IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 18 April 2023, accepted 4 May 2023, date of publication 9 May 2023, date of current version 24 May 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3274542

RESEARCH ARTICLE

A Novel Spatio-Temporal-Wise Network for Action Recognition

ZHENGBAO CAI
Department of Information, Anhui Vocational College of Defense Technology, Lu'an, Anhui 237011, China
School of Computer Science and Technology, Anhui University, Hefei, Anhui 230601, China
e-mail: caizhengbao2023@126.com

ABSTRACT Action recognition is a challenging task that requires understanding the temporal relationships between frames. However, capturing and processing spatio-temporal and motion features is computationally expensive, making it difficult to apply to practical situations. We propose a novel approach called the Spatio-Temporal-Wise (STW) network to address this problem. The STW network inserts STW blocks, consisting of a Spatio-Temporal Fusion Module and a Temporal-Wise Module, into an existing 2D CNN. This approach requires very little additional computational overhead but brings huge performance improvements in recognizing human actions. The proposed method is evaluated on several public datasets, including Something-Something v1 & v2, Kinetics-400, UCF101, and HMDB51. STW achieved comparable or better performance on these datasets compared to state-of-the-art methods. Notably, the STW network improves recognition accuracy by 26.6% and 34.6% on the Something-Something v1 & v2 datasets, respectively, with less than 2% additional computational overhead. The results demonstrate that the STW network can significantly improve performance in action recognition tasks while requiring only a small additional computational overhead, which represents a promising direction for developing more efficient and effective approaches to handling temporal reasoning in action recognition, which may have important applications in the future.

INDEX TERMS Action recognition, video understanding, temporal reasoning.

I. INTRODUCTION

Due to the rapid development of video technology, an enormous amount of video data is generated every day. This includes videos shared on social networking sites and monitoring videos. For instance, statistics show that YouTube uploads 300 hours of video data per minute. These videos have significant intrinsic values, and their analysis can promote technological and social progress. However, manual analysis of this vast amount of video data is impossible, which has led to the need for intelligent and efficient video analysis methods. Video understanding has become prevalent in many fields, including video recommendation and surveillance, and has gained extensive attention from both industry and academia. Action recognition is a crucial issue in video understanding, and researchers have extensively explored it in the past decades, as evidenced by several studies [1], [2], [3]. Human actions involve numerous factors, including body movements and human-object interaction. Unlike image recognition, video data is a high-dimensional and structured data type, and its processing requires consideration and mining of the temporal structure of the video. Thus, action recognition necessitates a strong temporal reasoning ability.

Convolutional networks have become the mainstream approach for many image-based tasks, such as image classification, and have been extended to video motion recognition. However, recent work by Adam [4] argues that such powerful deep learning architectures may not be well-suited for temporal reasoning, even though they are powerful static vision processors. Videos differ from static images in that they contain a temporal dimension. The temporal and spatial structures of videos are different, and simply using image convolution methods may not effectively capture the relationship information across time. For example, as shown in Figure 1, it is impossible to distinguish between "Pushing something from right to left" and "Pushing something

The associate editor coordinating the review of this manuscript and approving it for publication was Byung-Gyu Kim.

from left to right'' using RGB pictures. Many methods that are effective for scene-related tasks may not achieve good results for temporal-related tasks. Therefore, action recognition requires special processing of the temporal dimension.

Currently, there are two main types of action recognition methods. The first uses a two-stream neural network [2], with RGB frames as the spatial stream and optical flow as the temporal stream. Optical flow can capture object motion information between adjacent frames, which can avoid interference from factors such as the background. For action recognition, sparse sampling is often used to obtain semantic information on a larger temporal scale since the semantic changes between adjacent frames are slow, and action often spans dozens of frames [1]. However, optical flow can only represent motion information between adjacent frames and cannot capture motion information over a larger temporal span. In the case of sparse sampling, the optical flow's performance is not significantly better than that of RGB pictures, and it must be used in combination with RGB streams to achieve good results. However, extracting optical flow requires significant computing resources, making it challenging to apply two-stream methods to online action recognition. The second type of method uses a 3D convolutional network [3], [5], [6] to directly capture spatio-temporal information from RGB frames by extending the convolution in the temporal dimension. The performance of 3D convolution-based methods is better than 2D convolution-based methods using a single stream. Nevertheless, simply extending 2D CNN to 3D CNN results in a significant increase in computational overhead. Therefore, like the two-stream method, 3D CNN is also challenging to apply practically.

To achieve performance comparable to two-stream and 3D CNNs, while avoiding their huge computational overhead, we propose a novel Spatio-Temporal-Wise (STW) network. STW is composed of a Spatio-Temporal Fusion Module (STFM) and a Temporal-Wise Module (TWM). STFM is designed to capture spatio-temporal features by encoding adjacent frames and fusing them to create a frame that contains all the information of three frames for spatio-temporal fusion. TWM follows the idea of extracting optical flow to capture motion features. It enhances regions with significant motion changes and extracts motion features using element-subtraction temporal-wise. The two modules are combined into an STW block in parallel, which does not change the feature's shape, making it easy to insert into any existing 2D CNN network with very little computational overhead.

We conducted several experiments on temporal-related and scene-related datasets, and the results show that STW can achieve comparable or better performance than two-stream and 3D CNN methods by only using RGB images as input.

The main contributions of our work can be summarized as follows:

- We propose two specialized modules, the Spatio-Temporal Fusion Module (STFM) and the Temporal-Wise Module (TWM), to capture distinct spatio-temporal and motion features, respectively. The STFM is designed to fuse spatial and temporal information in video frames, while the TWM is specialized in capturing motion features. By separating the two types of features and processing them with dedicated modules, we achieve improved accuracy in action recognition tasks.
- The STW block incorporates both modules and can be seamlessly integrated into any 2D CNN with minimal computational overhead, making our approach highly practical and scalable. In addition, this is the first instance where temporal-wise has been employed to capture motion features for action recognition.
- Our proposed approach, STW, demonstrates significant improvements in recognition accuracy on the Something-Something v1 & v2 datasets [7], achieving gains of 26.6% and 34.6%, respectively, with less than 2% additional computational overhead. Additionally, on the Kinetics-400 [8], UCF101 [9], and HMDB51 [10] datasets, STW achieves comparable or superior performance to current state-of-the-art methods.

II. RELATED WORK

In the early years, action recognition mainly focused on recognizing scenes and objects. In recent years, the task of action recognition has gradually evolved to recognize abstract actions with temporal information.

A. CNNs IN ACTION RECOGNITION

1) 2D CNNs

2D CNNs [1], [2] are capable of recognizing images within a single frame. Many video classification methods [11] that are based on 2D CNNs aggregate predictions from different frames to classify videos. To capture both spatial and temporal features in a video, [2] developed a two-stream CNN that takes RGB images as input for spatial features and optical flow images as input for motion features. Reference [1] (TSN) proposed a sparse temporal sampling strategy that employs a two-stream structure and combines spatial and temporal stream networks using a weighted average.

In contrast to these methods, the Spatio-Temporal-Wise Network (STW) only uses RGB images as input, avoiding the high computational cost of optical flow extraction.

2) 3D CNNs

3D CNNs [3], [5], [6], [12], [13], [14], [15], [16], [17], [18], [19], [20], [21], [22] can capture spatio-temporal information directly from RGB frames by extending the convolution in the temporal dimension. To learn appearance and motion features from raw video volumes, [3] proposed C3D, a 3D CNN based on VGG models. To apply pre-trained 2D convolution filters to 3D convolutions, [8] developed an inflation technique. Other approaches, such as [23], combines 2D and 3D CNNs, while [5] and [18] employ 2D and 3D convolutions in different layers. Reference [16] decomposed the 3D CNN into a spatial 2D convolution and a temporal 1D convolution.

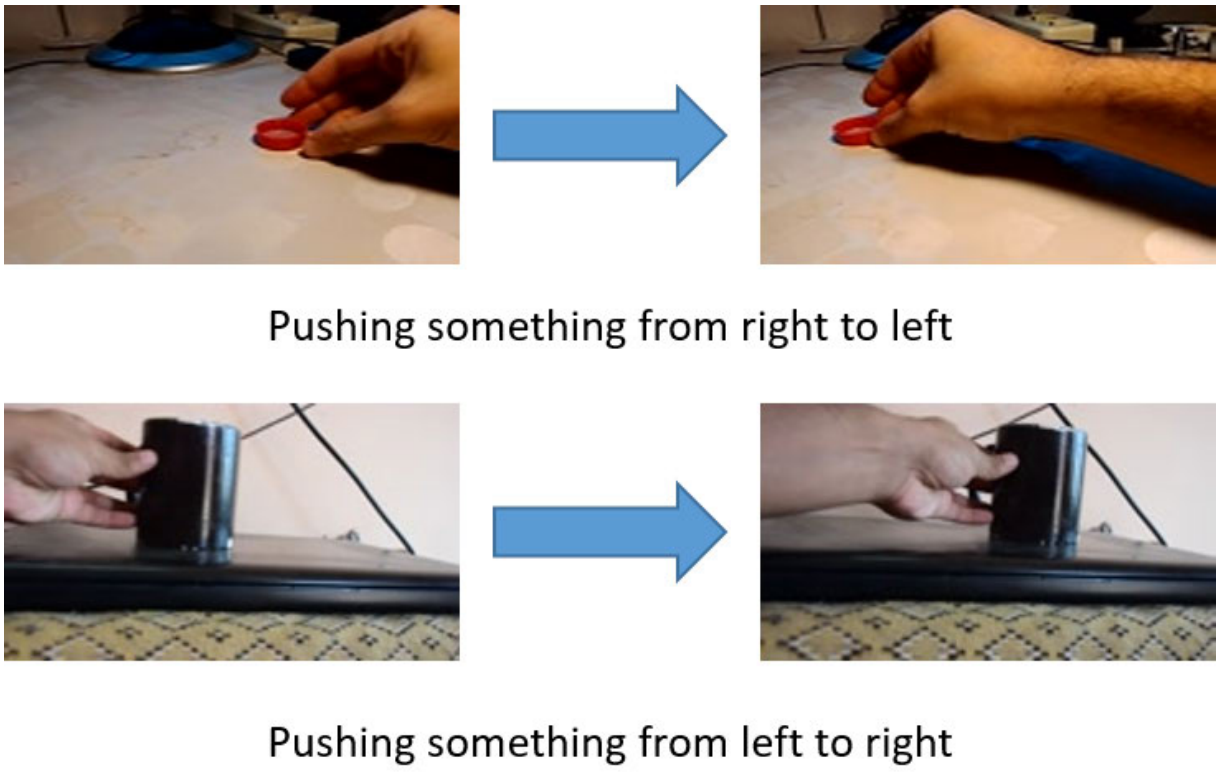

FIGURE 1. The actions in the Something-Something datasets have a strong temporal sequence. Some of the actions are similar and deceptive. It is impossible to distinguish "Pushing something from right to left" and "Pushing something from left to right" by scene information.

Another approach, [20], captures appearance and temporal information using parallel slow and fast paths.

In contrast to these methods, the Spatio-Temporal-Wise Network (STW) uses a 2D CNN as its backbone and captures temporal information by inserting STW blocks into the network. This approach achieves a comparable effect to 3D CNNs while saving a significant amount of computational overhead.

B. TEMPORAL MODELING IN ACTION RECOGNITION

Temporal information is crucial for recognizing actions involving motion over time, such as ''open the door'' versus ''close the door''. 3D CNNs provide a direct way to model temporal information. To extract and aggregate frame features over time, [24] combines CNN and LSTM. To capture temporal relations at multiple scales, [25] propose a temporal relation reasoning module that extracts the appearance features of different frames individually and uses MLPs to infer frame relations. Another approach, [26], shifts parts of the channels along the temporal dimension to facilitate information exchange between adjacent frames. Several other methods, such as [25], [27], [28], [29], [30], and [31], enable 2D networks to sense temporal changes by designing a temporal processing module. These modules extract temporal features by modeling temporal relations between frames and integrating them into the feature representation of each frame. With the emergence of Transformer [32] in computer vision, [33], [34], [35], [36] use the Transformer's encoder to replace the convolution module in the traditional CNN backbone and extract global temporal dependencies.

The Spatio-Temporal-Wise Network (STW) captures both spatio-temporal features and motion features through two parallel modules. The spatio-temporal module fuses spatio-temporal information, while the motion module extracts regions where motion is significant.

III. APPROACH

In this section, we will describe the proposed Spatio-Temporal-Wise Network (STW). Firstly, we will provide technical details of the Spatio-Temporal Fusion Module (STFM). Next, we will introduce the Temporal-Wise Module (TWM) in detail. Finally, we will show how to combine these two modules into an STW block and insert it into an existing 2D CNN.

A. SPATIO-TEMPORAL FUSION MODULE

The original 2D CNN makes separate inferences and predictions for each frame and combines the results at the end of the network. However, this approach does not allow for information exchange between frames during the inference process, which limits the capture of low-level spatio-temporal features. To address this issue, we propose the Spatio-Temporal

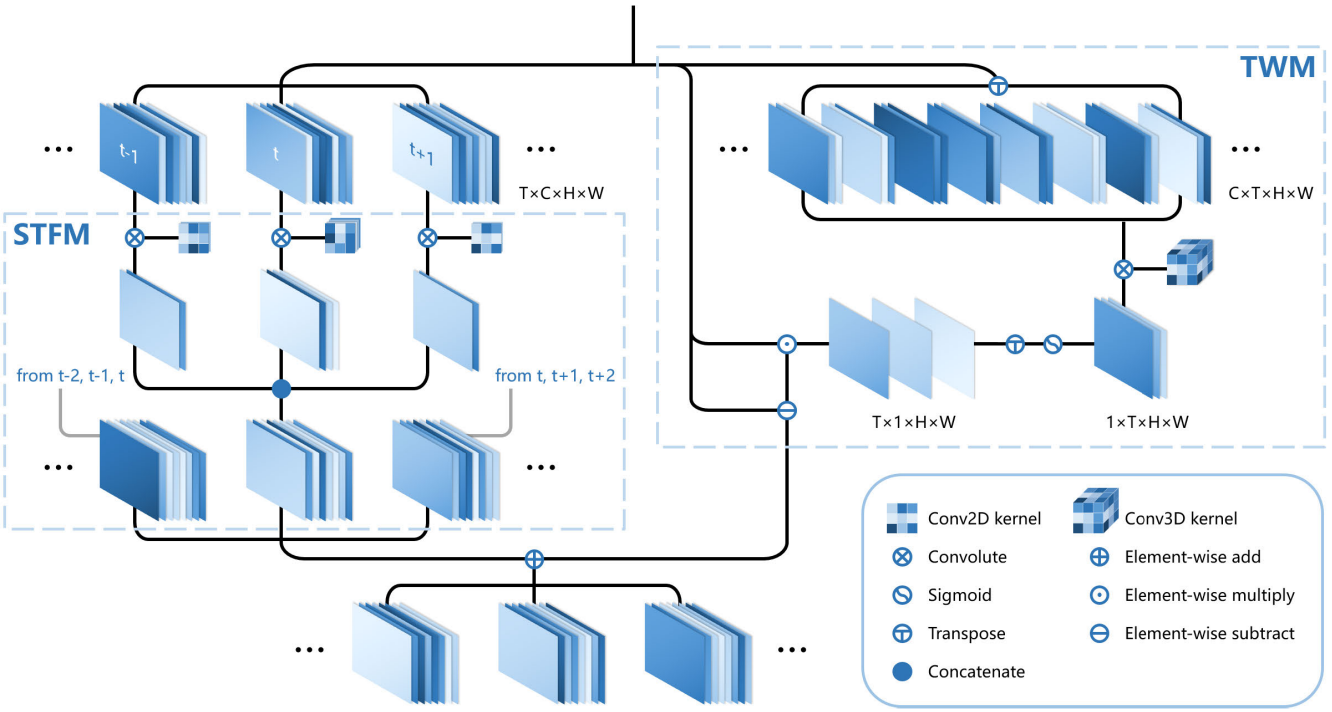

FIGURE 2. STW Block: *T* frame feature maps are fed into STFM and TWM for capturing spatio-temporal and motion features, respectively. The shape of the captured features matches the input features, and the features from both modules are combined using element-wise addition before being passed to the residual branch in the residual block.

Fusion Module (STFM), which encodes and merges adjacent channels during inference to capture both spatial and temporal information at each stage. Importantly, this module can be implemented with minimal computational cost.

As illustrated in Figure 2, given an input feature map squence $X = \{x_1, x_2, \ldots, x_T\}$, in which $x_t \in \mathbb{R}^{C\times H\times W}$, for three adjacent frames x_{t-1}, x_t and x_{t+1}, we use 2D convolution to reduce their number of channels to $\frac{C}{\alpha}$, $\frac{C}{\beta}$ and $\frac{C}{\gamma}$:

$$\begin{aligned} l_t &= \text{Conv2D}(x_{t-1}, \theta) \\ m_t &= \text{Conv2D}(x_t, \phi) \\ r_t &= \text{Conv2D}(x_{t+1}, \varphi) \end{aligned} \tag{1}$$

where θ, ϕ and φ are the parameters of the convolutional layers. To keep the shape of the output feature consistent with the shape of the input feature, we set $\frac{1}{\alpha} + \frac{1}{\beta} + \frac{1}{\gamma} = 1$.

Since x_t represents the current frame and plays a dominant role in capturing the information at time t, it requires a larger output channel number to retain more information about the current time step. On the other hand, x_{t-1} and x_{t+1} represent the frames of the previous and the next time steps, respectively, and play a supplementary role in capturing the information at time t. Therefore, their output channel numbers can be relatively smaller than that of x_t. As a result, we experimented with several settings and ultimately chose to set α, β, and γ to 4, 2, and 4, respectively. The results from experiments using different settings can be found in Section IV-D1.

Then we concatenate these three downsampled features maps $\{l_t, m_t, r_t\}$ together to obtain the final fused feature map y_t:

$$y_t = \text{Concat}(l_t, m_t.r_t) \tag{2}$$

The simplest way to fuse three frames is to average their values, but this approach does not produce good results. The Spatio-Temporal Fusion Module (STFM) achieves significant performance improvements by encoding and concatenating frames. This method of concatenating channels is similar to the TSM approach of shifting channels. However, unlike TSM, STFM encodes and stacks all channels of three adjacent frames, preserving all information from the three frames. TSM, on the other hand, only shifts part of the channels of adjacent frames. This manual method of moving channels based on experience cannot be optimized and only retains part of the channels in the current and adjacent frames, leading to information loss.

B. TEMPORAL-WISE MODULE

Videos typically contain a large number of scenes and objects. Both scene and object information can help recognize motion for action recognition tasks related to scenes. However, scene and object information may affect recognition performance for temporal-related tasks due to the need to capture subtle human actions accurately. Optical flow can extract motion information between frames and remove background information. Inspired by this, we aim to remove useless background information from each frame and only retain regions that contain motion changes. Channel-wise attention is widely used to enhance semantic information in static

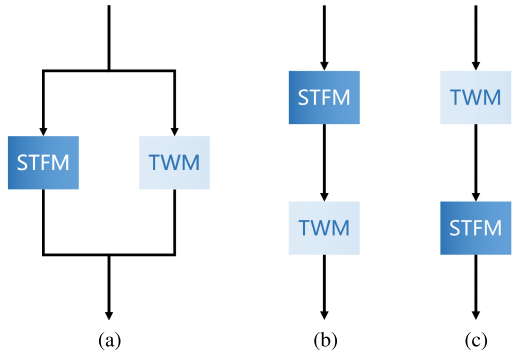

FIGURE 3. Three different architectures of block: (a) Parallel; (b) STFM⇒TWM; (c) TWM⇒STFM.

images, and we hope to use it to enhance motion regions and capture motion features of frames based on temporal information. Therefore, we propose the Temporal-Wise Module.

Given an input feature map $X \in \mathbb{R}^{T\times C\times H\times W}$, we first reshape its shape from $T \times C \times H \times W$ to $C \times T \times H \times W$:

$$X^{T\times C\times H\times W} \longrightarrow \tilde{X}^{C\times T\times H\times W} \tag{3}$$

Then we convolve $\tilde{X}$ with a $3 \times 1 \times 1$ convolutional layer with out-channel of only 1 and obtain $U^{1\times T\times H\times W}$:

$$U = \text{Conv3D}(\tilde{X}, \lambda) \tag{4}$$

where λ is the parameter of the convolutional layer Conv3D. The channel is reduced to 1 to save computational overhead. Its computational overhead is the same as depth-wise convolution, but it can aggregate the information of all channels.

After that, we perform a Sigmoid operation on U to get the weight S and element-wise multiply each channel of $\tilde{X}$ by S to obtain V:

$$\begin{aligned} S &= \text{Sigmoid}(U) \\ V &= \tilde{X} \cdot S \end{aligned} \tag{5}$$

Finally, we subtract V from $\tilde{X}$ to get the motion features Z and reshape Z from $C \times T \times H \times W$ to $T \times C \times H \times W$:

$$Z = \tilde{X} - V \tag{6}$$

Operating temporal-wise on the frame without subtracting the original frame does not yield desirable results. This is because the background interference remains despite enhancing the action regions. However, after subtracting the original frame, the extracted feature is similar to optical flow, capturing information from three frames. The unique advantage of this method is that these three frames can be sparsely sampled without requiring them to be adjacent. This operation eliminates background interference information, and the resulting motion features significantly improve the model's performance. This demonstrates that background interference plays a crucial role in motion recognition, and the proposed method effectively addresses this issue.

C. SPATIO-TEMPORAL-WISE NETWORK

This section describes how to construct two modules to create an STW block and develop a Spatio-Temporal-Wise network.

Figure 3 shows two ways to combine the STFM and TWM methods to form an STW block: vertical and parallel connections. We obtain two types of connections in the vertical connection by changing the order of STFM and TWM. For parallel connections, we aggregate the output features of STFM and TWM using element-wise summation. In our experiments, we found that the parallel combination was more effective than the vertical combination. Further details about the experimental results are described in Section IV-D3.

The overall architecture of the STW block is illustrated in Figure 2. We input T frame feature maps into STFM and TWM, respectively, for spatio-temporal feature capture and motion feature capture. Since the input and output features of STFM and TWM have consistent shapes, the STW block can be easily inserted into any existing 2D CNN. We choose ResNet-50, commonly used by state-of-the-art methods, as the backbone to insert the STW block to ensure a fair comparison. Similar to TSM [26], we place the STW block inside the residual branch of a residual block.

IV. EXPERIMENTS

The following section is divided into three parts. Firstly, we provide an overview of the datasets used and the implementation details of our proposed method. This is followed by an evaluation of the effectiveness of our proposed method, where we compare its performance with state-of-the-art methods on publicly available action recognition datasets. Finally, we conduct ablation studies to analyze the impact of various aspects of our proposed method.

A. DATASETS

Public action recognition datasets can be broadly classified into two categories based on their characteristics: temporal-related datasets and scene-related datasets. In this study, we primarily focus on Something-Something (v1 & v2) datasets [7], as they are particularly sensitive to temporal relations, and our proposed method, STW, is designed to extract temporal reasoning information. However, we also evaluate the performance of STW on scene-related datasets and achieve promising results. Our experimental evaluation is conducted on the following datasets.

1) SOMETHING-SOMETHING v1 & v2

Something-Something v1 & v2 emphasize modeling temporal relations and are therefore referred to as temporal-related datasets. Both versions of Something-Something consist of 174 action classes, with some classes being similar and potentially confusing. Most actions are challenging to recognize accurately without temporal information, such as differentiating between "Pushing something from left to right" and "Pushing something from right to left".

Something-Something v1 contains 108,499 videos, while v2 contains 220,847 videos.

2) KINETICS-400, UCF101, AND HMDB51

UCF101 [9], Kinetics-400 [8], and HMDB51 [10] are early and widely used datasets. These datasets are called scene-related datasets because they consist of many scene-based actions with weak temporal characteristics. Many of the videos in these datasets can be identified by the background or objects in the static frame. For instance, recognizing the basketball or the basketball court is sufficient for predicting the action of ''Playing Basketball'' accurately. Kinetics-400 comprises 240,000 training videos distributed across 400 classes, while UCF101 collected 13,320 videos representing 101 distinct actions. HMDB51 contains 51 classes with a total of 6,849 videos.

B. IMPLEMENTATION DETAILS

We employed ResNet-50 as the backbone of our model and incorporated the STW block into the residual branch of all residual blocks.

1) TRAINING

The model was trained using ResNet-50 as the backbone and the same training strategy as TSN [1]. Firstly, each video was divided into T segments of equal duration. Then, one frame was randomly sampled from each segment, and the resulting T frames were used as input into the network as a clip. The short side of the frame was resized to 256, followed by center cropping and scale-jittering to obtain a cropped image of size 224×224. The outputs of the T frames were averaged to produce the final result. We trained our model using a mini-batch size of 64. For the Something-Something and Kinetics-400 datasets, we used ImageNet pre-trained models for training. The model was trained for 50 epochs, with the learning rate starting from 0.01 and decreasing by 10 at 30, 40, and 45 epochs. For the UCF101 and HMDB51 datasets, we used models pre-trained on Kinetics for finetuning, set the learning rate to 0.001, reduced the learning rate by a factor of 10 for every 15 epochs, and trained for a total of 50 epochs. The optimizer is a mini-batch SGD with a momentum of 0.9 and weight decay of 5e-4. For the Something-Something datasets, we set T to 8 and 16, while for Kinetics-400, UCF101, and HMDB51, T was set to 16.

2) INFERENCE

To match the training strategy, we resized the shorter side to 256, extracted crops, and then resized them to 224×224. For the Something-Something datasets, we used only 1 clip as input, with each clip consisting of either 8 or 16 frames and each frame having 1 crop. On the other hand, for the Kinetics-400, UCF101, and HMDB51 datasets, we utilized 10 clips as input, with each clip containing 16 frames and each frame consisting of 3 crops: left, middle, and right.

C. COMPARISON WITH THE STATE-OF-THE-ART

We compared STW with the current state-of-the-art method, and to ensure a more intuitive and fair comparison, we listed the following details: backbones used, pre-training datasets, number of frames, and GFLOPs.

1) SOMETHING-SOMETHING

Table 1 displays the results of Something-Something v1. STW uses only one clip as input, with each clip consisting of either 8 or 16 frames, and only the middle crop is used for each frame. We only listed methods that utilized RGB as input to ensure a fair comparison. Furthermore, when using 8 frames as input, STW achieved a 26.6% higher accuracy than TSN, which served as the baseline in the validation set. Compared to TSM pre-trained on Kinetics-400, STW only used ImageNet as pre-training, but its accuracy with 8 frames already exceeded TSM accuracy with both 8 and 16 frames, and it was close to the ensemble TSM with 16 and 8 frames. Notably, STW_{16f+8f} achieved the highest validation and test sets accuracy. Compared with 3D methods such as I3D and S3D, STW demonstrated higher accuracy and lower computational overhead.

Table 2 presents the results of Something-Something v2. TSM's result was fused with 10 clips, each containing either 8 or 16 frames and 3 crops per frame, whereas STW utilized only 1 clip that contained 8 or 16 frames, with each frame using only 1 crop. With only 8 frames, STW achieved higher accuracy than all other methods while also having much lower computational overhead, demonstrating its strong temporal modeling capabilities.

It is worth noting that our STW is implemented by inserting STW blocks into the TSN network with ResNet2D-50 as the backbone network. As shown in the GFLOPs column of Table 1 and Table 2, when using ResNet2D-50 as the backbone network and 8 frames, the computation of the TSN network is 33 GFLOPs, while our STW computation is 33.5 GFLOPs. Therefore, in reality, STW only increases computation by less than 2% compared to the baseline, namely TSN ResNet2D-50.

2) KINETICS-400, UCF101, AND HMDB51

We compared STW with numerous competitive methods on Kinetics-400, UCF101, and HMDB51. Given that videos are longer, we used 16 frames as input. For UCF101 and HMDB51, we finetuned STW that was pre-trained on Kinetics-400, and all the accuracies were averaged over the three splits of the datasets. The experimental results are shown in Table 3 and Table 4.

The computational overhead of STW is among the smallest of all the listed methods, except for TSN. On Kinetics-400, STW achieved an accuracy that was 2.6% higher than that of densely sampled TSN. However, the performance improvement was less evident for the Something-Something dataset for two reasons. Firstly, because the temporal characteristics of scene-related datasets are weak, there is no strong temporal

TABLE 1. Comparison with the state-of-the-art on Something-Something v1.

Method	Backbone	Pretrain	Frames	GFLOPs	Top-1 Val	Top-1 Test
TSN [1]	ResNet2D-50	ImgNet	$8f$	33	19.7%	-
TRN-Multiscale [25]	BNInception	ImgNet	$8f$	33	34.4%	33.6%
TRN-Multiscale [25]	ResNet2D-50		$8f$	33	38.9%	-
S3D-G [18]	Inception	ImgNet	$64f$	71	48.2%	-
I3D [37]	ResNet3D-50	ImgNet+ K400	$32f \times 2$	306	41.6%	-
NL I3D [37]	ResNet3D-50			334	44.4%	-
NL I3D+GCN [37]	ResNet3D-50+GCN			606	46.1%	45.0%
ECO [23]	BNIncep+Res3D-18	K400	$16f$	64	41.6%	-
ECO [23]			$92f$	267	46.4%	-
TSM [26]	ResNet2D-50	ImgNet+ K400	$8f$	33	43.4%	-
TSM [26]			$16f$	65	44.8%	-
TSM_{En} [26]			$16f + 8f$	98	46.8%	-
STW_{8f}	ResNet2D-50	ImgNet	$8f$	34	46.3%	41.7%
STW_{16f}			$16f$	67	48.7%	43.9%
STW_{16f+8f}			$16f + 8f$	100	**49.9%**	**45.2%**

TABLE 2. Comparison with the state-of-the-art on Something-Something v2.

Method	Backbone	Pretrain	Frames	GFLOPs	Top-1 Val	Top-1 Test
TSN [1]	ResNet2D-50	ImgNet	$16f \times 10$	1950	30.0%	-
TRN [25]	BNInception	ImgNet	$8f$	33	48.8%	50.9%
TSM [26]	ResNet2D-50	ImgNet+K400	$8f \times 10$	990	59.1%	-
TSM [26]			$16f \times 10$	1950	59.4%	60.4%
TEFE [38]	ResNet2D-50	$8f$	ImgNet	90	59.6	-
TANet [31]	ResNet2D-50	$8f$	ImgNet	34	60.5	-
MKE-Net [39]	ResNet2D-50	$8f$	ImgNet	34	61.2	-
STW_{8f}	ResNet2D-50	ImgNet	$8f$	34	61.7%	60.8%
STW_{16f}			$16f$	67	63.0%	62.1%
STW_{16f+8f}			$16f + 8f$	100	**64.6%**	**63.7%**

TABLE 3. Comparison with the state-of-the-art on Kinetics-400.

Method	Backbone	Pre-train	GFLOPs$\times views$	Top-1	Top-5
I3D_{64f} [6]	Inception v1	ImgNet	$108 \times N/A$	72.1%	90.3%
I3D_{64f}+TSN [1]	Inception v1	ImgNet	$108 \times N/A$	73.5%	91.6%
NL+I3D_{32f} [40]	ResNet-50	ImgNet	70.5×30	74.9%	91.6%
NL+I3D_{128f} [40]	ResNet-101	ImgNet	359×30	77.7%	93.3%
LGD-3D_{128f} [41]	ResNet-101	ImgNet	$N/A \times N/A$	**7**9.4%	**9**4.4%
TSN [1]	Inception V3	ImgNet	3.2×250	72.5%	90.2%
ECO_{En} [23]	BNIncep+Res3D-18	From Scratch	$N/A \times N/A$	70.7%	89.4%
R(2+1)D_{32f} [5]	ResNet-34	Sports-1M	152×10	74.3%	91.4%
S3D-G_{64f} [18]	Inception v1	ImgNet	71.4×30	74.7%	93.4%
$\text{STW}_{16f\times10}$	ResNet-50	ImgNet	67×30	75.1%	91.7%

TABLE 4. Comparison with the state-of-the-art on UCF101 and HMDB51. The accuracies are averaged over all 3 splits.

Method	UCF101	HMDB51
TSN [1]	93.2%	-
I3D_{8f} [6]	95.6%	74.8%
P3D_{8f} [16]	88.6%	-
$\text{S3D-G}_{8f\times10}$ [18]	**96.8%**	75.9%
$\text{R(2+1)D}_{16f\times10}$ [5]	96.8%	74.5%
ECO_{En} [23]	94.8%	72.4%
$\text{STW}_{16f\times10}$	96.2%	73.3%

relationship between the sampled frames, and the effects obtained by interacting information between the frames are limited. Therefore, STW, designed specifically for temporal modeling, may degrade into a common TSN in some categories. Secondly, the accuracy on Kinetics-400 is very high and close to saturation accuracy, leaving little room for performance improvement. Compared with more sophisticated 3D methods such as S3D and R(2+1)D, STW outperformed them with higher accuracy by 0.4% and 0.5%, respectively. On UCF101, the accuracy of STW was higher than I3D but not as good as S3D and R(2+1)D. This could be attributed to the fact that the 3D network only performs 3D convolution on a few consecutive frames, thus providing local temporal modeling. In contrast, STW is a global temporal model for sparsely sampled frames of the entire video. This shows that the semantic change of the entire video is slow, and it is more effective to model atomic actions locally in temporal.

TABLE 5. Camparison of different fusion methods and setting in STFM.

Method	Top-1 Val	Top-5 Val
Average Pooling	32.8%	60.2%
Conv2d(16 : 8/7 : 16) & Concat	39.2%	68.9%
Conv2d(8 : 4/3 : 8) & Concat	41.5%	70.4%
Conv2D(4 : 2 : 4) & Concat	42.6%	73.5%

TABLE 6. Impact of STFM, TWM, and STW.

Method	Top-1 Val	Top-5 Val
TSN	19.7%	46.6%
TSM	42.1%	72.7%
STFM	42.6%	73.5%
TWM	41.9%	72.3%
STW	**46.3%**	**76.6%**

TABLE 7. Comparison of different stage effects.

Stage	Block	Top-1 Val	Top-5 Val
2	3	40.0%	70.2%
3	4	41.1%	71.5%
4	6	**43.9%**	73.7%
5	3	43.1%	**73.8%**
2-5	16	**46.3%**	**76.7%**
3-5	13	45.9%	76.1%
4-5	9	45.6%	76.1%

D. ABLATION STUDIES

In this section, we conducted ablation studies on Something-Something v1. All the models used ResNet-50, pre-trained on ImageNet, as the backbone with an input length of 8 frames.

1) FUSION METHODS AND SETTING IN STFM

Firstly, we conducted an ablation study on the spatio-temporal information fusion method of STFM. As shown in Table 5, the results of reducing the channel number of adjacent 3 frames using 2D convolution and concatenating them are significantly better than directly averaging the information of the three frames. We tried three different settings for channel reduction. When the α, β, and γ were set as $16 : 8/7 : 16$, the information contained in the previous and next frames was too limited because they only retained $1/16$ of the channel number. Thus, the temporal interaction effect was not satisfactory. As the parameters gradually adjusted to $4 : 2 : 4$, we obtained better results. Therefore, we set the STFM with a $4 : 2 : 4$ channel reduction ratio as the final setting.

2) IMPACT OF STFM AND TWM

Then we conducted experiments on STFM and TWM separately. As illustrated in Table 6, compared to training using only TSN, STFM and TWM showed performance improvements of 22.9% and 22.2%, respectively. This confirms that STFM and TWM can capture spatio-temporal and motion features. Furthermore, compared with TSM, STFM's accuracy was 0.5% higher, emphasizing the importance of preserving all channel information. After integrating the two modules, the accuracy of STW was improved by approximately 4%, proving that STFM and TWM exhibit certain coupling and complementarity.

TABLE 8. Comparison of three combinations.

Combination Type	Top-1 Val	Top-5 Val
Parallel	**46.3%**	**76.6%**
Vertical STFM⇒TWM	42.1%	73.3%
Vertical TWM⇒STFM	41.8%	73.5%

3) COMBINATION OF TWO MODULES

After establishing the effectiveness of the two modules, we experimented with different combinations of the modules. As illustrated in Figure 3, there were two ways to combine STFM and TWM: vertical and parallel. In the vertical connection, we experimented with changing the order of STFM and TWM. For parallel connections, we utilized element-wise sum operation to aggregate the output features of STFM and TWM.

As shown in Table 8, the performance of the parallel connection mode was much better than the vertical mode. Notably, the accuracy of vertical connections was lower than that of a single module. The accuracy of STFM⇒TWM was slightly lower than STFM, and the accuracy of TWM⇒STFM was also slightly lower than TWM. This could be attributed to the inherent differences in spatio-temporal features and motion features, and the modules below tended to disrupt the features captured by the modules above, thus reducing accuracy.

4) STAGE TO INSERT STW BLOCKS

We experimented by inserting STW blocks at different stages of ResNet-50. As indicated in Table 7, the performance improved as the stage increased. The performance improvement of stage 4 was greater than stage 5 because stage 4 had 6 blocks inserted, while stage 5 had only 3 blocks. The feature maps captured by ResNet's later layers had a higher semantic level, indicating that capturing spatio-temporal and motion features of high-level semantic features was better than capturing them at lower levels. Therefore, when computing resources were limited, inserting blocks in the later layers was preferable.

E. VISUALIZATION AND QUANTITATIVE ANALYSIS

We have visualized the classification accuracy of Baseline TSN and STW on some highly confusing categories in Something-Something v1, and the visualization results are shown in Figure 4. The categories in the figure are highly confusing, with only minor differences in left-right or front-behind directions. TSN, which only has temporal fusion but lacks temporal modeling and reasoning capabilities, performs poorly in these categories, but its accuracy improves significantly after the insertion of the STW Block. This further

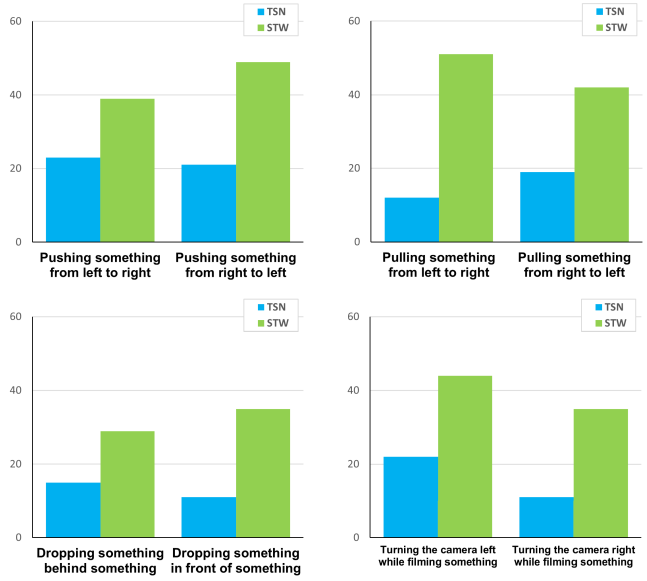

FIGURE 4. Comparing the accuracy of some confusing action categories on Something-Something v1.

demonstrates the STW Block's strong temporal reasoning ability.

V. CONCLUSION

In this paper, we propose a novel temporal modeling approach called the STW network, which is simple yet effective. The proposed model comprises two modules, namely STFM and TWM, each with a distinct role in temporal modeling. The STFM module integrates spatio-temporal information and extracts relevant features, while the TWM module focuses on improving motion characteristic recognition by enhancing the action region. Experimental results demonstrate that the STW network exhibits strong temporal modeling capabilities, outperforming existing state-of-the-art models on popular datasets, including Something-Something, Kinetics-400, UCF101, and HMDB. Furthermore, the proposed model achieves this with less than 2% additional computational overhead, making it a practical and efficient solution. STW network is a promising approach for temporal modeling, which can effectively capture spatio-temporal features and motion characteristics while maintaining computational efficiency.

REFERENCES

[1] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. V. Gool, "Temporal segment networks for action recognition in videos," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 11, pp. 2740–2755, Nov. 2019.

[2] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 568–576.

[3] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3D convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 4489–4497.

[4] A. Santoro, D. Raposo, D. G. T. Barrett, M. Malinowski, R. Pascanu, P. W. Battaglia, and T. Lillicrap, "A simple neural network module for relational reasoning," in *Proc. NIPS*, 2017, pp. 4967–4976.

[5] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri, "A closer look at spatiotemporal convolutions for action recognition," in *Proc. CVPR*, 2018, pp. 6450–6459.

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

[7] R. Goyal, S. E. Kahou, V. Michalski, J. Materzynska, S. Westphal, H. Kim, V. Haenel, I. Fründ, P. Yianilos, M. Mueller-Freitag, F. Hoppe, C. Thurau, I. Bax, and R. Memisevic, "The 'something something' video database for learning and evaluating visual common sense," in *Proc. ICCV*, 2017, pp. 5843–5851.

[8] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev, M. Suleyman, and A. Zisserman, "The kinetics human action video dataset," 2017, *arXiv:1705.06950*.

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

[10] H. Kuehne, H. Jhuang, E. Garrote, T. A. Poggio, and T. Serre, "HMDB: A large video database for human motion recognition," in *Proc. ICCV*, 2011, pp. 2556–2563.

[11] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and F. Li, "Large-scale video classification with convolutional neural networks," in *Proc. CVPR*, 2014, pp. 1725–1732.

[12] S. Ji, W. Xu, M. Yang, and K. Yu, "3D convolutional neural networks for human action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 221–231, Jan. 2013.

[13] G. Varol, I. Laptev, and C. Schmid, "Long-term temporal convolutions for action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 6, pp. 1510–1517, Jun. 2018.

[14] D. Tran, J. Ray, Z. Shou, S. Chang, and M. Paluri, "ConvNet architecture search for spatiotemporal feature learning," 2017, *arXiv:1708.05038*.

[15] C. Feichtenhofer, A. Pinz, and R. P. Wildes, "Spatiotemporal residual networks for video action recognition," in *Proc. NIPS*, D. D. Lee, M. Sugiyama, U. von Luxburg, I. Guyon, and R. Garnett, Eds. 2016, pp. 3468–3476.

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

[17] A. Diba, M. Fayyaz, V. Sharma, A. H. Karami, M. M. Arzani, R. Yousefzadeh, and L. V. Gool, "Temporal 3D convnets: New architecture and transfer learning for video classification," 2017, *arXiv:1711.08200*.

[18] S. Xie, C. Sun, J. Huang, Z. Tu, and K. Murphy, "Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification," in *Proc. ECCV*, 2018, pp. 318–335.

[19] A. Diba, M. Fayyaz, V. Sharma, M. M. Arzani, R. Yousefzadeh, J. Gall, and L. V. Gool, "Spatio-temporal channel correlation networks for action classification," in *Proc. ECCV*, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds., 2018, pp. 299–315.

[20] C. Feichtenhofer, H. Fan, J. Malik, and K. He, "Slowfast networks for video recognition," in *Proc. ICCV*, 2019, pp. 6201–6210.

[21] C. Feichtenhofer, "X3D: Expanding architectures for efficient video recognition," in *Proc. CVPR*, 2020, pp. 200–210.

[22] D. Tran, H. Wang, M. Feiszli, and L. Torresani, "Video classification with channel-separated convolutional networks," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Seoul, South Korea, Oct. 2019, pp. 5551–5560.

[23] M. Zolfaghari, K. Singh, and T. Brox, "ECO: Efficient convolutional network for online video understanding," in *Proc. ECCV*, in Lecture Notes in Computer Science, vol. 11206. Cham, Switzerland: Springer, 2018, pp. 713–730.

[24] J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, T. Darrell, and K. Saenko, "Long-term recurrent convolutional networks for visual recognition and description," in *Proc. CVPR*, 2015, pp. 2625–2634.

[25] B. Zhou, A. Andonian, A. Oliva, and A. Torralba, "Temporal relational reasoning in videos," in *Proc. ECCV*, 2018, pp. 831–846.

[26] J. Lin, C. Gan, and S. Han, "TSM: Temsporal shift module for efficient video understanding," in *Proc. ICCV*, 2019, pp. 7082–7092.

[27] Z. Liu, D. Luo, Y. Wang, L. Wang, Y. Tai, C. Wang, J. Li, F. Huang, and T. Lu, "TEINet: Towards an efficient architecture for video recognition," in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, no. 7, pp. 11669–11676, Apr. 2020.

[28] L. Wang, Z. Tong, B. Ji, and G. Wu, "TDN: Temporal difference networks for efficient action recognition," in *Proc. CVPR*, 2021, pp. 1895–1904.

[29] H. Shao, S. Qian, and Y. Liu, "Temporal interlacing network," in *Proc. AAAI*, 2020, pp. 11966–11973.

[30] C. Yang, Y. Xu, J. Shi, B. Dai, and B. Zhou, "Temporal pyramid network for action recognition," in *Proc. CVPR*, 2020, pp. 588–597.
[31] Z. Liu, L. Wang, W. Wu, C. Qian, and T. Lu, "TAM: Temporal adaptive module for video recognition," in *Proc. ICCV*, 2021, pp. 13688–13698.
[32] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. NIPS*, I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, and R. Garnett, Eds. 2017, pp. 5998–6008.
[33] A. Arnab, M. Dehghani, G. Heigold, C. Sun, M. Lucic, and C. Schmid, "ViViT: A video vision transformer," in *Proc. ICCV*, 2021, pp. 6816–6826.
[34] G. Bertasius, H. Wang, and L. Torresani, "Is space-time attention all you need for video understanding?" in *Proc. ICML*, M. Meila and T. Zhang, Eds. vol. 139, Jul. 2021, pp. 813–824.
[35] K. Li, Y. Wang, P. Gao, G. Song, Y. Liu, H. Li, and Y. Qiao, "Uniformer: Unified transformer for efficient spatiotemporal representation learning," 2022, *arXiv:2201.04676*.
[36] Z. Liu, J. Ning, Y. Cao, Y. Wei, Z. Zhang, S. Lin, and H. Hu, "Video Swin transformer," 2021, *arXiv:2106.13230*.
[37] X. Wang and A. Gupta, "Videos as space-time region graphs," in *Proc. ECCV*, 2018, pp. 413–431.
[38] J. Jiang and Y. Zhang, "An improved action recognition network with temporal extraction and feature enhancement," *IEEE Access*, vol. 10, pp. 13926–13935, 2022.
[39] Q. Tian, K. Wang, B. Liu, and Y. Wang, "Multi-kernel excitation network for video action recognition," in *Proc. ICSP*, vol. 1, 2022, pp. 155–159.
[40] X. Wang, R. B. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proc. CVPR*, 2018, pp. 7794–7803.
[41] Z. Qiu, T. Yao, C. Ngo, X. Tian, and T. Mei, "Learning spatio-temporal representation with local and global diffusion," in *Proc. CVPR*, 2019, pp. 12056–12065.

ZHENGBAO CAI received the bachelor's degree in computer science and technology from Anhui Jianzhu University, in 2006, and the master's degree in computer technology from Anhui University, in 2011.

He was a Research Fellow with the Key Laboratory of Computational Intelligence and Signal Processing, Ministry of Education, Anhui University. He has been a Visiting Scholar with Anhui University. He is currently an Associate Professor and a Research Expert in computer technology with the Anhui Vocational College of Defense Technology, China. His research interests include artificial intelligence technology and information security technology applications.