applied sciences

MDPI

Article

Video Human Action Recognition Based on Motion-Tempo Learning and Feedback Attention

Yalong Liu [1,2], Chengwu Liang [1,2,*], Songqi Jiang [1,2] and Peiwang Zhu [1,2]

[1] College of Electrical Engineering and New Energy, China Three Gorges University, Yichang 443002, China; 202208110021009@ctgu.edu.cn (Y.L.); 202208540021008@ctgu.edu.cn (S.J.); 202208540021056@ctgu.edu.cn (P.Z.)
[2] School of Electrical and Control Engineering, Henan University of Urban Construction, Pingdingshan 467036, China
* Correspondence: liangchengwu@huuc.edu.cn

Abstract: In video human action-recognition tasks, motion tempo describes the dynamic patterns and temporal scales of human motion. Different categories of actions are typically composed of sub-actions with varying motion tempos. Effectively capturing sub-actions with different motion tempos and distinguishing category-specific sub-actions are crucial for improving action-recognition performance. Convolutional Neural Network (CNN)-based methods attempted to address this challenge, by embedding feedforward attention modules to enhance the action's dynamic representation learning. However, feedforward attention modules rely only on local information from low-level features, lacking contextual information to generate attention weights. Therefore, we propose a Sub-action Motion information Enhancement Network (SMEN) based on motion-tempo learning and feedback attention, which consists of the Multi-Granularity Adaptive Fusion Module (MgAFM) and Feedback Attention-Guided Module (FAGM). MgAFM enhances the model's ability to capture crucial sub-action intrinsic information by extracting and adaptively fusing motion dynamic features at different granularities. FAGM leverages high-level features that contain contextual information in a feedback manner to guide low-level features in generating attention weights, enhancing the model's ability to extract more discriminative spatio-temporal and channel-wise features. Experiments are conducted on three datasets, and the proposed SMEN achieves top-1 accuracies of 52.4%, 63.3% on the Something-Something V1 and V2 datasets, and 76.9% on the Kinetics-400 dataset. Ablation studies, evaluations, and visualizations demonstrate that the proposed SMEN is effective for sub-action motion tempo and representation learning, and outperforms compared methods for video action recognition.

Keywords: action recognition; convolutional neural network; feedback attention; motion tempo; sub-action

check for updates

Academic Editors: João M. F. Rodrigues and Eleonora Iotti

Received: 26 February 2025
Revised: 3 April 2025
Accepted: 8 April 2025
Published: 10 April 2025

Citation: Liu, Y.; Liang, C.; Jiang, S.; Zhu, P. Video Human Action Recognition Based on Motion-Tempo Learning and Feedback Attention. *Appl. Sci.* **2025**, *15*, 4186. https://doi.org/10.3390/app15084186

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Video human action recognition and analysis is one key research topic pertaining to applications including healthcare monitoring, human–computer interaction, gait analysis, and security surveillance [1–4]. For example, in healthcare settings, recognizing human activities allows for understanding the motion patterns and habits of elderly individuals. Thus, timely emergency responses, through video action recognition, will contribute to the creation of safer and healthier living environments for communities. However, for sports and daily activities, action motion's visual tempo and temporal representation learning, similar action categories with long-range temporal dynamic structure, sub-action

sharing within similar action categories, and non-periodic interactions make video action recognition more challenging.

Early researchers strived to explore hand-crafted feature-extraction methods for action-recognition tasks, which perform well in simple actions but struggle to handle complex ones, especially for subtle actions characterized by rapid changes over short periods, such as slight hand movements or brief interactions with objects. Researchers have proposed various 2D CNNs [5–7] and 3D CNNs [8–10] approaches to learn motion representation for video action recognition, with end-to-end manner.

However, motion-tempo modeling at a distinct temporal scale which tends to describe the motion dynamics and the long-term or short-term motion tempos of actions, thus benefits for video action recognition, is often overlooked [11,12]. For instance, due to the distinctive age, gender, and objects interacted with, the activity habits and motion patterns of persons are different when performing the same action categories. Speed variations result in large intra-class variations. Failing to model the motion tempo and the representation learning of action's dynamic information deteriorates the action-recognition performance, especially for recognizing similar actions with less detailed dynamic or appearance differences, such as "Long jumping" and "Running", "Hugging" and "Approaching", "Pushing" and "Kicking", etc.

As shown in Figure 1, ubiquitous sub-action sharing phenomenons between similar action categories make video action recognition more challenging. For example, in the Kinetics-400 dataset [13], the action category "Triple jump" is often confused with similar categories like "High jump" and "Long jump", since they share the common sub-actions "Approach" and "Landing", which contribute less to distinguishing them. It noted that the unique category-specific sub-action "Flight" is crucial and contributes more to the action category "High jump".

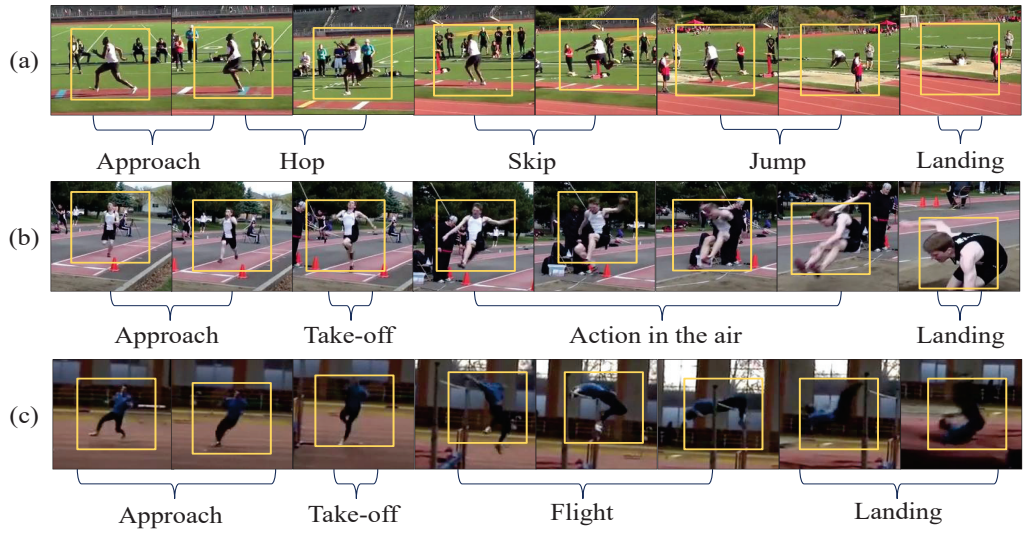


Figure 1. From the Kinetics-400 datasets, the eight video frames are sampled from similar action categories "(**a**) Triple jump", "(**b**) Long jump" and "(**c**) High jump". It can be observed that they contain the same sub-actions, and have varying motion tempos.

Actually, from Figure 1, it is observed that different action categories have varying motion tempos; "Take-off" is a fast-tempo sub-action, while the action-specific sub-actions, such as "Hop", "Skip", "Flight", and "Action in the air" have slower tempos. During the training of CNNs, more computational resources are allocated to extract the information of same video frames, but overlook the category-specific motion dynamic tempos, thus struggling to distinguish similar actions. In contrast, the unique sub-action information specific to each category is crucial for recognizing them. For instance, the sub-action "Action in the air" is crucial for recognition of the action category "Long jump". Therefore,

effectively capturing and distinguishing the crucial sub-actions of action categories is vital for video action classification.

To effectively enhance the ability of CNNs to capture sub-action information, two-stream methods [14] and temporal segment networks (TSNs) [15] utilize pre-extracted optical flow to extract fast motion-tempo information of sub-actions. However, time-consuming optical flow computation requires substantial computational resources. To this, Motion Squeeze [16] is proposed by employing lightweight motion-feature learning to replace the resource-intensive optical flow calculation, directly from adjacent video frames. Additionally, researchers have applied attention mechanisms to capture key sub-action information for video action recognition. For instance, attention modules [17,18] that assess the importance of different parts of the input features are utilized along the temporal, channel, or spatial dimensions by calculating a set of weights. The weights are then applied to the input features to extract critical information within the corresponding dimensions. The self-attention-based modules are proposed [19–22] for processing input features, by calculating the similarity between each position and all other positions of feature maps. Applying these weights to the input features allows us to capture global information more effectively.

However, on one hand, the models used to capture the aforementioned sub-action information can effectively extract the sub-action's fast-tempo information and struggle to capture slow tempos. On the other hand, these attention modules operate in a feed-forward manner, where information flows from the input layer to the output layer in a single direction, relying solely on low-level features before the attention module generates attention weights. The feedforward manner has the advantages of computational simplicity and ease of training but limits the attention mechanism to the representational capacity and low-level semantic information of a single feature. Actually, in the visual system [23], high-level semantic information and task-specific contextual information can be obtained from high-level features through feedback connections, in which high-level information is transmitted back to lower layers. Feedback connections have been proven effective in the visual cortex of primates by neuroscientists [24] and are considered to be a key component of visual attention [25,26]. Similarly, human visual perception transmits high-level semantic information through feedback connections and attention, selectively enhancing and suppressing neuronal activation [27].

Therefore, inspired and motivated by the aforementioned truths, we propose a Multi-Granularity Adaptive Fusion Module (MgAFM), to capture sub-action temporal information at various motion tempos and fuse it with the original RGB features for later representation learning. MgAFM captures crucial sub-action intrinsic information by adaptive motion dynamic features fusion at different granularities.

On the other hand, we propose a Feedback Attention-Guided Module (FAGM), which integrates the action semantic and action category-specific contextual information in a feedback manner. FAGM aims to guide the low-level feature learning by the high-level contextual information, focusing on sub-action spatio-temporal and channel-wise information, enhancing the model to focus on information crucial for action classification. In summary, we propose the Sub-action Motion information Enhancement Network (SMEN), mainly consisting of MgAFM and FAGM, designed to improve action classification performance by effectively capturing and discriminating sub-actions with diverse motion tempos. The main contributions of this paper can be summarized as follows:

- To capture sub-action motion tempos that characterize the dynamic motion information at different temporal granularities, we propose a Multi-Granularity Adaptive Fusion Module (MgAFM). This module captures sub-action temporal information

at various motion tempos and adaptively fuses it with the original RGB features for subsequent representation learning.
- The Feedback Attention-Guided Module (FAGM) is proposed, which utilizes high-level features containing action semantics and action category-specific contextual information to guide low-level features in a feedback manner, focusing on important spatio-temporal and channel-wise information for distinguishing category-specific sub-actions.
- Extensive experiments on Something-Something V1, Something-Something V2, and Kinetics-400 datasets achieve top-1 accuracies of 52.4%, 63.3%, and 76.9% respectively, demonstrating the effectiveness of the proposed model.

2. Related Work on Video Action Recognition Methods

According to neural network backbones, sub-action motion-tempo learning, and attention mechanisms for video action recognition, existing methods can be classified into CNN-based, sub-action motion learning-based, and attention mechanism-based methods.

2.1. Methods Based on CNNs

The existing CNN-based methods tailored for human action recognition mainly have two categories, 2D CNN and 3D CNN approaches. As a typical 2D CNN, TSN [15] leverages a sparse temporal sampling strategy to model long-term temporal dependencies. Meanwhile, Temporal Shift Module (TSM) [7] employs a zero-parameter mechanism that shifts specific channels along the temporal dimension, enhancing the flow of information between consecutive frames.

In contrast, 3D CNNs inherently possess the capability to directly model temporal information with 3D convolutional kernels. Convolutional 3D (C3D) [8] adopts a straightforward yet effective spatio-temporal feature-learning method, effectively capturing dynamic video content for video action recognition tasks. However, resource-intensive 3D CNNs often entail a substantial number of model parameters, leading to computational burden and training inefficiency. To this, Inflated 3D Convolutional Networks (I3D) [9] extends 2D convolutional kernels into 3D by expanding along the temporal axis, thus enabling the efficient processing of spatio-temporal features within video data. Pseudo-3D Networks (P3D) [28] decompose 3D convolutions into a 2D spatial and 1D temporal convolutions. I3D [9] and P3D [28] successfully mitigate the computational burden while maintaining robust spatio-temporal feature extraction.

Nevertheless, compared to 3D CNNs, 2D CNNs offer computational efficiency and simpler training pipelines, highlighting a trade-off between temporal modeling accuracy and resource demands.

2.2. Methods for Capturing Motion Tempos of Sub-Action

Accurately capturing sub-action information at varying motion tempos is essential for improving a model's ability to capture the temporal dynamics of video actions. Sub-actions often exhibit considerable differences in their motion tempos. To address this challenge, SlowFast [29] employs a dual-pathway architecture that captures motion information of slow–fast tempos. MotionSqueeze [16] utilizes adjacent RGB features to generate frame-level motion representations, which aids in capturing fast-tempo sub-action information, whereas struggles to adequately capture slow-tempo sub-actions. Similarly, AGPN [30] uses RGB features sampled at two different frame intervals to capture both fast-tempo and slow-tempo sub-action motion information.

However, these methods are limited to fully capture the sub-action information that is most pertinent to action classification. To capture important sub-actions with varying motion tempos, we propose a Multi-Granularity Adaptive Fusion Module (MgAFM) to

extract a crucial sub-action's intrinsic information with fast and slow motion tempos, and fuses them adaptively at different granularities.

2.3. Video Action Recognition Method Based on Attention Mechanism

Attention mechanism-based video action recognition methods have demonstrated significant improvements to capture key sub-action information. For instance, TEA [17] adopts an attention mechanism to propose the Motion Excitation (ME) module, which operates along the channel dimension, enhancing the weights of channels that focus on dynamic motion patterns. The famous non-local network [20] utilizes a feedforward self-attention mechanism to design a non-local module for capturing long-range dependencies.

However, these methods lack the exploitation of high-level features that contain semantic and task-specific contextual information. In contrast, we proposed FAGM to focus on the sub-action motion-tempo information that is essential for video action classification.

3. The Proposed Critical Action Information Extraction Network

In this section, we will describe the proposed SMEN. As depicted in Figure 2, SMEN uses ResNet50 as its backbone and is composed of two main components: the Multi-Granularity Adaptive Fusion Module (MgAFM) and the Feedback Attention-Guided Module (FAGM). Initially, the T visual frames of the action videos are processed via convolution operations and encoded into feature maps. To capture motion features of sub-actions with varying tempos, the designed MgAFM is introduced following the layer Res3. The temporal dynamics are explored across multiple temporal spans, incorporating various temporal granularities, and then are fused with RGB appearance information adaptively. To identify the most important contextually relevant feature maps, a feedback attention module (FAGM) is designed for video representation learning along temporal, channel, and spatial dimensions.

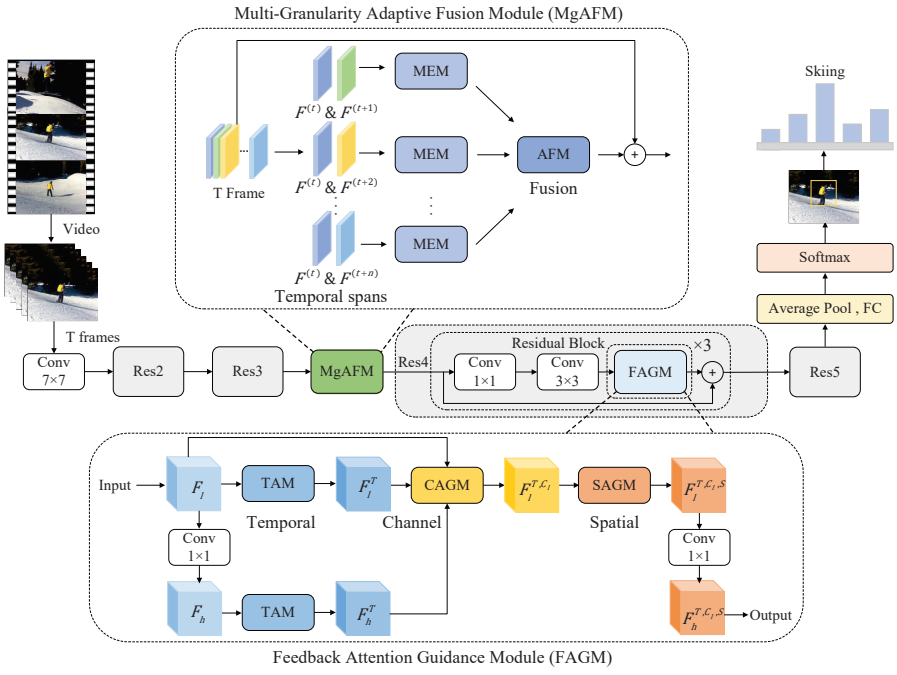


Figure 2. The architecture of our SMEN. It takes ResNet50 [31] as its backbone and primarily consists of MgAFM and FAGM modules. The SMEN takes T visual frames of the action videos as input and transforms them into frame-wise appearance features. In MgAFM, the RGB frames are utilized to capture sub-action motion dynamics with fast and slow tempos. In FAGM, multi-level features are combined as input and then are passed through attention modules along the temporal, channel, and spatial dimensions to generate attention weights. These weights guide low-level features to focus on essential sub-action information.

3.1. Multi-Granularity Adaptive Fusion Module (MgAFM)

The MgAFM aims to capture sub-action motion dynamics with fast and slow tempos of varying motion amplitudes and speeds, from RGB features. MgAFM includes the Motion-Extraction Module (MEM) and the Adaptive Fusion Module (AFM), which extract and adaptively fuse sub-action motion features with varying tempos.

3.1.1. Motion-Extraction Module (MEM)

The MEM is responsible for extracting motion features from sub-actions at different tempos. As illustrated in Figure 3, we employ motion transformations with $(n-1)$ frame intervals to represent sub-actions motion features at different tempos. For the video data, T frames are sampled as input to the model and subsequently converted into feature representations through convolution calculation. The features corresponding to different frame intervals are represented as $\mathbf{F}^t \in \mathbb{R}^{C\times H\times W}$ and $\mathbf{F}^{t+n} \in \mathbb{R}^{C\times H\times W}$, where C, H, and W represent the channel dimension, height, and width, respectively. The C dimensional features at spatial position x are denoted by $\mathbf{F}_x$. The correlation score for position x with a displacement p is defined as:

$$s(x,p,t) = \mathbf{F}_x^t \cdot \mathbf{F}_{x+p}^{t+n}. \tag{1}$$

where $\cdot$ denotes the dot product. To enhance computational efficiency, x is computed only within a neighborhood of size $P = 2R+1$, limiting the displacement to $p \in [-R, R]^2$. For the t_{th} frame, the resulting correlation tensor $\mathbf{S}^{(t)}$ has dimensions $H \times W \times P^2$. In our implementation, considering the spatial resolution of 28×28, we set $P = 15$.

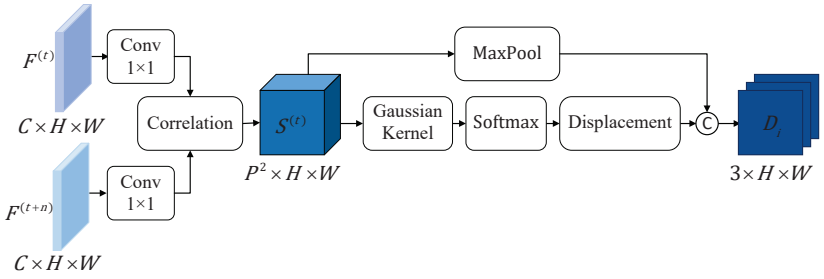


Figure 3. The Motion-Extraction Module (MEM). The module takes two RGB video frame-level features, $\mathbf{F}^t$ and $\mathbf{F}^{t+n}$, with an interval of $n-1$, as input. Through correlation computation, it estimates motion changes and transforms them into the correlation tensor $S^{(t)}$. This tensor is then used as input to estimate the displacement tensor D_i via a Gaussian kernel.

To establish the motion relationship between video frames at different intervals, the correlation tensor $\mathbf{S}^{(t)}$ is utilized to generate a displacement field that describes the motion trajectory. To mitigate the influence of noisy outliers in the correlation tensor on the displacement estimation, we adopt the kernel-soft-argmax method [32]. The kernel-soft-argmax for displacement computation is defined as:

$$d(x,t) = \sum_p \frac{\exp(g(x,p,t)s(x,p,t)/\tau)}{\sum_{p'}\exp(g(x,p',t)s(x,p',t)/\tau)} p. \tag{2}$$

where

$$g(x,p,t) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left(\frac{p-\text{argmax}_p s(x,p,t)}{\sigma^2}\right). \tag{3}$$

the function $g(x,p,t)$ represents a Gaussian kernel, with the standard deviation σ empirically set to 5. The temperature factor τ is introduced to adjust the softmax distribution, and based on empirical observations in our experiments, τ is set to 0.01. Additionally, a

confidence map of the correlation tensor is applied as supplementary motion information. This confidence map is generated by pooling the highest correlation at each position x:

$$s^*(x,t) = \max_p s(x,p,t). \tag{4}$$

To represent sub-action motion features at different tempos, we define a displacement tensor $\mathbf{D}_i \in \mathbb{R}^{H\times W\times 3}$, which is formed by concatenating the single-channel confidence map and the two-channel displacement map.

3.1.2. Adaptive Fusion Module (AFM)

The AFM adaptively fuses motion features from sub-actions at different tempos to capture the varying dynamics. As shown in Figure 4, the displacement tensor resulting from MEM processing is denoted as $\mathbf{D}_1, \mathbf{D}_2, \ldots, \mathbf{D}_n$. Initially, the displacement tensors from the n branches, each corresponding to different motion tempos sub-actions, are connected and passed through average pooling to obtain a global representation. This global representation is then fed into MLP and sigmoid function to compute the adaptive weight vector:

$$\mathbf{D}_c = [\mathbf{D}_1; \mathbf{D}_2; \cdots; \mathbf{D}_n] \in R^{3n\times H\times W} \tag{5}$$

$$\mathbf{W}_c = \sigma(\mathrm{MLP}(\mathrm{AvgPool}(\mathbf{D}_c))) \in R^{3n\times 1\times 1} \tag{6}$$

where σ denotes the sigmoid function and $\mathbf{D}_c$ represents the features concatenated along the channel dimension. Secondly, the concatenated $\mathbf{D}_c$ is element-wise multiplied with the weight vector $\mathbf{W}_c$ to yield the feature vector $\mathbf{D}_f$:

$$\mathbf{D}_f = \mathbf{D}_c \odot \mathbf{W}_c \in R^{3n\times H\times W} \tag{7}$$

where $\odot$ denotes the element-wise multiplication. Finally, we use three $1\times 3\times 3$ separable convolutions to convert the displacement tensor $\mathbf{D}_f$ into features with the same dimensions as the original RGB features.

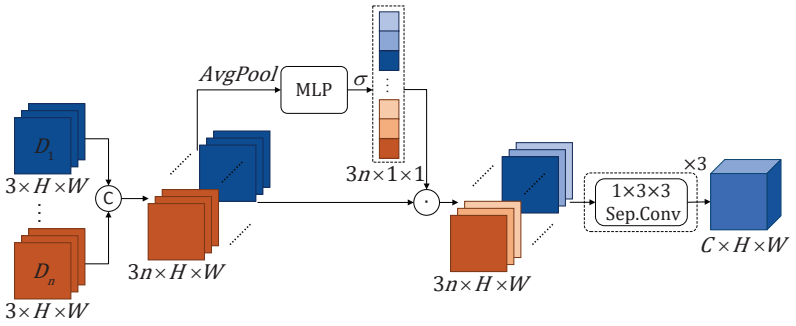


Figure 4. Adaptive Fusion Module (AFM). For the displacement tensor $\mathbf{D}_i$ with different frame intervals, we first concatenate them along the channel dimension, then apply average pooling, MLP, and the sigmoid function to generate adaptive weights. These adaptive weights are then added to the concatenated displacement tensor. Finally, we use three $1\times 3\times 3$ separable convolutions to convert it into features with the same dimensions as the original RGB features.

3.2. Feedback Attention-Guided Module (FAGM)

To further enhance the capability of CNNs to capture sub-action information, we propose a Feedback Attention-Guided Module (FAGM). The FAGM consists of attention modules applied to the temporal, channel, and spatial dimensions. Firstly, the low-level and high-level features are separately fed into the Temporal Attention Module (TAM), to take more attention to frames that are informative with salient motion regions. Next, the features processed by TAM are then combined and passed into the Channel Attention-Guided

Module (CAGM) and the Spatial Attention-Guided Module (SAGM). CAGM and SAGM modules leverage contextual information to guide the low-level feature learning, to focus on the channels and spatial regions most relevant for promoting video action classification.

3.2.1. Temporal Attention Module (TAM)

As shown in Figure 2, within FAGM, the features before and after the convolution 1×1 operation are divided into low-level and high-level features, denoted as $\mathbf{F}_l \in \mathbb{R}^{N \times C_l \times H_l \times W_l}$ and $\mathbf{F}_h \in \mathbb{R}^{N \times C_h \times H_h \times W_h}$, respectively. For the low-level features $\mathbf{F}_l$, as illustrated in Figure 5, we first perform a feature reshape to expand $\mathbf{F}_l$ to $\mathbf{F}_l{}' \in \mathbb{R}^{N_1 \times T \times C_l \times H_l \times W_l}$ ($N_1 = N/T$). Secondly, $W_l \in \mathbb{R}^{N_1 \times T \times 1 \times 1 \times 1}$ is obtained through the operation of average pooling $\mathbf{F}'_l$ along the temporal dimension and processing it with a sigmoid function. Finally, after performing element-wise multiplication on W_l and $\mathbf{F}'_l$, the result is added to $\mathbf{F}'_l$ and reshaped into the feature $\mathbf{F}_l^T \in \mathbb{R}^{N \times C_l \times H_l \times W_l}$, which contains key frame information. The high-level features $\mathbf{F}_h$ are processed in the same way to obtain the feature $\mathbf{F}_h^T$. The specific formulas are as follows:

$$\mathbf{F}'_l = \text{reshape}(\mathbf{F}_l) \tag{8}$$

$$W_l = \sigma(\text{AvgPool}(\mathbf{F}'_l)) \tag{9}$$

$$\mathbf{F}_l^T = \text{reshape}\left(W_l \odot \mathbf{F}'_l + \mathbf{F}'_l\right) \tag{10}$$

After processing with TAM, the model can focus on video frames' rich motion information, reducing the impact of frames with less noticeable motion changes on video action recognition.

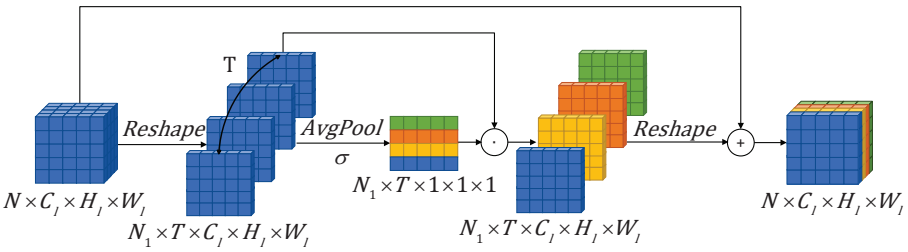


Figure 5. Temporal Attention Module (TAM). The module first expands the time dimension of the original feature with shape $\mathbb{R}^{N \times C_l \times H_l \times W_l}$ and reshapes it to the feature with shape $\mathbb{R}^{N_1 \times T \times C_l \times H_l \times W_l}$. Then, average pooling is applied along the time dimension to generate attention weights for each frame, which are assigned to each frame. Finally, the result is transformed back to the original dimensions and added to the original features for subsequent computations.

3.2.2. Channel Attention-Guided Module (CAGM)

As shown in Figure 6, the features $\mathbf{F}_l^T$ and $\mathbf{F}_h^T$ are first average-pooled along the channel dimension to generate feature vectors $V_l \in \mathbb{R}^{N \times C_l \times 1 \times 1}$ and $V_h \in \mathbb{R}^{N \times C_h \times 1 \times 1}$. Next, the number of channels is reduced into C_l/α through separate Multilayer Perceptron (MLP). The resulted vectors are then concatenated along the channel dimension and processed through another MLP to generate the visual attention-guided feature $V_g \in \mathbb{R}^{N \times C_l \times 1 \times 1}$. Finally, V_g is fed into a sigmoid function to produce the channel attention weight vector $W_c \in \mathbb{R}^{N \times C_l \times 1 \times 1}$. This weight vector W_c is used for element-wise multiplication with the low-level feature $\mathbf{F}_l$ to obtain the feature $\mathbf{F}_l^{T,C_l} \in \mathbb{R}^{N \times C_l \times H_l \times W_l}$. The specific formulas are as follows:

$$V_l = \text{AvgPool}(\mathbf{F}_l^T) \tag{11}$$

$$V_h = \text{AvgPool}(\mathbf{F}_h^T) \tag{12}$$

$$V_g = \text{MLP}_3[\text{MLP}_1(V_l); \text{MLP}_2(V_h)] \tag{13}$$

$$W_c = \sigma(V_g) \tag{14}$$

$$\mathbf{F}_l^{T,C_l} = W_c \odot \mathbf{F}_l \tag{15}$$

After processed by the CAGM, different channels of the low-level features are assigned different weights by the guidance of high-level contextual information leverage.

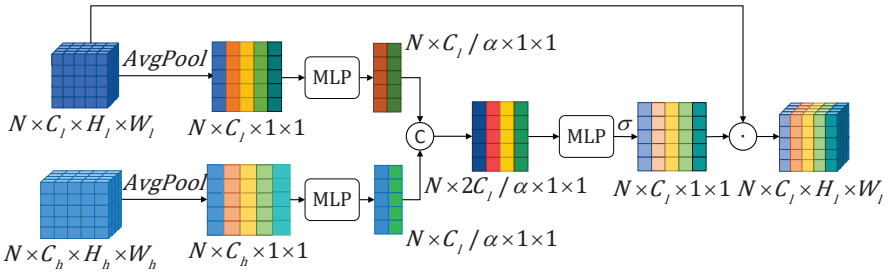


Figure 6. The Channel Attention-Guided Module transformed multi-level features into same dimension along the channel dimension, which are then concatenated accordingly. Then attention weights are generated along the channel dimension via an MLP and applied to the original low-level features. In this way, the contextual information contained in the high-level features is integrated into the low-level features and guides the low-level features to focus on important channel information.

3.2.3. Spatial Attention-Guided Module (SAGM)

As demonstrated in Figure 7, $\mathbf{F}_l^{T,C_l}$ undergoes pooling separately to capture global and local salient information. Then, the results are concatenated to obtain the feature $\mathbf{F}_s \in \mathbb{R}^{N \times 2 \times H_l \times W_l}$. Afterward, $\mathbf{F}_s$ is processed through a convolutional layer for dimension reduction, followed by a sigmoid function. Finally, the result is then element-wise multiplied with $\mathbf{F}_l^{T,C_l}$ to obtain $\mathbf{F}_l^{T,C_l,S} \in \mathbb{R}^{N \times C_l \times H_l \times W_l}$. The specific formulas are as follows:

$$\mathbf{F}_s = [\text{AvgPool}(\mathbf{F}_l^{T,C_l}); \text{MaxPool}(\mathbf{F}_l^{T,C_l})] \tag{16}$$

$$\mathbf{F}_l^{T,C_l,S} = \sigma[Conv_{7\times7}(\mathbf{F}_s)] \odot \mathbf{F}_l^{T,C_l} \tag{17}$$

where $\text{Conv}_{7\times7}$ represents a convolution operation with a kernel size of 7×7. By iterating through the above steps multiple times, the model can fully exploit the contextual information within high-level features.

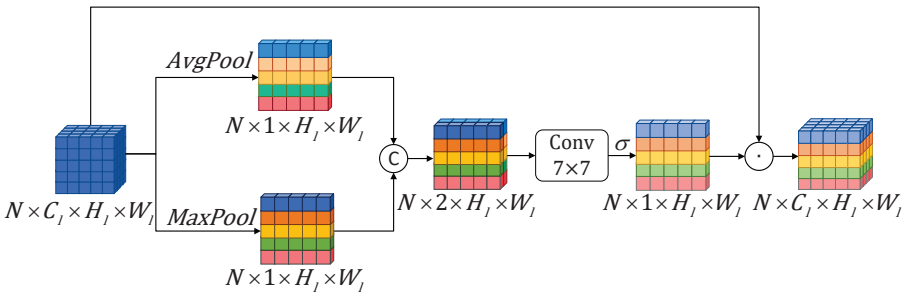


Figure 7. The low-level features, enriched with contextual information, are subjected to max pooling and average pooling along the spatial dimension to generate a spatial attention map. The values of the spatial attention map are then constrained within the range of [0, 1] using a sigmoid function. Finally, the spatial attention weights are applied to the original features through element-wise multiplication, effectively emphasizing the crucial appearance information.

To enhance the ability of CNNs to activate key sub-actions and motion tempos learning, we propose SMEN. SMEN employs ResNet50 as its backbone and consists of two important components: MgAFM and FAGM. MgAFM is designed to capture motion features of sub-actions with varying tempos and motion dynamics, and adaptively fuse sub-action

motion features with multi-granularity. FAGM leverages contextual information and salient motion frames from high-level features, to guide low-level feature learning, thus focusing on channel and spatial information which facilitates video action classification.

4. Experiments

Experiments are performed on three large action datasets: Something-Something V1, Something-Something V2 [33], and Kinetics-400 [13]. Furthermore, ablation studies are carried out to evaluate the impact of different motion granularities, and feedback attention on the performance of our proposed methods. We also compare our approach with the state of the art and perform an activation map visualization to explore the working mechanism of the proposed method.

4.1. Datasets and Experiment Setup

4.1.1. Datasets

To verify the effectiveness of the proposed method, experiments, and evaluations were conducted on three large publicly available human video action-recognition datasets.

Something-Something V1 is a richly annotated video action dataset that contains basic interactions between humans and various objects. It includes a wide range of subtle actions characterized by rapid changes over short periods, such as slight hand movements or brief interactions with objects. The dataset comprises 174 action categories and 108,499 videos, where 86,017 are used for training, 11,522 are used for validation, and 10,960 for testing.

Something-Something V2 is an expanded action datasets, based on Something-Something V1, including 174 action categories and total 220,847 videos, with 168,913 training videos, 24,777 validating videos, and 27,157 testing videos. Figure 8 illustrates sample frames from three action categories from Something-Something V2.

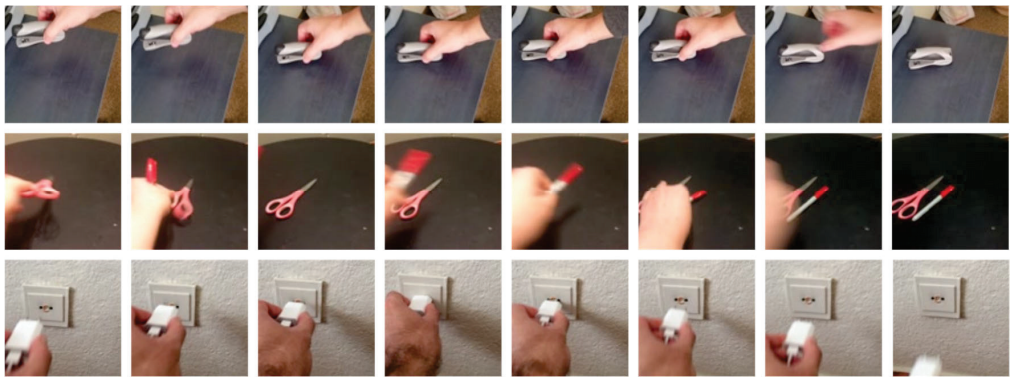


Figure 8. Video frames of Something-Something V2 datasets, from categories "Putting something on a surface" (**top**), "Putting something and something on the table" (**middle**), and "Plugging something into something but pulling it right out as you remove your hand" (**bottom**). These actions include sub-actions with subtle variances, characterized by rapid changes over short periods and brief interactions with objects.

Kinetics-400 is a dataset collected from the YouTube video containing 400 action classes, with each class featuring at least 400 clips of around 10 s. Kinetics-400 presents a significant challenge due to background clutter, complex motion tempos, and long-range or short-range temporal dependencies. Figure 9 shows sampled frames from three action categories within Kinetics-400.

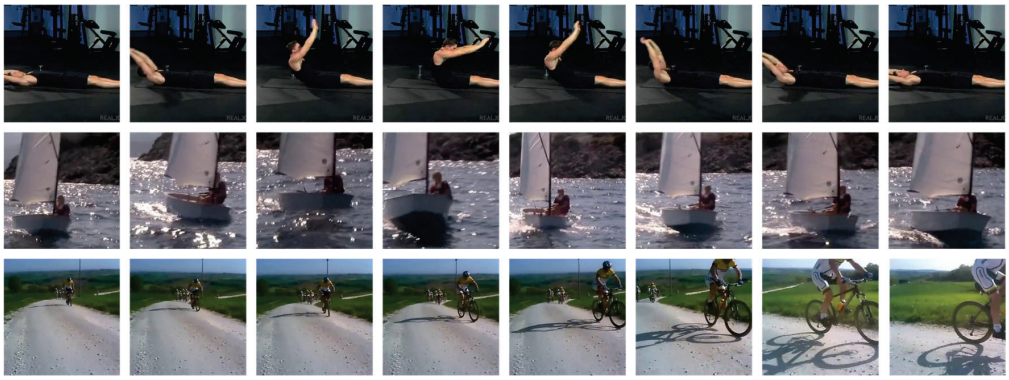


Figure 9. The sampled frames of three action categories from Kinetics-400 datasets, with categories "Situp" (**top**), "Sailing" (**middle**), and "Riding a bike" (**bottom**).

4.1.2. Experiment Setup

In the training stage, we adopt TSM as the baseline network, utilizing ResNet50 pre-trained on ImageNet [34] as the backbone. The same data augmentation strategy used in TSM is adopted. Each video is sampled with 8 or 16 frames as input. Training is performed on 8 NVIDIA GeForce RTX 3090 GPUs using distributed learning, with a per-GPU batch size of 16. The server includes an Intel Xeon CPU, 128 GB of RAM, and SSD storage for fast data loading. The experimental software includes Python 3.9, PyTorch 1.10, and CUDA 11.1, running on Ubuntu 20.04. Optimization is conducted via stochastic gradient descent (SGD) with a momentum of 0.9 and a dropout rate of 0.5. For the Something-Something V1 and V2 datasets, the initial learning rate is set to 0.02 with a weight decay of 5×10^{-4}. Training spans 60 epochs, with the learning rate reduced by a factor of 10 at epochs 30, 40, and 45. For Kinetics-400, the initial learning rate is set to 0.01 with a weight decay of 5×10^{-4}. Training extends to 100 epochs, with the learning rate decreasing by a factor of 10 at epochs 50, 70, and 90.

In the evaluation stage, the experimental software and hardware conditions remain consistent with the training stage. For the Something-Something V1 and V2 datasets, we sample a single clip with T frames from each video as input. Frames are uniformly sampled with a center crop of size 224 $\times$ 224 (center crop $\times$ 1 clip). For the Kinetics-400 dataset, 10 clips are sampled with T frames from each video. Each frame is uniformly sampled using three crops of 256 $\times$ 256 (three crops $\times$ 10 clips).

4.2. Ablation Study

Ablation study is a commonly used analysis method in the field of artificial intelligence. Its goal is to analyze the contribution of different parts of a model to its overall performance by gradually removing or adding certain components or functionalities of the model. This section presents ablation experiments and analysis on the Something-Something V1 dataset to assess the effectiveness of each module. This paper primarily uses Top-1 (%) and Top-5 (%) as model performance evaluation metrics, which are widely used in human action-recognition research [7,11,35]. Among them, the human action-recognition model classifies input samples and ranks multiple predicted results in descending order based on probability. Top-1 (%) refers to the percentage of test samples where the predicted category ranked first matches the actual category. Top-5 (%) represents the percentage of test samples where the actual category appears within the top five predicted rankings.

4.2.1. The Ablation of FAGM and MgAFM

As illustrated in Table 1, to more directly demonstrate the necessity of each module, we use ΔTop-1 (%) to represent the impact of adding our proposed modules to the baseline

model on action-recognition accuracy. Incorporating the FAGM into the baseline yields a 4.0% improvement in Top-1 accuracy. Adding the MgAFM to TSM baseline [7] results in a 5.7% increase in Top-1 accuracy, as MgAFM effectively captures sub-action motion information across varying tempos, thereby improving the model's capacity to recognize essential sub-action details. When both FAGM and MgAFM are integrated into the baseline, Top-1 accuracy rises by 6.8%, indicating that the two modules complement each other and work synergistically to boost video action-recognition performance.

Table 1. Performance comparison with different modules.

FAGM	MgAFM	Top-1 (%)	ΔTop-1 (%)
		45.6	Baseline (TSM)
✓		49.6	+4.0
	✓	51.3	+5.7
✓	✓	52.4	+6.8

4.2.2. The Impact of Motion Tempos at Different Temporal Granularities

The MgAFM improves the model's ability to model motion information by integrating sub-action motion features with different tempos into the original input features. To verify which granularities of motion features contribute to improving model performance, we conducted ablation experiments by incrementally adding motion features with different frame intervals to the model. As shown in Table 2, when only the fast-tempo motion features extracted from adjacent frames are added to the original input, the Top-1 accuracy is merely 51.5%. As the types of motion features increase, the action-recognition accuracy gradually improves. When motion features with four granularities (frame intervals 0, 1, 2, and 3) are added to the original input, the Top-1 accuracy reaches a maximum of 52.4%.

When the motion features with frame intervals of 4, 5, and 6 are further added to the original RGB features, the model's performance begins to decline. We analyzed that the correlation of larger temporal granularity cannot provide useful motion change information for subsequent representation learning. Using such motion features as input can negatively impact the model's recognition performance.

Table 2. The impact of motion features at different temporal granularities on the accuracy of video action recognition.

Branch	Temporal Granularities	Top-1 (%)
1	0	51.5
2	0, 1	51.7
3	0, 1, 2	52.3
4	0, 1, 2, 3	52.4
5	0, 1, 2, 3, 4	52.3
6	0, 1, 2, 3, 4, 5	52.1
7	0, 1, 2, 3, 4, 5, 6	51.9

4.2.3. The Impact of Feedback Iteration Times

As illustrated in Table 3, setting the times of FAGM iterations to 1 improves Top-1 accuracy by 0.5% compared to the baseline. A peak Top-1 accuracy of 52.4% is achieved with 3rd iterations. However, increasing the iterations further results in accuracy drops. We speculate that as the time of iterations grows, the computational steps increase, potentially causing a decrease in gradient values during training (similar to the "vanishing gradient" issue), which could limit the effectiveness of attentional guidance and reduce model performance.

Table 3. The impact of feedback iteration times for video action-recognition accuracy.

Iteration Times	Top-1 (%)
0	51.3
1	51.8
2	52.1
3	52.4
4	52.2
5	51.6
6	49.9
7	49.0

4.3. Comparison with the State of the Art

Utilizing ImageNet for pre-training has been shown to speed up model convergence on new datasets [36], significantly lowering computational costs compared to training from scratch. Most video action-recognition models with -50 backbones pre-trained on ImageNet. To achieve a fair comparison with existing methods, we also use ImageNet for pre-training.

4.3.1. Something-Something V1

Table 4 shows our results on the Something-Something V1 datasets. The table is divided into three parts, where the top section lists methods based on 3D CNN [9,10,37] the middle section lists methods based on 2D CNN [7,11,15–17,30,35,38,39] and the proposed SMEN-based method is presented in the bottom section. The "Params" in the table represents the number of learnable parameters in each model, mainly including the weights of the convolution kernels and the bias terms. The larger the number of parameters, the more memory the model requires.

Table 4. Performance comparison on Something-Something V1, where "-" denotes a result that is not given in the original references.

Method	Backbone	Pre-Trained	Frame × Clip × Scrops	Params	Top-1 (%)	Top-5 (%)
3DCNN:						
ECO_{En}Lite [37]	BNIncep+3D ResNet18	Kinetics-400	92 × 1 × 1	150 M	46.4	-
I3D from [9]	ResNet50	ImageNet+ Kinetics-400	32 × 2 × 1	28.0 M	41.6	72.2
NL-I3D from [9]	ResNet50	ImageNet+ Kinetics-400	32 × 2 × 1	35.3 M	44.4	76.0
NL-I3D + GCN [9]	ResNet50	ImageNet+ Kinetics-400	32 × 2 × 1	62.2 M	46.1	76.8
3D DenseNet-121 [10]	3D DenseNet-121	ImageNet	16 × 1 × 1	21.4 M	50.2	78.9
2DCNN:						
TSN [15]	BNInception	ImageNet	8 × 1 × 1	10.7 M	19.5	-
TSM [7]	ResNet50	ImageNet	8 × 1 × 1	24.3 M	45.6	74.2
TSM [7]	ResNet50	ImageNet	16 × 1 × 1	24.3 M	47.2	77.1
TANet [39]	ResNet50	ImageNet	16 × 1 × 1	25.6 M	47.6	77.7
TCM [11]	ResNet50	ImageNet	8 × 1 × 1	24.5 M	52.2	80.4
MSNet [16]	ResNet50	ImageNet	8 × 1 × 1	24.6 M	50.9	80.3
AGPN [30]	ResNet50	ImageNet	8 × 1 × 1	27.6 M	51.6	80.9
TEA [17]	ResNet50	ImageNet	16 × 1 × 1	24.5 M	51.9	80.3
MDAF [35]	ResNet50	ImageNet	8 × 1 × 1	24.5 M	49.1	78.0
MT-Net [38]	ResNet50	ImageNet	16 × 1 × 1	-	53.4	81.3
SMEN (ours)	ResNet50	ImageNet	8 × 1 × 1	26.1 M	52.4	80.6
SMEN (ours)	ResNet50	ImageNet	16 × 1 × 1	26.1 M	54.4	82.7

Compared to 3D CNN-based methods [9,37], our model has fewer parameters and achieves higher accuracy in video action-recognition tasks. Compared to 2D CNN-based methods [7,11,15–17,30,39], our SMEN achieves top-1 accuracies of 52.4% and 54.4% when using 8 or 16 video frames as input, respectively, showing improvements of 6.8% (52.4% vs. 45.6%) and 7.2% (54.4% vs. 47.2%) over the TSM baseline. Furthermore, even in comparison

with recent 2D CNN-based methods [35,38], our model continues to show competitive performance improvement for video action-recognition tasks.

Especially, on one hand, compared to feedforward attention-based methods [17,39], our model effectively exploits contextual information from high-level features through the proposed FAGM attention mechanism. The contextual semantic information is leveraged to guide the low-level feature learning significant sub-action information, leading to higher recognition accuracy. On the other hand, compared to methods that capture sub-action information [11,16,30], our model can adaptively fuse sub-actions with different motion tempos, focusing more on key sub-action information.

4.3.2. Something-Something V2

To evaluate the robustness of the proposed method, we test it on the Something-Something V2 datasets. Table 5 compares our results with 2D CNN and 3D CNN-based methods. As shown in the Table 5, when using 8 or 16 video frames as input, our SMEN reports top-1 accuracies of 63.3% and 65.0% respectively, achieving improvements of 4.5% (63.3% vs. 58.8%) and 1.6% (65.0% vs. 63.4%) over the TSM baseline. Compared to feedforward attention-based methods [12,18,40] and recent 2D CNN-based methods [30,35], the proposed SMEN outperforms them in video action-recognition tasks.

Table 5. Performance comparison on Something-Something V2, where "-" denotes a result that is not given in the original references.

Method	Backbone	Pre-Trained	Frame × Clips × Crops	Params	Top-1 (%)	Top-5 (%)
3DCNN:						
3D DenseNet-121 [10]	3D DenseNet121	ImageNet	16 × 1 × 1	21.4 M	62.9	88.0
CIDC [41]	ResNet50	ImageNet	32 × 1 × 1	87 M	56.3	83.7
RubiksNet [42]	ResNet50	ImageNet	8 × 1 × 1	-	58.8	85.6
2DCNN:						
TSN [15]	BNInception	ImageNet	8 × 1 × 1	10.7 M	33.4	-
TSM [7]	ResNet50	ImageNet	8 × 1 × 1	24.3 M	58.8	85.4
TSM [7]	ResNet50	ImageNet	16 × 1 × 1	24.3 M	63.4	88.5
STM [12]	ResNet50	ImageNet	16 × 1 × 1	35.3 M	64.2	89.8
MSNet [16]	ResNet50	ImageNet	8 × 1 × 1	24.6 M	63.0	88.4
AGPN [30]	ResNet50	ImageNet	8 × 1 × 1	27.6 M	63.1	88.6
ACTION-Net [18]	ResNet50	ImageNet	16 × 1 × 1	28.1 M	64.0	89.3
GSFNet [43]	ResNet50	ImageNet	8 × 1 × 1	24.0 M	62.1	-
MDAF [35]	ResNet50	ImageNet	8 × 1 × 1	24.5 M	62.3	87.8
TEINet [40]	ResNet50	ImageNet	16 × 1 × 1	30.4 M	61.3	-
SMEN (ours)	ResNet50	ImageNet	8 × 1 × 1	26.1 M	63.3	88.8
SMEN (ours)	ResNet50	ImageNet	16 × 1 × 1	26.1 M	65.0	89.5

4.3.3. Kinetics-400

Kinetics-400 is a larger dataset with diverse appearances and backgrounds, compared to Something-Something V1 and V2. To assess the proposed method's effectiveness in modeling appearance features and interactions, we take performance comparisons on the Kinetics-400 dataset, as shown in Table 6.

Compared to the 3D CNN-based method SlowFast [29], our SMEN model also obtains remarkable results in the Top-5 accuracy (93.3% vs. 92.1%) and its Top-1 accuracy is higher (76.9% vs. 75.6%) since our model can acquire more granular motion features. Compared to TSM, our method achieves a Top-1 accuracy of 76.9%, an improvement of 2.8%. For other 2D CNN-based methods, SMEN provides further performance gains with only a slight increase in model parameters, demonstrating its efficiency and effectiveness in appearance-based action datasets. Furthermore, our SMEN model obtains competitive performances on the Kinetics-400 datasets. Specifically, its Top-1 and Top-5 accuracy outperforms the state-of-the-art 2D CNN-based method MDAF [35] by 0.7% (76.9% vs. 76.2%) and

0.8% (92.8% vs. 92.0%), respectively. All these results highlight that our SMEN is more effective and accurate than the feedforward network in modeling the sub-action motion tempos and representation learning for video action-recognition classification.

Table 6. Performance comparison with the state of the art on the Kinetics-400 dataset. The best results are highlighted. "N/A" and "-" denote a result that is not given in the original references.

Method	Backbone	Pre-Trained	Frame × Clip × Crops	Params	Top-1 (%)	Top-5 (%)
3DCNN:						
ECO [37]	BNIncep+3D-ResNet-18	Scratch	92 × 1 × 1	150 M	70.0	89.4
I3D [9]	Inception	ImageNet	64 × N/A × N/A	28.0 M	72.1	90.3
SlowFast [29]	ResNet50	Scratch	(4 + 32) × 1 × 3	-	75.6	92.1
2DCNN:						
TSN [15]	BNInception	ImageNet	25 × 1 × 10	10.7 M	72.5	90.2
TSM [7]	ResNet50	ImageNet	8 × 10 × 3	24.3 M	74.1	91.2
TCM [11]	ResNet50	ImageNet	8 × 10 × 3	24.5M	76.1	92.3
STM [12]	ResNet50	ImageNet	16 × 10 × 3	24.0 M	73.7	91.6
MSNet [16]	ResNet50	ImageNet	8 × 10 × 3	24.5 M	75.0	-
GSFNet [43]	ResNet50	ImageNet	8 × 10 × 3	24.0M	74.8	-
MDAF [35]	ResNet50	ImageNet	8 × 10 × 3	24.5 M	76.2	92.0
TEINet [40]	ResNet50	ImageNet	8 × 10 × 3	30.4 M	74.9	91.9
MVFNet [44]	ResNet50	ImageNet	8 × 10 × 3	24.3 M	76.0	92.4
GTDNet [45]	ResNet50	ImageNet	8 × 10 × 3	33.9 M	75.5	92.5
GTDNet [45]	ResNet50	ImageNet	16 × 10 × 3	33.9 M	76.8	92.3
SMEN (ours)	ResNet50	ImageNet	8 × 10 × 3	26.1 M	76.9	92.8
SMEN (ours)	ResNet50	ImageNet	16 × 10 × 3	26.1 M	77.9	93.3

4.4. Visualization

To further understand the working mechanism of the proposed SMEN, Grad-CAM [46] is applied to visualize activation maps and the most discriminative feature locations on the Kinetics-400 dataset. As presented in Figure 10, the feature visualization results of TSM [7] and our SMEN are illustrated for the actions "Triple jump", "Long jump", and "High jump". In this activation map, the redder the area, the stronger the model's attention to that region. The input frames are sampled at equal intervals and cropped to 256 × 256.

In Figure 10, for the first action category "Triple jump", it can be observed that the TSM method focuses more on the "take-off line" rather than the motion tempos of human motion. In contrast, our method keeps its attention on human motion dynamics and tempos. Our method effectively focuses attention on the temporal information of the sub-actions "Hop", "Skip", and "Jump", which are more critical for classifying human actions. On the other hand, for the second action category "Long jump", it can be observed that the TSM method pays more attention to the "Take-off line" from frames T0 to T7, rather than to the motion tempos in human motion. In contrast to TSM, our method effectively captures part of the information related to the key sub-actions "Action in the air" and the fast-paced action "Take-off". For the third action category "High jump", from frames T3 to T7, it can be observed that TSM places more attention on the "High jump bar" rather than the various motion tempos of action. Our method focuses in a balanced manner on both action motion tempos and the action-specific contextual information "High jump bar", which can be observed from the brighter blue areas compared to the background in the class activation map.

Besides, we employed Grad-CAM [46] to visualize class activation maps on the Something-Something V2 dataset. As presented in Figure 11, the feature visualization results of TSM [7] and our SMEN are illustrated on the Something-Something V2 dataset for the action categories "Covering something with something" and "Holding something over something". In these visualizations, we sampled eight frames from each video as input. Since the Something-Something V2 dataset primarily consists of short videos with durations of about 3 seconds, the tempo of sub-action motions within the videos changes rapidly.

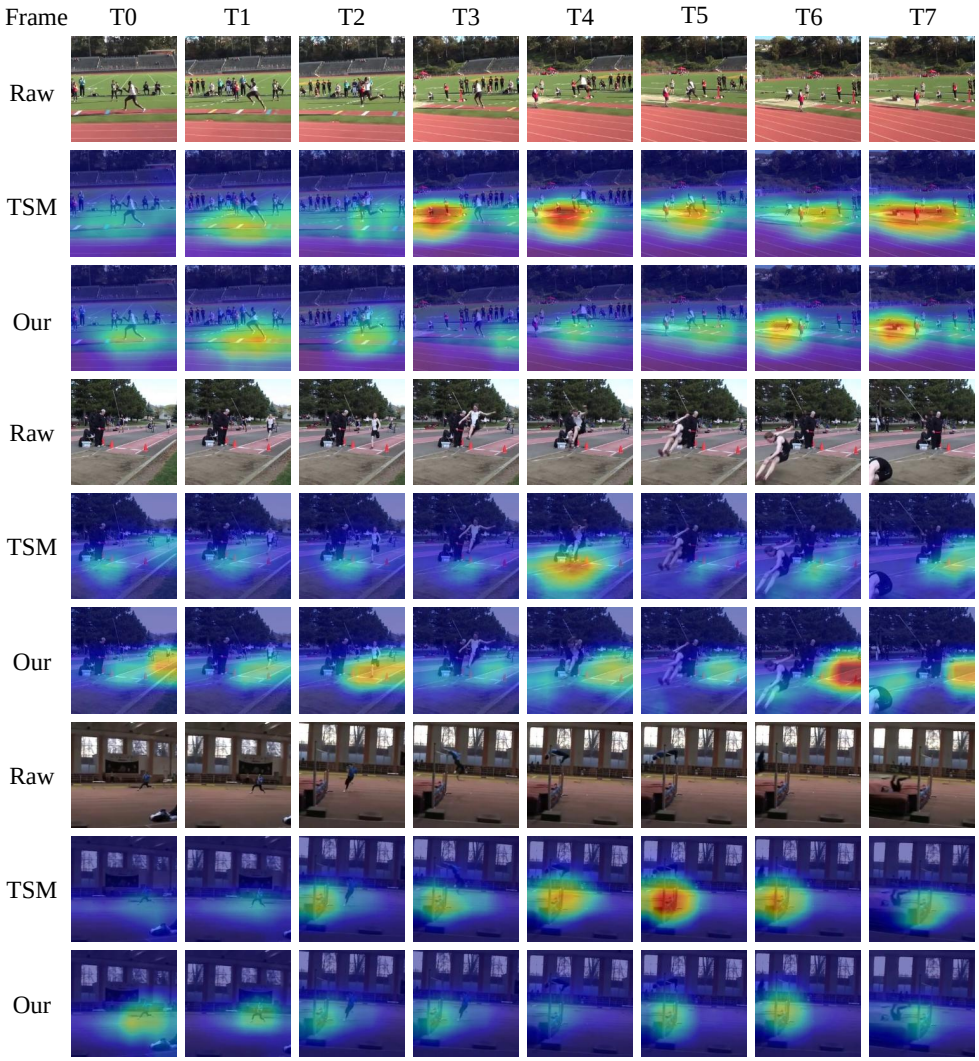


Figure 10. On the Kinetics-400 dataset, we use Grad-CAM [46] to visualize the activation maps for the actions "Triple jump", "Long jump" and "High jump", with deep features extracted using TSM and the proposed SMEN method, respectively.

The visualization results of Figure 11 show that, for the first action category, compared to TSM, the proposed SMEN captures the motion tempos and the temporal changes of sub-action "Picked up" more promptly. Furthermore, for the subsequent sub-action "Covering" the cup with a towel, our model captures the entire "Covering" tempo more accurately. For the second action category, SMEN outperforms TSM by more effectively capturing the sub-actions "Placing one object on another" and "Removing the arm" while focusing more attention on these dynamic sub-actions. Our method avoids the allocation of resources to static information that has less impact on video action classification.

Ablation studies and results confirm the effectiveness of modules in the proposed SMEN method. The performance comparisons demonstrate the effectiveness of the proposed SMEN, due to the model's capacity to capture and represent key sub-action information with different motion tempos more effectively. Additionally, the visualization analysis offers a more intuitive insight into the working mechanism of SMEN.

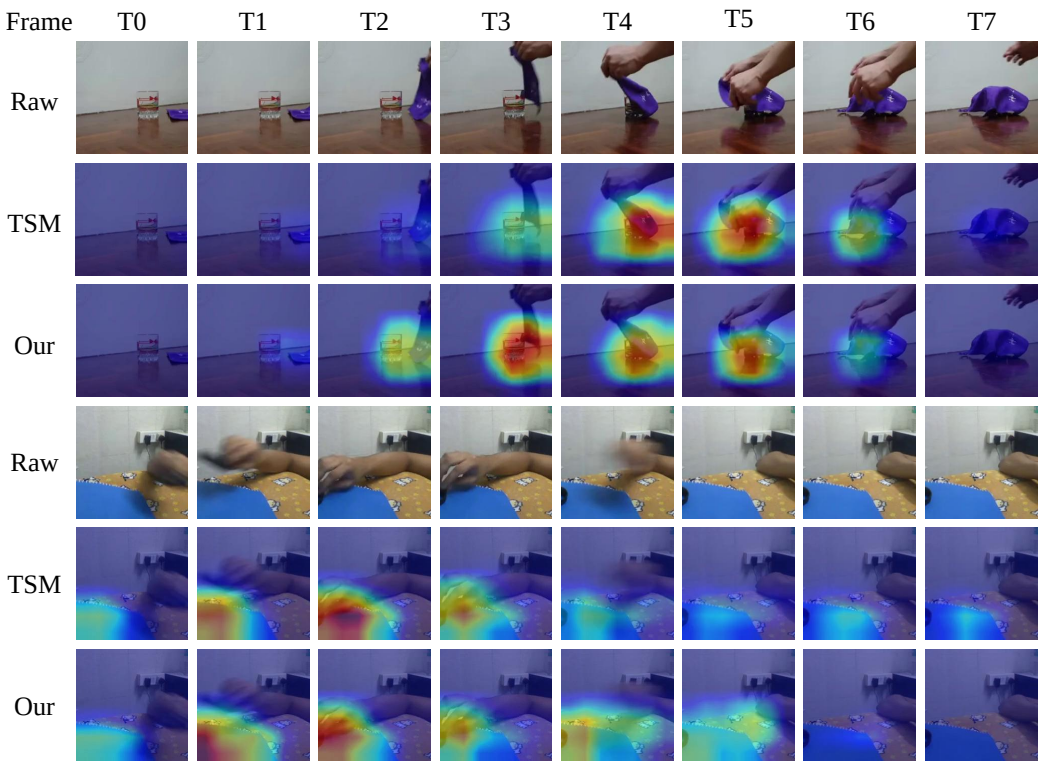


Figure 11. On the Something-Something V2 dataset, we use Grad-CAM [46] to visualize the activation maps for the actions "Covering something with something" and "Holding something over something", with deep features extracted using TSM and the proposed SME Nm ethod, respectively.

5. Conclusions

In this paper, to explore the potential of the human action-recognition model for capturing and representing vital sub-action information with different motion tempos, we propose a Sub-action Motion information Enhancement Network (SMEN) based on motion tempo learning and feedback attention, comprising two key components: the Multi-Granularity Adaptive Fusion Module (MgAFM) and the Feedback Attention-Guided Module (FAGM). The MgAFM captures the intrinsic information of sub-actions with motion tempos at different granularities and adaptively fuses it with the original RGB features. The FAGM leverages global contextual information to guide low-level feature learning with a feedback mechanism, focusing on vital spatio-temporal and channel-wise information. Extensive experiments on three public datasets, ablation studies, and visualization analyses demonstrate that the proposed method outperforms compared methods, especially for videos containing sub-actions with different motion tempos.

The proposed SMEN focuses less on the contextual background information of human actions and the object categories involved in interactive actions, which is useful for complex human activities. The future plan is to optimize motion-extraction methods to better capture the contextual background of actions or explore the generalization of SMEN using heatmaps [47], which are converted from human skeleton data for skeleton-based human action recognition.

Author Contributions: Conceptualization, C.L. and Y.L.; Data curation, Y.L.; Formal analysis, Y.L.; Funding acquisition, C.L.; Investigation, Y.L., S.J. and P.Z.; Methodology, Y.L. and C.L.; Project administration, C.L.; Resources, C.L.; Software, Y.L. and S.J.; Supervision, C.L.; Validation, C.L. and Y.L.; Visualization, Y.L., S.J. and P.Z.; Writing—original draft preparation, Y.L.; Writing—review and editing, C.L., S.J. and P.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was founded in part by the National Natural Science Foundation of China under Grant 62176086, and in part by the Science and Technology Development of Henan Province under Grant 242102211055 and 242102210032, the Soft Science Research Program of Henan Province under Grant 252400410618.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The Something-Something V1 dataset can be downloaded from https://aistudio.baidu.com/datasetdetail/44373 (accessed on 20 February 2025). The Something-Something V2 dataset can be downloaded from https://www.qualcomm.com/developer/software/something-something-v-2-dataset (accessed on 20 February 2025). The Kinetics-400 dataset can be downloaded from https://github.com/cvdfoundation/kinetics-dataset (accessed on 20 February 2025).

Acknowledgments: The authors sincerely thank Ying Su for her constructive comments, which substantially improved the overall quality of the manuscript.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Karim, M.; Khalid, S.; Aleryani, A.; Khan, J.; Ullah, I.; Ali, Z. Human Action Recognition Systems: A Review of the Trends and State-of-the-Art. *IEEE Access* **2024**, *12*, 36372–36390. [CrossRef]
2. Ji, Y.; Yang, Y.; Shen, F.; Shen, H.T.; Li, X. A Survey of Human Action Analysis in HRI Applications. *IEEE Trans. Circuits Syst. Video Technol.* **2019**, *30*, 2114–2128. [CrossRef]
3. Sun, Z.; Ke, Q.; Rahmani, H.; Bennamoun, M.; Wang, G.; Liu, J. Human Action Recognition From Various Data Modalities: A Review. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *45*, 3200–3225. [CrossRef]
4. Wang, L.; Huynh, D.Q.; Koniusz, P. A Comparative Review of Recent Kinect-Based Action Recognition Algorithms. *IEEE Trans. Image Process.* **2019**, *29*, 15–28. [CrossRef]
5. Liu, S.; Ma, X. Attention-Driven Appearance-Motion Fusion Network for Action Recognition. *IEEE Trans. Multimed.* **2022**, *25*, 2573–2584. [CrossRef]
6. Jaiswal, S.; Fernando, B.; Tan, C. TDAM: Top-Down Attention Module for Contextually Guided Feature Selection in CNNs. In Proceedings of the European Conference on Computer Vision (ECCV), Tel Aviv, Israel, 23–27 October 2022; pp. 259–276.
7. Lin, J.; Gan, C.; Han, S. TSM: Temporal Shift Module for Efficient Video Understanding. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 7083–7093.
8. Du, T.; Bourdev, L.; Fergus, R.; Torresani, L.; Paluri, M. Learning Spatiotemporal Features with 3D Convolutional Networks. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Santiago, Chile, 11–18 December 2015; pp. 4489–4497.
9. Carreira, J.; Zisserman, A. Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 6299–6308.
10. Zhou, Y.; Sun, X.; Luo, C.; Zha, Z.-J.; Zeng, W. Spatiotemporal Fusion in 3D CNNs: A Probabilistic View. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 14–19 June 2020; pp. 9826–9835.
11. Liu, Y.; Yuan, J.; Tu, Z. Motion-Driven Visual Tempo Learning for Video-Based Action Recognition. *IEEE Trans. Image Process.* **2022**, *31*, 4104–4116. [CrossRef] [PubMed]
12. Jiang, B.; Wang, M.; Gan, W.; Wu, W.; Yan, J. STM: SpatioTemporal and Motion Encoding for Action Recognition. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; pp. 2000–2009.
13. Kay, W.; Carreira, J.; Simonyan, K.; Zhang, B.; Hillier, C.; Vijayanarasimhan, S.; Viola, F.; Green, T.; Back, T.; Natsevc, P.; et al. The kinetics human action video dataset. *arXiv* **2017**, arXiv:1705.06950.
14. Simonyan, K.; Zisserman, A. Two-stream convolutional networks for action recognition in videos. *Adv. Neural Inf. Process. Syst.* **2014**, *1*, 568–576.
15. Wang, L.; Xiong, Y.; Wang, Z.; Qiao, Y.; Lin, D.; Tang, X.; Van Gool, L. Temporal Segment Networks: Towards Good Practices for Deep Action Recognition. In Proceedings of the European Conference on Computer Vision (ECCV), Amsterdam, The Netherlands, 8–16 September 2016; pp. 20–36.
16. Kwon, H.; Kim, M.; Kwak, S.; Cho, M. MotionSqueeze: Neural Motion Feature Learning for Video Understanding. In Proceedings of the European Conference on Computer Vision (ECCV), Glasgow, UK, 23–28 August 2020; pp. 345–362.

17. Li, Y.; Ji, B.; Shi, X.; Zhang, J.; Kang, B.; Wang, L. TEA: Temporal Excitation and Aggregation for Action Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 14–19 June 2020; pp. 909–918.
18. Wang, Z.; She, Q.; Smolic, A. ACTION-Net: Multipath Excitation for Action Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 19–25 June 2021; pp. 13214–13223.
19. Ramachandran, P.; Parmar, N.; Vaswani, A.; Bello, I.; Levskaya, A.; Shlens, J. Stand-alone self-attention in vision models. In Proceedings of the Advances in Neural Information Processing Systems 32 (NeurIPS 2019), Vancouver, BC, Canada, 8–14 December 2019; Volume 32.
20. Wang, X.; Girshick, R.; Gupta, A.; He, K. Non-local Neural Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 7794–7803.
21. Wang, Z.; Liu, Z.; Li, G.; Wang, Y.; Zhang, T.; Xu, L.; Wang, J. Spatio-Temporal Self-Attention Network for Video Saliency Prediction. *IEEE Trans. Multimed.* **2021**, *25*, 1161–1174. [CrossRef]
22. Nasaoui, H.; Bellamine, I.; Silkan, H. Improving Human Action Recognition in Videos with Two-Stream and Self-Attention Module. In Proceedings of the 7th IEEE Congress on Information Science and Technology, Agadir, Morocco, 16–22 December 2023; pp. 215–220.
23. Gilbert, C.D.; Li, W. Top-down influences on visual processing. *Nat. Rev. Neurosci.* **2013**, *14*, 350–363. [CrossRef] [PubMed]
24. Kok, P.; Bains, L.J.; van Mourik, T.; Norris, D.G.; de lange, F.P. Selective Activation of the Deep Layers of the Human Primary Visual Cortex by Top-Down Feedback. *Curr. Biol.* **2016**, *26*, 371–376. [CrossRef] [PubMed]
25. Kreiman, G.; Serre, T. Beyond the feedforward sweep: Feedback computations in the visual cortex. *Ann. N. Y. Acad. Sci.* **2020**, *1464*, 222–241. [CrossRef] [PubMed]
26. Hochstein, S.; Ahissar, M. View from the top: Hierarchies and reverse hierarchies in the visual system. *Neuron* **2002**, *36*, 791–804. [CrossRef] [PubMed]
27. Qiao S, Chen L.C.; Yuille A. Detectors: Detecting objects with recursive feature pyramid and switchable atrous convolution. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 19–25 June 2021; pp. 10213–10224.
28. Qiu, Z.; Yao, T.; Mei, T. Learning Spatio-Temporal Representation with Pseudo-3D Residual Networks. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 5534–5542.
29. Feichtenhofer, C.; Fan, H.; Malik, J.; He, K. SlowFast Networks for Video Recognition. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; pp. 6202–6211.
30. Chen, Y.; Ge, H.; Liu, Y.; Cai, X.; Sun, L. AGPN: Action Granularity Pyramid Network for Video Action Recognition. *IEEE Trans. Circuits Syst. Video Technol.* **2023**, *33*, 3912–3923. [CrossRef]
31. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
32. Lee, J.; Kim, D.; Ponce, J.; Ham, B. SFNet: Learning Object-aware Semantic Correspondence. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019; pp. 2278–2287.
33. Goyal, R.; Kahou, S.E.; Michalski, V.; Materzynska, J.; Westphal, S.; Kim, H.; Haenel, V.; Fruend, I.; Yianilos, P.; Mueller-Freitag, M.; et al. The "something something" video database for learning and evaluating visual common sense. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 5842–5850.
34. Jia, D.; Wei, D.; Socher, R.; Li-Jia, L.; Kai, L.; Li, F.-F. ImageNet: A large-scale hierarchical image database. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Miami, FL, USA, 20–25 June 2009; pp. 248–255.
35. Wang, B.; Chang, F.; Liu, C.; Wang, W.; Ma, R. An efficient motion visual learning method for video action recognition. *Expert Syst. Appl.* **2024**, *255*, 124596. [CrossRef]
36. He, K.; Girshick, R.; Dollár, P. Rethinking imagenet pre-training. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; pp. 4918–4927.
37. Zolfaghari, M.; Singh, K.; Brox, T. ECO: Efficient Convolutional Network for Online Video Understanding. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 695–712.
38. Sheng, X.; Li, K.; Shen, Z.; Xiao, G. A Progressive Difference Method for Capturing Visual Tempos on Action Recognition. *IEEE Trans. Circuits Syst. Video Technol.* **2022**, *33*, 977–987. [CrossRef]
39. Liu, Z.; Wang, L.; Wu, W.; Qian, C.; Lu, T. TAM: Temporal Adaptive Module for Video Recognition. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 11–17 October 2021; pp. 13688–13698.
40. Liu, Z.; Luo, D.; Wang, Y.; Wang, L.; Tai, Y.; Wang, C.; Li, J.; Huang, F.; Lu, T. TEINet: Towards an Efficient Architecture for Video Recognition. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 11669–11676.
41. Li, X.; Shuai, B.; Tighe, J. Directional Temporal Modeling for Action Recognition. In Proceedings of the European Conference on Computer Vision (ECCV), Glasgow, UK, 23–28 August 2020; pp. 275–291.

42. Fan, L.; Buch, S.; Wang, G.; Cao, R.; Zhu, Y.; Niebles, J.C.; Li, F.-F. RubiksNet: Learnable 3D-Shift for Efficient Video Action Recognition. In Proceedings of the European Conference on Computer Vision (ECCV), Glasgow, UK, 23–28 August 2020; pp. 505–521.
43. Sudhakaran, S.; Escalera, S.; Lanz, O. Gate-Shift-Fuse for Video Action Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2023**, *45*, 10913–10928. [CrossRef] [PubMed]
44. Wu, W.; He, D.; Lin, T.; Li, F.; Gan, C.; Ding, E. Mvfnet: Multi-view fusion network for efficient video recognition. In Proceedings of the AAAI Conference on Artificial Intelligence, Vancouver, BC, Canada, 2–9 February 2021; Volume 35, pp. 2943–2951.
45. Xie, Z.; Chen, J.; Wu, K.; Guo, D.; Hong, R. Global Temporal Difference Network for Action Recognition. *IEEE Trans. Multimed.* **2022**, *25*, 7594–7606. [CrossRef]
46. Selvaraju, R.R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; Batra, D. Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 618–626.
47. Duan, H.; Zhao, Y.; Chen, K.; Lin, D.; Dai, B. Revisiting skeleton-based action recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 2969–2978.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.