Noname manuscript No.
(will be inserted by the editor)

An Efficient Attention Module for 3D Convolutional Neural Networks in Action Recognition

Guanghao Jiang [1] · Xiaoyan Jiang [1] · Zhijun Fang [1] · Shanshan Chen [1]

Abstract Due to illumination changes, varying postures, and occlusion, accurately recognizing actions in videos is still a challenging task. A three-dimensional convolutional neural network (3D CNN), which can simultaneously extract spatio-temporal features from sequences, is one of the mainstream models for action recognition. However, most of the existing 3D CNN models ignore the importance of individual frames and spatial regions when recognizing actions. To address this problem, we propose an efficient attention module (EAM) that contains two sub-modules, that is, a spatial efficient attention module (EAM-S) and a temporal efficient attention module (EAM-T). Specifically , without dimensionality reduction, EAM-S concentrates on mining category-based correlation by local cross-channel interaction and assigns high weights to important image regions, while EAM-T estimates the importance score of different frames by cross-frame interaction between each frame and its neighbors. The proposed EAM module is lightweight yet effective, and it can be easily embedded into 3D CNN-based action recognition models. Extensive experiments on the challenging HMDB-51 and UCF-101 datasets showed that our proposed module achieves state-of-the-art performance and can significantly improve the recognition accuracy of 3D CNN-based action recognition methods.

Keywords Effective attention module · 3D CNN · Deep learning · Action recognition

1 Introduction

Action recognition is an important component in video understanding. It has been applied to various scenarios, for example, intelligent video surveillance, video retrieval, human-computer interaction, and autonomous driving. The core problem of action recognition, which can identify interesting actions in time and space, is how to effectively extract semantic and temporal information. In unconstrained environments, due to illumination changes, varying postures, and occlusion, it is challenging to accurately recognize actions in videos . Before the development of deep learning, traditional methods based on hand-crafted features, such as 3D HOG [1], HOF [2], MBH [3], dense trajectory (DT) [4], and improved dense trajectory (IDT) [5], were widely used in action recognition tasks. In the past few years, the powerful feature learning ability of convolutional neural network (CNN) has also demonstrated remarkable results in many computer vision tasks, such as image classification [6], object detection [7], and action recognition [8]. Recent mainstream CNN-based action recognition models include two-stream CNN [9–12], long short-term memory (LSTM) [13, 14], and 3D CNN [15–22].

Two-stream CNN contains two independent two-dimensional (2D)CNNs with unshared parameters, namely, a spatial network and a temporal network. The spatial network uses RGB frames as input to extract appearance features, while the temporal network uses optical flow as input to process motion information. Although two-stream CNN-based methods obtain high action recognition performance, extracting the optical flow from images in advance is usually computationally expensive. To avoid the high computation cost, researchers proposed LSTM-based action recognition methods, in which the spatio-temporal features are mod-

✉ Xiaoyan Jiang
xiaoyan.jiang@sues.edu.cn

[1] School of Electronic and Electrical Engineering, Shanghai University of Engineering Science, Shanghai, China

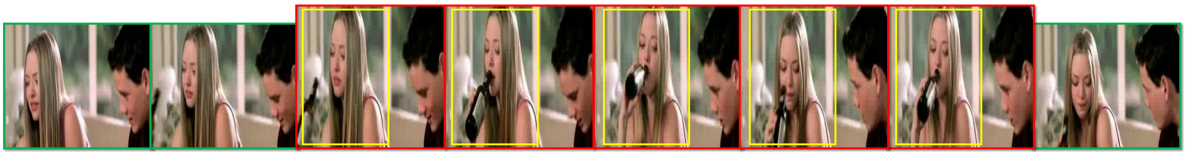


Fig. 1: An example of a ”Drink” sequence. Video frames with red rectangles (especially the yellow region in video frames colored in red) are more relevant to the action category than the video frames with green rectangles.

eled by an LSTM network. However, such methods only model high-level features from the top convolution layers while useful low-level information from earlier convolution layers is lost. Besides, three-dimensional (3D) CNN-based action recognition methods can directly extract spatio-temporal features from consecutive video frames simultaneously, which enables both low-level and high-level temporal modeling. For the past few years, the primary reason for the slow development of 3D CNN-based action recognition methods is the lack of relatively large-scale datasets used to optimize the immense number of parameters in 3D CNN. Recently, with the publicity of the Kinetics dataset [16] and the development of computation ability, the performance of 3D CNN-based action recognition approaches has been greatly promoted.

However, the feature representations learned by most 3D CNN-based methods are still not discriminative for action recognition, for example, there is no significant attention mechanism focusing on extracting spatial regions or video frames related to the action to be recognized. An example of a ”Drink” sequence from the HMDB-51 dataset [23] is shown in Fig. 1. There are two people in the video and the lady’s action is “Drink.” Frames with red rectangles and image regions drawn with yellow rectangles are more discriminative for the semantic representation of ”Drink.” In contrast, the appearance characteristics of the green-colored frames indicate a weak correlation with ”Drink” but show a strong correlation with ”Talk” and ”Laugh.” These frames have no positive influence on distinguishing between the actions of ”Drink,”, ”Talk,” or ”Laugh.” Rather, they have negative effects on feature extraction for action recognition and reduce performance.

To solve the above-mentioned problems and inspired by ECA-Net [24], we propose an EAM module, a spatial efficient attention module (EAM-S) and a temporal efficient attention module (EAM-T), for 3D CNN-based action recognition approaches. In practice, many actions generally show similar features that confuse the action recognition systems, and discriminative features help distinguish between actions. In this paper, we focus on mining the distinct importance of different frames and spatial regions in the image on recognizing actions. In the spatial dimension, we further consider the difference between various channels in the 3D CNN for spatial appearance representation to develop spatial attention at the channel-level. Previous works, [25] have applied channel dimensionality reduction as well as global cross-channel interaction strategy followed by two fully convolutional layers to compute channel attention. In contrast, to learn the correlation score of each channel containing a certain action feature, EAM-S captures local cross-channel interaction by considering each channel and its neighbors. The goal is to highlight the spatial regions relevant to a certain action class, while suppressing the irrelevant ones. Similarly, in the temporal dimension, without dimensionality reduction, our proposed EAM-T captures the local cross-frame interaction by considering each frame and its neighbors to estimate the importance score of different frames. The aim is to focus on the keyframes containing relevant information to a certain action category and reduce the recognition performance interference of unrelated frames.

The main contribution of this paper is design of a lightweight yet effective attention module. The EAM module can be easily implemented and embedded into 3D CNN-based action recognition models and trained end-to-end. Without dimensionality reduction, the EAM module captures spatial and temporal attention by local cross-channel and cross-frame interaction to assign high importance scores to spatial regions and keyframes that are relevant to the action category. Our proposed approach achieved state-of-the-art performance on two standard datasets, HMDB-51[23] and UCF-101[26].

The remainder of this paper is organized as follows: We first review related work in Section 2. Afterward, we introduce the proposed approach in detail in Section 3. Experimental results and analysis are given in Section 4. Finally, we conclude our work in Section 5.

2 Related work

Feature extraction. Traditional action recognition methods apply hand-crafted features to represent static and motion information in videos. Wang et al. [4] proposed the dense trajectory (DT) method, which densely samples feature points in each frame and tracks them in videos based on optical flow. Multiple descriptors are computed along the trajectories of the feature points to capture the shape, appearance, and motion information. However, in typical video scenarios, the camera motion generates many irrelevant trajectories in the background. Wang et al. [5] proposed IDT to eliminate the effect of background movement, which forces the model to focus on human motion description. Improved dense trajectories perform the best before deep learning is adopted for action recognition.

Due to the great success of CNN in computer vision, researchers have applied deep learning methods to action recognition tasks. Simonyan et al. [9] first proposed a two-stream CNN architecture with spatial and temporal networks that extract appearance features and motion features. However, it usually concentrates on appearances and short-term motions and lacks long-range temporal modeling ability. A temporal segment network (TSN) [10] extracts short snippets over a long video sequence with a sparse sampling scheme, and then the class scores of different snippets are generated through the two-stream CNN. Finally, the classification results are obtained by fusing the class scores of each stream. The TSN provides an effective and efficient way to capture long-term temporal structures. Despite the good performance for two-stream CNN-based action recognition, it is necessary to extract optical flow from the image in advance, which is usually computationally intensive. Furthermore, the training of two independent 2D CNNs is separate, which is time-consuming and cannot achieve end-to-end training. To address this, LSTM was proposed and has been successfully employed to model spatio-temporal relationships for action recognition. Ng et al. [13] and Donahue et al. [14] leveraged an LSTM network to aggregate frame-level information of a 2D CNN from the top convolution layers and discover long-range temporal relationships for learning spatio-temporal features. However, such methods lose useful low-level information from earlier convolution layers.

Another type of method tries to learn spatio-temporal features from RGB frames directly with a 3D CNN. Tran et al. [15] first proposed a model to learn spatio-temporal features using a deep 3D CNN, namely, C3D. To capture spatio-temporal features and model motion features with another flow stream, Carreira et al. [16] explored inflating all the 2D convolutional filters in the InceptionV1 model [27] into 3D convolutional filters. However, using small-scale datasets optimizes the tremendous parameters in a 3D CNN, which leads to overfitting of the model. Hara et al. [17] proposed a 3D residual network (3D ResNet) by inflating all the 2D convolutional filters in the ResNet [28] into 3D convolutional filters to capture spatio-temporal features. Moreover, they empirically demonstrated that using a very deep 3D CNN trained on a large-scale Kinetics dataset [16] retraces the successful history of the 2D CNN and ImageNet. To reduce the number of parameters of a 3D CNN, Qiu et al.[19] proposed another solution for 3D CNNs based on ResNet through simulating 3 × 3 × 3 convolutions with 1 × 3 × 3 convolutional filters on a spatial domain plus 3 × 1 × 1 convolutions to construct temporal connections on adjacent feature maps in time, namely, a pseudo-3D network (P3D). Here, R(2+1)D [20] explicitly factorizes the 3D convolutional filters into separate 2D spatial convolution and 1D temporal convolution to reduce the cost of 3D CNN. Besides, MiCT-Net [21] integrates 2D CNN with the 3D CNN to generate deeper and more informative feature maps while reducing the complexity of each round of spatio-temporal fusion by using the cross-domain residual connection. Another way to save computational costs was devised by Tran et al. [22], where a 3D channel-separated network (CSN) is used in which all the convolutional operations are separated into either pointwise 1×1×1 or depth-wise 3×3×3 convolutions.

Some researchers have proposed some interesting fusion methods. Feichtenhofer et al. [11] studied fusion strategies in the middle of the two streams to fuse spatial and temporal cues at several levels of granularity in feature abstraction, with spatial as well as temporal integration. Lin et al. [12] introduced an asynchronous fusion network to fuse information at different time points. Imran et al. [23] proposed a three-stream architecture consisting of RGB and inertial and skeleton streams for action recognition. A 1D CNN is used for inertial sensor gyroscope data, a 2D CNN is used for stacked dense flow difference image classification, and a bidirectional gated recurrent unit (BiGRU) based recurrent neural network (RNN) is used for skeletal classification. In the end, the outputs of all the streams are combined by late fusion to predict the final class label. Wei et al. [24] proposed fusion strategies to combine video images with simultaneously captured inertial signals using one 3D CNN for RGB video and one 2D CNN for inertial signal images for action recognition.

Attention mechanism. In recent years, attention mechanisms have received increasing attention in different computer vision tasks, such as image classification [25, 29, 30], visual tracking [31, 32], person re-identification [33–36], semantic segmentation [37–40], and action recognition[41–47]. Hu et al. [25] proposed a squeeze-and-excitation network (SE-Net) based on channel attention. It is composed of pooling and two fully convolutional layers, employs squeeze and excitation operations, and adopts the strategies of channel dimensionality reduction as well as global cross channel interaction to accomplish the function of calculating channel attention. Besides, GE-Net [29] incorporates context throughout the architecture of a deep network, and it uses depth-wise convolution to calculate spatial attention by gather-excite operators. Inspired by SE-Net, Woo et al. [30] designed the convolution block attention module (CBAM), which integrates spatial and channel attention modules to refine convolutional features independently in the spatial and channel dimensions. Gao et al. [31] proposed a Siamese lightweight hourglass network with a cross-attention module to selectively highlight meaningful information and boost the representation power of feature maps in visual tracking. Another study [32] used a hierarchical attention module to leverage both inter- and intra-frame attention at each convolutional layer to effectively highlight informative representations and suppress redundancy in visual tracking. Wang et al. [41] first proposed a non-local module to capture long-range dependencies directly by computing the correlation matrix between each spatial point in the feature map. However, such a method requires extensive computation, thus being very inefficient. Consequently, CC-Net [39] harvests capturing long-range contextual information in the horizontal and vertical directions through a novel criss-cross attention module while reducing FLOPs by about 85% of the non-local module in computing long-range dependencies. Similarly, Zhu et al. [40] devised an asymmetric non-local module that can dramatically improve the efficiency and decrease the memory consumption of the non-local module without sacrificing performance. Zhang et al. [33] proposed an effective relation-aware global attention (RGA) module capturing the global structural information for better attention learning. Moreover, Li et al. [37] introduced the expectation-maximization attention network (EMA) that computes an attention map by iteratively executing the EM algorithm from context information. Du et al. [45] proposed an effective interaction-aware self-attention model inspired by PCA to learn attention maps. Based on attention clusters, Long et al. [43] proposed a local feature integration framework that generates an effective global representation by aggregating local features. However, the complexity of the above attention module is relatively high. The ECA-Net [24] model has shown that avoiding dimensionality reduction and controlling the kernel size of 1D convolution to achieve local cross-channel interaction are effective for learning channel attention.

3 The proposed approach

In practice, many video actions show similar features, which confuses the action recognition system. Extracting discriminative features is essential to distinguishing actions, especially ambiguous ones. Thus, to mine image regions related to certain action category and keyframes containing action-related information in sequences, we propose an EAM module with spatial and temporal attention.

3.1 The architecture of EAM

As shown in Fig. 2, composed of an EAM-S and EAM-T module, EAM jointly learns attention weights for different channels in the spatial dimension and attention weights of different frames in the temporal dimension. The 4D cost volume $V \in R^{H\times W\times T\times C}$ is passed into these two modules in sequence. EAM sequentially infers a 3D channel attention map $M_c \in R^{1\times 1\times 1\times C}$ and a 3D temporal attention map $M_t \in R^{1\times 1\times 1\times T}$. The processing flow of EAM can be expressed as

$$V' = T_{trans}(M_c(V) \otimes V), \tag{1}$$

$$V'' = T_{trans}(M_t(V') \otimes V'), \tag{2}$$

and

$$\widehat{V} = V + V'', \tag{3}$$

where $\otimes$ denotes element-wise multiplication; and $\widehat{V}$ is the final refined output, which is passed into the next 3D CNN module. The details of each attention module are described below.

3.2 Spatial efficient attention module

The various channels in 3D CNN models can be regarded as a spatial appearance representation of a certain action, and thus we can explore spatial attention on the channel-level, which helps learn discriminative features for action recognition. Inspired by ECA-Net [24], we designed an EAM-S module to learn the correlation score of each channel containing a certain action feature in the 3D CNN. The model highlights spatial regions relevant to the certain action category with high scores while suppressing the irrelevant regions with low scores. To capture the spatial attention map efficiently at the channel-level, we first squeeze the spatial and temporal dimensions of the input feature map to extract channel descriptors. For aggregating spatial information, global average pooling has been adopted for previous recalibration methods, such as ECA-Net [24]

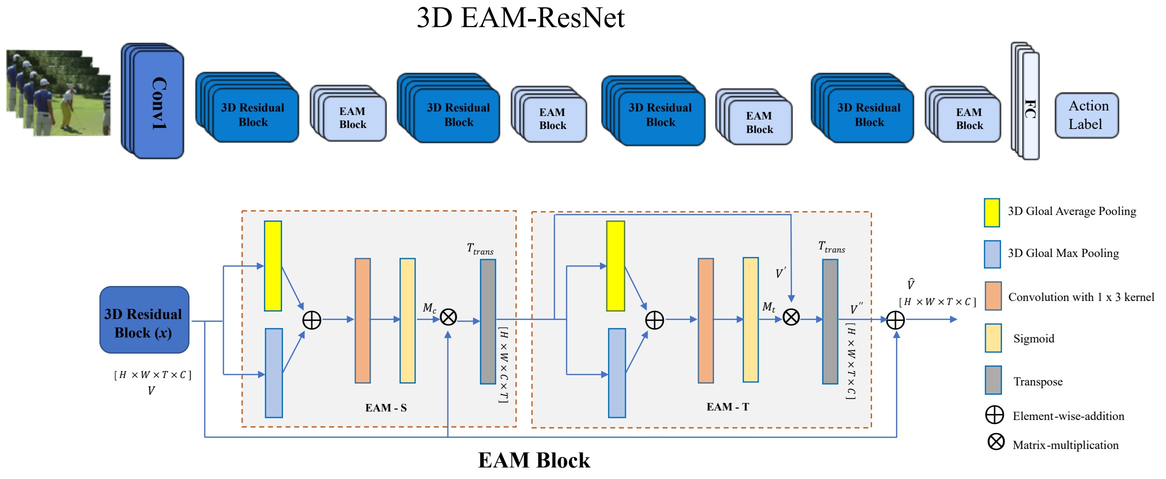
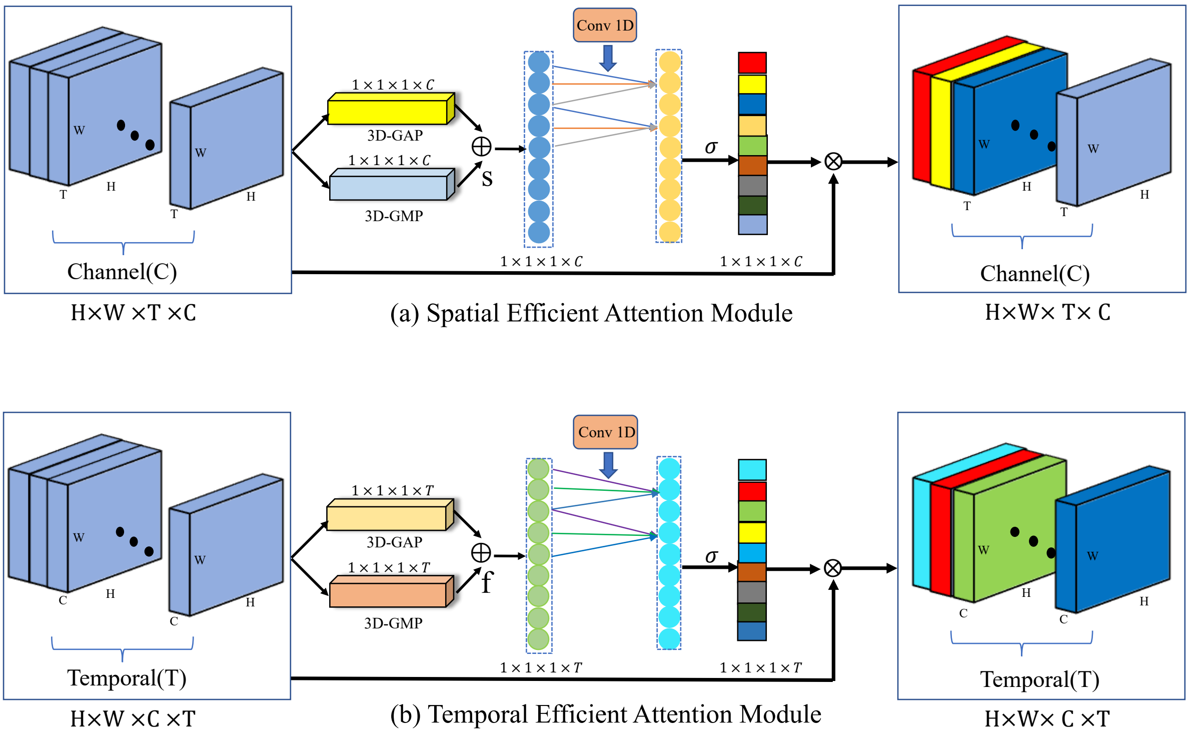


Fig. 2: EAM-ResNet. The EAM block is applied to the 3D ResNet. The 3D network uses video clips as input. The attention module has two sequential sub-modules: the spatial module EAM-S and the temporal module EAM-T. They respectively focus on spatial regions related to the action category on the channel-level and keyframes that are relevant to the action category on the frame-level.



(a) Spatial Efficient Attention Module

(b) Temporal Efficient Attention Module

Fig. 3: Diagram of two attention modules. Both the spatial module and temporal module combine outputs from 3D-GAP and 3D-GMP by element-wise summation, to generate attention weights by performing a 1D convolution and the sigmoid activation function.

and SE-Net [25]. In practice, we argue that global max pooling gathers another important clue for distinguishing object features. The CBAM [30] study confirmed that using both global average pooling and global max pooling features simultaneously can greatly improve the representation ability of CNN models rather than using each independently. Thus, as shown in Fig. 3 (a), the input feature map $V_c = [v_1, v_2, ..., v_c]$ can be treated as a combination of channels $v_c \in R^{H\times W\times T}$. We first aggregate spatial and temporal information of the input feature map by using both 3D global average pooling (3D-GAP) and 3D global max pooling (3D-GMP,) to obtain two different channel descriptors $a_c \in R^{1\times1\times1\times C}$ and $m_c \in R^{1\times1\times1\times C}$, respectively. The formula of the aggregate operation is

$$a_c = GAP_{3D}(V_c) = \frac{1}{HWT}\sum_{i=1}^{H}\sum_{j=1}^{W}\sum_{t=1}^{T} V_c(i,j,t), \tag{4}$$

and

$$m_c = GMP_{3D}(V_c) = \max_{\substack{i=1,\ldots,H\\ j=1,\ldots,W\\ t=1,\ldots,T}} \{V_c(i,j,t)\}, \tag{5}$$

where GAP_{3D} and GMP_{3D} represent 3D-GAP and 3D-GMP, respectively; i, j, and t are the spatio-temporal indexes; c is the index of the channel; $c \in [1,2,...,C]$; and H, W, and T represent the height, weight, and temporal, respectively. Two descriptors a_c and m_c count the global context and local discrimination information in each channel. The results from two pooling operations are first combined by element-wise summation to obtain the final channel feature descriptor $s \in R^{1\times1\times1\times C}$ as follows:

$$s = a_c + m_c. \tag{6}$$

Then, only considering the local interaction between each channel and its k_c adjacent channels, the weight w_c of $s_c \in s$ can be calculated as

$$w_c = \sigma(\sum_{j=1}^{k_c} \beta^j s_c^j), s_c^j \in \Omega_c^k, \tag{7}$$

where Ω_c^k indicates the set of k_c adjacent channels of s_c, β^j is the parameter of a 1D convolution with the kernel size of k_c, and σ is a sigmoid activation function.

3.3 Temporal efficient attention module

For recognizing the action in a video, different frames generally make different contributions. Some frames are more relevant to the action category, whereas other frames are more likely to be irrelevant or less relevant to the action category and may reduce the final recognition performance by introducing noise. By focusing on small but informative segments of the action video, instead of the entire video, the action recognition model is more robust. Therefore, introducing a temporal attention module plays a key role in learning discriminative features. To effectively sort out frames containing the information related to the certain action category and calculate the temporal attention score, we squeeze the channel and spatial dimensions of the input feature map to extract temporal descriptors. As shown in Fig. 3 (b), the input feature map $V_t{}' = [v_1', v_2', ..., v_t']$ is regarded as a combination of temporal series $v_t' \in R^{H\times W\times C}$. We simultaneously compress the channel and spatial dimensions by 3D-GAP and 3D-GMP to obtain two different temporal feature descriptors $a_t \in R^{1\times1\times1\times T}$ and $m_t \in R^{1\times1\times1\times T}$. The operation is as follows:

$$a_t = GAP_{3D}(V_t{}') = \frac{1}{HWC}\sum_{i=1}^{H}\sum_{j=1}^{W}\sum_{k=1}^{C} V_t{}'(i,j,k), \tag{8}$$

and

$$m_t = GMP_{3D}(V_t{}') = \max_{\substack{i=1,\ldots,H\\ j=1,\ldots,W\\ k=1,\ldots,C}} \{V_t{}'(i,j,k)\}, \tag{9}$$

where the outputs of transformation a_t and m_t represent a collection of global and local descriptors expressive for the entire video, respectively; i, j, and k are the spatial and channel indexes, respectively; t is the index of temporal; $t \in [1,2,...,T]$; and H, W, and C represent the height, width, and channel of the feature map, respectively. Then, to obtain the final temporal feature descriptor $f \in R^{1\times1\times1\times T}$, the model aggregates two different temporal feature descriptors by element-wise summation , and the operation is as follows:

$$f = a_t + m_t. \tag{10}$$

Similar to the method of channel activation in EAM-S, only the local interaction between each frame and its k_t adjacent frames is considered. Besides, w_t can be achieved by 1D convolution with the kernel size k_t followed by a sigmoid activation function as follows:

$$w_t = \sigma(C1D_{k_t}(f)), \tag{11}$$

where $C1D$ indicates a 1D convolution that only involves k_t parameters, σ is a sigmoid activation function, and $w_t \in R^{1\times1\times1\times T}$ is between $[0,1]$.

3.4 Joint efficient attention module

The EAM-S and EAM-T modules play complementary roles. The EAM-S and EAM-T modules can be combined to constitute the EAM module for 3D CNN models. The proposed model can focus on the video frame containing the action category on the premise of highlighting the spatial region related to the action class. The EAM module can be applied to any stage of a CNN and trained in an end-to-end manner without any additional auxiliary supervision. Experimental results show that jointly using them sequentially can achieve higher performance than the parallel method for 3D CNN-based action recognition methods. Taking the sequential spatial-temporal combination as an example, the given intermediate feature map V can be rescaled by EAM-S to obtain the feature map V'; then, EAM-T is derived from feature map V' and applied on feature map V' to obtain the output feature map V'', finally, the output feature map of the EAM module is $\widehat{V} = V + V''$. In the experimental part, we will discuss the results of using each alone versus in combination as well as the results of the proposed model with parallel aggregation and sequential aggregation.

4 Experiments

Experiments were conducted on two challenging action recognition datasets, that is, HMDB-51 [23] and UCF101 [26]. We fine-tuned our model by pre-training it on the Kinetics dataset [16] and applied it to the experiments on the two datasets to make a comparison with the state-of-the-art methods. Finally, we show visualization results to analyze and prove the effectiveness of the proposed EAM.

4.1 Dataset and evaluation metric

Typical action examples in the HMDB-51 and UCF-101 datasets are shown in Fig. 4. Both datasets have three sub-datasets, of which 70% were used for training and 30% were used for testing. Our evaluation results were based on the standard evaluation metrics of video accuracy, which is the mean video accuracy over three testing splits.

HMDB-51. The HMDB-51[23] dataset contains 6,849 clips divided into 51 action categories. Each action class contains a minimum of 101 clips collected from various sources, mostly from movies, and a small proportion from public databases, such as YouTube and Google videos. The dataset is full of challenges with higher intra-class variations and smaller inter-class variations. The action categories can be grouped into five types: general facial actions, facial actions with object manipulation, general body movements, body movements with object interaction, and body movements for human interaction.

UCF-101. The UCF-101[26] dataset is an action recognition dataset with realistic action videos. Collected from YouTube, the dataset contains 13,320 video clips belonging to 101 action categories. It contains various challenging scenarios, such as extreme illumination conditions, cluttered backgrounds, and large variations in camera motion. The videos were temporarily cut to remove non-action frames. The average duration of each video is about seven seconds and the action categories can be divided into five types: human-object interaction, body-motion only, human-human interaction, playing musical instruments, and sports.

4.2 Implementation details

The proposed EAM module can be easily integrated into existing 3D CNN-based action recognition models. In the experiments, we used PyTorch to implement EAM and integrated it into the popular ResNet and ResNeXt 3D ConvNet architectures. We fine-tuned the pre-trained models provided by a prior study [17], namely, ResNet-50, ResNeXt-101(16f), and ResNeXt-101(64f) for the HMDB-51 and UCF-101 datasets. These networks contained an independent convolution layer (conv1) at the forefront, which was followed by four residual blocks (res2, res3, res4, and res5). Following the settings from a prior study [17], we randomly obtained training samples generated from videos in training data for data augmentation. We randomly sampled a 112 × 112 crop in the aforementioned training samples from a random clip. Each sample crop was randomly flipped horizontally with a size of 3 channels × 16 frames × 112 pixels × 112 pixels ability 50%. We also subtracted the mean values of ActivityNet [48] from the sample for each color channel.

To mitigate overfitting and accelerate convergence in training, we only updated res5, the fully convolutional layer, and EAM. We used a stochastic gradient descent optimizer with a weight decay of 1e-5 and 0.9 for momentum. Then, the initial learning rates of the ResNet-50 network and ResNext-101 network were set to be 0.001 and 0.002, respectively. After a linear warm-up strategy in the first 15th epochs, the learning rate gradually increased to the configured value. The learning rate was also reduced by a factor of 10 at 45 and 75 epochs. Training of the model stopped at 100 epochs. The learning rate $lr(t)$ at epoch t was computed as follows:

$$lr(t) = \begin{cases} 1.0 \times 10^{-3} \times \frac{t}{15} & \text{if } t \leq 15 \\ 1.0 \times 10^{-3} & \text{if } 15 < t \leq 45 \\ 1.0 \times 10^{-4} & \text{if } 45 < t \leq 75 \\ 1.0 \times 10^{-5} & \text{if } 75 < t \leq 100 \end{cases} \tag{12}$$

In testing, we decomposed each video into non-overlapped 16 frame clips that were fed as input. Each clip was cropped from the center and resized to 3 channels × 16 frames × 112 pixels × 112 pixels. We then input each clip into the model to estimate the clip class scores,

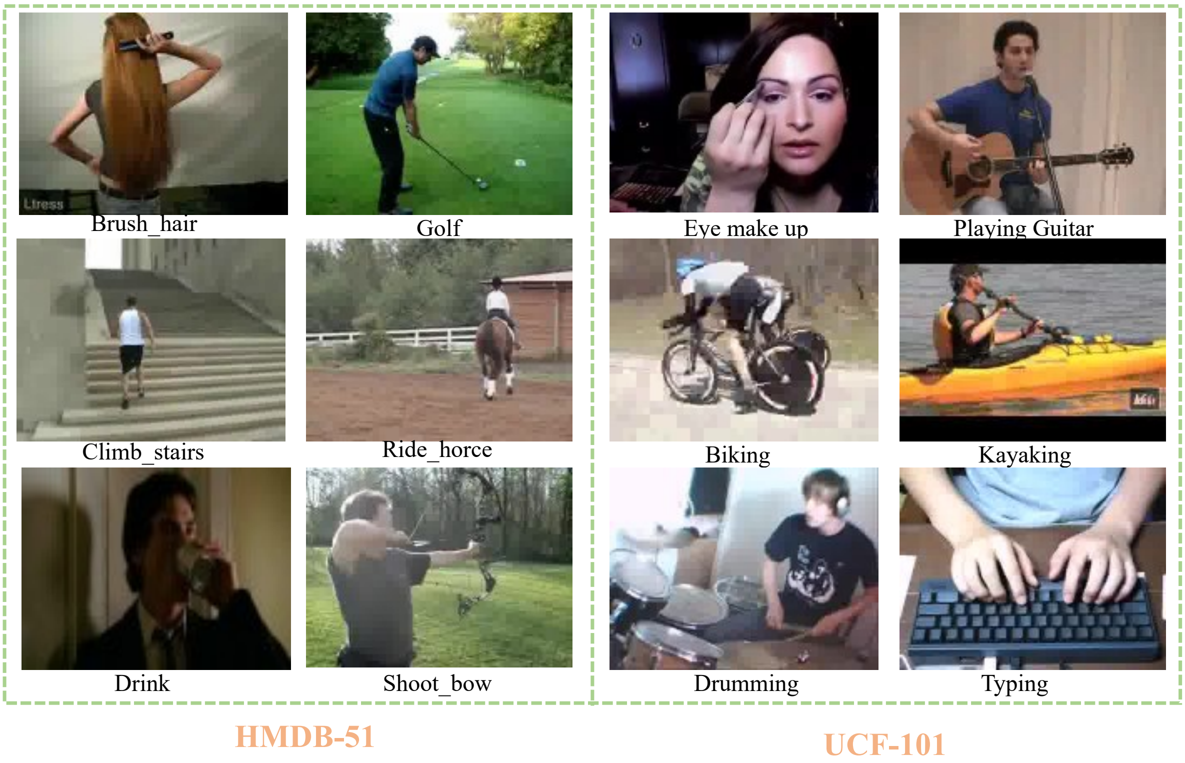


Fig. 4: Action examples from the HMDB-51 and UCF-101 datasets.

which were averaged over all the clips of the video to make a video-level prediction.

4.3 Ablation studies

To verify the effectiveness of our EAM module, we fine-tuned the pre-trained 3D ResNet-50 on the HMDB-51(Split 1) dataset for ablation studies.

Effect of kernel size on the EAM module. As shown in Eqs. (7) and (11), our EAM module involved two parameters that is, the kernel size of 1D convolution k_c and k_t for EAM-S and EAM-T, respectively. As shown in Table 1, we evaluated the effect of 1D convolution kernel size k in the EAM module. Table 1 shows the experimental results by setting k from 3 to 9. As shown, our EAM module improved the performance over the powerful baseline by using different kernel sizes. The EAM module using 1D convolution with a fixed kernel size of 3 obtained the highest performance by outperforming the baseline for 2.9%. The above results indicate that EAM is effective to mine the distinct importance of different frames and spatial regions in the image on recognizing actions.

Table 1: Performance (%) comparison of the proposed EAM module with varying kernel sizes

Model	k (kernel size)	HMDB-51
ResNet-50	N/A	62.5
ResNet-50+EAM	3	**65.4**
	5	63.2
	7	63.6
	9	63.2

EAM models versus the baseline. Table 2 shows a performance comparison of our EAM-S, EAM-T, and mixed EAM by both EAM-S and EAM-T, with the baseline evaluated on the HMDB-51(Split 1) dataset. We studied three ways of model combination: parallel with fusion (EAM-S//T), sequential spatial-temporal (EAM-ST), and sequential temporal-spatial (EAM-TS).

As shown in Table 2, the performances of both EAM-S and EAM-T were significantly improved over the baseline. EAM-S, EAM-T, and the combined EAM-ST significantly outperformed the baseline by 1.3%, 1.9%, and 2.9%, respectively. The above results show that focusing on the spatial areas that have high relevance to the action category and the keyframes containing the action class can enhance the prediction robustness of the model. Temporal attention achieved higher accuracy than spatial attention. This phenomenon demonstrates that temporal attention plays a dominant role

Table 2: Performance (%) comparison of our model with the baseline

	Model	HMDB-51
Baseline	ResNet-50	62.5
Spatial	EAM-S	63.8
Temporal	EAM-T	64.4
Both	EAM-S//T	63.7
	EAM-TS	64.9
	EAM-ST	**65.4**

Table 3: Performance (%) comparison of EAM in different stages of 3D ResNet-50

Model	HMDB-51
ResNet-50	62.5
ResNet-50 + EAM(res2)	63.4
ResNet-50 + EAM(res3)	63.7
ResNet-50 + EAM(res4)	64.4
ResNet-50 + EAM(all res)	**65.4**

Table 4: Recognition results of different attention methods on the HMDB-51(Split 1) dataset .

Model	Param.	GFLOPs	HMDB-51
ResNet-50 [17]	46.30M	10.102G	62.5
+ SE [25]	46.99M	10.105G	63.7
+ ECA [24]	46.30M	10.104G	64.1
+ CBAM [30]	47.00M	10.106G	64.3
+ EAM	46.30M	10.106G	**65.4**
ResNext-101(16f) [17]	47.62M	9.616G	63.7
+ SE [25]	48.32M	9.619G	64.3
+ ECA [24]	47.62M	9.619G	64.5
+ CBAM [30]	48.32M	9.620G	64.8
+ EAM	47.62M	9.620G	**65.1**
ResNext-101(64f) [17]	47.62M	38.466G	70.1
+ SE [25]	48.32M	38.476G	70.1
+ ECA [24]	47.62M	38.475G	70.2
+ CBAM [30]	48.32M	38.476G	70.3
+ EAM	47.62M	38.483G	**70.7**

in EAM, and it can select the keyframes related to the action category instead of reducing the interference of the recognition performance of unrelated frames.

As shown in Table 2, sequential spatial-temporal EAM-ST achieved the best performance, with an enhancement of 1.6% and 1.0% compared to EAM-S and EAM-T, respectively. Parallel optimization is more difficult than sequential . The above results validate that EAM-S and EAM-T play complementary roles, and the proposed model takes the advantages of both EAM-S and EAM-T to further improve the recognition performance of 3D CNN-based action recognition methods. Moreover, our model can focus on the keyframes containing the action category on the premise of highlighting the spatial regions related to the action class, which promotes the robustness of the model to extract features. Our EAM module containing sequential the spatial-temporal attention module was consistently used in all the experiments.

Where to embed EAM. We compared the performance of adding the EAM module to different stages of 3D ResNet-50 and checked which layer had more impact in terms of analyzing action, that is, res2, res3, res4, and after each layer. As we expected, when the stage increased, the performance of the model gradually improved. The reason for this is that the extracted information is more abstract and representative as the number of layers increases. As shown in Table 3, the addition of EAM produced better performance at each layer.

4.4 Experimental results

Comparison with the SOTA attention methods. We evaluated the experiments on the HMDB-51 (Split 1) dataset. According to previous experiments, the SE [25], ECA [24], CBAM [30], and EAM modules respectively were embedded into all the stages of these models, that is, ResNet-50, ResNeXt-101(16f), and ResNeXt-101(64f). As shown in Table 4,compared with the baseline and SOTA attention modules, our models achieved better performance. The reason for this is that our models pay more attention to the keyframes with high relevance to the action category and the spatial regions related to the action class when extracting spatio-temporal information, which improves the performance of network extraction features. Additionally, we further evaluated the complexity of our methods. Table 4 shows the overall overhead of the EAM module is quite small in terms of both parameters and computation, which indicates that the proposed EAM module is lightweight.

Comparison with the state-of-the-art methods. As shown in Table 5, we compared our results with the state-of-the-art methods over all three splits of the HMDB-51 and UCF-101 datasets. For comparison, the two-stream CNN-based methods including TSN [10], CO2FI+ASY [12], and ST-Multiplier Net[49]. The 3D CNN-based methods included C3D [15], P3D [19], and MiCT-Net [21]. Attention-based methods included STC-

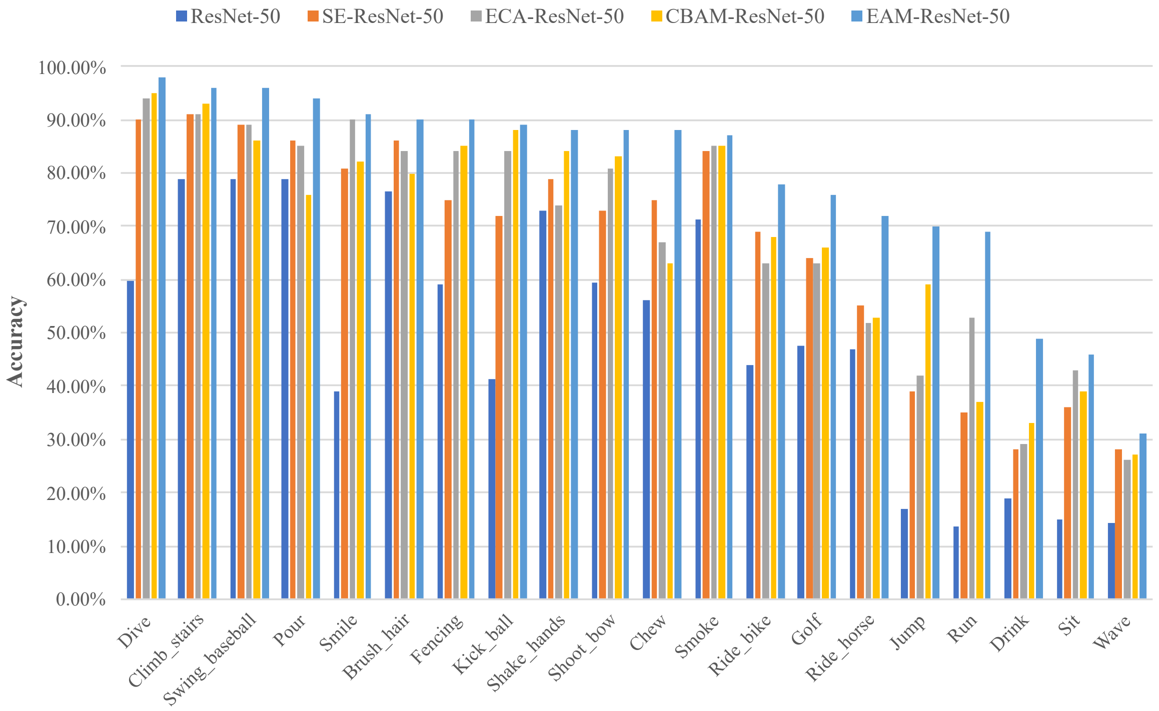


Fig. 5: Comparisons of accuracy (%) for the top-20 classes on the HMDB-51 dataset (Split 1) between the the EAM-ResNet-50 with ResNet-50, SE-ResNet-50, ECA-ResNet-50, and CBAM-ResNet-50 models.

ResNet [42], Attention Cluster [43], STA-CNN [44], and Pyramid Attention Network [45]. Our best result outperformed many methods on both the HMDB-51 dataset and the UCF-101 dataset, which indicates the importance of the attention mechanism and demonstrates the effectiveness of the EAM module. The performances of the 3D CNN-based methods were lower than those of the two-stream CNN-based methods for action recognition. It should be noted that in our method, we simply insert our EAM module into 3D CNN without too much additional computation, and the recognition performance could be significantly improved, and it outperformed the two-stream CNN-based methods[10, 12, 49]. Our model can distinguish the spatio-temporal feature representation to highlight the spatial regions that are more relevant to the action category in frames and focus on the keyframes relevant to the action category by spatial and temporal attention. Besides, our model also outperformed the latest attention-based methods, such as STC-ResNet [42] and Attention Cluster [43], on both datasets. However, STA-CNN [44] and Pyramid Attention Network [45] achieved better performances than our method on the UCF-101 dataset. This is because these methods use expensive optical-flow maps in addition to RGB input-frames. It is necessary to extract optical flow from the image in advance, which is usually computationally intensive and therefore difficult to obtain for large scale datasets. We observed that the EAM method appears to work better for HMDB-51 than UCF-101 (1.2% vs. 0.6%). The reason is because the video quality of UCF-101 is higher than that of HMDB-51. In the UCF-101 dataset, frames irrelevant to a certain action in a video are less. Hence, it is more difficult for EAM-T to select key frames.

To further verify the performance we compared our model with the baseline and other SOTA attention models. As shown in Fig. 5, the recognition accuracy of the top-20 classes from our model significantly outperformed the baseline and other SOTA attention models as, “Fencing”, ”Brush_hair, “Jump,” and “Climb_stairs”. The performance was significantly improved compared to the original model. This means that our model faithfully focuses on the spatial regions related to the action class in frames and captures the keyframes containing information that is relevant to the action category. Several action examples from the HMDB-51 dataset, such as ”Brush_hair, ”Drink, “Fencing,” “Jump,” and “Thow,” are shown in Fig. 6. They were incorrectly recognized by the baseline and other SOTA attention models, whereas these actions could be accurately rec-

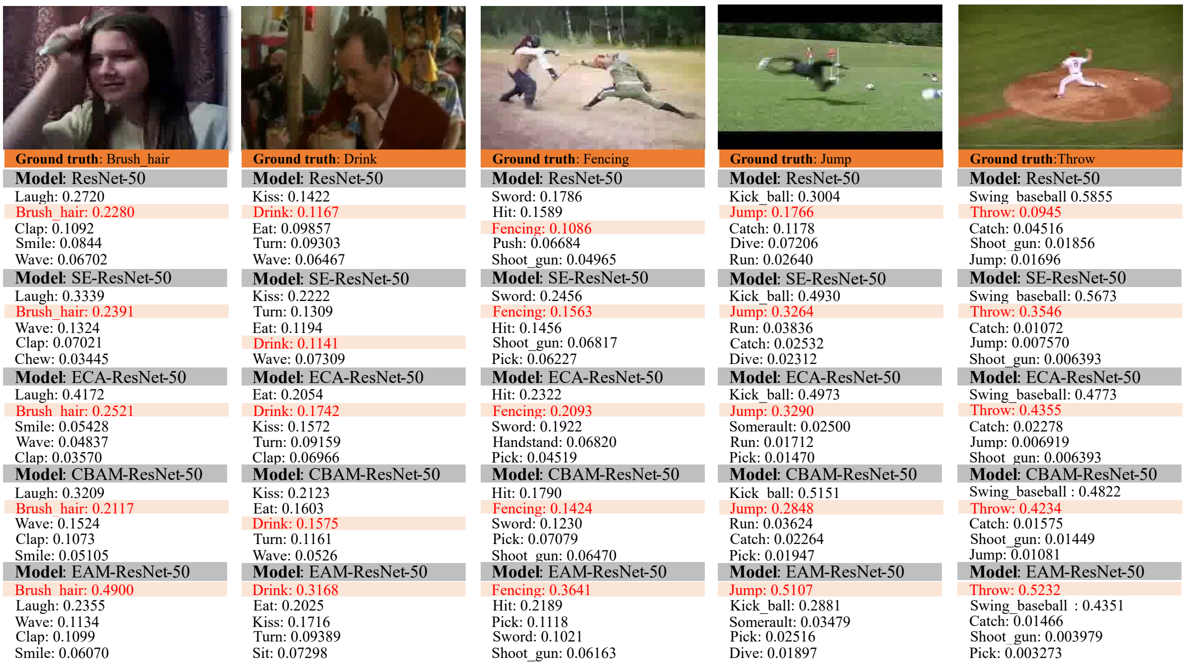


Fig. 6: Examples of action recognition, in which EAM-ResNet-50(Ours) succeeded while the original ResNet-50, SE-ResNet-50, ECA-ResNet-50, and CBAM-ResNet-50 failed.

Table 5: Performance (%) comparison with the state-of-the-art methods on the HMDB-51 and UCF-101 datasets

Method	HMDB-51	UCF-101
ResNet-50(RGB)[17]	61.0	89.0
ResNeXt-101(RGB)[17]	63.5	90.7
ResNeXt-101+64f(RGB)[17]	69.5	94.0
TBN(RGB) [50]	69.4	93.6
TSN (RGB+Flow)[10]	69.4	94.2
CO2FI+ASY(RGB+Flow)[12]	69.0	94.3
ST-Multiplier Net (RGB+Flow)[49]	68.9	94.2
C3D(RGB)[15]	56.8	82.3
P3D(RGB)[19]	N/A	88.6
MiCT-Net (RGB+Flow)[21]	70.5	94.7
STC-ResNet-101+64f (RGB)[42]	70.5	93.7
Attention Cluster (RGB+Flow)[43]	69.2	94.6
STA-CNN (RGB+Flow)[44]	70.2	95.3
Pyramid Attention Network (RGB+Flow)[45]	70.5	95.3
EAM-ResNet-50	63.4	89.8
EAM-ResNeXt-101	64.4	91.0
EAM-ResNeXt-101+64f	**70.7**	**94.6**

ognized after putting the EAM module in the model, which indicates that our EAM module can exploit the discriminative information at the channel level and frame level and improve the capability of 3D CNNs with a more powerful spatio-temporal feature learning.

4.5 Visualization analysis

To understand the role of the attention mechanism more intuitively, Fig. 7 shows the visualization of spatial-temporal attention weights from the EAM-ResNet-50 on video sequences from the HMDB-51 dataset using channel feature heatmaps and temporal attention weights. According to Fig. 7, the temporal attention weights indicate distinguishing the importance of different frames. The frames containing the information that is relevant to the action category can be identified by the larger weights. The channel feature maps for the specified frame are shown, and different feature maps have different contributions. As shown, EAM is effective for focusing on the feature maps that are more relevant to the action category. As shown in Fig. 8, we applied the Grad-CAM [51] to different networks using some video sequences from the HMDB-51 validation set. We compared the visualization results of EAM-ResNet-50 with ResNet-50, SE-ResNet-50, ECA-ResNet-50, and CBAM-ResNet-50. The softmax scores for a target class are also shown in the figure. From the Grad-CAM mask that covers the object regions in the input, the regions that the network considered as important for predictions can be seen. Compared to the other methods, EAM-ResNet-50 generated more accurate mask regions for predictions, and the target class scores also increased accordingly. That is, EAM-ResNet-50 learned well to

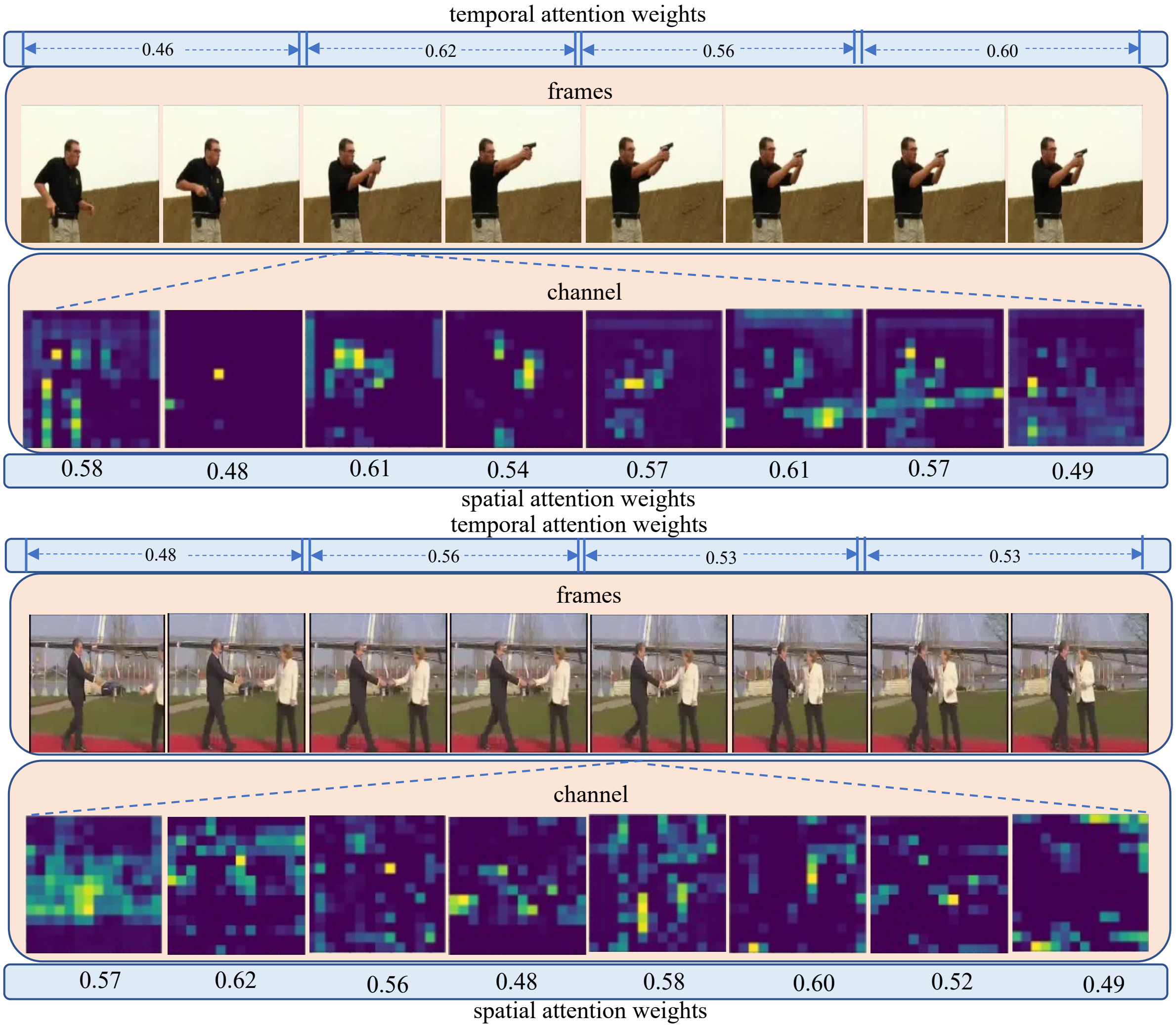


Fig. 7: Visualization of spatial and temporal attention weights extracted from EAM-ResNe-50 on the "Shoot_gun" and "Shake_hands" actions.

mine image regions related to certain action category and keyframes containing action-related information in sequences.

5 Conclusion

In this paper, a novel module with spatial and temporal attention for 3D CNN-based action recognition is proposed. This method uses a 3D CNN to extract the basic deep features and then mines discriminative features between actions using the proposed attention model. The proposed attention mechanism assigns high importance scores to spatial regions and keyframes that are more relevant to the action category by local cross-channel and cross-frame interaction strategies without dimensionality reduction. Our EAM can be expediently added into 3D CNN-based action recognition models with only a minor increase in computational complexity. State-of-the-art performance can be achieved in action recognition tasks, and extensive experiments proved the effectiveness of the proposed EAM module.

Acknowledgements The work is supported by the following projects:National Natural Science Foundation of China (NSFC) Essential project, Nr.: U2033218, 61831018; NSFC, Nr.: 61772328, 61702322.

References

1. Klaser A, Marszalek M, Schmid C (2008) A spatio-temporal descriptor based on 3d-gradients. In: British Machine Vision Conference (BMVC)

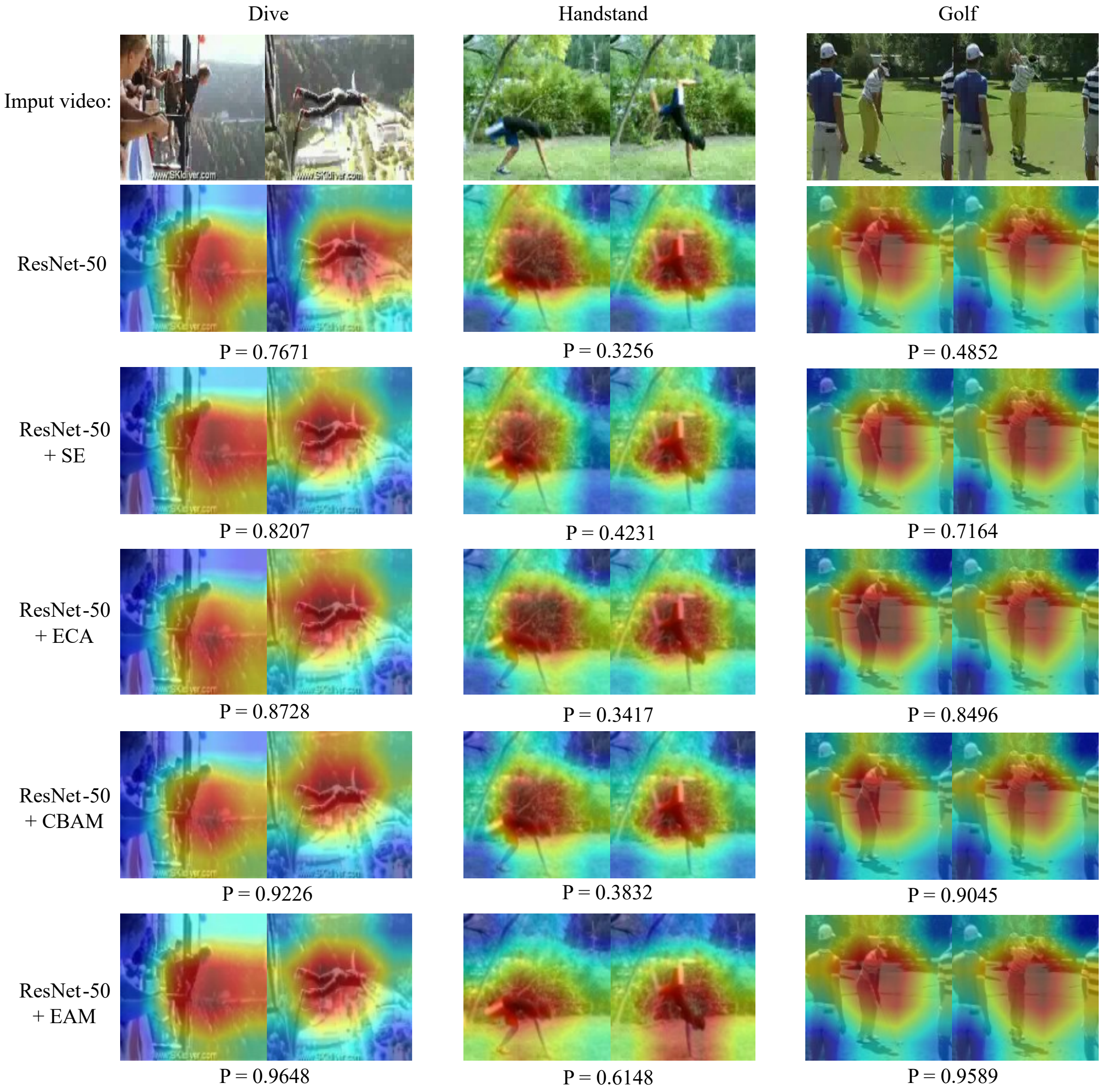


Fig. 8: Grad-CAM [51] visualization results on the HMDB-51 validation set. We compared the visualization results of EAM-ResNet-50 with ResNet-50, SE-ResNet-50, ECA-ResNet-50, and CBAM-ResNet-50. Here, P denotes the softmax score of each network for the ground-truth class.

2. Brox T, Malik J (2010) Large displacement optical flow: descriptor matching in variational motion estimation. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) 33(3):500–513
3. Dalal N, Triggs B, Schmid C (2006) Human detection using oriented histograms of flow and appearance. In: European Conference on Computer Vision (ECCV)
4. Wang H, Kläser A, Schmid C, Liu CL (2011) Action recognition by dense trajectories. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
5. Wang H, Schmid C (2013) Action recognition with improved trajectories. In: IEEE international conference on computer vision (ICCV)
6. Yu Z, Li T, Luo G, Fujita H, Yu N, Pan Y (2018) Convolutional networks with cross-layer neurons for image recognition. Information Sciences 433:241–254
7. Pérez-Hernández F, Tabik S, Lamas A, Olmos R, Fujita H, Herrera F (2020) Object detection binary classifiers methodology based on deep learning to identify small objects handled similarly: Application in video surveillance. Knowledge-Based Sys-

tems (KBS) p 105590

8. Liu L, Wang S, Hu B, Qiong Q, Wen J, Rosenblum DS (2018) Learning structures of interval-based bayesian networks in probabilistic generative model for human complex activity recognition. Pattern Recognition (PR) 81:545–561
9. Simonyan K, Zisserman A (2014) Two-stream convolutional networks for action recognition in videos. In: Advances in Neural Information Processing Syetems (NIPS)
10. Wang L, Xiong Y, Wang Z, Qiao Y, Lin D, Tang X, Van Gool L (2018) Temporal segment networks for action recognition in videos. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) 41:2740–2755
11. Feichtenhofer C, Pinz A, Zisserman A (2016) Convolutional two-stream network fusion for video action recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
12. Lin W, Zhang C, Lu K, Sheng B, Wu J, Ni B, Liu X, Xiong H (2018) Action recognition with coarse-to-fine deep feature integration and asynchronous fusion. In: AAAI Conference on Artificial Intelligence (AAAI)
13. Yue-Hei Ng J, Hausknecht M, Vijayanarasimhan S, Vinyals O, Monga R, Toderici G (2015) Beyond short snippets: Deep networks for video classification. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
14. Donahue J, Anne Hendricks L, Guadarrama S, Rohrbach M, Venugopalan S, Saenko K, Darrell T (2015) Long-term recurrent convolutional networks for visual recognition and description. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
15. Tran D, Bourdev L, Fergus R, Torresani L, Paluri M (2015) Learning spatiotemporal features with 3d convolutional networks. In: IEEE International Conference on Computer Vision (ICCV)
16. Carreira J, Zisserman A (2017) Quo vadis, action recognition? a new model and the kinetics dataset. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
17. Hara K, Kataoka H, Satoh Y (2018) Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet? In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
18. Wang L, Li W, Li W, Van Gool L (2018) Appearance-and-relation networks for video classification. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
19. Qiu Z, Yao T, Mei T (2017) Learning spatio-temporal representation with pseudo-3d residual networks. In: IEEE International Conference on Computer Vision (ICCV)
20. Tran D, Wang H, Torresani L, Ray J, LeCun Y, Paluri M (2018) A closer look at spatiotemporal convolutions for action recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
21. Zhou Y, Sun X, Zha ZJ, Zeng W (2018) Mict: Mixed 3d/2d convolutional tube for human action recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
22. Tran D, Wang H, Torresani L, Feiszli M (2019) Video classification with channel-separated convolutional networks. In: IEEE International Conference on Computer Vision (ICCV)
23. Kuehne H, Jhuang H, Garrote E, Poggio T, Serre T (2011) Hmdb: A large video database for human motion recognition. In: IEEE International Conference on Computer Vision (ICCV)
24. Wang Q, Wu B, Zhu P, Li P, Zuo W, Hu Q (2020) Eca-net: Efficient channel attention for deep convolutional neural networks. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
25. Hu J, Shen L, Sun G (2018) Squeeze-and-excitation networks. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
26. Soomro K, Zamir AR, Shah M (2012) Ucf101: A dataset of 101 human actions classes from videos in the wild. CoRR
27. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2015) Going deeper with convolutions. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
28. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
29. Hu J, Shen L, Albanie S, Sun G, Vedaldi A (2018) Gather-excite: Exploiting feature context in convolutional neural networks. In: Conference on Neural Information Processing Syetems (NIPS)
30. Woo S, Park J, Lee JY, So Kweon I (2018) Cbam: Convolutional block attention module. In: European conference on computer vision (ECCV)
31. Gao P, Yuan R, Wang F, Xiao L, Fujita H, Zhang Y (2020) Siamese attentional keypoint network for high performance visual tracking. Knowledge-based systems (KBS) 193:105448
32. Gao P, Zhang Q, Wang F, Xiao L, Fujita H, Zhang Y (2020) Learning reinforced attentional representation for end-to-end visual tracking. Information Sciences 517:52–67

33. Zhang Z, Lan C, Zeng W, Jin X, Chen Z (2020) Relation-aware global attention for person re-identification. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
34. Li Y, Jiang X, Hwang JN (2020) Effective person re-identification by self-attention model guided feature learning. Knowledge-Based Systems (KBS) 187:104832
35. Chen B, Deng W, Hu J (2019) Mixed high-order attention network for person re-identification. In: IEEE International Conference on Computer Vision (ICCV)
36. Li S, Bak S, Carr P, Wang X (2018) Diversity regularized spatiotemporal attention for video-based person re-identification. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
37. Li X, Zhong Z, Wu J, Yang Y, Lin Z, Liu H (2019) Expectation-maximization attention networks for semantic segmentation. In: IEEE International Conference on Computer Vision (ICCV)
38. Fu J, Liu J, Tian H, Li Y, Bao Y, Fang Z, Lu H (2019) Dual attention network for scene segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
39. Huang Z, Wang X, Huang L, Huang C, Wei Y, Liu W (2019) Ccnet: Criss-cross attention for semantic segmentation. In: IEEE International Conference on Computer Vision (ICCV)
40. Zhu Z, Xu M, Bai S, Huang T, Bai X (2019) Asymmetric non-local neural networks for semantic segmentation. In: IEEE International Conference on Computer Vision (ICCV)
41. Wang X, Girshick R, Gupta A, He K (2018) Non-local neural networks. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
42. Diba A, Fayyaz M, Sharma V, Mahdi Arzani M, Yousefzadeh R, Gall J, Van Gool L (2018) Spatio-temporal channel correlation networks for action classification. In: European Conference on Computer Vision (ECCV)
43. Long X, Gan C, De Melo G, Wu J, Liu X, Wen S (2018) Attention clusters: Purely attention based local feature integration for video classification. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
44. Yang H, Yuan C, Zhang L, Sun Y, Hu W, Maybank SJ (2020) Sta-cnn: Convolutional spatial-temporal attention learning for action recognition. IEEE Transactions on Image Processing (TIP) 29:5783–5793
45. Du Y, Yuan C, Li B, Zhao L, Li Y, Hu W (2018) Interaction-aware spatio-temporal pyramid attention networks for action classification. In: European Conference on Computer Vision (ECCV)
46. Chen Y, Rohrbach M, Yan Z, Shuicheng Y, Feng J, Kalantidis Y (2019) Graph-based global reasoning networks. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
47. Li J, Liu X, Zhang M, Wang D (2020) Spatio-temporal deformable 3d convnets with attention for action recognition. Pattern Recognition (PR) 98:107037
48. Caba Heilbron F, Escorcia V, Ghanem B, Carlos Niebles J (2015) Activitynet: A large-scale video benchmark for human activity understanding. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
49. Feichtenhofer C, Pinz A, Wildes RP (2017) Spatiotemporal multiplier networks for video action recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
50. Li Y, Song S, Li Y, Liu J (2019) Temporal bilinear networks for video action recognition. In: AAAI Conference on Artificial Intelligence (AAAI)
51. Selvaraju RR, Cogswell M, Das A, Vedantam R, Parikh D, Batra D (2017) Grad-cam: Visual explanations from deep networks via gradient-based localization. In: IEEE International Conference on Computer Vision (ICCV)