fig.6 Gated Feature Fusion Pyramid

During the fusion process, we also cleverly incorporate the gating mechanism $G=\{G_1, G_2, \ldots, G_L\}$. The process can be expressed as:

$$Z_m = G_L(Conv(Z_L) \otimes Conv(Z_{L-1})) \tag{10}$$

This mechanism modifies the influence of the feature map transitioning from the bottom layer to the final feature map as required, enabling the model to concentrate more intently on pivotal feature data, thus further improving its discriminative ability. Through this design, our Gated Feature Fusion FPN model demonstrates outstanding performance and stability in temporal action detection tasks.

3.5 Detection Head

After the fine processing by the Gated Feature Fusion Pyramid, we further utilize a detection head integrated with both classification and regression modules to accurately predict the action boundary points at each temporal location and perform action classification.

3.5.1 Classification Head

The classification head works in close collaboration with the feature pyramid, thoroughly scrutinizing each time node t in all layers of the pyramid, and estimating the likelihood of action manifestation at those specific temporal instances. To accomplish this, we employ a three-layer 1D Convolutional Neural Network. Additionally, we integrate the ReLU activation function to enhance the expressive capability of the model. For each output dimension, we apply the sigmoid function to achieve an accurate prediction of the probability of an action class. Notably, the classification head connects to each layer of the feature pyramid and implements parameter sharing, which significantly improves computational efficiency and model consistency.

3.5.2 Regression Head

The regression head's task is to detect each time step t across all L layers of the pyramid. Once it is determined that a time step is inside an action, the regression head is activated immediately to predict the offset distance between the start and end of the action, that is, the range of the output regression. For accurate distance estimation, we add a Rectified Linear Unit (ReLU) activation function at its end.

3.6 Training and Inference

Our output has three key data points at each time step t: the action category $p(a_t)$, the start time d_t^s and end time d_t^e. In this process, we adopt two loss functions: Focal Loss[45] for the classification task to better handle class imbalance issues, and Distance-Intersection over Union (Distance-IoU) Loss[46] for the distance regression task to improve the accuracy of boundary prediction. After the classifier has determined the action category at the current time location and the regression head predicts the offset distances for the action boundaries, we employ the Soft Non-Maximum Suppression (SoftNMS) [47] algorithm on these candidate segments of actions during post-processing, suppressing the redundant information to obtain an accurate and reliable result of the detection.

4 Experiments

4.1 Datasets and Metrics

We conducted a set of experiments on the THUMOS14[48] and ActivityNet1.3[49] datasets. The THUMOS14 dataset contains 20 action categories, including 200 validation set videos (containing 3, 007 action clips) and 213 test set videos (containing 3, 358 action clips). We use the validation sets for model training and the test sets to evaluate the final detection performance. In the existing studies, mean average precision (mAP) is generally calculated at various temporal intersections over union (tIoU). On the THUMOS14 dataset, we set the tIoU threshold as [0.3: 0.7:0.1].

ActivityNet1.3 contains 200 activity categories, 849 hours of YouTube videos, 19, 994 unedited videos, and the training set is divided into three parts in a 2:1:1 ratio. It has an average of 137 uncut videos in each category and contains 1.41 activities per video. In this work, we follow the setting of previous work and evaluate the performance using mean average precision, with tIoU threshold setting to [0.5: 0.5: 0.95], and we set tIoU threshold setting to [0.5: 0.95: 0.05] for average mAP on ActivitiyNet1.3 dataset

4.2 Implementation Details

We chose the Adaptive Moment Estimation (Adam) optimizer[50] for the warm-up training phase. The warm-up training improves the stability and efficiency of training. To ensure consistency of data processing, we set a strict limitation on the maximum length of input sequences, which is 2304. This balances the consumption of computing resources with the integrity of information. For input sequences longer than the limit, we utilized a cropping strategy to get the core parts to fit within the limit of length, and for those shorter than the limit, we padded them to the required length. Besides, we set the dimension of input data to 2048 to balance the richness of information and computation efficiency. According to the characteristics of different datasets, corresponding adjustments were made to the model architecture.

Therefore, we set the window size of the local multi-head attention to 19. This larger window allows the model to capture a broader temporal context. We selected an initial learning rate that could provide a good balance between training speed and model convergence stability.

We employed a more detailed attention window setting for the larger ActivityNet1.3 dataset with more action categories. Considering the density and diversity of action fragments in this dataset, we adjust the size of the local multi-head attention window to 7. This setting enables the model to capture more precise key action segments while avoiding unnecessary computation overhead. Also, an initial learning rate is tuned carefully for this dataset, which will result in a very stable convergence to the optimal solution in the process of training.

4.3 Comparison with Other Methods

We benchmarked our model against leading-edge action detection techniques on the THUMOS14 and ActivityNet1.3, presenting the outcomes in Tables 1 and 2 respectively. These tables also detail the feature extraction methodologies employed by each model.

Table 1 presents the performance metrics obtained on the THUMOS14 dataset. Our method achieved an average mAP of 68.0%. At tIoU = 0.3, the mAP was 82.5%, at tIoU = 0.5, the mAP was 72.1%, and at tIoU = 0.7, the mAP was 45.5%. Compared with CNN-based methods, our results clearly outperform these methods. When compared to some transformer-based methods, our approach also shows an advantage, particularly in comparison with ActionFormer[11], where the average mAP improved by 1.2%. Pure CNN methods (such as CDC and Temporal Context Aggregation Network (TCANet)) are limited by the local receptive field and have inherent flaws in cross-frame dependency modeling. However, a single global attention model (such as TadTR with a pure Transformer architecture) leads to blurred short-action localization due to the lack of local constraints. It is particularly worth noting that under the strict detection standard with an IoU threshold of 0.7, TransCLGA still maintains an outstanding performance of 45.5%, which is 1.6 percentage points higher than the current optimal ActionFormer (43.9%). This fully

Table 1 Performance of temporal action detection on the THUMOS14 measured by mean Average Precision (mAP, %) at various tIoU thresholds. where bold text indicates the best result.

Model	Feature	tIoU					Avg.
		0.3	0.4	0.5	0.6	0.7	
BMN[51]	TSN	56.0	47.4	38.8	29.7	20.5	38.5
BC-GNN[52]	TSN	57.1	49.1	40.4	31.2	23.1	40.2
TCANet[25]	TSN	60.6	53.2	44.6	36.8	26.7	44.3
ReAct[53]	TSN	69.2	65.0	57.1	47.8	35.6	55.0
VTCS[24]	C3D	52.3	49.5	44.3	39.3	30.0	43.1
AFSD[54]	I3D	67.3	62.4	55.5	43.7	31.1	52.0
DRN[55]	I3D	69.2	64.7	57.5	46.9	30.8	53.8
TadTR[8]	I3D	74.8	69.1	60.1	46.6	32.8	56.7
RTD-Net[14]	I3D	68.3	62.3	51.9	38.8	23.7	49.0
BasicTad[56]	R50-SlowOnly	75.5	70.8	63.5	50.9	37.4	59.6
ActionFormer[11]	I3D	82.1	77.8	71.0	59.4	43.9	66.8
LTP[13]	I3D	82.3	78.3	72.1	59.8	44.2	67.3
Hao[57]	I3D	82.4	78.8	71.5	59.8	44.3	67.4
LGAFormer[28]	I3D	82.4	78.9	71.8	60.4	45.2	67.7
TransCLGA(ours)	I3D	**82.5**	**78.9**	**72.1**	**60.8**	**45.5**	**68.0**

demonstrates its excellent ability in precise boundary positioning. This advantage mainly stems from the model's unique hybrid attention mechanism and the synergistic effect of the channel convolution module. The former achieves a balanced modeling of local details and global context through a step-by-step differentiation strategy, while the latter effectively reduces the interfering information during the feature extraction process.

Table 2 shows the results on ActivityNet1.3. Our method achieved an average mAP of 36.8%. Many videos in ActivityNet1.3 contain complex scenes and multiple participants, where multiple actions may occur simultaneously or overlap. This poses higher demands on the model's efficiency and accuracy. From the perspective of average mAP, our model did not show a large improvement on this challenging dataset, but at tIoU = 0.5, the mAP was 54.9%, which is excellent compared to previous transformer-based methods at this threshold. It outperforms 53.5% of ActionFormer and 49.1% of TadTR.

Table 2 Performance of temporal action detection on the ActivityNet1.3 measured by mean Average Precision (mAP, %) at various tIoU thresholds. where bold text indicates the best result.

Model	Feature	tIoU			Avg.
		0.5	0.75	0.95	
BMN[51]	TSN	50.1	34.8	8.3	33.9
BC-GNN[52]	TSN	50.6	34.8	**9.4**	34.3
TCANet[25]	TSN	52.3	36.7	6.9	35.5
ReAct[53]	TSN	49.6	33.0	8.6	32.6
AFSD[54]	I3D	52.4	35.2	6.5	34.3
TadTR[8]	I3D	49.1	32.6	8.5	32.3
RTD-Net[14]	I3D	47.2	30.7	8.6	30.8
ActionFormer[11]	I3D	53.5	36.2	8.2	35.6
ActionFormer[11]	R(2+1)D	54.7	37.8	8.4	36.6
LTP[13]	R(2+1)D	54.7	37.9	9	36.7
TransCLGA(ours)	R(2+1)D	**54.9**	**37.9**	8.6	**36.8**

4.4 Ablation Experiment

We conducted ablation experiments on the THUMOS14 dataset to verify the effects of different model designs and hyperparameters on the final results.

4.4.1 The Effectiveness of Channel Convolutional Block

First, we validated the effectiveness of the Channel Convolutional Block. The results obtained were compared through comparative experiments with the previous attention modules CBAM and SE. We find that the Channel Conv Block proposed in this paper improves the performance of the model. As shown in Table 3, a significant improvement was observed on the THUMOS14 dataset, with mAP increasing by 0.5%. While reducing interfering information, CCB significantly enhances the representation ability of features, providing higher-quality input for subsequent attention modules and feature fusion.

Table 3 Compare whether to use the CCB module and compare it with CBAM/SE

Method	tIoU			Avg.
	0.3	0.5	0.7	
CCB	**82.5**	**72.1**	**45.5**	**68.0**
CBAM	81.7	71.6	44.5	67.0
SE	81.3	70.6	43.9	66.3
None	82.4	71.7	45.5	67.5

4.4.2 The Effectiveness of Hybrid Attention Module

Next, we validated the effectiveness of the Hybrid Attention Module on the THUMOS14 dataset. The Hybrid Attention Module consists of both global and local multi-head attention modules. We needed to explore the effectiveness of each individual module. And explore the connection methods of the two attention modules. The "&" represents weighted fusion and the "+" represents serial connection, which is also the method we designed. The results, as shown in Table 4, confirm that the Hybrid Attention Module is effective for this dataset. Compared with 66.8% mAP using global attention (G_EN) alone or 67.4% local attention (L_EN) using local attention, the serially connected mixed attention (G+L) achieved a breakthrough performance of 68.0%. This design overcomes the problem that local attention loses the global context in the processing of long sequences. In addition, we also attempted the weighted fusion of two types of attention, but the results were not satisfactory.

Table 4 A comparison of the use of local attention and global attention and an exploration of the connection methods between the two

method	tIoU			Avg.
	0.3	0.5	0.7	
G_EN	81.8	70.5	44.5	66.8
L_EN	82.4	71.4	45.5	67.4
G&L	81.6	70.2	44.2	66.4
G+L(ours)	**82.5**	**72.1**	**45.5**	**68.0**

4.4.3 Local Multi-Head Attention Window Settings

In local multi-head attention, the window size affects the range of the input sequence the model can attend to when calculating attention, i.e., the length of the context that can be viewed. Therefore, the window size is crucial. We predetermined five distinct window sizes for assessment, examining their effect on detection performance, as outlined in Table 5. The results indicate that a window size of 19 provided a clear advantage in performance. This window size can balance the richness of information and computational efficiency in a local context. It not only avoids the insufficiency of context information caused by a too small window but also prevents the noise interference introduced by a too large window. In addition, the adaptive downsampling strategy further optimizes the computational efficiency of local attention, enabling the model to maintain high performance even when processing long videos.

Table 5 Comparison of different window sizes in local multi-head attention.

Win size	tIoU			Avg.
	0.3	0.5	0.7	
7	81.3	71.1	44.2	66.9
13	81.7	71.3	45.2	67.3
19	**82.5**	**72.1**	**45.5**	**68.0**
24	82.0	70.8	45.0	67.1
36	81.9	71.2	44.3	67.0

4.4.4 The Effectiveness of Gating Mechanism

The gating mechanism incorporated in the Feature Pyramid dynamically adjusts the contribution of features from different layers to the final feature representation. We compared our design with the classical FPN structure and a simplified FPN from[11], and the gated feature fusion pyramid model outperformed both. And I delved deeply into the effectiveness of the components within the gated FPN, including the gating mechanism and the element multiplication mechanism. M represents the multiplication mechanism and G represents gating. The results, shown in Table 6, The 66.0% mAP of the traditional FPN was significantly surpassed by our design with a gating mechanism (68.0%). Specifically, the performance can reach 67.3% just by introducing element-by-element multiplication (FPN+M), and it is further improved by 0.7% after combining the gating mechanism (FPN+M+G). This progress stems from the precise regulation of the feature flow by the gating mechanism. Compared with the simplified FPN used in ActionFormer, our design has improved the positioning accuracy by 1.6%, fully demonstrating the key value of dynamic feature selection in sequential action detection.

Table 6 Comparison results of different multi-scale fusion methods.

Neck	tIoU			Avg.
	0.3	0.5	0.7	
identity	82.7	71.4	44.4	67.4
FPN	81.5	70.1	42.7	66.0
FPN+M	82.0	72.0	45.1	67.3
FPN+M+G(ours)	**82.5**	**72.1**	**45.5**	**68.0**

4.4.5 Level Number of Feature Pyramid

Within the neck section of our model, we introduced a Feature Fusion Pyramid. The tier count within this pyramid has a bearing on the model's overall detection capabilities. We pre-set seven different pyramid layer numbers (N) to analyze the impact, with results shown in Table 7. The findings suggest that when no pyramid structure is used (N=1), the model's detection performance significantly drops. When the pyramid structure (N=2) is employed, there is a substantial improvement in performance. Further comparison across different values of N indicates that the model achieves the best detection results when N=6.

Table 7 Comparison of the number of different pyramid layers.

N	tIoU			Avg.
	0.3	0.5	0.7	
1	72.5	51.7	16.5	47.8
2	75.7	59.0	25.5	54.3
3	79.9	66.2	36.2	62.2
4	80.2	68.9	40.9	64.5
5	82.0	71.4	43.9	66.9
6	**82.5**	**72.1**	**45.5**	**68.0**
7	82.2	71.3	45.2	67.1

4.5 Visualization

In the analysis of visualization results, TransCLGA demonstrated a more accurate temporal action localization capability than the benchmark model. As shown in Figure 7, the model can not only accurately identify the action boundaries but also maintain stable classification performance in complex scenarios. For instance, in the THUMOS14 test set, for short-duration (<1

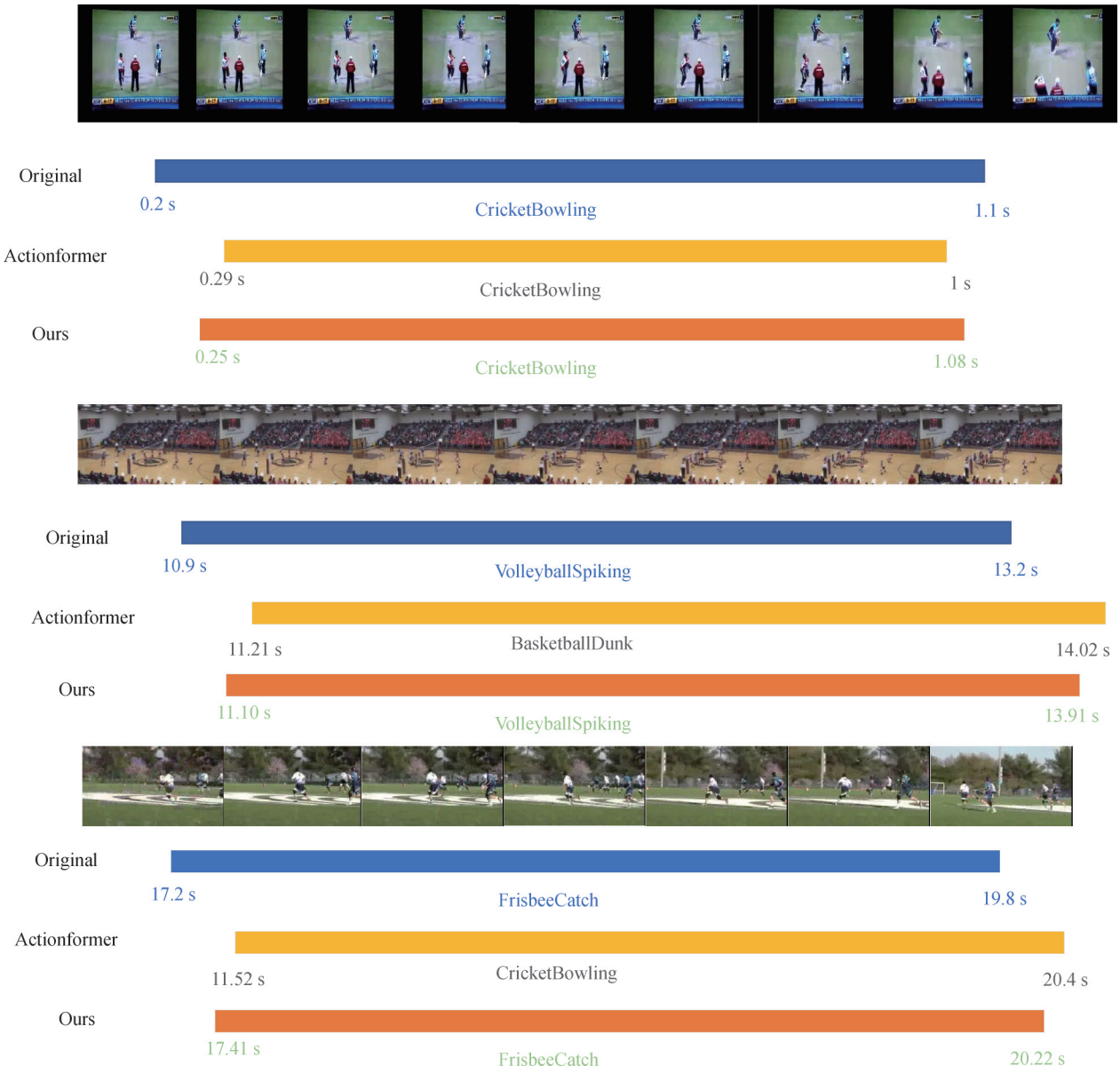


Fig.7 Sample qualitative results for the THUMOS14 test set

second) actions (such as "CricketBowling"), its positioning error is reduced by approximately 15% compared to ActionFormer. These visualization results are mutually corroborated with quantitative analysis, intuitively demonstrating the model's advantages in time series modeling and feature fusion. Beyond quantitative improvements, these visualization results also illustrate practical application scenarios. For example, the precise detection of short-duration sports actions such as "CricketBowling" and "BasketballDunk" shows the potential of TransCLGA in sports video analysis and training systems, where fine-grained motion recognition is crucial for performance evaluation. Similarly, the ability to detect multi-stage and overlapping actions (e.g., the preparation and hitting phases of "FrisbeeCatch") demonstrates the model's applicability in intelligent surveillance and video indexing, where it is important to identify complete and continuous action processes in untrimmed long videos. Furthermore, this ability to recognize multi-stage actions may have application potential in the field of medical surgery. For some rapid and precise surgical actions, our model can also attempt to handle them. Therefore, the visualization results can be used as case studies, further verifying the practical value of our method in real-world applications.

4.6 Error Analysis

We used the tool from [58] to analyze the test results on the THUMOS14 dataset, focusing on false positives, false negatives, and sensitivity.

4.6.1 False Positive Analysis

Following the method in[58], we classify false positives into five categories of errors: Double Detection Error (DD), Wrong Label Error (WL), Localization Error (LOC), Confusion Error (CON), and Background Error (BG). and then analyze the false Positive distribution in the first ten Ground Truth instances of each category, and analyze the True Positive results. Compared to ActionFormer, TransCLGA demonstrates significant improvements in handling false positives, particularly in reducing localization errors (LOC) and background errors (BG). As shown in Figure 8, TransCLGA exhibits fewer instances of double detection (DD) and wrong label (WL) errors, which can be attributed to the Channel Convolutional Block (CCB) and Hybrid Attention Module. The CCB effectively suppresses interference during feature extraction by recalibrating channel weights

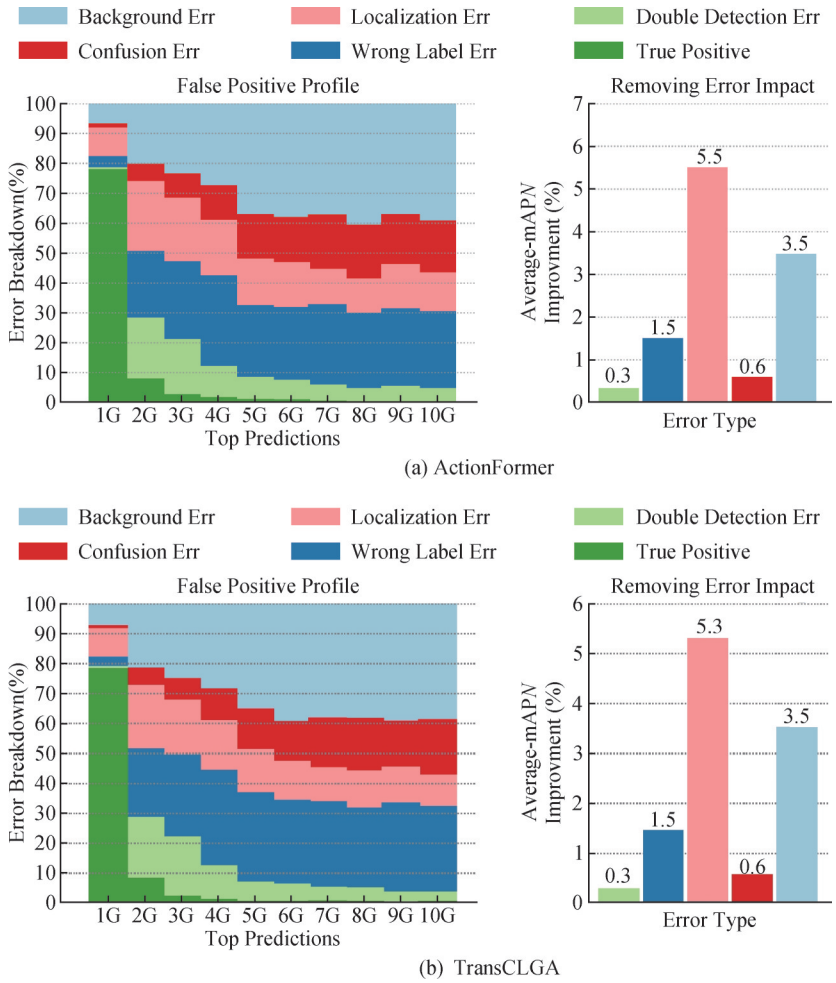


Fig.8 Comparison of false positive error distributions between different methods:
(a) Results from the ActionFormer baseline. (b) Results from our proposed TransCLGA model

and enhancing local spatiotemporal patterns, while the Hybrid Attention Module combines global and local attention to capture both fine-grained details and long-range dependencies. This dual attention mechanism ensures more accurate action boundary predictions and reduces misclassifications caused by irrelevant background segments.

4.6.2 False Negative Analysis

Then we analyse reasons for the false negative errors, defined as cases where no matching test is found at higher than 0.05 confidence level. We follow the analysis method of[58], and define the following three important metrics: Coverage, Length, and Instance Count. Coverage is the proportion of an action instance in a complete video, and it falls into five ranges: Extra Small, Small, Medium, Large, and Extra Large. In contrast, Length refers to the absolute duration of an action instance and then is divided into five ranges: Extra Small, Small, Medium, Large, and Extra Large. Lastly, there is the Instance Count, defined as the total number of action instances in a video and is stratified into four classes, namely Extra Small, Small, Medium, and Large. TransCLGA addresses the limitations of ActionFormer in detecting extremely short or long action segments (Figure 9). The stepwise differentiated attention strategy in the Hybrid Attention Module plays a critical role here. For short actions, the local attention window (size=19) focuses on precise temporal patterns, while the global attention ensures contextual coherence. For long actions, the adaptive downsampling in local attention (controlled by `n_ds_strides`) reduces noise and maintains temporal resolution integrity. The multi-scale depth-wise convolutions in CCB also enhance the model's ability to capture actions of varying durations. As a result, TransCLGA shows a lower false negative rate for "Extra Small" and "Extra Large" segments compared to ActionFormer, particularly in dense action sequences.

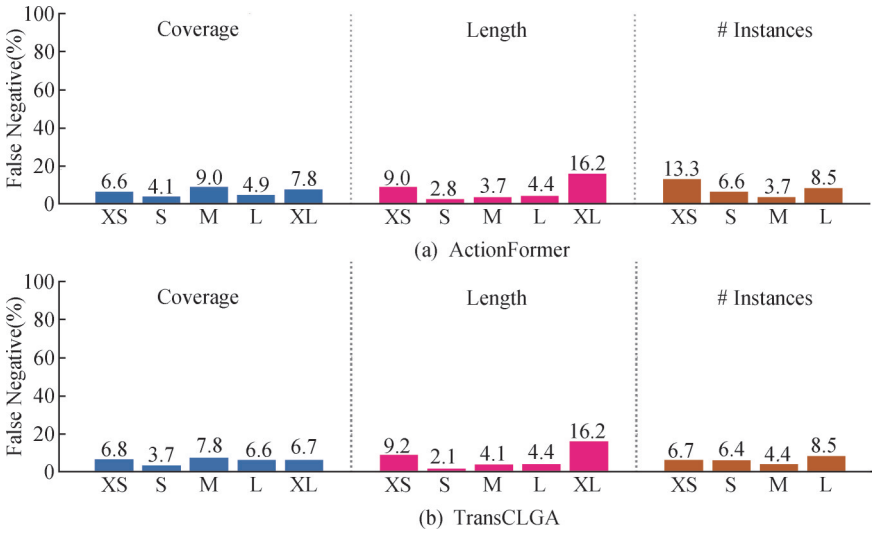


Fig.9 Comparison of false negative error distributions between different methods:
(a) Results from the ActionFormer baseline. (b) Results from our proposed TransCLGA model.

4.6.3 Sensitivity Analysis

In the sensitivity analysis, we compared the performance of TransCLGA and ActionFormer across different action characteristics (Coverage, Length, and Instance Count) at tIoU=0.5, as shown in Figure 10. The left plot reveals that TransCLGA achieves superior mean mAP for videos with "Medium" coverage and instance counts, demonstrating the effectiveness of our gated feature fusion pyramid in handling moderately complex scenarios. While both models show comparable performance for "Extra Large" coverage or "Large" instance counts-indicating the persistent challenge of exceptionally long or dense actions-TransCLGA exhibits more stable performance across "Small" and "Medium" length actions thanks to the Hybrid Attention Module's balanced local-global feature extraction. The right plot highlights TransCLGA's significantly lower variance in mAP, showcasing its remarkable consistency across varying action densities. These results collectively demonstrate that TransCLGA not only matches but surpasses ActionFormer in handling typical video scenarios while maintaining more reliable performance across diverse conditions.

4.7 Analysis of Efficiency-Performance Trade-off

We conducted a comprehensive assessment of the efficiency of the TransCLGA model, with a particular focus on the trade-off relationship between computational cost and detection performance. As shown in Table 8, we compared the model proposed in this paper with the strong baseline method ActionFormer in terms of average mAP, parameter count, computational complexity (Floating Point Operations, FLOPs), and Frames Per Second (FPS).

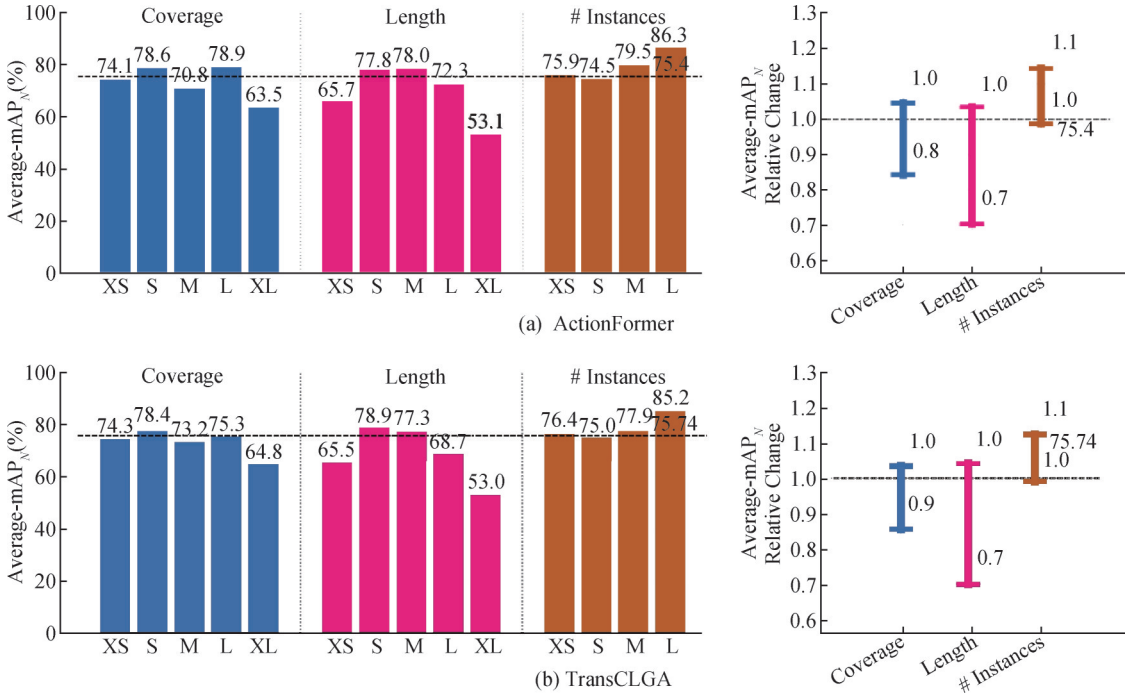


Fig.10 Comparison of sensitivity analysis between different methods:
(a) Results from the ActionFormer baseline. (b) Results from our proposed TransCLGA model.

The experimental results show that TransCLGA has reached the current optimal level in detection performance (68.0% mAP), but its computational cost has increased compared with ActionFormer. Specifically, the number of parameters and FLOPs of TransCLGA increased by approximately 74% and 63% respectively, while the inference speed (FPS) dropped from 15.65 to 12.05, a decrease of about 23%. We believe that this trade-off is reasonable and intentional, mainly for the following reasons:

(1) Precision critical application requirements: In practical application scenarios such as sports video analysis, the accuracy of motion localization and classification is of vital importance. A slight increase in mAP (especially at high IoU thresholds, such as 45.5% vs 43.9% when tIoU=0.7) may significantly affect the actual utility of the system. TransCLGA has improved the average mAP by 1.2% compared to ActionFormer, indicating a significant reduction in the number of false detections and missed detections in actual scenarios.

(2) Applicability of offline processing: Many practical applications (such as sports event video analysis or surveillance video processing) are usually in offline or near-line processing mode, with relatively loose restrictions on processing time. In these scenarios, TransCLGA offers more accurate and reliable solutions.

Although TransCLGA is not specifically designed for ultra-low latency applications, its design concept prioritizes maximizing detection accuracy. The experimental results show that the structural innovation we proposed effectively converts the increased computing resources into significant and valuable performance improvements.

Furthermore, through module ablation experiments, we further revealed the contribution degree of each component to the final performance and the corresponding computational overhead. It can be seen from the experimental results that the introduction of the channel convolution module (CCB) alone can bring a 0.3% improvement in mAP, an increase of 10.43M in parameter quantity, and a decrease of 3.32 frames in FPS. This indicates that this module effectively improves the feature quality at a moderate computational cost. The contribution of the Hybrid Attention Module (HAM) is the most significant. When used alone, it can increase the mAP by 0.6%, but it also brings the greatest computational overhead (15.09M parameters and 19.99G FLOPs), and reduces the FPS by 5.54 frames. This demonstrates the powerful ability of its complex dual-attention mechanism in capturing multi-scale temporal dependencies. When used alone, the performance improvement of the Gated Feature Fusion Pyramid (GFFP) is limited, but its computational cost is not high either, and the FPS only drops by 1 frame. This indicates that its gating mechanism is a balanced approach. When the three modules worked together, the model achieved an optimal performance of 68.0%, which was greater than the simple sum of the individual improvements of each module. Although the FPS further dropped to 12.05, this trade-off between performance and efficiency is reasonable in precision-critical applications. Overall, TransCLGA has achieved a significant improvement in detection performance through the organic combination of modules while maintaining an acceptable inference speed, demonstrating its practical value and potential in complex video understanding tasks.

Table 8 The performance improvements, computational overhead and efficiency impacts brought by each module.

CCB	HAM	GFFP	mAP	Parameters	FLOPs	FPS
-	-	-	66.8	35.90M	34.61G	15.65
√	-	-	67.1	+10.43M	+0.55G	-3.32
-	√	-	67.4	+15.09M	+19.99G	-5.54
-	-	√	67.0	+1.32M	+1.31G	-1.00
√	√	√	68.0	+26.84M	+21.85G	-3.60

4.8 Long-term Stability Assessment

To further verify the stability and reliability of the proposed system under long-term training, in the dataset THUMOS14, we present the loss curves of TransCLGA and the baseline ActionFormer across 35 epochs (see Fig. 11). Both methods are able to converge within the training process, demonstrating the effectiveness of the underlying framework. Notably, TransCLGA exhibits a smoother convergence trend with relatively smaller fluctuations and reaches a stable low loss value (0.04) after around 30 epochs. This indicates that our approach maintains stable performance during extended training and achieves reliable convergence.

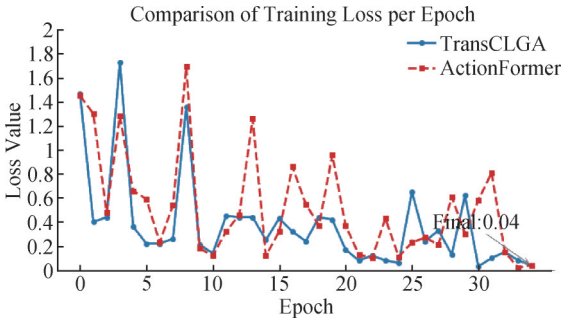


Fig.11 Comparison of training losses per Epoch for TransCLGA and ActionFormer

5 Conclusion

In this paper, we introduced the Channel Convolution Module, designed to extract both rich and detailed channel and spatial information from the input data, while minimizing the interference from noise generated during feature extraction. By cleverly integrating both global and local multi-head attention mechanisms, we are able to extract rich global-local feature information that is crucial for accurately capturing actions in videos. To further enhance the feature representation capability, we employed an advanced gating pyramid structure, enabling effective fusion of multi-scale features. This step significantly facilitates the flow and complementarity of information across different layers.

We developed a Transformer-based anchor-free single-stage temporal action detection model called TransCLGA. The model is not only simple in design, but also has excellent performance, and can efficiently and accurately identify action periods in videos without relying on predefined anchor frames and boxes. To evaluate its performance, we conducted experiments on two datasets: THUMOS14 and ActivityNet1.3. The results showed that our model performed well on both datasets, with particularly impressive results on THUMOS14.

However, in our pursuit of excellence, we remain humble and introspective. While the model showed considerable strength in most test scenarios, its performance against the larger and more complex ActivityNet1.3 dataset still needs to be improved. Moreover, the pursuit of higher accuracy leads to increased computational costs compared to the baseline. This reflects a deliberate trade-off, where we prioritize detection performance for precision-critical applications. Future work will focus on model compression and distillation techniques to enhance efficiency without significantly sacrificing the achieved performance gains. This discovery prompts us to deeply reflect on the limitations of the current model and consider it as an important direction for future research. To address the performance bottleneck, we plan to conduct more in-depth ablation experiments to analyze the differences in the performance of model components on large data sets, so as to accurately locate the improvement space.

The modular architecture of TransCLGA is itself designed with future scalability and integration in mind. Its design opens avenues for practical application and expansion. For instance, the model can be extended to process multi-modal data by incorporating features from audio or inertial sensors, which would be invaluable in scenarios where visual information is ambiguous. In addition, TransCLGA shows potential application as a "candidate clip" generator in video analysis. This method is expected to assist in locating the time periods that may be worth paying attention to in the video and adapt its output results to the existing workflow. This might help downstream AI systems or manual reviews focus more efficiently on certain key segments, thereby enhancing the overall analysis efficiency.

In addition, we will actively explore new algorithmic ideas and techniques, such as introducing more advanced attention mechanisms, optimizing feature fusion strategies, or using additional context information to enhance the generalization ability of the model. Through these efforts, we expect to further improve the model's performance on large data sets, push the boundaries of temporal motion detection technology, and contribute more to the field of video understanding and intelligent analysis.

Author Contribution:

Bin Zhang: Methodology, Software, Validation, Resources, Datacuration, Writing-originaldraft; Yinfeng

Fang: Conceptualization, Validation, formal analysis, Supervision, Writing-review&editing; Xuguang Zhang: Conceptualization, formal analysis, Methodology, Supervision, Validation, Writing-originaldraft, Writing-review&editing; Yun Zhang: Resources, formal analysis, Writing-review&editing,

Foundation Information:

This research was supported by the National Natural Science Foundation of China (no. 61771418). This research was supported by the Zhejiang Key Laboratory of Film and TV Media Technology under Grant 2024E10023.

Data Availability:

The authors declare that the main data supporting the findings of this study are available within the paper and its Supplementary Information files.

Conflicts of Interest:

The authors declare no competing interests.

Dates:

Received 07 June 2025; Accepted 17 November 2025; Published online 31 March 2026

References

[1] Jin C-B, Li S, Kim H. Real-Time Action Detection in Video Surveillance using Sub-Action Descriptor with Multi-CNN. *Journal of Institute of Control, Robotics and Systems* , **2018**, 24(3): 298-308.

[2] Liu J, Che Y. Action recognition for sports video analysis using part-attention spatio-temporal graph convolutional network. *Journal of Electronic Imaging* , **2021**, 30(3): 033017-033017.

[3] Liciotti D, Bernardini M, Romeo L, Frontoni E. A sequential deep learning application for recognising human activities in smart homes. *Neurocomputing* **2020**, 396: 501-513.

[4] Liu Q, Wang Z. Progressive boundary refinement network for temporal action detection. *Proceedings of the AAAI conference on artificial intelligence* , New York, NY, USA, February 7-12, **2020**: 11612-11619.

[5] Shou Z, Chan J, Zareian A, Miyazawa K, Chang S-F. Cdc: Convolutional-de-convolutional networks for precise temporal action localization in untrimmed videos. *Proceedings of the IEEE conference on computer vision and pattern recognition* , Honolulu, HI, USA, July 21-26, **2017**: 5734-5743.

[6] Simonyan K, Zisserman A. Two-stream convolutional networks for action recognition in videos. *Advances in neural information processing systems* , Montreal, Canada, December 8-13, **2014**, 27.

[7] Gao Z, Cui X, Zhuo T, Cheng Z, Liu A-A, Wang M, Chen S. A Multitemporal Scale and Spatial-Temporal Transformer Network for Temporal Action Localization. *IEEE Trans Hum-Mach Syst* **2023**, 53(3): 569-580.

[8] Liu X, Wang Q, Hu Y, Tang X, Zhang S, Bai S, Bai X. End-to-end temporal action detection with transformer. *IEEE Transactions on Image Processing* , **2022**, 31: 5427-5441.

[9] Yang J, Wei P, Zheng N. Cross time-frequency transformer for temporal action localization. *IEEE Transactions on Circuits and Systems for Video Technology* , **2023**, 34(6): 4625-4638.

[10] Dosovitskiy A. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint* arXiv: 2010.11929, **2020**.

[11] Zhang C-L, Wu J, Li Y. ActionFormer: Localizing Moments of Actions with Transformers. *European Conference on Computer Vision* . Cham: Springer Nature Switzerland, Tel Aviv, Israel, October 23-27, **2022**: 492-510.

[12] Lin KQ, Zhang P, Chen J, Pramanick S, Gao D, Wang AJ, Yan R, Shou MZ. UniVTG: Towards Unified Video-Language Temporal Grounding. Proceedings of the IEEE/CVF International Conference on Computer Vision, Paris, France, Oct 2-6, **2023**: 2794-2804.

[13] Kim J, Lee M, Heo J-P. Long-term Pre-training for Temporal Action Detection with Transformers. *Pattern Recognition* , **2025**: 112144.

[14] Tan J, Tang J, Wang L, Wu G. Relaxed Transformer Decoders for Direct Action Proposal Generation. *Proceedings of the IEEE/CVF international conference on computer vision. Montreal* , QC, Canada, Oct 11-17, **2021**: 13526-13535.

[15] Neimark D, Bar O, Zohar M, Asselmann D. Video transformer network. *Proceedings of the IEEE/CVF international conference on computer vision* , Montreal, QC, Canada, Oct 11-17, **2021**: 3163-3172.

[16] Arnab A, Dehghani M, Heigold G, Sun C, Lučić M, Schmid C. Vivit: A video vision transformer. *Proceedings of the IEEE/CVF international conference on computer vision* , Montreal, QC, Canada, Oct 11-17, **2021**: 6836-6846.

[17] Vaswani A. Attention is all you need. *Advances in neural information processing systems* , Long Beach, CA, USA, December 4-9, **2017**, 30.

[18] Beltagy I, Peters ME, Cohan A. Longformer: The Long-Document Transformer. *arXiv preprint* arXiv:2004.05150, **2020**.

[19] Wang S, Li BZ, Khabsa M, Fang H, Ma H. Linformer: Self-Attention with Linear Complexity. *arXiv preprint* arXiv: 2006.04768, **2020**.

[20] Zaheer M, Guruganesh G, Dubey KA, Ainslie J, Alberti C, Ontanon S, Pham P, Ravula A, Wang Q, Yang L. Big bird: Transformers for longer sequences. *Advances in neural information processing systems* , Vancouver, Canada, December 6-12, **2020**, 33: 17283-17297.

[21] Hu J, Shen L, Sun G. Squeeze-and-Excitation Networks. *Proceedings of the IEEE conference on computer vision and pattern recognition* , Salt Lake City, UT, USA, June 18-23, **2018**: 7132-7141.

[22] Lin T-Y, Dollár P, Girshick R, He K, Hariharan B, Belongie S.

Feature pyramid networks for object detection. *Proceedings of the IEEE conference on computer vision and pattern recognition* , Honolulu, HI, USA, July 21-26, **2017**: 2117-2125.

[23] Gao J, Yang Z, Nevatia R. Cascaded Boundary Regression for Temporal Action Detection. *arXiv preprint* arXiv: 1705.01180, **2017**.

[24] Murtaza F, Yousaf MH, Velastin SA, Qian Y. Vectors of temporally correlated snippets for temporal action detection. *Computers & Electrical Engineering* , **2020**, 85: 106654.

[25] Qing Z, Su H, Gan W, Wang D, Wu W, Wang X, Qiao Y, Yan J, Gao C, Sang N. Temporal context aggregation network for temporal action proposal refinement. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* , Nashville, TN, USA, June 19-25, **2021**: 485-494.

[26] Lin T, Zhao X, Shou Z. Single Shot Temporal Action Detection. *Proceedings of the 25th ACM international conference on Multimedia* , Mountain View, CA, USA, October 23-27, **2017**: 988-996.

[27] Yang L, Peng H, Zhang D, Fu J, Han J. Revisiting anchor mechanisms for temporal action localization. *IEEE Transactions on Image Processing* , **2020**, 29: 8535-8548.

[28] Zhang H, Zhou F, Wang D, Zhang X, Yu D, Guan L. LGAFormer: transformer with local and global attention for action detection. *Journal of Supercomputing* , **2024**, 80(12): 17952-17979.

[29] Lei Z, Wang J, Chen S, Niu D. Global Two-Stream Network for Temporal Action Proposal Generation. *Journal of Physics: Conference Series* . IOP Publishing, **2021**, 1920(1): 012063.

[30] Lin J, Gan C, Han S. Tsm: Temporal shift module for efficient video understanding. *Proceedings of the IEEE/CVF international conference on computer vision* , Seoul, South Korea, Oct 27 - Nov 2, **2019**: 7083-7093.

[31] Chen X, Kan S, Zhang F, Cen Y, Zhang L, Zhang D. Multiscale spatial temporal attention graph convolution network for skeleton-based anomaly behavior detection. *J Vis Commun Image Represent* **2023**, 90: 103707.

[32] Nawhal M, Mori G. Activity Graph Transformer for Temporal Action Localization. *arXiv preprint* arXiv:2101.08540, **2021**.

[33] Lowe DG. Distinctive Image Features from Scale-Invariant Keypoints. *International journal of computer vision* , **2004**, 60 (2): 91-110.

[34] Itti L, Koch C, Niebur E. A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on pattern analysis and machine intelligence* , **2002**, 20(11): 1254-1259.

[35] Ren S. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems* , Montreal, QC, Canada, December 7-12, **2015**: 28.

[36] Aslam A, Sargano AB, Habib Z. Attention-based multimodal sentiment analysis and emotion recognition using deep neural networks. *Applied Soft Computing* , **2023**, 144: 110494.

[37] Carreira J, Zisserman A. Quo vadis, action recognition? a new model and the kinetics dataset. *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* , Honolulu, HI, USA, July 21-26, **2017**: 6299-6308.

[38] Wang L, Xiong Y, Wang Z, Qiao Y, Lin D, Tang X, Van Gool L. Temporal segment networks for action recognition in videos. *IEEE transactions on pattern analysis and machine intelligence* , **2018**, 41(11): 2740-2755.

[39] Huang H, Chen Z, Zou Y, Lu M, Chen C, Song Y, Zhang H, Yan F. Channel prior convolutional attention for medical image segmentation. *Computers in Biology and Medicine* , **2024**, 178: 108784.

[40] Woo S, Park J, Lee J-Y, Kweon IS. Cbam: Convolutional block attention module. *Proceedings of the European conference on computer vision (ECCV)* , Munich, Germany, September 8-14, **2018**: 3-19.

[41] Pecoraro R, Basile V, Bono V. Local multi-head channel self-attention for facial expression recognition. *Information* , **2022**, 13(9): 419.

[42] Kumar A, Vepa J. Gated mechanism for attention based multi modal sentiment analysis. ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, Barcelona, Spain, May 4-8, **2020**: 4477-4481.

[43] Deng C, Wang M, Liu L, Liu Y, Jiang Y. Extended feature pyramid network for small object detection. *IEEE Transactions on Multimedia* , **2021**, 24: 1968-1979.

[44] Kim S-W, Kook H-K, Sun J-Y, Kang M-C, Ko S-J. Parallel feature pyramid network for object detection. *Proceedings of the European conference on computer vision (ECCV)* , Munich, Germany, September 8-14, **2018**: 234-250.

[45] Lin T. Focal Loss for Dense Object Detection. *Proceedings of the IEEE international conference on computer vision* , Venice, Italy, Oct 22-29, **2017**: 2980-2988.

[46] Rezatofighi H, Tsoi N, Gwak J, Sadeghian A, Reid I, Savarese S. Generalized intersection over union: A metric and a loss for bounding box regression. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* , Long Beach, CA, USA, June 15-20, **2019**: 658-666.

[47] Bodla N, Singh B, Chellappa R, Davis LS. Soft-NMS-improving object detection with one line of code. *Proceedings of the IEEE international conference on computer vision* , Venice, Italy, Oct 22-29, **2017**: 5561-5569.

[48] Idrees H, Zamir AR, Jiang Y-G, Gorban A, Laptev I, Sukthankar R, Shah M. The thumos challenge on action recognition for videos "in the wild." *Computer Vision and Image Understanding* , **2017**, 155: 1-23.

[49] Caba Heilbron F, Escorcia V, Ghanem B, Carlos Niebles J. Activitynet: A large-scale video benchmark for human activity understanding. *Proceedings of the ieee conference on computer vision and pattern recognition* , Boston, MA, USA, June 7-12, **2015**: 961-970.

[50] Loshchilov I, Hutter F. Fixing weight decay regularization in adam. *arXiv preprint arXiv* :1711.05101, **2017**, 5(5): 5.

[51] Lin T, Liu X, Li X, Ding E, Wen S. Bmn: Boundary-matching

network for temporal action proposal generation. *Proceedings of the IEEE/CVF international conference on computer vision* , Seoul, South Korea, Oct 27 - Nov 2, **2019**: 3889-3898.

[52] Bai Y, Wang Y, Tong Y, Yang Y, Liu Q, Liu J. Boundary Content Graph Neural Network for Temporal Action Proposal Generation. *European conference on computer vision* . Cham: Springer International Publishing, Glasgow, UK, August 23-28, **2020**, 121-137.

[53] Shi D, Zhong Y, Cao Q, Zhang J, Ma L, Li J, Tao D. ReAct: Temporal Action Detection with Relational Queries. *European conference on computer vision* . Cham: Springer Nature Switzerland, Tel Aviv, Israel, October 23-27, **2022**: 105-121.

[54] Lin C, Xu C, Luo D, Wang Y, Tai Y, Wang C, Li J, Huang F, Fu Y. Learning salient boundary feature for anchor-free temporal action localization. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* , Nashville, TN, USA, June 19-25, **2021**: 3320-3329.

[55] Xia K, Wang L, Zhou S, Hua G, Tang W. Dual relation network for temporal action localization. *Pattern Recognition* , **2022**, 129: 108725.

[56] Yang M, Chen G, Zheng Y-D, Lu T, Wang L. Basictad: an astounding rgb-only baseline for temporal action detection. *Computer Vision and Image Understanding* , **2023**, 232: 103692.

[57] Wang H, Wei P, Liu M, Zheng N. Temporal Deformable Transformer for Action Localization. International Conference on Artificial Neural Networks. Cham: Springer Nature Switzerland, Heraklion, Crete, Greece , September 26-29, **2023**: 563-575.

[58] Alwassel H, Heilbron FC, Escorcia V, Ghanem B. Diagnosing error in temporal action detectors. *Proceedings of the European conference on computer vision (ECCV)* , Munich, Germany, September 8-14, **2018**: 256-272.