Computer Systems
Science and Engineering

Tech Science Press

DOI: 10.32604/csse.2023.040132
Article

Check for updates

MSF-Net: A Multilevel Spatiotemporal Feature Fusion Network Combines Attention for Action Recognition

Mengmeng Yan[1], Chuang Zhang[1,2,*], Jinqi Chu[1], Haichao Zhang[1], Tao Ge[1] and Suting Chen[1]

[1]School of Electronic and Information Engineering, Nanjing University of Information Science and Technology, Nanjing, 210044, China
[2]Jiangsu Key Laboratory of Meteorological Observation and Information Processing, Nanjing, 210044, China
*Corresponding Author: Chuang Zhang. Email: ChZhang2023@outlook.com
Received: 06 March 2023; Accepted: 17 April 2023; Published: 28 July 2023

Abstract: An action recognition network that combines multi-level spatiotemporal feature fusion with an attention mechanism is proposed as a solution to the issues of single spatiotemporal feature scale extraction, information redundancy, and insufficient extraction of frequency domain information in channels in 3D convolutional neural networks. Firstly, based on 3D CNN, this paper designs a new multilevel spatiotemporal feature fusion (MSF) structure, which is embedded in the network model, mainly through multilevel spatiotemporal feature separation, splicing and fusion, to achieve the fusion of spatial perceptual fields and short-medium-long time series information at different scales with reduced network parameters; In the second step, a multi-frequency channel and spatiotemporal attention module (FSAM) is introduced to assign different frequency features and spatiotemporal features in the channels are assigned corresponding weights to reduce the information redundancy of the feature maps. Finally, we embed the proposed method into the R3D model, which replaced the 2D convolutional filters in the 2D Resnet with 3D convolutional filters and conduct extensive experimental validation on the small and medium-sized dataset UCF101 and the large-sized dataset Kinetics-400. The findings revealed that our model increased the recognition accuracy on both datasets. Results on the UCF101 dataset, in particular, demonstrate that our model outperforms R3D in terms of a maximum recognition accuracy improvement of 7.2% while using 34.2% fewer parameters. The MSF and FSAM are migrated to another traditional 3D action recognition model named C3D for application testing. The test results based on UCF101 show that the recognition accuracy is improved by 8.9%, proving the strong generalization ability and universality of the method in this paper.

Keywords: 3D convolutional neural network; action recognition; MSF; FSAM

1 Introduction

Multimedia information is exploding in popularity as a result of the quick development of computer and network technology. Also, a lot of important information is present in the video as a medium for information transfer. Throughout daily life, there is a category of video content regarding human behavior and activities. The technology to recognize and categorize the actions in the video is also known as Human Action Recognition (HAR) technology. The ability of machines to comprehend the world and human action depends in large part on the HAR technology [1], which has important applications in the fields of intelligent security, human-computer interaction, and intelligent education. Early videos typically used RGB or grayscale video as HAR input [2]. However, recently, multimodal HAR frameworks consisting of the skeleton, infrared sequence, point cloud, radar, and WiFi as multiple sensor inputs have become a research hotspot. Unimodal action recognition techniques still have a lot of research value since unimodal data in the form of RGB is more typical in practical applications [3]. Deep learning has been extensively used in the field of action recognition in recent years, as a result of the advancement of artificial intelligence technology. When DeepVideo [4] was first proposed, it was intended for each video frame to independently use the 2DCNN model. Although though there are now various research techniques that use still photos as input [5] to increase recognition effectiveness, utilizing solely images as input will overlook the temporal connection between video frames [6], leading to the loss of temporal dimensional properties. C3D [7] was proposed to directly extract the features of time and space dimensions to better capture time information. This approach had a strong generalization ability in addition to producing good recognition results. Additionally, the author successfully introduced residual structure into the field of human action recognition by replacing 2D-Resnet [8] with a 3D convolution kernel. However, 3D CNN is ineffective and has a lot of redundant information because of the numerous weight parameters and computation requirements. According to the literature [9], which is based on C3D, the Efficient residual block replaces the two cascaded $3 \times 3 \times 3$ convolutional filters and adds a time attention mechanism that can efficiently reduce redundant information by lowering model parameters. However, the convolution filters in this model only have a single kernel size and don't perform a hierarchical analysis of spatiotemporal features at various scales. The reference [10] uses a feature pyramid structure on time to study frame rate and realize multi-scale feature fusion of time dimension in order to extract features at various levels. The multi-level spatiotemporal feature extraction method needs to be improved, though, because the feature pyramid's structure is intricate and challenging to optimize during training.

In the literature [11], grouping convolution and depth-separable convolution in 2DCNN are used for the spatiotemporal feature extraction step of 3D convolution. The findings indicate that the extraction of spatial-temporal features is more sufficient the more feature information interacts across channels. The interdependence of features across channels should therefore also be taken into consideration. By giving each channel a different weight at the beginning of the process, SENet [12], which has the advantages of low computing cost and superior performance, can efficiently reduce redundant information. However, it was demonstrated by FcaNet [13] that the pre-processing GAP operation in SENet only took into account the information of the channel's lowest frequency component, causing the loss of information for higher frequency components. This paper proposes a multilevel spatiotemporal feature fusion network model combining attention since none of them can fully address the issues of single spatiotemporal feature scale extraction, information redundancy, and inadequate extraction of frequency domain information in channels. The model is based on Res3D-18, which decomposes the second convolution filter of the residual block into a multi-level convolution structure organized by channels. The primary method by which this structure accomplishes the fusion

of multi-scale features is by splicing the features following one set of convolution with the features prior to the subsequent set of convolution. Additionally, the model incorporates spatial and temporal attention modules as well as multi-frequency channel attention to improve the perception of channel and spatiotemporal features.

A large number of experiments have been carried out on the UCF101 dataset using the above network model, and the results amply support the efficacy of the method suggested in this paper. Three major contributions are made in this paper compared to other studies:

1. A novel multi-level spatiotemporal feature fusion (MSF) module is designed, which allows for the fusion of short-medium-long time series of temporal data based on the reduction of model parameters, as well as the equivalent expansion of the perceptual field of the feature map from the spatial dimension.

2. We propose a new multi-frequency channel and spatiotemporal attention module (FSAM) that enables the model to capture key action information in multiple frequency channels by extending the pre-processing GAP operation in SENet to the frequency domain and spatial attention to the spatiotemporal dimension.

3. For experimental validation and generalization capability testing, our two proposed modules are ported to R3D and C3D structures. It is implied that the method can be incorporated into other 3D neural networks to achieve the effect of reducing the number of parameters while increasing the recognition accuracy.

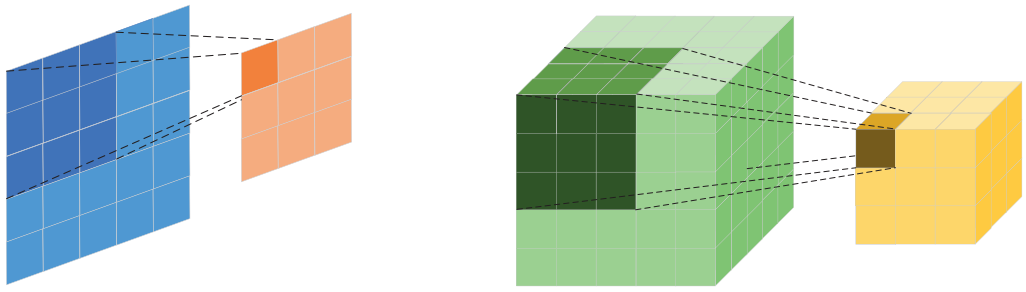
2 Related Work

2.1 Video Action Recognition

Action recognition has long been a significant and difficult research value in the area of video comprehension. Its major duty is to categorize actions for a given segmented video clip, such as making out, playing football, sprinting, etc. The extraction of video temporal and spatial information has always been the focus of research since video classification includes both spatial and temporal features. Improved Dense Trajectories (IDT) [14] is one of the most time-honored and successful methods, and it was initially utilized by action recognition to extract characteristics [15]. The IDT algorithm is primarily divided into three sections: track-based feature extraction, feature track tracking, and intensive sampling of feature points. IDT's great precision and strong durability have allowed it to dominate the field of video processing. Traditional action identification techniques, however, suffer from drawbacks such a heavy reliance on sample data and a susceptibility to noise [16]. Deep learning-based approaches for recognizing human action increasingly demonstrate their superiority. 2DCNN [5] was initially suggested as a solution for action recognition, but this approach struggles to capture motion information, necessitating the development of a more suitable approach. The Two-Stream Convolutional Network [17] incorporates information on optical flow and spatial flow for action recognition since optical flow can be utilized to represent the motion properties of objects. Nevertheless, Two-Stream has significant computation and storage needs, making it difficult to install and train large amounts of data. In order to model the time information in videos, 3DCNN became a crucial tool.

Ji et al. [18] initiated the 3DCNN pioneering work by primarily using a 3D convolution filter to extract time and space data simultaneously. The primary distinction between 3D convolution and 2D convolution is the requirement for the three-dimensional sliding of the 3D filter. Fig. 1 depicts the neural network's convolution procedure using a single channel as an example. The 2D

convolution in Fig. 1a has the following number of parameters: $C_{out} \times \left(3^2 \times C_{in} + 1\right)$, and the 3D convolution in Fig. 1b has the following number of parameters: $C_{out} \times \left(3^3 \times C_{in} + 1\right)$, where C_{in} and C_{out} denote the number of input and output channels, respectively. It is clear that 3D convolution requires exponentially more parameters and processing resources than 2D convolution, which makes network training time-consuming and challenging to achieve the low latency requirements in real-world applications. Therefore, many networks focused on enhancing the effectiveness and 3DCNN's recognition accuracy. One 3D residual structure to avoid gradient explosion in the deep network is R3D [19]. To minimize the number of parameters, P3D [20] and R(2 + 1)D [21] were created to break down the 3D convolution filter into 1D time convolution and 2D space convolution. Furthermore, I3D [22] enhances action recognition performance by combining optical flow with conventional 3D convolution. Nevertheless, not only do these techniques have redundant information, but they can also only extract local short-range temporal information, which omits the impact of medium- and long-range temporal information on the action recognition model.



(a) The process of 2D convolution (b) The process of 3D convolution

Figure 1: Illustration of the convolution of a neural network, the filter of the 2D convolution is 3 × 3 and of the 3D convolution is 3 × 3 × 3

We have begun implementing effective deep learning algorithms coupled with vision systems on established hardware as a result of the aforementioned significant research that is primarily centered on action recognition algorithms [23]. However, as vision systems generally need to operate in real-time and with high recognition accuracy, the suggested algorithms must meet greater requirements for both the number of parameters and the volume of floating-point calculations. It indicates that 3D convolution still has tremendous space for development in recognition accuracy and efficiency. In contrast to other works, we applied the multilevel decomposition of the 3D convolution kernel and were motivated by the multi-scale feature fusion approach in 2D convolution to accomplish the fusing of multi-scale spatial information and short, medium, and long time series features. Experimental results support the simplicity and efficiency of this approach.

2.2 Attention Mechanism

The attention mechanism was first proposed in the area of visuals. In 2014, the Google Mind team used the Attention mechanism to identify images using an RNN network [24]. Subsequently, Bahdanau et al. [25] used an attention-like mechanism to perform translation and alignment simultaneously on a machine translation task. There are already a number of highly developed Attention mechanisms in the field of image processing, such the Channel Attention Mechanism (SENet) [12], which employs compression and incentive implementation, and CBAM [26] and BAM [27], which combine channel attention with spatial attention. Nevertheless, because only the lowest frequency component is maintained in the Global Average Pooling (GAP) process and the information from

other frequencies is ignored, such channel attention uses GAP as a pre-processing approach, which results in information loss. The majority of attention research in the area of action recognition has been concentrated on attention to temporal sequences since videos provide an additional temporal dimension above pictures. Li et al. [28] introduced an attention mechanism in a network combining LSTM and CNN to achieve better performance than the traditional LSTM; Li et al. [29] also proposed nested spatiotemporal attention blocks (NST) by embedding temporal attention into spatial attention and exploiting the interactions and nested relationships between time and space. However, when acting solely on a neural network, such attention mechanisms are unable to prevent information redundancy of other dimensional features.

Many 3D networks have attention mechanisms that deal with a single dimension, typically focused on time series or spatial data, as opposed to the attention mechanism in 2DCNN. We expand space attention into the space-time dimension in addition to generalizing the pre-processing of the channel attention mechanism in the frequency domain. We enhance the action recognition performance by combining the enhanced channel attention with space-time attention.

3 Approach

Multilevel features considerably enhance the visual task effects, as demonstrated in various research. In this paper, we propose a CNN model that combines attention and multilevel spatiotemporal feature fusion. In particular, the class residual blocks of multilevel spatiotemporal feature fusion are used to extract the spatiotemporal features at various scales, which are then combined with multi-frequency channels and spatiotemporal attention mechanisms to assign various weights to information of various importance and produce various prediction values using the weighted fusion. The method's implementation specifics are the main topic of this section.

3.1 Multilevel Spatiotemporal Feature Fusion

The majority of conventional 3D convolutional neural networks use fixed convolutional kernels of a single size, which not only results in the loss of fine-grained spatial features as the network gets deeper but also reduces recognition accuracy when the temporal input is too uniform. To solve these problems, this section proposes a multilevel spatiotemporal feature fusion (MSF) structure. The main source of inspiration for this module was the Inception network [30], which employs a $3 \times 3 \times 3$ convolutional kernel for concurrent multilevel spatiotemporal feature decomposition to expand the receptive field of the current layer. The MSF method, however, employs a fixed convolutional kernel size for every level. The specific structure of the module is shown in Fig. 2. As determined by the convolutional neural network's current layer's perceptual field's calculation:

$$RF_i = RF_{i-1} + (k-1)\prod_{i=0}^{i-1} S_{i-1} \tag{1}$$

where RF_i denotes the perceptual field of the current layer, RF_{i-1} denotes the perceptual field of the previous layer, k is the size of the convolution kernel, and S defines the step size. Its structure primarily deepens the neural network by multi-level decomposition and convolutional equivalence, hence increasing the model's recognition performance and enlarging the comparable spatiotemporal field of vision.

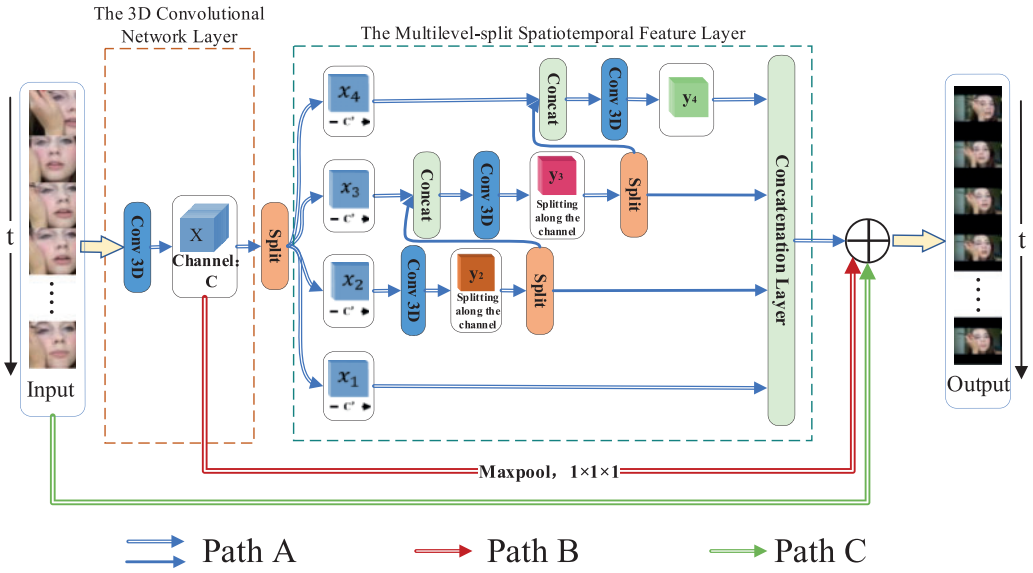


Figure 2: The structure diagram of MSF module

The Multilevel-split spatiotemporal Feature Layer (MSL) and the 3D convolutional layer make up the majority of the MSF module. Specifically, for a particular input feature, the low-level semantic data is first retrieved in the residual block after the first $3 \times 3 \times 3$ convolution kernel as the input feature X of the MSL. We split the feature maps into S groups (S = 4 in the Fig. 1.), each feature subset is denoted by x_i, where $i \in \{1, 2, \ldots, S\}$. The number of channels C' is $\frac{1}{S}$ of the number of input feature channels, which can be written as $C' = \frac{C}{S}$. During spatiotemporal features separation, the feature subset x_1 is directly output as y_1 without transformation. Each of the other subsets x_i corresponds to a hierarchically connected branch, and each branch has a $3 \times 3 \times 3$ convolution, denoted by $f^{3\times3\times3}()$. We denote by y_i the output of $f^{3\times3\times3}()$. The feature subset x_i is concatenated with the output of $f^{3\times3\times3}()$, and then fed into $f^{3\times3\times3}()$. Thus, y_i can be written as:

$$y_i = \begin{cases} x_i, i = 1 \\ f^{3\times3\times3}\left(cat\left[x_i, y_{(i-1)2}\right]\right), 1 < i < s \end{cases} \tag{2}$$

In the multilevel spatiotemporal feature fusion process, y_i will be divided into two sets of sub-feature maps y_{i1} and y_{i2} by channel again, where y_{i1} is directly spliced to the final output and y_{i2} is directly spliced to the next set of feature maps x_{i+1}, in order to increase the semantic and information fusion between branches and different frame rates. This is because numerous feature maps that are very similar and highly redundant are created when the neural network performs feature extraction. In order to expand the spatiotemporal perceptual field of the convolution layer and decrease the number of model parameters and information redundancy, the features after convolution are connected hierarchically with the features before convolution by feature reuse, which was inspired by the Ghost module [31]. It turns out that a 3D convolution kernel's parameters are calculated as follows: $Param1 = C_{in} \times 3^3 \times C_{out}$, The input channel of each branch is taken to be W when the channel C_{in} is split into S groups. After that, $C_{in} = W \times S$. The input channels of the remaining branches of the MSL

become $C_{in} = w + \frac{2^{n-1}-1}{2^{n-1}}w$ since S_1 is immediately spliced. Following that, the MSL parameters are determined as follows:

$$\begin{aligned} Param2 &= \sum_{n=1}^{s-1}\left(w + \frac{2^{n-1}-1}{2^{n-1}}w\right) \times w \times 3^3 \\ &= \left[\sum_{n=1}^{s-1}\left(1 - \frac{1}{2^{n-1}}\right) + S - 1\right] \times w \times w \times 3^3 \\ &< (2S-2) \times w \times w \times 3^3 < S^2 \times w \times w \times 3^3 \\ &= Param1 \end{aligned} \tag{3}$$

where C_{in} and C_{out} are the number of input and output channels, respectively. Eq. (3) shows that following the substitution, the convolution kernel's parameter count decreases in MSL $C_{in} = C_{out}$. The final output Y of the MSL can be written as:

$$Y = cat[x_1, y_{22}, y_{32}, \ldots, y_{(i-1)2}, y_i] \tag{4}$$

The benefits of the MSL are primarily evident in spatial and temporal dimensions, as opposed to the original structure of two fixed convolutional kernel cascades. After multi-level cascading, the 3D convolutional kernel can combine the spatial feature information from the deep and shallow layers of the network and can similarly broaden the range of the feature map perceptual field. Fig. 3a depicts the analogous change process of the MSL structure's spatial perceptual field. The feature maps of all other branches, besides the branch where x_1 is located, are combined using the unconvolved feature maps from this set and the sub-feature maps left over after the prior set of convolution, as can be seen from the figure. The stitched feature maps are once more run through the convolutional neural network, and the receptive field of each branch is corresponding enlarged by increasing the depth of the convolutional neural network of a portion of the channel feature maps at the expense of negligible computational effort. The MSL structure gives the model the ability to efficiently capture the global features of high-level semantics as well as shallow detailed features, considerably enhancing the feature extraction capability of the original network. In addition, the MSL structure allows for the fitting of extended temporal elements in the time dimension. The MSL structure decomposes from the original single convolutional kernel in multiple layers into equivalent three convolutional kernels, and the decomposed convolutional kernels' step sizes are all (1, 1, 1) so that the information of each moment of the output of the convolutional layer fits the featured letter of the corresponding three moments of the previous layer. Each cube produced by conv3 after three convolutional layers equally fit the feature data from seven moments of the input. Fig. 3b illustrates how the convolutional block fits the information equally along the time direction.

The top $3 \times 3 \times 3$ convolutional filter in the original residual structure is kept in order to maintain the fusion capability of the stitched feature maps, resulting in a mixture of standard 3D convolutional kernels and multilayer spatiotemporally separated convolutional kernels connected alternately. Residual mapping is regarded as an effective procedure in standard 3D residual networks. However, the $1 \times 1 \times 1$ convolution with a step size of 2 is typically employed for downsampling when fusing convolutional layers with various feature sizes. Path B of this module employs a maximum pooling operation to preserve more texture information because a downsampling operation of this kind results in three-quarters of the information loss. As path B of the backbone utilizes a pooling layer for downsampling while path A of the backbone uses $3 \times 3 \times 3$ convolution with a step size of

(2, 2, 2), combining soft downsampling and hard downsampling can further increase the use of valuable information. Paths B and C further expand the diversity of timing information by combining short, medium, and long timing features. The main role of path C is to further fuse the feature maps of various convolutional layers. As a result, the final output of the MSF module is: $Output = X + Y + Input$. We discover that the MSF module may effectively increase the performance of model action recognition while minimizing the number of specific parameters. It can not only learn richer spatiotemporal information.

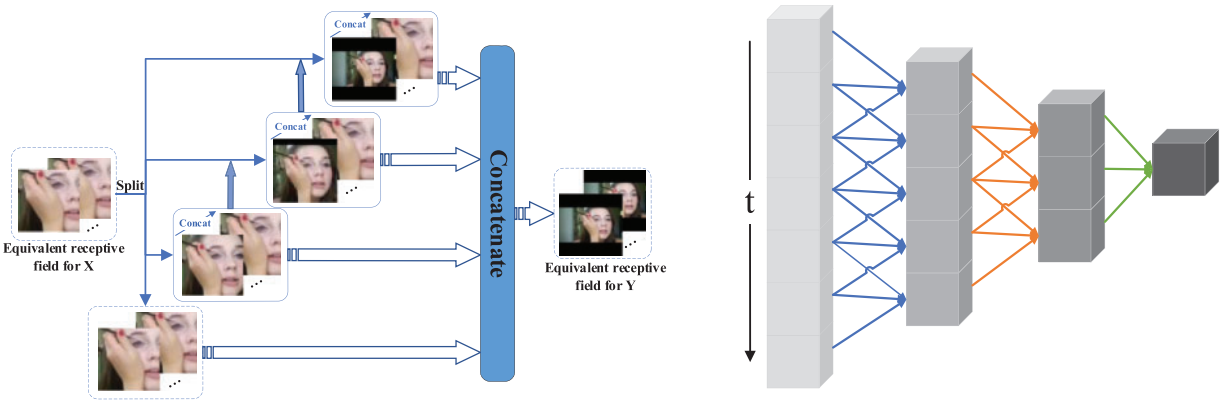


Figure 3: Illustration of spatiotemporal information optimization of MSL structure. (a) Illustration of the equivalent change of spatial receptive field, (b) Equivalent fitting process for time information

3.2 Frequency Channel and Spatiotemporal Attention

The fundamental goal of the attention mechanism is to "dynamically weight" the important areas of interest by autonomously learning a set of weight coefficients through the network. Channel attention, spatial attention, and self-attention are currently the three primary mechanisms of attention. The most prevalent attention mechanism among them is SENet, which has found widespread application in the field of computer vision thanks to its benefits of cheap computing cost and few parameter numbers. However, SENet adopts GAP as the pre-processing method, and this averaging method will lose a lot of useful information. On this basis, Fcanet [13] was created from research into the relationship between DCT and GAP, which demonstrates that GAP is a particular instance of two-dimensional DCT. The Frequency Channel Attention Module (FAM) was so suggested. In other words, the channel attention preprocessing method introduces a two-dimensional DCT transform so that the network may focus on the characteristics of various frequencies.

The structure of FAM is shown in Fig. 4, specifically, the input feature maps are split into many parts along the channel dimension first. Denote $[X^0, X^1, \ldots, X^{n-1}]$ as the parts, in which $X^i \in \mathbb{R}^{C\times H\times w}$, $i \in \{0, 1, \ldots, n-1\}$, $C = \frac{c_{in}}{n}$. After assigning two-dimensional frequency components to each component, splicing will yield the final preprocessing vector:

$$Freq = cat\left(\left[Freq^0, Freq^1, \ldots, Freq^{n-1}\right]\right) \tag{5}$$

In which $Freq \in \mathbb{R}^C$, The whole FAM framework can be written as:

$$Fc\_att = sigmoid\left(fc\left(Freq\right)\right) \tag{6}$$

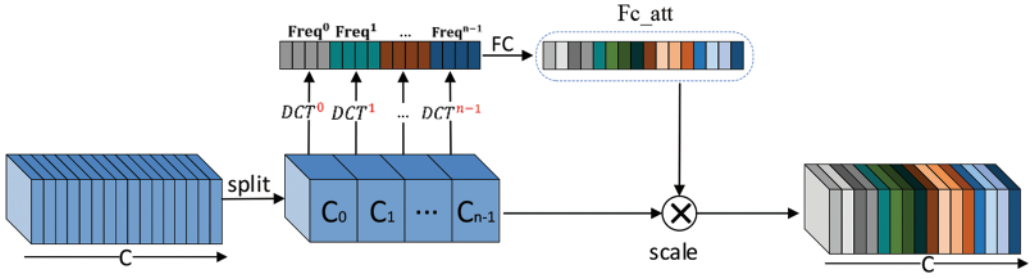


Figure 4: Diagram of FAM architecture

This article extends spatial attention to the time dimension and proposes a Spatiotemporal Attention Module (SAM), as seen in Fig. 5. CBAM is a combination of channel attention and spatial attention. After the global max pooling of channel dimensions and the global average pooling, respectively, the input feature $F \in \mathbb{R}^{C\times T\times H\times w}$ will produce two 3D feature maps: $F_{Max} \in \mathbb{R}^{1\times T\times H\times w}$ and $F_{Avg} \in \mathbb{R}^{1\times T\times H\times w}$. A 3D spatiotemporal attention map is created by concatenating and convolving them using a 3D convolution. The computation of the spatial attention is, in brief:

$$ST\_att = sigmoid\left(f^{7\times7\times7}\left(cat\left[MaxPool\left(F\right), AvgPool(F)\right]\right)\right) \tag{7}$$

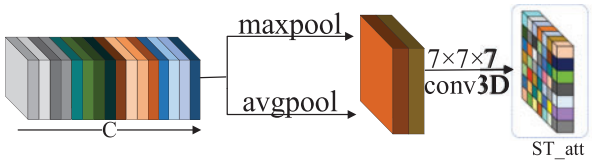


Figure 5: Diagram of SAM architecture

The SAM preserves information on the temporal dimension and can focus on the "when" and "where" of essential elements, whereas the FAM primarily focuses on the "what" of important features. The two types of attention working together can compensate for any crucial information that was lost. As a result, we suggested in this work that the frequency channel and the spatial-temporal attention module be integrated (FSAM).To provide the optimized feature results, feature maps apply the attention mechanism in the channel and spatiotemporal dimensions sequentially. According to Fig. 6, given a feature map $F \epsilon \mathbb{R}^{C\times T\times H\times W}$ as input and fed into FSAM, after outputting the relevant frequency channel attention feature maps, broadcast along the spatiotemporal dimensions, and finally output the attention feature maps through the SAM. The entire attention module can be distilled into the following:

$$Refinedfeature = F \otimes Fc\_att \otimes ST\_att \tag{8}$$

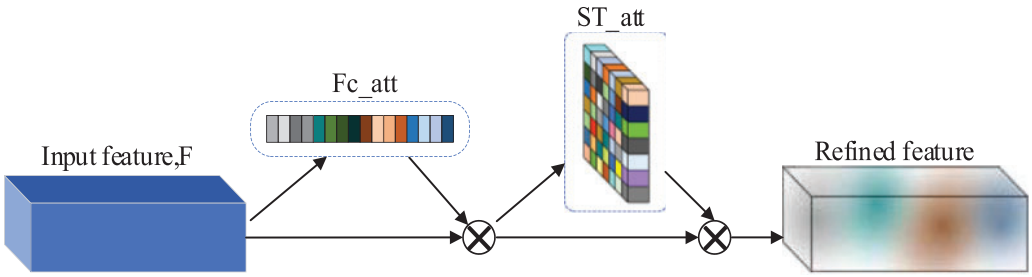


Figure 6: Diagram of FSAM architecture

3.3 The Architecture of MSF-R3D Combined with FSAM

The R3D network model serves as the primary foundation for the module design and technique validation in this article. The MSF module and FSAM make up the majority of the final network model, and Fig. 7 illustrates its precise structure. With the exception of the first layer, which employs $3 \times 7 \times 7$ convolutional kernels with step sizes (1, 2, 2) to extract shallow features, all of the model's convolutional kernels are of size $3 \times 3 \times 3$, and the step size used for the downsampling layer is (2, 2, 2). The first two MSF modules are not downsampled and are padded using padding after convolution in order to preserve more spatiotemporal characteristics. The multi-frequency channel attention can operate successfully when there are a maximum number of channels in the network output layer, which is why the FSMA module is primarily placed in the intermediate and final network output layers. The spatiotemporal attention does, however, play a limited function since the spatial and temporal dimensions of the characteristics at the output are minimized, but if it were to be placed at the network input, it would result in information redundancy. The network model's performance is significantly enhanced by the inclusion of FSAM in the middle of the network model in this study, which effectively solves the issue at hand.

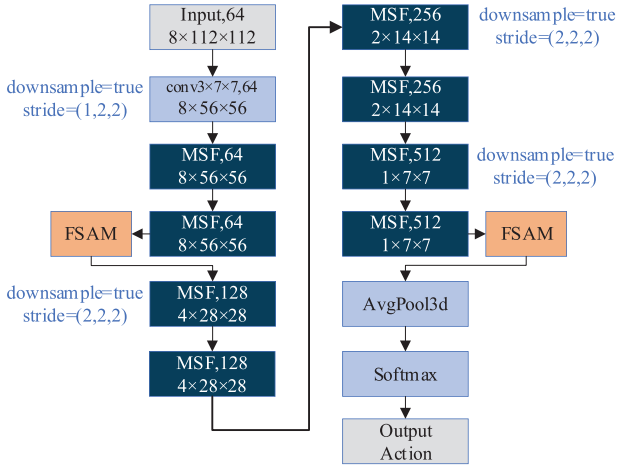


Figure 7: The architecture of MSF-R3D combined with FSAM. The parameters of each layer are represented in the following order: convolution kernel, number of output channels, and output feature map size. Downsample represents the parameters of the first 3D convolution kernel in the MSF module

4 Experiments and Analysis

The experiments in this paper based on the UCF101 dataset are all implemented using the Python 3.6 deep learning framework and an NVIDIA Geforce GTX 1660 GPU hardware platform. The experiments based on the Kinetics-400 dataset are implemented using the Python 3.8 deep learning framework and an NVIDIA GeForce RTX 3090 GPU hardware platform. To demonstrate the efficacy of the network suggested in this paper, a large number of ablation experiments on various models are examined in this section after a description of the relevant dataset, data preprocessing, and experimental parameter configuration. Particularly, the UCF101 dataset served as the foundation for all of our ablation studies.

4.1 Dataset Introduction

We primarily used the dataset UCF101 [32] and Kinetics-400 [33] to test the model. The UCF101 was gathered via YouTube and videotaped in a real environment. The movies are split into 25 groups, each of which has 4−7 different action categories, for a total of 101 different sorts of actions. Because the UCF101 dataset was recorded in a real scenario with camera movements, different lighting conditions, and low-quality frames, it is both representative and demanding for action evaluation. We use the first division scheme between training and test samples supplied by UCF to prevent the recognition accuracy from being too high due to the similarity of features like people and backgrounds in the videos. A sizable dataset called kinetics-400 contains 400 human movements. Each video clip in this collection, which covers at least 400 actions, is at least 10 s long. There are nearly 20 k test movies and about 240 k training videos in the dataset.

4.2 Experimental Settings

In the pre-processing of the dataset, to extract 8 video frames at random intervals, the video in the dataset should first be deconstructed into images frame by frame while maintaining its structural integrity. The image is then randomly cropped to $112 * 112$ size as the input to the network, after being scaled to $171 * 128$ size. Two techniques, level flipping, and contrast enhancement are utilized for data expansion of the training set to avoid overfitting caused by insufficient samples. After preprocessing the dataset, the network is fed $8 * 112 * 112$ images as input during the training phase. Using the Adam optimizer and cross-entropy loss function, the initial learning rate is set to 1e-4 during training, and it is decreased to one-tenth of the original rate for every 10 training epochs.

4.3 Ablation Analysis of the MSF Structure

To verify the efficacy of the suggested strategy, We empirically demonstrate the usefulness of our design modules. The best MSF module is initially chosen in this section, after which we examine the outcomes of various attention mechanisms created using this model and contrast them with the suggested FASM attention.

The MSF Module. We discovered that the MSL at various points in the MSF structure had a substantial effect on the model's ultimate recognition performance during the initial design phase. As can be seen in Fig. 8, we used four different strategies for our comparison trials, (a) demonstrates that MSL is located in the first layer of the MSF module; method (b) demonstrates the method used in this paper and demonstrates that MSL is located in the second layer of the MSF module; method (c) illustrates that the $3 \times 3 \times 3$ convolutional kernel in the downsampling layer is retained to prevent information loss and that all other convolutional kernels are decomposed into MSL; The first convolutional kernel in the residual block is divided into Parts using method (d), which is based on the approach described in this research, to further cut down the number of parameters needed to

create cascaded patterns of temporal and spatial convolution. Table 1 displays the outcomes of the comparison experiments.

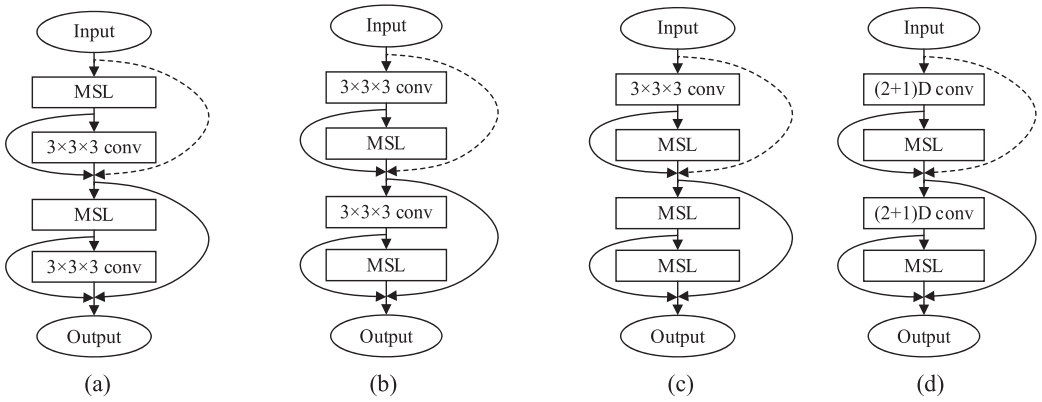


Figure 8: Scheme for multilevel decomposition with different convolution kernels

Table 1: Experiments comparing multilevel-split convolution filters at various positions (%). Where the scale parameter S = 4

Method	Parameter	Accuracy
(a)	26.66 M	48.0
(b)	21.82 M	49.5
(c)	16.09 M	47.1
(d)	15.53 M	45.2

Method (b), as indicated in Table 1, eliminates some of the information loss by keeping the 3 × 3 × 3 convolution of the downsampling layer. Meanwhile, the 3D convolution layer after MSL may complete the information interaction between channels and fulfill the function of spatiotemporal feature fusion, resulting in the best recognition effect of the four architectures.

Since channel grouping of input features is the primary operation of multi-scale spatiotemporal feature separation, channel segmentation parameter S is introduced. Table 2 shows the experimental findings of ablation tests on the selection of channel grouping coefficient S.

Table 2: Experimental results of different S (%)

Setting	Parameter	Accuracy
S = 3	21.18 M	46.0
S = 4	21.84 M	49.5
S = 5	21.80 M	48.0
S = 6	21.50 M	47.6

When S = 2, the spatiotemporal feature map is separated into two groups based on channel, with one group being immediately spliced and output and the other being output after convolution. There

is no information interaction between channels, and feature reuse accounts for half of the output's spatiotemporal features. As a result, the scaling coefficient is not tested in this part. As can be seen in Table 3, when S = 4, the network's test performance is the greatest, and as S increases, the accuracy rate declines. When S > 4, the receptive field of the last set of channels is larger than $9 \times 9 \times 9$, and the image begins to be smaller than the effective receptive field of this layer of convolution, resulting in a fall in recognition accuracy.

Table 3: Action recognition performance comparison of various attention on the UCF101 (%)

Attention	Parameter	Accuracy
SE-Net [12]	21.88 M	50.1
CBAM [26]	21.88 M	45.5
NST [29]	21.85 M	48.2
FSAM	21.88 M	52.8

The FSAM Attention. This section primarily compares experiments with conventional channel attention, CBAM, and embedded space-time attention mechanisms since the FSAM introduced in this study is improved by merging channel and spatial attention. To assure the fairness of the experimental comparison, the placements and quantities of different attention modules are the same, and the experimental results are provided in Table 3.

As shown in Table 3, SENet performs well in terms of recognition, but CBAM performs worse due to the loss of most spatial-temporal data during the 3D convolution process and the retention of only low-frequency components in the channel attention dimension. The inter-channel dependency is disregarded by NST. We can deduce from the statistics above that inter-channel information reliance is significantly more significant in action recognition than spatiotemporal information. Consequently, our method adds a frequency channel attention before the spatiotemporal attention, which can focus on keyframes and significant information in the video in addition to the location of critical features and achieves the best recognition effect when compared to other attention methods.

4.4 Comparisons with Other Methods

Table 4 displays the outcomes of our comparative tests using the R3D-based model on the Kinetics-400 dataset. Additionally, the MSF module suggested in this paper is extended to the C3D network in order to further confirm the generalization of the network structure presented in this paper. The actual implementation is as follows: the test experiment is run on UCF101 after replacing the $3 \times 3 \times 3$ convolution of two cascades in the C3D network with the MSF module and leaving the rest of the network components unchanged. The adoption of zero training is caused by the fact that not all of them employ pre-training weight. Simultaneously, the residual block of the R3D-18 network was replaced by the MSF module for testing in order to demonstrate the effectiveness of the MSF module and FSAM, respectively. Thereafter, the FSAM suggested in this study was incorporated on the basis of MSF-R3D. The same experiment was carried out on the basis of the C3D network to confirm the generalizability of the method described in this research. In order to ensure that R3D and C3D comparisons are fair, the number of network layers was not expanded following these advancements.

Table 4: Comparison of action recognition on the Kinetics-400 (%)

Method	GFLOPs	Parameter	Accuracy
C3D	38.5	78.41 M	56.1
Two-stream CNN [22]			61.0
[34]	254	–	**72.5**
3D-TDC(101) [35]	19.22	–	67.5
STILT(VGG-19) [36]	–	–	64.8
MSF-R3D	**15.1**	**21.84 M**	66.7
MSF-R3D+FSAM	**15.1**	21.88 M	67.8

As seen in Table 5, our method outperforms both the traditional network and more contemporary networks in terms of recognition performance. Although the MSF-R3D+FSAM model is less accurate at recognizing objects on the Kinetics-400 dataset than the top model, it is simpler to train and requires just one-sixteenth of the floating-point calculation. Moreover, each MSF structure and the FSAM has the ability to enhance network performance. We not only lower the number of parameters by 34% after using our method on the R3D network structure, but we also increase recognition accuracy by 7.2% on the UCF101 dataset. We used our technique in the C3D network model for generalization ability testing to further strengthen the confidence of the experimental results. The test results reveal that the recognition accuracy is enhanced by 8.9%, totally reflecting the universality of our method. It is possible that our approach can be adaptably incorporated into any 3DCNN architecture to reduce the overall number of parameters while enhancing recognition accuracy.

Table 5: Comparison of action recognition on the UCF101 (%). The 'Accuracy' column means the accuracy of action recognition, and figures in '( )' indicate the accuracy rate of reproduction based on the equipment in this paper

Method	GFLOPs	Parameter	Accuracy
C3D(baseline)	38.5	78.41 M	44.9
R3D(baseline)	19.3	33.18 M	45.6
R(2+1)D [21]	152.4	33.23 M	(41.6)
3Dresnet50 [37]	10.1	47.02 M	45.52
Dense-3D [38]	**8.4**	18.8M	47.3
3D-MobileNetV3(CBAM) [39]		**5.31 M**	50.36
SR3D [40]	44.3	34.5 M	47.7
MSF-R3D	15.1	21.84 M	49.5
MSF-R3D+FSAM	15.1	21.88 M	52.8
MSF-C3D	32.9	69.20 M	52.1
MSF-C3D+FSAM	32.9	69.25 M	**53.8**

5 Conclusion

The action recognition network model we propose in this paper combines FSAM and MSF structure. Initially, by equally increasing the spatiotemporal perceptual field of convolution through hierarchical class residual connections, the MSF structure is used to achieve the merging of multi-scale spatial features and short-medium-length temporal information. This study also suggests a new attention module called FSAM, which primarily increases attention to keyframes and significant information in video sequences while extending preprocessing of channel attention to the frequency domain to explicitly model the interdependence between channels of convolutional features. In order to implement an end-to-end framework for action recognition, the suggested multi-level spatiotemporal feature fusion module and attention mechanism are integrated into the 3DCNN architecture.

The R3D network design is used to validate the method first, and the results demonstrate that performance may be improved while using fewer model parameters. Using the same dataset, the approach is applied to the C3D network architecture to evaluate its generalizability, which likewise greatly enhances the network's performance at recognition. It is clear that the proposed method may be used with any 3D CNN architecture, demonstrating its enormous potential. On small and medium-sized datasets, this model performs better, but on large datasets, it only slightly improves recognition accuracy. The top-performing MSF-C3D+FSAM, however, includes a significant number of parameters that are challenging to use in real-world scenarios. Consequently, a key issue we will need to consider in the future is how to better balance the model's parameter count and action recognition accuracy.

Acknowledgement: I want to thank everyone who contributed to the research and let them know how much I appreciate it. Firstly, my supervisor gave me advice on how to write the research questions and techniques in a professional manner. Also, I would want to express my gratitude to the lab senior, who has been a tremendous help with the article's composition and proofreading. Without their assistance, I would not have been able to complete this dissertation.

Funding Statement: This work was supported by the General Program of the National Natural Science Foundation of China (62272234); the Enterprise Cooperation Project (2022h160); and the Priority Academic Program Development of Jiangsu Higher Education Institutions Project.

Author Contributions: The authors confirm contribution to the paper as follows: study conception and design: Mengmeng Yan, Chuang Zhang; data collection: Mengmeng Yan, Chuang Zhang; analysis and interpretation of results: Mengmeng Yan, Jinqi Chu, Haichao Zhang; draft manuscript preparation: Mengmeng Yan, Chuang Zhang, Haichao Zhang, Tao Ge, Suting Chen. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The data used to support the findings of this study are included in the article.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] S. Yuan, Y. J. Qiao, H. Su, Q. H. Chen and X. Liu, "A review of behavior recognition methods based on deep learning," *Microelectronics & Computer*, vol. 39, no. 8, pp. 1–10, 2022.

[2] Z. Sun, Q. Ke, H. Rahmani, M. Bennamoun, G. Wang *et al.,* "Human action recognition from various data modalities: A review," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 3, pp. 3200–3225, 2023.

[3] M. M. Islam, S. Nooruddin, F. Karray and G. Muhammad, "Multi-level feature fusion for multimodal human activity recognition in internet of healthcare things," *Information Fusion*, vol. 94, no. 12, pp. 17–31, 2023.

[4] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar *et al.,* "Large-scale video classification with convolutional neural networks," in *Proc. CVPR*, Columbus, OH, USA, pp. 1725–1732, 2014.

[5] D. K. Vishwakarma and T. Singh, "A visual cognizance based multi-resolution descriptor for human action recognition using key pose," *International Journal of Electronics and Communications*, vol. 107, no. 3, pp. 513–521, 2019.

[6] S. Herath, M. Harandi and F. Porikli, "Going deeper into action recognition: A survey," *Image and Vision Computing*, vol. 60, no. 2, pp. 4–21, 2017.

[7] D. Tran, L. Bourdev, R. Fergus, L. Torresani and M. Paluri, "Learning spatiotemporal features with 3D convolutional networks," in *Proc. ICCV*, Santiago, Chile, pp. 4489–4497, 2015.

[8] K. He, X. Zhang, S. Ren and J. Sun, "Deep residual learning for image recognition," in *Proc. CVPR*, Las Vegas, NV, USA, pp. 770–778, 2016.

[9] H. C. Zhang and C. Zang, "Research on lightweight action recognition networks integrating attention," *Journal of Electronic Measurement and Instrumentation*, vol. 36, no. 5, pp. 173–179, 2022.

[10] C. Yang, Y. Xu, J. Shi, B. Dai and B. Zhou, "Temporal pyramid network for action recognition," in *Proc. CVPR*, Seattle, WA, USA, pp. 588–597, 2020.

[11] D. Tran, H. Wang, M. Feiszli and L. Torresani, "Video classification with channel-separated convolutional networks," in *Proc. ICCV*, Seoul, Korea (South), pp. 5551–5560, 2019.

[12] J. Hu, L. Shen and G. Sun, "Squeeze-and-excitation networks," in *Proc. CVPR*, Salt Lake City, UT, USA, pp. 7132–7141, 2018.

[13] Z. Qin, P. Zhang, F. Wu and X. Li, "FcaNet: Frequency channel attention networks," in *Proc. ICCV*, Montreal, QC, Canada, pp. 763–772, 2021.

[14] H. Wang and C. Schmid, "Action recognition with improved trajectories," in *Proc. ICCV*, Sydney, NSW, Australia, pp. 3551–3558, 2013.

[15] H. Wang, A. Kläser, C. Schmid and C. L. Liu, "Action recognition by dense trajectories," in *Proc. CVPR*, Colorado Springs, CO, USA, pp. 3169–3176, 2011.

[16] Y. Zhu, J. K. Zhao, Y. N. Wang and B. B. Zheng, "A review of human action recognition based on deep learning," *Acta Automatica Sinica*, vol. 42, no. 6, pp. 848–857, 2016.

[17] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Proc. NIPS*, Cambridge, MA, USA, MIT Press, pp. 568–576, 2014.

[18] S. Ji, W. Xu, M. Yang and K. Yu, "3D convolutional neural networks for human action recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 221–231, 2013.

[19] D. Tran, R. Jamie, S. Zheng, C. Shih-Fu and P. Manohar, "ConvNet architecture search for spatiotemporal feature learning," arXiv preprint arXiv: 1708.05038, 2017.

[20] Z. Qiu, T. Yao and T. Mei, "Learning spatio-temporal representation with Pseudo-3D residual networks," in *Proc. ICCV*, Venice, Italy, pp. 5534–5542, 2017.

[21] D. Tran, H. Wang, L. Torresani, J. Ray and Y. LeCu, "A closer look at spatiotemporal convolutions for action recognition," in *Proc. CVPR*, Salt Lake City, UT, USA, pp. 6450–6459, 2018.

[22] J. Carreira and A. Zisserman, "Quo Vadis, action recognition? A new model and the kinetics dataset," in *Proc. CVPR*, Honolulu, HI, USA, pp. 4724–4733, 2017.

[23] M. M. Islam, S. Nooruddin, F. Karray and G. Muhammad, "Human activity recognition using tools of convolutional neural networks: A state of the art review, data sets, challenges, and future prospects," *Computers in Biology and Medicine*, vol. 149, pp. 1–20, 2022.

[24] V. Mnih, N. Heess, A. Graves and K. Kavukcuoglu, "Recurrent models of visual attention," in *Proc. NIPS*, Cambridge, MA, USA, MIT Press, pp. 2204–2212, 2014.

[25] D. Bahdanau, K. Cho and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *Proc. ICLR*, San Diego, United States, 2015.

[26] S. Woo, J. Park, J. Y. Lee and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. ECCV*, Munich, Germany, pp. 3–19, 2018.

[27] J. Park, S. Woo, J. Y. Lee and I. S. Kweo, "BAM: Bottleneck attention module," in *Proc. BMVC*, Newcastle, UK, 2018.

[28] M. Li, D. J. Ning and J. C. Guo, "Attention mechanism-based CNN-LSTM model and its application," *Computer Engineering and Applications*, vol. 55, no. 13, pp. 20–27, 2019.

[29] J. Li, P. Wei and N. Zheng, "Nesting spatiotemporal attention networks for action recognition," *Neurocomputing*, vol. 459, no. 6, pp. 338–348, 2021.

[30] C. Szegedy, W. Liu, Y. Q. Jia, P. Sermanet, S. Reed *et al.,* "Going deeper with convolutions," in *Proc. CVPR*, Boston, MA, USA, pp. 1–9, 2015.

[31] K. Han, Y. Wang, Q. Tian, J. Guo, C. Xu *et al.,* "GhostNet: More features from cheap operations," in *Proc. CVPR*, Seattle, WA, USA, pp. 1577–1586, 2020.

[32] K. Soomro, A. R. Zamir and M. Shah, "UCF101: A dataset of 101 human actions classes from videos in the wild," arXiv preprint arXiv: 1212.0402, 2012.

[33] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier *et al.,* "The kinetics human action video dataset," arXiv preprint arXiv: 1705.06950, 2017.

[34] Y. Zhou, X. Sun, C. Luo, Z. -J. Zha and W. Zeng, "Spatiotemporal fusion in 3D CNNs: A probabilistic view," in *Proc. CVPR*, Seattle, WA, USA, pp. 9826–9835, 2020.

[35] Y. Ming, F. Feng, C. Li and J. H. Xue, "3D-TDC: A 3D temporal dilation convolution framework for video action recognition," *Neurocomputing*, vol. 450, no. 11, pp. 362–371, 2021.

[36] T. Liu, Y. Ma, W. Yang, W. Ji, R. Wang *et al.,* "Spatial-temporal interaction learning based two-stream network for action recognition," *Information Sciences*, vol. 606, no. 3, pp. 864–876, 2022.

[37] K. Hara, H. Kataoka and Y. Satoh, "Towards good practice for action recognition with spatiotemporal 3D convolutions," in *Proc. ICPR*, Beijing, China, pp. 2516–2521, 2018.

[38] G. Li, X. Liu and G. H. Gu, "Three-dimensional convolution dense network for action recognition," *China Science Paper*, vol. 13, no. 14, pp. 1634–1638+1663, 2018.

[39] X. G. Hu, "Research on algorithm of lightweight human action recognition based on video," M.S. Thesis, University of Science and Technology of China, China, 2021.

[40] P. F. Xu, P. C. Zhang, Y. H. Liu and S. F. Ge, "A human action recognition algorithm based on SR3D network," *Computer Knowledge and Technology*, vol. 18, no. 1, pp. 10–11, 2022.