Available Online: 28 May 2025 DOI: 10.54254/2977-3903/2025.23586

A human action recognition method based on spatiotemporal information interaction

Tian Wei

Shangluo University, Shangluo, China

1908790127@qq.com

Abstract. In the field of deep learning, current human action recognition algorithms often treat temporal information, spatial information, and background information equally, which leads to limited recognition accuracy. To address this issue, this paper proposes a human action recognition algorithm based on spatiotemporal information interaction. First, a dual-pathway network is proposed to learn spatial and temporal information at different refresh rates. The network includes a sparse pathway operating at a low frame rate to capture spatial semantic information, and a parallel dense pathway operating at a high frame rate to capture temporal motion information. Second, to extract more discriminative features from videos, a cross-dual attention interaction model is introduced to focus on key regions of video segments and explicitly exchange spatiotemporal information between the two pathways. Experimental results show that the proposed algorithm achieves recognition accuracies of 97.6% on the UCF101 dataset and 78.4% on the HMDB51 dataset, outperforming the novel SlowFast algorithm by 1.8% and 1.4%, respectively. Combined with a nighttime image enhancement algorithm based on MDIFE-Net curve estimation, the method achieved an accuracy of 83.2% on the ARID nighttime dataset—an improvement of 22.9% over the performance before image enhancement. This demonstrates the method's strong potential for real-world nighttime action recognition applications.

Keywords: image enhancement, action recognition, illumination curve estimation, spatiotemporal information interaction, attention mechanism

1. Introduction

Action recognition aims to detect human activities in video footage and has broad application prospects in fields such as smart homes, intelligent security, human-computer interaction, and video retrieval. With the deepening of research in deep learning models, the accuracy of human action recognition models based on deep learning now significantly surpasses that of early traditional algorithms. These models simulate the human nervous system using bionic knowledge, automatically extract image features, and perform classification through end-to-end multi-round training.

1.1. Action recognition algorithms based on deep learning

Currently, deep learning-based human action recognition algorithms can be broadly classified into the following categories: two-stream convolutional-based algorithms, 3D Convolutional Neural Networks (3DCNN), and algorithms based on Long Short-Term Memory (LSTM) networks.

(1) Two-Stream Convolutional Action Recognition Algorithms: Simonyan et al. were the first to propose the two-stream convolutional network model in the field of video action recognition [1]. This model effectively extracts and aggregates spatial and temporal features by processing spatial information using RGB video frames and temporal information using optical flow frames. The spatial and optical flow prediction results obtained from the two-stream convolutional network are fused to generate the final classification output. Although this model achieves promising results on both public and small-sample datasets, its computational cost is high. To address this issue, Chen et al. proposed a lightweight and efficient neural network based on Graph Convolutional Networks (GCN), named NLB-ACSE [2]. This model consists of two branches: a non-local block branch focusing on long-range features and an adaptive cross-spatiotemporal edge branch focusing on short-range features. Both branches extract information across time and space, focusing respectively on long-term and short-term dependencies, ultimately achieving promising results. Ji et al. introduced the Temporal Shift Module (TSM), a model based on the two-stream structure that improves

video action recognition performance by enhancing convolutional neural networks [3]. The temporal shift operation extends to the video understanding domain by shifting certain channels along the temporal axis and combining information from neighboring frames with the current frame. This allows for temporal modeling through the exchange of channel information between adjacent frames and has led to state-of-the-art performance. Feichtenhofer et al. proposed a SlowFast network model [4], which integrates a slow, high-resolution channel for extracting static spatial features with a fast, low-resolution channel for capturing dynamic temporal features. These two parallel convolutional networks operate on the same video segment, achieving significant performance improvements. Pang et al. introduced a novel deep network model [5], the GCN-Transformer Network (ConGT), which integrates spatial and temporal modules in parallel. ConGT comprises two parallel streams: the Spatiotemporal Graph Convolution stream (STG), which aims to preserve the natural topological structure of human skeletons, and the Spatiotemporal Transformer Stream (STT), which captures global relationships between human joints. Additionally, a contrastive learning paradigm is introduced to enhance action features by maximizing mutual information between the two streams. A Cyclical Focal Loss (CFL) is also proposed to focus on confident samples in early training and hard samples in mid-training stages. These contributions result in state-of-the-art performance.

(2) Action Recognition Algorithms Based on 3D Convolutional Neural Networks (3DCNN): Since video data contain both spatial and temporal dimensions, action recognition in videos necessitates consideration of temporal information in addition to spatial features, unlike static 2D images. Owing to its innovative temporal operations, 3D convolutional models have been widely applied in the domain of video-based action recognition. Tran et al. proposed a 3D Convolutional Network (C3D) for action recognition, which extracts spatiotemporal features from video clips using 3DCNN, thereby enhancing recognition accuracy [6]. This model achieved promising results on the UCF101 dataset. However, as the network depth increased, the model's error rate also rose. To address the problems of gradient explosion and vanishing during backpropagation in deep networks, Tran et al. [7] further introduced the Res3D network, which not only reduced the number of parameters by half compared to C3D but also achieved better performance, effectively alleviating the gradient-related issues caused by deeper architectures. Carreira et al. [8] proposed the Inflated 3D Convolutional Network (I3D), which integrates 3D convolutions with the two-stream approach. Using video clips as input, I3D stacks multiple 3D convolutional layers to perform action recognition, yielding significant results on public datasets. Christoph et al. [9] extended the depth and width of existing models and optimized image resolution and parameters to propose the eXpand3D Convolutional Network (X3D), which achieved excellent results with minimal computational cost. Li et al. [10] developed a residual network model incorporating multi-scale feature fusion and global average pooling. First, a multi-scale feature extraction module is used to obtain features at various scales, thereby enriching spatiotemporal information. Then, a global average pooling layer is adopted at the network's output stage in place of fully connected layers to reduce excessive parameters, ultimately resulting in superior performance. Although 3DCNNs can directly extract spatiotemporal information from RGB images or video segments, their high computational cost and parameter complexity make them less practical in real-world applications. Furthermore, their increased parameter counts often lead to overfitting. Consequently, many hybrid models have been proposed to balance spatiotemporal modeling capability and computational efficiency.

(3) Action Recognition Algorithms Based on Long Short-Term Memory (LSTM): Compared with convolution-based methods, Recurrent Neural Networks (RNNs) incorporate hidden state data from previous time steps into current computations, effectively preserving temporal dependencies and making them more suitable for sequential data. Donahue et al. [11] combined the strengths of CNNs in image processing with the sequential modeling capability of LSTMs by first using a CNN to extract spatial features from video clips and then feeding these features into an LSTM to model temporal dynamics, eventually fusing both to determine the action class. Si et al. [12] utilized Graph Convolutional Networks (GCNs) to extract spatial features and LSTMs to model temporal dynamics, achieving strong performance. However, these methods treat spatial and temporal information separately and fail to capture their joint representations. To overcome this limitation, Li et al. [13] proposed a method that integrates convolutional operations with LSTM networks. Recognizing the pixel-level correlations between adjacent video frames, they replaced the standard LSTM operations with convolutional ones, allowing the model to simultaneously capture spatial and temporal features, leading to better results. While RNNs offer unique advantages in handling temporal data, they are prone to gradient vanishing or explosion. Bidirectional LSTM (BiLSTM) networks address this issue by using memory cells to store long-term dependencies. Aljarrah et al. [14] proposed a BiLSTM-based action recognition model combined with Principal Component Analysis (PCA) for dimensionality reduction, achieving high recognition accuracy. Wang et al. [15] proposed a two-stream BiLSTM architecture that extracts spatiotemporal features from video frames. They introduced an optical flow discrimination loss function to better capture changes in motion, and ultimately achieved high recognition performance on public datasets.

1.2. Action recognition algorithms integrating attention mechanisms

In addition to the above, attention mechanisms have been widely applied across various tasks, especially in image classification and video analysis. Due to their ability to enhance model performance with minimal additional parameters, many researchers have begun incorporating attention mechanisms into action recognition frameworks. Attention modules are capable of capturing long-range spatial dependencies and long-duration temporal relationships. By embedding attention mechanisms into networks that facilitate mutual learning between components, the accuracy and performance of video action recognition can be significantly improved. Vaswani et al. [16] introduced the Transformer, a neural network architecture based entirely on attention mechanisms, replacing traditional RNNs and CNNs. Wang et al. [17] proposed a non-dimensionality-reducing local inter-channel interaction strategy based on fast 1D convolution, and designed an adaptive method to select kernel sizes to determine the local receptive field

for inter-channel interactions. Woo et al. [18] proposed a lightweight and generic attention module that sequentially applies attention in the channel and spatial dimensions. This module reallocates weights for both spatial and channel features and multiplies them with the original feature maps, implementing a self-adaptive feature weighting scheme. Fu et al. [19] proposed a dual-attention mechanism to model and fuse both local and global features in the spatial and temporal domains. Pan et al. [20] proposed a co-attention mechanism to address temporal misalignment. Liu et al. [21] focused on temporal weight modeling and enhanced crucial motion features through channel attention mechanisms, thereby supplementing temporal contextual information. Wang et al. [22] proposed a method based on searching for optimal spatiotemporal attention units, which can be inserted at arbitrary positions within a network to enhance spatiotemporal features. Experimental results demonstrated that this method significantly improved video understanding performance.

2. Model overview

2.1. Spatiotemporal interaction network concept

There are two primary challenges in recognizing human actions from video clips: extracting discriminative features and leveraging temporal information to model dynamic changes in actions throughout the video. The Two-Stream Network addresses these by proposing a dual-branch architecture that extracts temporal and spatial features separately. One branch receives individual RGB frames to capture spatial information, while the other receives multiple optical flow frames to capture temporal information. Together, they represent the complete information of a video sequence, allowing for action recognition through the fusion of spatial and temporal features. However, optical flow relies on handcrafted features, which are not conducive to end-to-end learning. This separation of spatial and temporal processing has inspired new directions in the design of video action recognition models.

Compared to static images, videos are primarily different in the temporal dimension. When handling video data, it is usually represented in the form of (x, y, t), where t denotes time. Since variations along the temporal and spatial axes do not occur with equal probability, spatial and temporal information cannot be treated identically. Spatial changes tend to be visually gradual, leading to slower sampling and recognition, whereas action changes often occur much more rapidly than changes in the appearance of the subject. For example, under a fixed background, a person's action category may be "walking," but during the video, the action may shift to "running," "jumping," or "climbing," while the person's appearance remains largely unchanged. This implies that the appearance of the object does not require a high refresh rate, whereas action recognition does. Inspired by this observation, this study proposes a dual-path network that processes temporal and spatial information separately at different refresh rates for better spatiotemporal modeling of actions in videos.

In deep learning, integrating attention mechanisms into network architectures allows the model to intelligently focus on relevant regions during recognition tasks [23], thereby enhancing performance and offering high research value. In the domain of action recognition, adaptively learning specific temporal and spatial features remains a significant challenge. To extract more discriminative features from images and suppress irrelevant background noise, attention mechanisms are introduced to emphasize important areas in video frames, ultimately improving model accuracy.

To address the above challenges, this chapter proposes a human action recognition algorithm based on spatiotemporal information interaction. The innovations are summarized as follows:

(1) A dual-path network is proposed to extract spatiotemporal features. A sparse path captures spatial semantic information using low frame rates and slow refresh speeds, while a parallel dense path captures rapidly changing motion information using high frame rates and fast refresh speeds.

(2) A Cross Dual-Attention Interaction Module is proposed to focus attention on critical regions within video clips and facilitate explicit exchange of spatiotemporal information between the two paths. A 3D spatiotemporal self-attention mechanism is employed to compute attention maps from the sparse path to the dense path, while a 3D efficient channel attention mechanism assigns importance weights across channels. The final spatiotemporal features are fused via a cross-connected structure.

2.2. Description of the spatiotemporal interaction network model

This section presents the spatiotemporal interaction network model, as illustrated in Figure 1. First, the video is input into both the sparse and dense paths, which process data simultaneously. The sparse path samples video frames with a large temporal stride, extracting spatial features at a low frame rate and slow refresh speed. The dense path samples frames with a small temporal stride to extract temporal features at a high frame rate and fast refresh speed. Meanwhile, the Cross Dual-Attention Interaction Module focuses on key areas of video clips. After four rounds of spatiotemporal interaction, the spatial and temporal features are fused. The fused features are passed through a Softmax function for action classification and labeling. The following subsections will detail the key components of the proposed algorithm: the dual-path network and the cross dual-attention interaction module.

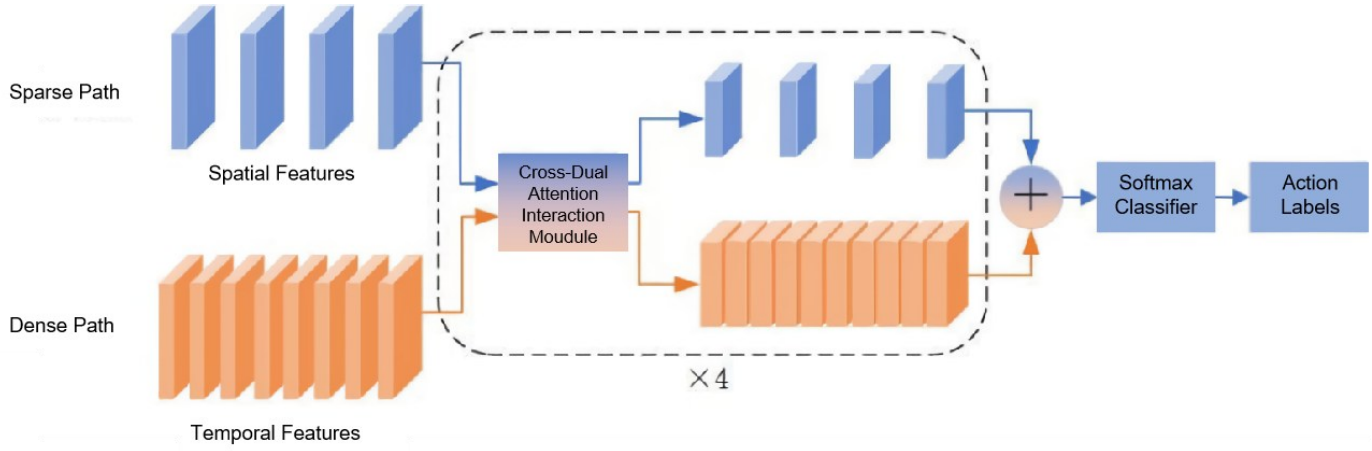


Figure 1. Cross dual-attention interaction module

2.2.1. Dual-pathway network

Feichtenhofer et al. discovered that RGB video data can be divided into two distinct components [24]: slowly changing static regions and rapidly changing dynamic regions. This finding suggests that two different types of features can be extracted from the same RGB video data. Inspired by this idea, we propose a dual-pathway network for action recognition. The network consists of two branches: a sparse pathway that extracts spatial semantic features at a low frame rate and slow refresh rate, and a dense pathway that captures temporal motion features at a high frame rate and fast refresh rate.

The initial pathway allocation strategy draws on insights from bionics. Studies have shown that in the visual systems of primates [25], approximately 80% of retinal ganglion cells are parvocellular (P-cells), while the remaining 20% are magnocellular (M-cells). M-cells exhibit high dynamic temporal processing capabilities, allowing them to effectively perceive temporal changes, but they are not sensitive to spatial detail. In contrast, P-cells are highly effective at processing static spatial information but lack accuracy in capturing temporal changes. Inspired by this biological analogy, the dual-pathway network is designed to emulate this division: the dense pathway is analogous to M-cells and the sparse pathway to P-cells. The concepts of cell size and quantity are mapped to frame sampling rates and channel numbers, respectively. Enhancing the temporal modeling ability (frame rate) of one branch while reducing its spatial modeling complexity (channel width) not only reduces computational load but also allows each branch to specialize in its respective role.

The 3DResNet architecture is employed as the backbone network due to its ability to extract mixed spatiotemporal features and its use of skip connections to alleviate the vanishing gradient problem [25]. Both pathways use the same backbone architecture but operate at different temporal speeds. The specific design is as follows:

1. The input is low-frame-rate video data. This branch operates in a spatiotemporal convolutional manner and primarily captures spatial semantic information. It runs at a slow refresh rate with a large temporal stride. For every τ frames in the input video ($\tau > 1$), only one frame is processed. In our design, τ is set to 16. If the sparse pathway samples T frames, then the total length of the original video clip is $T \times \tau$ frames.
2. The dense pathway operates in parallel with the sparse pathway, processing the same video segment but with high-frame-rate input. Its primary function is to capture rapidly changing temporal motion information, operating at a fast refresh rate and small temporal stride. The input frame stride for this branch is defined as τ/α ($\alpha > 1$), where α denotes the ratio of the dense pathway's frame rate to that of the sparse pathway. In this study, α is set to 8. Unlike the sparse pathway, the dense pathway does not require strong spatial semantic modeling capabilities. Therefore, the number of channels in this branch is reduced to β times that of the sparse pathway ($\beta < 1$), and we set $\beta = 1/8$. Although the dense pathway extracts frames at a higher frequency, the reduced number of channels significantly lowers its computational cost. As a result, it consumes only about 20% of the total computational load, ensuring that the sub-network remains lightweight. The parameter configuration of the spatiotemporal interaction network is presented in Table 1. The spatial-temporal dimensions are denoted as $T \times S^2$ where T is the temporal duration and S is the side length of the spatial crop.

Table 1. Parameters of the spatiotemporal interaction network

Stage	Sparse Pathway (3DResNet50)	Dense Pathway (3DResNet50)	Output Size
Raw Input	-	-	64×224^2
Sampling	Stride16, 1^2	Stride 2, 1^2	Sparse: 4×224^2 Dense: 32×224^2
Conv Layer	1×7^2, 64 Stride 1, 2^2	5×7^2, 8 Stride 1, 2^2	Sparse: 4×112^2 Dense: 32×112^2
Pooling	MaxPool 1×3^2 Stride 1, 2^2	MaxPool 1×3^2 Stride 1, 2^2	Sparse: 4×56^2 Dense: 32×56^2
Residual Block 1	$\begin{bmatrix}1\times1^2,64\\1\times3^2,64\\1\times1^2,256\end{bmatrix}\times3$	$\begin{bmatrix}3\times1^2,8\\1\times3^2,8\\1\times1^2,32\end{bmatrix}\times3$	Sparse: 4×56^2 Dense: 32×56^2
Cross Dual-Attention Interaction Module			
Residual Block 2	$\begin{bmatrix}1\times1^2,128\\1\times3^2,128\\1\times1^2,512\end{bmatrix}\times3$	$\begin{bmatrix}3\times1^2,16\\1\times3^2,16\\1\times1^2,64\end{bmatrix}\times3$	Sparse: 4×28^2 Dense: 32×28^2
Cross Dual-Attention Interaction Module			
Residual Block 3	$\begin{bmatrix}3\times1^2,256\\1\times3^2,256\\1\times1^2,1024\end{bmatrix}\times3$	$\begin{bmatrix}1\times1^2,32\\1\times3^2,32\\1\times1^2,128\end{bmatrix}\times3$	Sparse: 4×14^2 Dense: 32×14^2
Cross Dual-Attention Interaction Module			
Residual Block 4	$\begin{bmatrix}3\times1^2,512\\1\times3^2,512\\1\times1^2,2048\end{bmatrix}\times3$	$\begin{bmatrix}1\times1^2,64\\1\times3^2,64\\1\times1^2,256\end{bmatrix}\times3$	Sparse: 4×7^2 Dense: 32×7^2
Cross Dual-Attention Interaction Module			
Global Avg Pooling, Feature Fusion, Fully Connected Layer			#Action Categories

The input to the network consists of RGB video frames with a resolution of 64×224^2. A convolutional layer with a stride of 16×12 is applied to the input frames to produce feature maps of size 4×224^2 for the sparse pathway. Simultaneously, a convolutional layer with a stride of 2×12 is applied to produce 32×224^2 feature maps for the dense pathway. The next convolutional layer for the sparse pathway uses 1×7 kernels with 64 channels and a stride of 1×2, resulting in output feature maps of size 4×112^2. For the dense pathway, 5×7 kernels with 8 channels and the same stride (1×2) are applied, producing 32×112^2 feature maps. Following this, both pathways apply max pooling layers with a kernel size of 1×3 and a stride of 1×2, downsampling the output feature maps to 4×56^2 for the sparse pathway and 32×56^2 for the dense pathway.

Residual Block 1 consists of three residual units, each comprising three convolutional layers: a 1×1 convolution with 64 filters to adjust channel dimensions, a 1×3 convolution for spatial feature extraction, and another 1×1 convolution with 256 filters for residual addition. This design enables the network to align and combine feature maps of varying dimensions efficiently. A cross dual-attention interaction module is inserted after Residual Block 1 to enhance feature representation by enabling interaction between the two pathways. The outputs remain at 4×56^2 and 32×56^2, respectively. Residual Blocks 2, 3, and 4 follow the same design pattern, with cross dual-attention modules inserted after each block. Ultimately, the feature maps are downsampled to 4×7^2 for the sparse pathway and 32×7^2 for the dense pathway.

After feature extraction through the residual blocks, global average pooling is applied to each feature map to compress each channel into a single value. The outputs from the sparse and dense pathways are then fused to form a single feature vector, which is passed through a fully connected layer to produce the final action class prediction.

2.2.2. Cross Dual-Attention Interaction Module

As shown in Figure 2, this chapter proposes a Cross Dual-Attention Interaction Module, which facilitates information exchange between the sparse and dense pathways via a combination of 3D spatiotemporal self-attention and 3D efficient channel attention mechanisms.

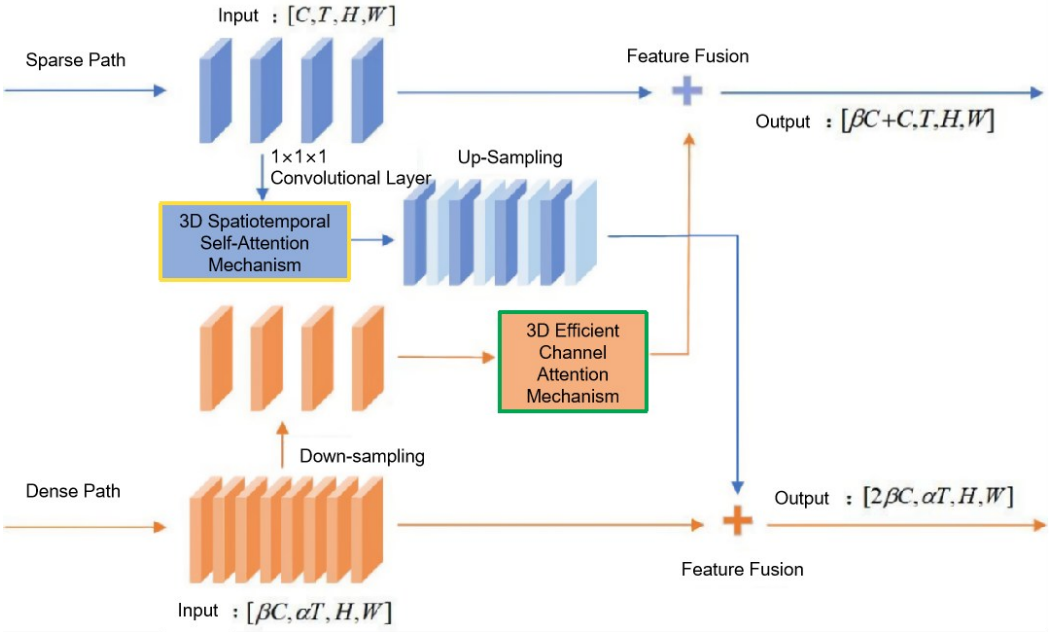


Figure 2. Cross dual-attention interaction module

2.2.2.1. Attention mechanism

The human visual system can rapidly scan the environment to locate salient regions and focus cognitive resources on these areas for detailed processing, while ignoring irrelevant information. Inspired by this mechanism, attention models in deep learning aim to highlight task-relevant information from a large pool of inputs, effectively improving model performance.

To enhance the network's representational capacity and leverage the rich spatiotemporal information inherent in action videos, this chapter proposes a 3D spatiotemporal self-attention mechanism for generating attention maps from the sparse to the dense pathway, and a 3D efficient channel attention mechanism to assign varying importance weights across channels. These two mechanisms are integrated through cross-connections to enable dynamic information exchange between pathways.

(1) 3D Spatiotemporal Self-Attention Mechanism: Given an input feature tensor $I[C,T,H,W]$, the raw input video is first processed using a 1×1×1 convolutional kernel to generate three sets of video features. These are then used to adaptively assign weights along the height, width, and temporal dimensions, respectively. In each dimension, the darker the color in the corresponding heatmap block, the more critical that region is for classifying the video action category, and thus the model allocates more attention to it. After two rounds of feature selection and filtering via the 3D spatiotemporal self-attention mechanism, attention weights are distributed over all pixels, further refining the extracted features. Additionally, a residual connection is incorporated to preserve static background information and prevent degradation in the model's learning capacity. As illustrated in Figure 3, the detailed process is as follows:

The 3D spatiotemporal self-attention mechanism generates attention weights across both spatial and temporal dimensions. Conceptually, this process can be interpreted as unsupervised estimation of the influence exerted by surrounding neighboring pixels on a given pixel. The attention computation begins with the width-direction, where all pixels along this axis interact with each other to compute individual attention weights. The same operation is applied to pixels that lie in the same row, column, or share the same location across the temporal sequence. Through iterative traversal, each pixel ultimately captures global contextual information from all other pixels.

Specifically, three independent attention operations are performed along the width, height, and temporal directions to implement the 3D spatiotemporal self-attention. For instance, in width-direction attention, the input feature tensor is first reshaped to $I[\{T,H\},W$,C$]I[\backslash\{T,H\backslash\},W,C]I[\{T,H\},W,C]$, followed by three 1×1 convolutions to generate the query (Q), key (K), and value (V) matrices, each of the same shape $I[\{T,H\},W,C]$. The attention weights for the width direction are then computed via the dot product of Q and the transpose of K, and normalized using the Softmax function, as shown in Equation (1). The final output of the 3D spatiotemporal self-attention module is a refined feature tensor $O[C,T,H,W]$, calculated as follows:

$$Attention(Q,K,V) = soft\,max(\frac{QK^T}{\sqrt{d}})V \tag{1}$$

$$O^{[C,T,H,W]} = Attention(Q,K,V) + I^{[C,T,H,W]} \tag{2}$$

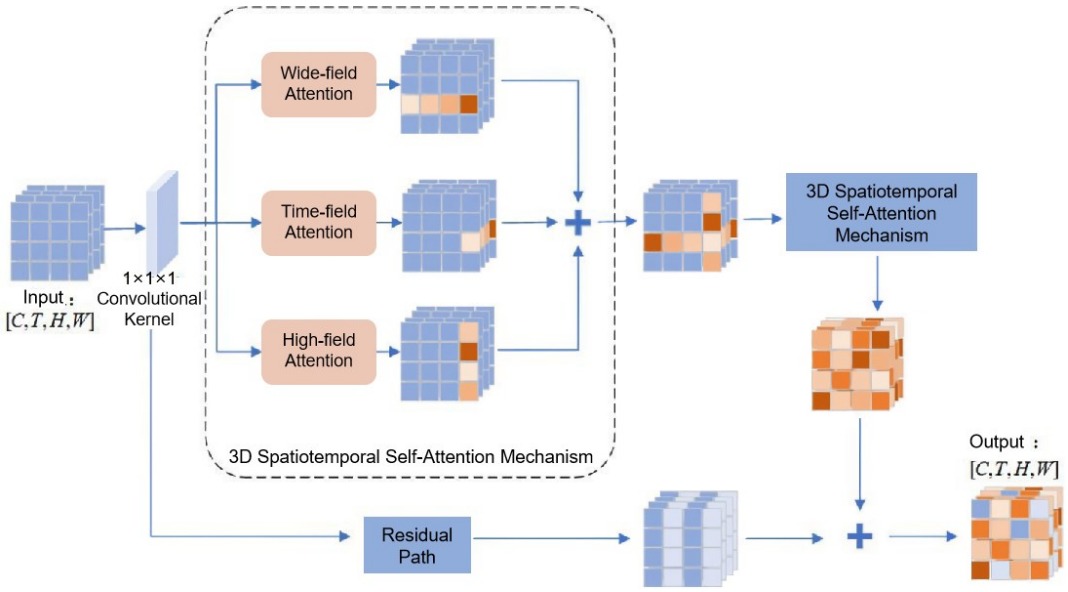


Figure 3. 3D spatiotemporal Self-Attention mechanism

Where $Q = F \cdot Wq, K = F \cdot Wk, V = F \cdot Wv, Q, K, V$, represent Query, Key and Value respectively. Wq, Wk, Wv denote the learnable projection matrices. I represents the input video feature tensor, and O denotes the output feature tensor.

By analogy, the height-direction attention is computed starting from the reshaped feature map $I^{[\{T,W\},H,C]}$, while the temporal-direction attention begins with $I^{[\{W,H\},T,C]}$. The attention features from the width, height, and temporal directions are then added to the original input features to obtain the final fused feature representation. This fused feature captures both spatial information from width and height directions and temporal information from the time direction. After the first round of computation, each pixel is influenced by all other pixels along the same row, column, and temporal position. This attention mechanism can be recursively applied, enabling each pixel to attend to all other pixels across the entire feature map, thereby capturing comprehensive global context. Notably, the two self-attention modules share parameters, which effectively reduces memory consumption in practice and introduces minimal computational overhead.

(2) 3D Efficient Channel Attention Mechanism: Given an input feature tensor $I^{[C,T,H,W]}$, the original video features are first passed through global average pooling to remove spatial and temporal dimensions, preserving only the channel-wise information. Next, a 1D convolution is applied to model the correlations between adjacent channels, enabling local cross-channel information interaction. This process can be interpreted as a sliding window that aggregates nearby channel features. The output is then passed through a Sigmoid activation function to generate the channel attention weights, thereby enhancing feature representations. In the resulting attention map, darker-colored channel blocks indicate higher importance, signifying that the model allocates greater weights to these channels. Furthermore, a residual connection is introduced to preserve static background information and ensure the model's learning capacity remains intact. The overall process is illustrated in Figure 4.

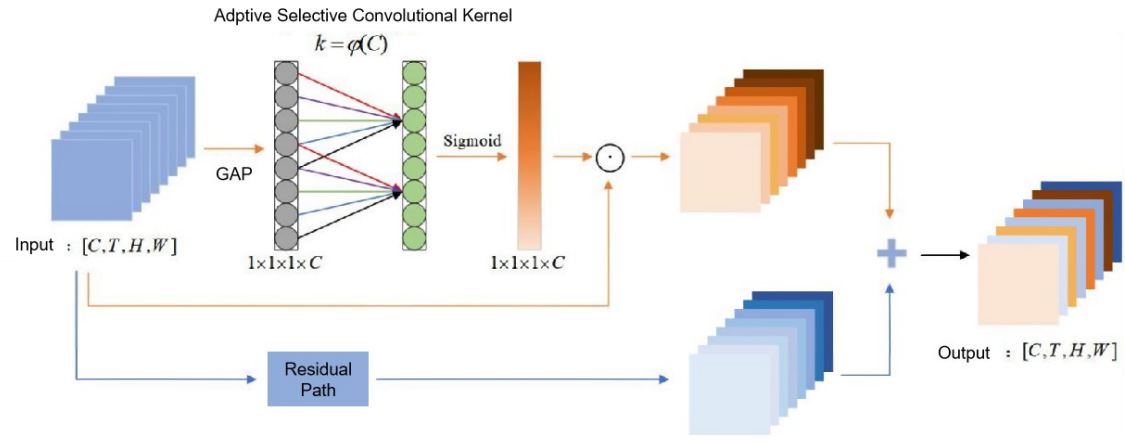


Figure 4. 3D Efficient Channel Attention Mechanism

The 3D Efficient Channel Attention Mechanism captures inter-channel and multi-channel interactions while avoiding dimensionality reduction. After performing global average pooling, a vector of size 1×1×1×C is obtained, retaining only channel-wise information. An adaptive kernel size k is then computed and used in a 1D convolution to calculate the weight of each channel. The number of weights determines the kernel size k. To reduce parameter count, the module adopts weight sharing, reducing the number of weights from $k \times C$ to just k. The value of k varies depending on the number of channels C, and once C is determined, the adaptive kernel size k can be calculated using the following formulas:

$$C = \phi(k) = 2^{\gamma \cdot k - b} \tag{3}$$

$$k = \varphi(C) = \left| \frac{log_2(C)}{\gamma} + \frac{b}{\gamma} \right|_{odd} \tag{4}$$

Here, γ =2 and b =1, $|t|_{odd}$ denotes the nearest odd number to ensure symmetry in convolution.

Next, a 3D global average pooling operation is applied to remove the spatial and temporal dimensions, preserving only the channel-wise information, as defined by:

$$Avg(I^{[C,T,H,W]}) = \frac{1}{T*H*W} \sum_{i,j,k}^{T,H,W} I_{i,j,k} \tag{5}$$

Where Avg ($I^{[C,T,H,W]}$) denotes the result of 3D global average pooling. Based on this, the final output of the module is defined as:

$$O^{[C,T,H,W]} = \sigma(CID_k(Avg(I^{[C,T,H,W]}))) \cdot I^{[C,T,H,W]} + I^{[C,T,H,W]} \tag{6}$$

Where CID denotes the 1D convolution, k represents the adaptive kernel size, and σ denotes the Sigmoid activation function.

2.2.2.2. Feature fusion method

As shown in Figure 2, in order to fuse information between the sparse and dense paths, it is first necessary to align the dimensions of the obtained features—specifically, the channel (C) and temporal (T) dimensions in $[C, T, H, W]$. In the spatiotemporal information interaction network proposed in this chapter, the input feature map of the sparse path is $[C, T, H, W]$, while the input feature map of the dense path is $[\beta C, \alpha T, H, W]$, where C represents the number of convolutional channels, T, H, and W denote the temporal length, spatial height, and width, respectively. α and β represent the temporal speed ratio and channel ratio, respectively. Feature fusion is achieved through cross-connections between the sparse and dense paths.

(1) Fusion from the Dense Path to the Sparse Path: In this direction, the dense path contains more frames than the sparse path—specifically, α times more. Therefore, a 3D temporal max pooling operation with a kernel size of [α,1,1] and a stride of [α,1,1] is used to downsample the dense path from αT frames to T frames. This results in a new spatiotemporal feature map, in which each pixel represents the most salient feature among the α frames. Next, channel attention is computed to assign appropriate weights to each channel, further enhancing performance. Finally, the temporal features of the dense path are laterally connected to the spatial features of the sparse path in the next stage. The resulting fused feature map has the shape $[\beta C + C, T, H, W]$.

(2) Fusion from the Sparse Path to the Dense Path: In this direction, a 1×1×1 convolutional layer is first applied to reduce and match the number of channels. Subsequently, a spatiotemporal attention map is computed for specific spatiotemporal features, and the execution order is optimized so that computationally expensive operations are performed on smaller feature maps. Then, nearest-neighbor interpolation is used along the temporal axis to upsample the sparse path by a factor of [α,1,1], expanding the T frames to αT frames. Finally, the spatial features of the sparse path are fused with the temporal features of the original dense path. To achieve dimensional consistency between the two paths, lateral connections are used. Currently, three lateral connection strategies are considered:

- Time-to-Channel: Compresses α frames into a single frame by increasing the number of channels, transforming $[\beta C, \alpha T, H, W]$ into $[\alpha\beta C, T, H, W]$;
- Temporal Interval Sampling: Randomly samples one frame out of every α frames, converting $[\beta C, \alpha T, H, W]$ into $[\beta C, T, H, W]$;
- Temporal Interval Convolution: Applies a 3D convolution with a kernel size of 5×1×2, outputting $2\beta C$ channels, and a stride of α in the temporal dimension.

Given the stochastic nature of video actions, which are not always evenly distributed over time, the first two methods may result in the loss of critical action segments, thereby negatively affecting recognition accuracy. Therefore, the temporal interval convolution approach is adopted for lateral connection, resulting in a fused feature map with shape $[\beta C + \beta C, T, H, W]$.

2.3. Experiments and analysis

The algorithm proposed in this chapter is implemented using the deep learning framework PyTorch, and the operating system used is Windows 10. The Central Processing Unit (CPU) is an Intel(R)Core(TM)i9-9820XCPU@3.30GHz, and the Graphics Processing Unit (GPU) is an NVIDIAGeForceRTX2080Ti, with 16 GB of RAM. The algorithm is optimized using the SGD optimizer, with an initial learning rate set to 0.01. A cosine annealing strategy is used to decay the learning rate, with a momentum of 0.9, weight

decay of 0.0006, and a dropout rate of 0.5. The number of training epochs is 150, and the batch size is 8. The following sections introduce the experimental setup and datasets, followed by the presentation and analysis of the experimental results.

2.3.1. Dataset description

The experiments and analysis in this section are based on the algorithm presented in Section 2.2. The publicly available UCF101, HMDB51, and ARID datasets are used for evaluation. The UCF101 dataset consists of video clips collected from YouTube [26], introduced by Soomro et al. in 2012. Sample video clips from the dataset are shown in Figure 5. The dataset contains 13,320 videos across 101 human action categories, each with a resolution of 320×240. These categories can be grouped into five major types: Instrument playing, Sports, Actions by children, Human-human interaction, Human-object interaction.

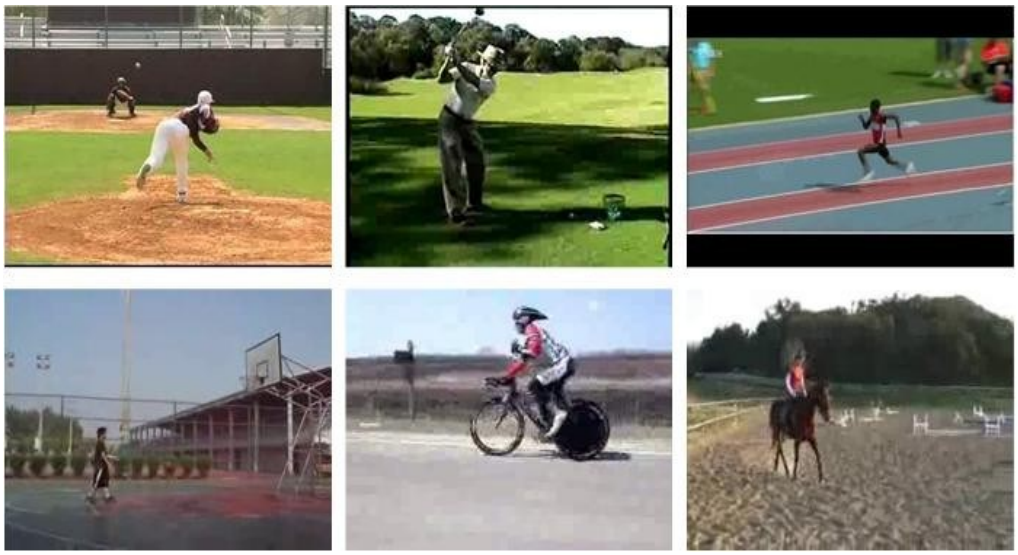


Figure 5. Sample action clips from the UCF101 dataset

The HMDB51 dataset is primarily compiled from various released films [27], with a smaller portion collected from public datasets. It contains 6,849 video clips across 51 human action categories, with each category having at least 101 video clips.

Sample clips from the dataset are shown in Figure 6. These action categories can be grouped into five main types: Human interactions, General body movements, Body-object interactions, General facial actions, Facial actions involving object manipulation

Figure 6. Sample action clips from the HMDB51 dataset

The ARID dataset [28] is the first dataset focused on human actions in dark environments. It consists of over 3,780 video clips covering 11 action categories, including: drinking, jumping, picking, pouring, pushing, running, sitting down, standing up, turning (left or right), walking, and waving—as shown in Figure 7. These clips were recorded in 11 different scenes (24 indoor and 12 outdoor environments) with more than 12 volunteers participating. The video clips are significantly affected by illumination, noise, and other challenging conditions.

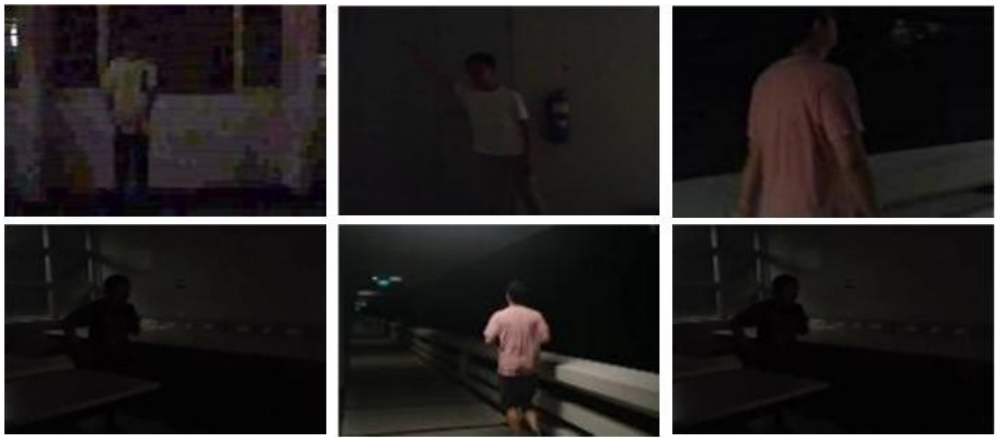


Figure 7. Sample action clips from the ARID dataset

2.3.2. Evaluation metrics

In binary classification tasks, there are four types of prediction outcomes: True Positive (TP): A positive sample is correctly predicted as positive; False Negative (FN): A positive sample is incorrectly predicted as negative; True Negative (TN): A negative sample is correctly predicted as negative; False Positive (FP): A negative sample is incorrectly predicted as positive. When the prediction results are TP or TN, the recognition is considered correct. Based on the above conditions, the confusion matrix is shown in Table 2.

Table 2. Confusion matrix

Confusion Matrix		Ground Truth	
		Positive	Negative
Prediction	Positive	*TP*	*FP*
	Negative	*FN*	*TN*

In this chapter, $Accuracy$ is adopted as the evaluation metric for assessing model performance, defined as:

$$Accuary = \frac{TP+TN}{TP+FP+FN+TN} \times 100\% \tag{7}$$

where TP and TN denote correctly predicted samples, and FP and FN denote misclassified samples.

2.3.3. Ablation study

To verify the effectiveness of each module in the proposed algorithm, ablation experiments were conducted on the HMDB51 and UCF101 datasets. The experimental results are shown in Table 3, where " √ " indicates the module is used and "×" indicates it is not.

From Table 3, it can be seen that using only the dense path yields relatively low accuracy. This is because the dense path is a lightweight model with fewer channels—accounting for only 20% of the overall network's computation—thus weak in spatial information processing. In contrast, using only the sparse path achieves higher accuracy due to its higher computational load and richer feature extraction. When both paths are used and their spatial and temporal features are fused, accuracy is significantly improved, demonstrating the feasibility of the dual-path network. With the cross dual-attention interaction module integrated, the model achieves the highest accuracy. This improvement stems from the dual-attention mechanism and the clearly defined bidirectional information exchange, which enable the sparse path to focus more effectively on important contextual semantic regions, while the dense path gains enhanced perceptual capabilities. As a result of these enhancements, the spatiotemporal interaction network achieves superior performance.

Table 3. Ablation experiment results

Dense Path	Sparse Path	Cross Dual-Attention Interaction Module	Accuracy (%)	
			UCF101	**HMDB51**
√	×	×	78.5	51.7
×	√	×	90.3	72.6
√	√	×	95.8	77.0
√	√	√	**97.6**	**78.4**

2.3.4. Comparative experiments

To validate the effectiveness of the proposed Spatiotemporal Interaction Network, comparative experiments were conducted on the public HMDB51 and UCF101 datasets. The model was compared against several mainstream human action recognition models, including Res3D [25], Two-stream [1], LRCN [29], and SlowFast [4]. The results are presented in Table 4.

Table 4. Comparative results on public datasets

Method	**Accuracy(%)**	
	UCF101	**HMDB51**
Res3D	85.8	54.9
Two-stream	86.9	58.0
LRCN	88.6	61.0
Slowfast	95.8	77.0
Ours	**97.6**	**78.4**

As shown in Table 4, the proposed method achieves the best classification performance. Specifically, it outperforms the SlowFast model by 1.8% on the HMDB51 dataset and 1.4% on the UCF101 dataset, demonstrating the effectiveness of the proposed Spatiotemporal Interaction Network.

2.3.5. Application experiments

In this section, the proposed Spatiotemporal Interaction Network is tested on the nighttime ARID dataset both before and after image enhancement [28]. The action recognition model, trained on the HMDB51 dataset, is evaluated on 11 action categories from 12 outdoor scenes in the ARID dataset, with 12 video clips selected for each category. The experimental results are presented in Table 5, and Figure 3 provides visual comparisons from 10 experimental groups.

Table 5. Action recognition accuracy on the ARID dataset

Action Category	Accuracy Before Enhancement (%)	Accuracy After Enhancement (%)
drink	61.5	84.3
jump	62.3	86.5
pick	45.1	79.8
pour	35.6	63.6
push	42.3	76.9
run	87.6	98.6
sit	35.1	62.1
stand	41.9	75.3
turn	75.3	91.2
walk	89.6	99.5
wave	87.2	97.5
Average (All 11)	**60.3**	**83.2**

As shown in Table 5, the recognition accuracy for every action category significantly improves after image enhancement. The average recognition accuracy increases by 22.9%, indicating that the enhanced images greatly improve action recognition

performance. This is because nighttime surveillance videos typically suffer from low brightness and contrast, resulting in blurry content and lost details, which make feature extraction and action recognition difficult and inaccurate. However, after enhancement using the nighttime image enhancement model described in Chapter 3, the videos exhibit improved contrast, brightness, clarity, and color saturation. These improvements enable the extraction of more meaningful features, leading to a substantial increase in recognition accuracy.

As shown in Figure 8, Figures 8(b) and 8(d) illustrate the recognition results after enhancement of Figures 8(a) and 8(c), respectively. The results clearly demonstrate that the recognition accuracy improves significantly after image enhancement using the Spatiotemporal Interaction Network. This is because the pre-enhancement videos suffer from poor brightness and contrast, making the actions hard to discern. In contrast, the enhanced videos illuminate extremely dark regions while preserving well-lit areas, thereby enhancing clarity and visibility, which facilitates the extraction of richer features and results in higher recognition performance. In summary, applying image enhancement before action recognition significantly improves the performance of nighttime human action recognition.

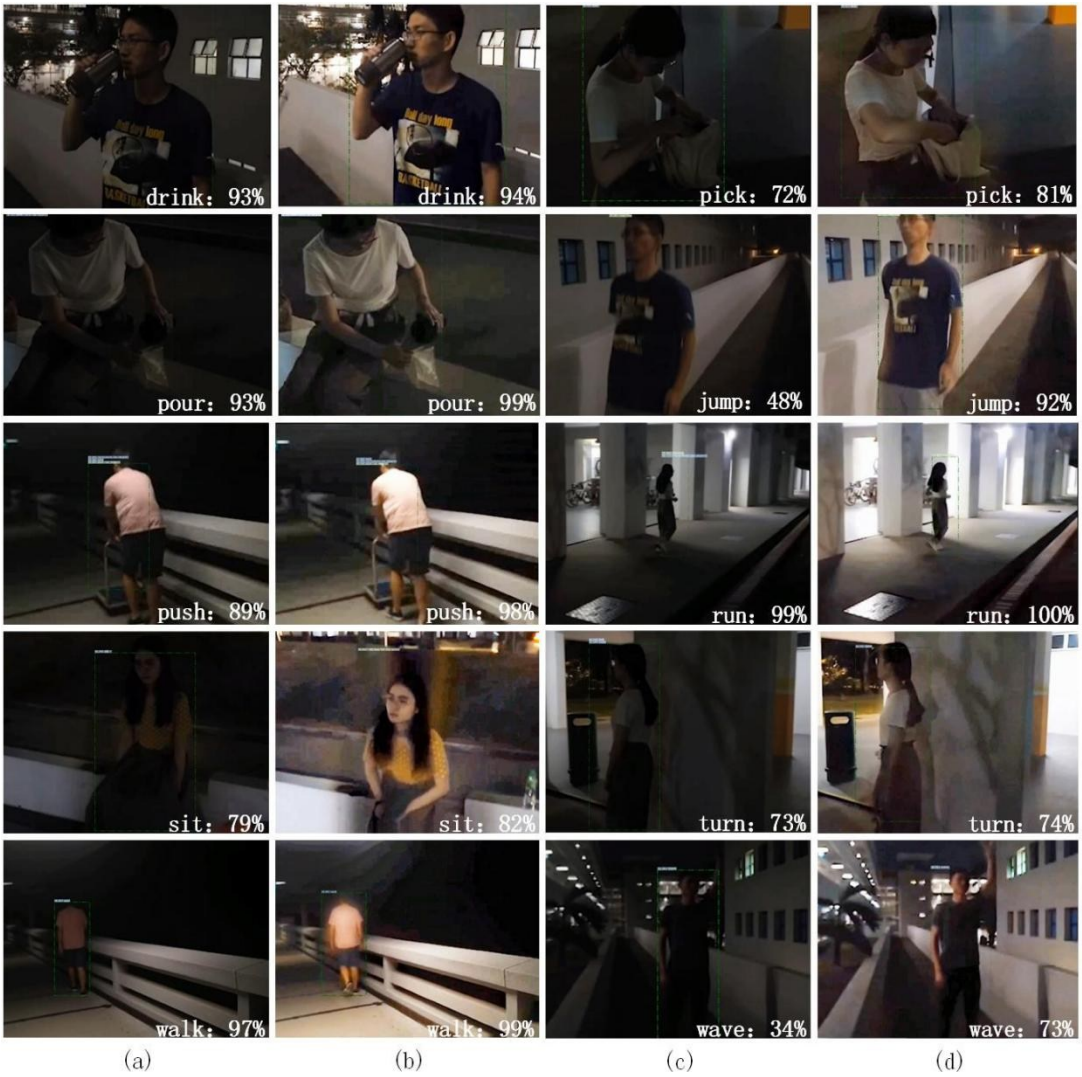


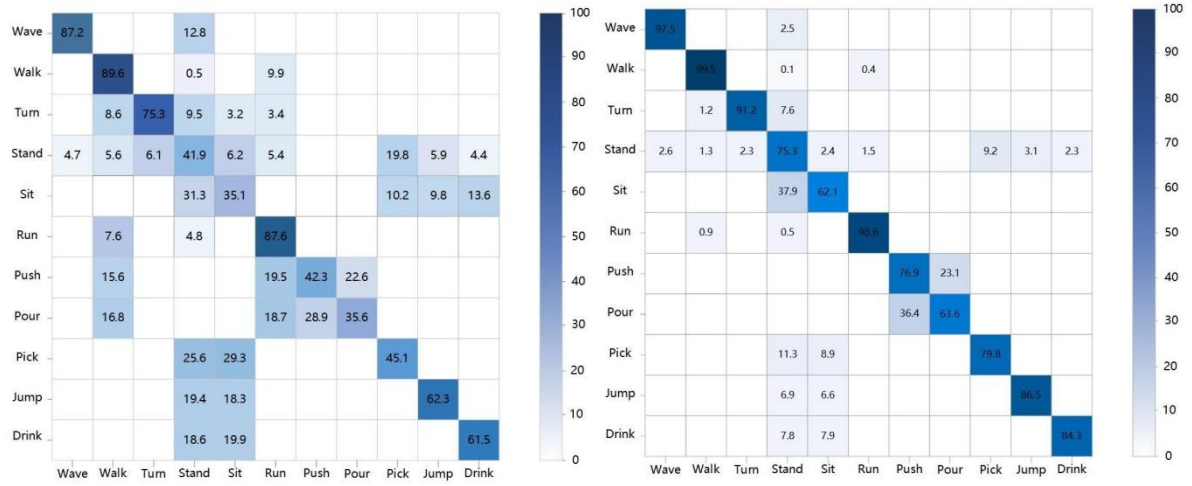
Figure 8. Human action recognition results before and after nighttime image enhancement

To further analyze the experimental results, confusion matrices are used. In a confusion matrix, each row represents the proportion of actual video categories, while each column represents the proportion of predicted categories. The darker the cell color and the higher the percentage, the more accurately that class is recognized. The confusion matrices for the 11 outdoor action categories in the original ARID dataset are shown in Figure 9, where Figure 9(a) displays the confusion matrix before enhancement, and Figure 9(b) shows the matrix after enhancement. The following analysis is based on these two confusion matrices.

As shown in Figure 9, the misclassification rate in Figure 9(a) is significantly higher than that in Figure 9(b). This is mainly because the original nighttime ARID videos suffer from serious visual degradation, including low brightness, poor contrast, and unclear details, which reduce the amount of useful features extractable during the recognition stage, resulting in increased

misclassification. In contrast, the enhanced videos eliminate the adverse effects of insufficient lighting, thereby substantially improving recognition accuracy.

In addition, the confusion matrices show variation in recognition accuracy across different action categories. This can be attributed to factors such as: whether the action involves significant visual changes, how distinguishable it is from other actions, and the complexity of the environment. For example: The actions pick and drink exhibit subtle changes, relatively static postures, and overlap with actions like stand and sit, resulting in lower recognition accuracy. Although jump involves more pronounced movement, it shares visual similarities with stand and sit, leading to frequent confusion and reduced accuracy. Pour and push show larger motion changes but are visually similar to each other, which also lowers their classification accuracy.



(a) Before enhancement

(b) After enhancement

Figure 9. Confusion matrices on the ARID dataset

3. Conclusion

This chapter proposes a human action recognition model based on spatiotemporal information interaction. Firstly, a dual-path network is introduced to independently learn spatial and temporal features. It consists of a sparse pathway that operates at a low frame rate to capture semantic spatial information, and a parallel dense pathway that runs at a high frame rate to capture temporal motion information. Secondly, to extract more discriminative features from the video, a cross dual-attention interaction module is proposed to focus attention on key regions of the video segments and to explicitly exchange spatiotemporal information between the two pathways. Comparative experiments conducted on the public datasets HMDB51 and UCF101 demonstrate that the proposed method achieves higher accuracy than four mainstream action recognition algorithms. Additionally, ablation studies validate the rationality and necessity of each component in the model. Finally, a set of comparative experiments on the nighttime ARID dataset—between models applied to pre-enhancement and post-enhancement video—show a significant increase in recognition accuracy after enhancement, thereby verifying the feasibility and effectiveness of the proposed method.

References

[1] Simonyan, K., & Zisserman, A. (2014). Two-stream convolutional networks for action recognition in videos. *Advances in Neural Information Processing Systems*, *27*, 1-9.

[2] Chen, H., Li, M., Jing, L., & Cheng, Z. (2021). Lightweight long and short-range spatial-temporal graph convolutional network for skeleton-based action recognition. *IEEE Access*, *9*, 161374-161382. https://doi.org/10.1109/ACCESS.2021.3133045

[3] Lin, J., Gan, C., & Han, S. (2019). TSM: Temporal shift module for efficient video understanding. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 7083-7093.

[4] Feichtenhofer, C., Fan, H., Malik, J., & He, K. (2019). SlowFast networks for video recognition. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 6202-6211.

[5] Pang, C., Lu, X., & Lyu, L. (2023). Skeleton-based action recognition through contrasting two-stream spatial-temporal networks. *IEEE Transactions on Multimedia*, *1*(4), 1-4.

[6] Tran, D., Bourdev, L., Fergus, R., Torresani, L., & Paluri, M. (2015). Learning spatiotemporal features with 3D convolutional networks. *Proceedings of the IEEE International Conference on Computer Vision*, 4489-4497.

[7] Tran, D., Ray, J., Shou, Z., Chang, S. F., & Paluri, M. (2017). ConvNet architecture search for spatiotemporal feature learning. *arXiv preprint arXiv:1708.05038*, 1-12.

[8] Carreira, J., & Zisserman, A. (2017). Quo vadis, action recognition? A new model and the kinetics dataset. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 6299-6308.

[9] Feichtenhofer, C. (2020). X3D: Expanding architectures for efficient video recognition. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 203-213. https://doi.org/10.1109/CVPR42600.2020.00028

[10] Li, J., Han, Y., Zhang, M., Li, G., & Zhang, B. (2022). Multi-scale residual network model combined with global average pooling for action recognition. *Multimedia Tools and Applications*, *81*(1), 1375-1393.

[11] 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 2625-2634).

[12] Si, C., Jing, Y., Wang, W., Wang, L., & Tan, T. (2018). Skeleton-based action recognition with spatial reasoning and temporal stack learning. *Proceedings of the European Conference on Computer Vision (ECCV)*, 103-118.

[13] Li, Z., Gavrilyuk, K., Gavves, E., Jain, M., & Snoek, C. G. (2018). VideoLSTM convolves, attends and flows for action recognition. *Computer Vision and Image Understanding*, *166*, 41-50.

[14] Aljarrah, A. A., & Ali, A. H. (2019). Human activity recognition using PCA and BiLSTM recurrent neural networks. *2019 2nd International Conference on Engineering Technology and its Applications (IICETA)*, 156-160.

[15] Chenhao, W., Yongquan, W. E. I., Dong, G. U. O., & Jun, G. (2020). Human behavior recognition under occlusion based on two-stream network combined with BiLSTM. *2020 Chinese Control And Decision Conference (CCDC)*, 3311-3316.

[16] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, *30*, 1-11.

[17] Wang, Q., Wu, B., Zhu, P., Li, P., Zuo, W., & Hu, Q. (2020). ECA-Net: Efficient channel attention for deep convolutional neural networks. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11534-11542.

[18] Woo, S., Park, J., Lee, J. Y., & Kweon, I. S. (2018). CBAM: Convolutional block attention module. *Proceedings of the European Conference on Computer Vision (ECCV)*, 3-19.

[19] Fu, J., Liu, J., Tian, H., Li, Y., Bao, Y., Fang, Z., & Lu, H. (2019). Dual attention network for scene segmentation. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3146-3154.

[20] Pan, B., Cao, Z., Adeli, E., & Niebles, J. C. (2020). Adversarial cross-domain action recognition with co-attention. *Proceedings of the AAAI Conference on Artificial Intelligence, 34*(7), 11815-11822.

[21] Liu, Z., Luo, D., Wang, Y., Wang, L., Tai, Y., Wang, C., Li, J., Huang, F., & Lu, T. (2020). TEINet: Towards an efficient architecture for video recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence, 34*(7), 11669–11676.

[22] Wang, X., Xiong, X., Neumann, M., Piergiovanni, A. J., Ryoo, M. S., Angelova, A., Kitani, K. M., & Hua, W. (2020). AttentionNAS: Spatiotemporal attention cell search for video classification. *Computer Vision–ECCV 2020: 16th European Conference*, 449-465.

[23] Niu, Z., Zhong, G., & Yu, H. (2021). A review on the attention mechanism of deep learning. *Neurocomputing*, *452*, 48-62.

[24] Feichtenhofer, C., Fan, H., Malik, J., & He, K. (2019). SlowFast networks for video recognition. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 6202-6211.

[25] Picaud, S., Dalkara, D., Marazova, K., Goureau, O., Roska, B., & Sahel, J. A. (2019). The primate model for understanding and restoring vision. *Proceedings of the National Academy of Sciences*, *116*(52), 26280-26287.

[26] Dai, W., Chen, Y., Huang, C., Gao, M. K., & Zhang, X. (2019). Two-stream convolution neural network with video-stream for action recognition. *2019 International Joint Conference on Neural Networks (IJCNN)*, 1-8.

[27] Wu, M. C., Chiu, C. T., & Wu, K. H. (2019). Multi-teacher knowledge distillation for compressed video action recognition on deep neural networks. *ICASSP 2019-2019 IEEE International Conference on Acoustics*, 2202-2206.

[28] Xu, Y., Yang, J., Cao, H., Mao, K., Yin, J., & See, S. (2021). ARID: A new dataset for recognizing action in the dark. *Deep Learning for Human Activity Recognition: Second International Workshop*, 70-84.

[29] Lore, K. G., Akintayo, A., & Sarkar, S. (2017). LLNet: A deep autoencoder approach to natural low-light image enhancement. *Pattern Recognition*, *61*, 650-662.