IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 29 December 2023, accepted 7 February 2024, date of publication 9 February 2024, date of current version 1 March 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3364810

RESEARCH ARTICLE

Short-Term Action Learning for Video Action Recognition

LIU TING-LONG
Center for Information Technology, Dalian Polytechnic University, Dalian 116034, China
e-mail: liutl@dlpu.edu.cn

This work was supported by Liaoning Province Science and Technology Plan Project under Grant 2020JH2/10100032.

ABSTRACT For a long-term complex Action, it is typically composed of various short-term Actions. The speed and importance of these short-term Actions directly affect the recognition results. Current two-stream neural networks have already achieved good recognition results on Action recognition datasets. However, previous two-stream networks have focused more on Action modeling, neglecting the impact of the speed and importance of different short-term Actions on the results of Action recognition. This has directly limited the model's ability to model different short-term Actions, thereby affecting the effectiveness of Action recognition. To address this issue, this paper proposes a Short-term Action Spatio-Temporal Attention (STASTA) module based on the two-stream network structure. The STASTA module is capable of focusing on the differences in importance and speed between different short-term Actions. By extracting the differences in importance and speed of different short-term Actions in the video and then fusing the features, the aim is to enrich spatio-temporal features and improve Action recognition performance. The proposed method is evaluated on the Something-Something v1 & v2 and Charades datasets. A large number of experimental results indicate that the method proposed in this paper achieves state-of-the-arts results among video Action recognition methods.

INDEX TERMS Short-term action, two-stream CNN, action visual tempo, action visual importance, video action recognition.

I. INTRODUCTION

Video action recognition is one of the most challenging tasks in the field of computer vision [1], [2], [3], [4], [5], [6], [7], [8], [9], [10]. The aim of video action recognition is to extract a large amount of action information from the raw video through effective spatio-temporal modeling techniques. In real life, a long video action is typically composed of multiple short-term actions. These short-term actions influence the longer action in two main aspects: first, the longer action appears differently due to the varying speeds of the short-term actions; and second, different short-term actions exhibit different levels of importance for the final judgment of the longer action. These short-term actions typically appear in the video as consecutive video frames; they provide rich content information for the final video action recognition. For example, in the video action of "a person opens the fridge, drinks milk, and closes the fridge," there are multiple short-term activities such as "walking to the fridge," "opening the fridge door," "taking out the milk bottle," "opening the milk cap," "drinking milk," and "closing the fridge door." These short-term activities provide rich content information for the final recognition result. But these short-term activities differ in importance; Compared to other short-term activities, walking to the refrigerator is relatively less important. For instance, "a person throws a cushion at another person" and "a person hands a cushion to another person." Although both include short-term movement Actions such as "a person raises the cushion," "the cushion moves," and "another person holds the cushion," the speed of the cushion's movement is not the same; the speed of the "throwing" action is faster than that of the "passing." From the two examples above, we can see that in order to accurately recognize a video action,

The associate editor coordinating the review of this manuscript and approving it for publication was Chuan Li.

it is necessary to fully consider the importance of multiple short-term actions and the issue of movement speed for different short-term actions. Therefore, effectively modeling the information of these short-term actions is an important aspect of video action recognition.

Recently, some methods have been proposed to address these issues [3], [6], [11], [12], [13]. SlowFast [12] By using two parallel Convolutional Neural Networks to analyze video clips, two distinct pathways are established for the purpose of video action recognition. The slow pathway functions at a reduced frame rate, whereas the fast pathway operates at an elevated frame rate. The skeleton network adeptly merges the fast and slow streams of information, resulting in a substantial enhancement of performance across both temporal dimensions for the processed Action instances. However, this method does not fully consider the impact of fine-grained features in low-level features on the Action. Additionally, using different frame sampling rates leads to relatively high computational complexity. TCM [3] extract motion speed features from low-level features and analyze global expression features through speed variations. TCM resolves the issue of classifying Actions with high similarity in speed variations and actions. However, this method does not fully consider the impact of different factors in the importance of short-term activities on Action recognition, limiting the model's capacity to model multiple short-term activities. TPN [11] constructs a temporal pyramid to obtain visual speed information at the feature level. TPN has demonstrated sustained enhancements on certain action datasets, yet they place an undue burden on the temporal modeling prowess of the estimation network, which limits the ability to capture low-level features and thus affects the results of action recognition. These excellent network models can effectively model actions and speed, but they ignore the importance and speed of different short-term activities, leading to a limitation in the ability to model multiple short-term activities. Recently, FSformer [13] uses an optimized Transformer approach to learn the importance of different short-term activities and achieves good accuracy for Actions containing multiple short-term activities. FSformer divides the input into high-frame and low-frame video paths, and the proposed model separates spatial features from temporal features, fusing spatial features into temporal features through an attention mechanism. The model achieves the best results in video action recognition performance, but it only separates high and low video frames overall without fully considering the impact of different short-term Action visual speeds within video frames on action recognition. Inspired by these observations, we consider designing a model on a two-stream network to extract visual speed and importance features of different short-term Actions, thereby achieving the goal of improving recognition accuracy and reducing computational overhead.

To address the issue of insufficient extraction of short-term actions features in existing models, this paper designs a two module architecture based on a two-stream network, seeking a balance between computational accuracy and computational overhead. In the first module, the model fully considers the features of the importance of short-term activities. In the second module, the model is designed to consider the speed variation features of different short-term actions. Finally, residual connections are used to fuse the features, and the fused features are used as the input for Action classification, which is performed by a classification model. To demonstrate the effectiveness of the model, we implement it using ResNets. Experiments were conducted on the Something-Something v1 & V2 [14] and Charades [15], and the comparison results with recent state-of-the-arts models are shown in Fig.1. With a relatively small sacrifice in computational cost, we achieved the best recognition results. We also performed ablation experiments and visualization analysis, which fully demonstrated the effectiveness of each module of the proposed model. As summarized, our main contributions include the following three points:

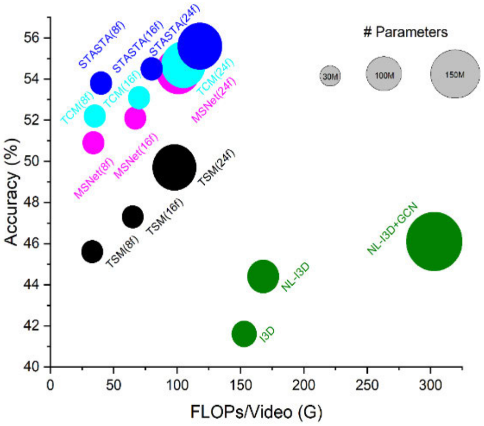

FIGURE 1. **Comparison of Top 1 values for video action classification on the Something-Something V1 dataset, along with computational complexity and model size.**

- We designed the Short-term Action Spatio-Temporal Attention network architecture to capture pixel-level fine-grained temporal dynamics information, including both slow visual tempo and fast visual tempo actions. This addresses the issue of inaccurate recognition that arises from previous two-stream network methods neglecting the impact of the speed and importance of different short-term actions on the results of action recognition.
- We proposed the Short-Term Action Feature Attention and Fast-Slow Feature Fusion (ST-AFS-FF) module and the Displacement Map Temporal Attention (DM-TA) module. The ST-AFS-FF module is an action recognition module that can learn the different importance of various short-term action features. The DM-TA module is a module that can learn pixel-level action feature information and enhance the expression capability of temporal features; it can effectively capture the visual speed information of different short-term actions.

- We evaluate the proposed method on two challenging action datasets: Something-Something v1 & v2 and Charades. Experimental results show that we achieved state-of-the-arts performance.

II. RELATED WORK

A. VIDEO ACTION RECOGNITION

At present, deep learning methods [16], [17], [18], [19], [20], [21], [22], [23] are the mainstream in video action recognition. The prevailing deep learning methodologies can be broadly categorized into two distinct realms: the first being 3D Convolutional Neural Networks, and the second, 2D Convolutional Neural Networks. 3D Convolutional Neural Networks model spatial and temporal semantic information through 3D convolutional kernels. Additionally, to enhance performance, many related variants have emerged [4], [24], [25], [26]. Among these, the most famous is the Non-local [4]. The Non-local network utilizes a non-local operation to better leverage long-range temporal dynamics information. However, the biggest drawback of 3D models is their high computational demand, which requires excessive computational resources. 2D Convolutional Neural Networks use 2D kernels to obtain spatial semantic information and then aggregate temporal dynamic information through a module [27], [28], [29], [30]. The most famous 2D Convolutional Neural Network is the two-stream network [31]. In the two-stream network, one stream is used to learn RGB appearance feature information, while the other stream is used to learn optical flow motion feature information. Finally, the spatiotemporal information is aggregated through average pooling. Scholars have researched and developed numerous models based on this design to enhance the feature extraction capabilities of temporal information [10], [30], [32]. TEA [30] calculates the temporal differences in the feature layers from spatiotemporal features, and uses these differences to enhance the channels in the feature map that are related to motion information. MotionSqueeze [32] proposes an internal and lightweight motion feature learning module that can effectively extract motion features while reducing the computational cost more effectively than optical flow feature extraction. AGPN [9] is an Action Granularity Pyramid Network; AGPN can effectively utilize multi-granular spatiotemporal information of Actions; it has good recognition effects for Action identification of complex spatiotemporal content information; this model pays more attention to the fine-grained information in spatiotemporal information. MVFNet [10] is a visual fusion network that strives to balance the performance and efficiency metrics of the model; however, MVFNet does not consider the importance of the correlation between multi-scale spatiotemporal features for action recognition. TDN [33] can effectively obtain multi-scale temporal information; it can make full use of temporal difference operations and systematically evaluate their impact in short-term and long-term motion models. TDN aggregates short-term features into long-term features, resulting in an inability to effectively extract short-term features. In summary, although these methods provide the ability to learn fine-grained temporal dynamics of adjacent frames, they all neglect the learning ability of motion visual speed, a key element.

B. VIDEO ACTION SPEED LEARNING MODEL

Recently, scholars have proposed a large number of models for video action speed learning [3], [6], [11], [12], [13]. SlowFast [12] uses two pathways, slow and fast, to represent motion visual speed. Features from both pathways are processed separately, and then temporal information features are integrated into spatial information features through lateral connections. Although this approach improves accuracy, the presence of multiple branch networks leads to excessive computational complexity. Additionally, it does not make full use of the varying importance of short-term actions and the different features of action speed in different Actions, resulting in less than ideal effects for long Actions containing multiple short-term actions. TPN [11] proposes a temporal residual network based on feature layers that can be easily integrated into 2D or 3D skeleton networks, capable of effectively obtaining features of Action instances at different speeds. However, TPN highly depends on the temporal modeling capabilities of the skeleton network, which greatly limits its performance. TPN cannot utilize low-level features and cannot make full use of long-range, fine-grained temporal dynamics features from higher levels at a distance. To address this issue, correlation operations can be used to establish pixel-level matching values for features at different scales and utilize the information of motion visual speed in the video. FSformer [13] uses a temporal convolutional network to encode the original video, taking the fast and slow paths of RGB as input, and distinguishes the importance of short-term movements in video actions through different attention modules for short-term Actions, effectively enhancing the accuracy of action recognition. However, the model does not consider the impact of different short-term Action speeds on the action recognition results, and the input of dual-path frame resolution increases the computational complexity of the model. Inspired by this model, to address the issue that existing two-stream network models do not fully consider the different importance of short-term Actions and the insufficient modeling capabilities of the overall Action recognition due to the different speeds of short-term Actions, we propose using correlation operations to establish pixel-level matching values for features at different scales. We extract the importance of different short-term activities from the features of the fast and slow paths, followed by feature fusion. The processed features are input into a motion estimation feature attention module to extract features of different short-term motion speeds. Finally, the fused features and the original features are connected residually for output.

III. THE APPROACH

The overall architecture of video action recognition is shown in Fig.2 Below, we will describe our proposed method in detail. First, the visual content of the input video is processed through a temporal action skeleton network(I3D [34]) encoded into a sequence of features; Temporal dynamics features at fast and slow speeds are extracted from the feature sequence using correlation operations. Then, the extracted fast and slow temporal features are input into our core module, the Short-Term Action Spatial-Temporal Attention Module. The module is mainly composed of the Short-Term Action Feature Attention and Fast-Slow Feature Fusion (ST-AFS-FF) module and the Displacement Map Temporal Attention (DM-TA) module. The ST-AFS-FF module primarily learns the different importance of short-term Actions on the slow temporal features and then fuses them into the fast temporal features. Finally, it merges the sequence features of the two speeds. The DM-TA module mainly accomplishes the acquisition of visual speed information for short-term Actions. It takes the slow temporal features and fast temporal features generated by the ST-AFS-FF as inputs for the DM-TA module to estimate the visual speed similarity, deciding the most effective visual speed information. Finally, the fused Action visual features and the original input features are merged as the input for the final result prediction.

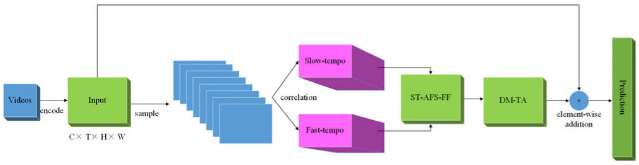

FIGURE 2. Overall architecture of the STASTA model.

A. ST-AFS-FF MODULE

The ST-AFS-FF module is an action recognition module that can learn the different importance of various short-term motion features. These motion features are densely distributed throughout the video frames, providing rich semantic information, which is precisely these short semantic snippets that constitute the long-range motion Action information. We encode the features of the slow path and the fast path into input features for different temporal dimensions T through the action recognition network. Then, we use the ST-AFS-FF module to learn the different importance of short-term Actions within these input features. The architecture of the ST-AFS-FF module is shown in Fig.3.

1) ST-ATA MODULE

The Short-Term Action Feature Attention Module (ST-ATA) is capable of effectively learning the importance of different short-term Actions, dividing them into positive correlations and negative correlations. The ST-ATA module takes slow features as input because they possess higher spatial features compared to fast features, which enables the provision of more abundant motion and semantic information, thus better distinguishing between positive and negative correlations. As shown in the lower left part of Fig.3, the ST-ATA includes an up-sampling layer, a linear layer projection, and a temporal convolution layer. The up-sampling layer's role is to convert the temporal dimensions of the slow features to the same dimensionality as the fast feature temporal dimensions. The linear layer projection is used to project the channel size to match the fast features. Finally, the temporal convolution layer generates attention weights within the features. These generated weights are applied to the fast feature path features through matrix addition operations, thus completing the internal feature attention mechanism. The ST-ATA module distinguishes the importance of different short-term Actions by adding more biases at the beginning of the path. This approach enables the model to focus more on learning short-term Action feature information that is positively correlated with the action recognition results, thereby achieving a more effective action recognition model.

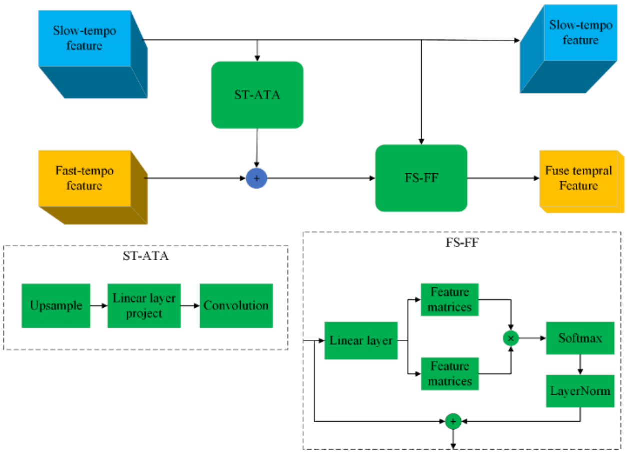

FIGURE 3. Architecture of the ST-AFS-FF module. The upper part includes the ST-ATA module and the FS-FF module. The lower part shows the detailed structure of the ST-ATA and FS-FF modules.

2) FS-FF MODULE

The Fast and Slow Feature Fusion (FS-FF) module is designed to integrate the original features from the input slow path and the fused features from the ST-ATA module from a global perspective, in order to better learn the semantic information of motion. As shown in the lower right part of Fig.2, the FS-FF module mainly includes a linear layer, a Softmax layer, and a LayerNorm layer. The module uses residual connections to fuse the original features with the processed features. The input to the FS-FF module is the output of the ST-ATA module and the original features from the slow path. The original slow path features need to be upsampled first to match the temporal dimensions of the fast path features. After linear layer projection, two feature matrices are generated. The features processed by the Softmax layer serve as an intermediate layer, ensuring that the relevance of all input features is calculated. This achieves the calculation of the correlation between each input feature and the fused features from the ST-ATA module through the FS-FF module. Then, the LayerNorm layer enables the model to maintain stable training. Finally, by using residual connections to fuse the input features from both paths with the output of the LayerNorm, the module maximizes the utilization of spatiotemporal feature information.

B. DM-TA MODULE

The Displacement Map Temporal Attention (DM-TA) module is a module that can learn pixel-level motion feature information and enhance temporal feature expression, effectively capturing visual speed information of different short-term activities. By utilizing temporal cross-interaction operations to supervise the supervision of unimportant information, followed by 1D temporal convolution operations, the obtained action visual features and the original input features are connected through residual connections, which serve as the final prediction inputs. The detailed structure of the model is shown in Fig.4.

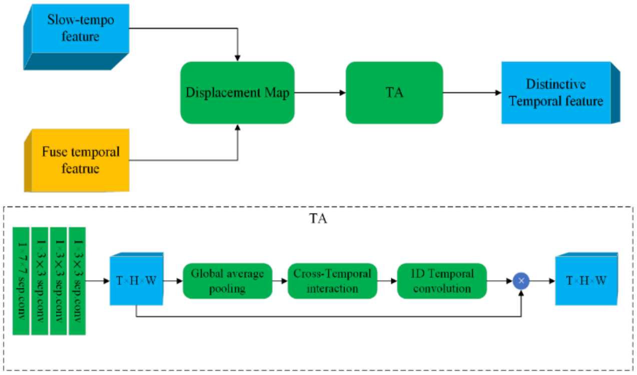

FIGURE 4. Architecture of the DM-TA module. It mainly includes the displacement map module and the temporal attention module.

1) DISPLACEMENT MAP

The Displacement Map includes slow spatial information and temporal motion information that can distinguish the importance of short-term actions. In order to have the ability to estimate motion features, it is necessary to establish associations between various cross-features and direct features. Following the method of MotionSqueeze [32], this can be completed by calculating the confidence map extracted from the fused temporal feature and its motion information. The two-channel fused temporal feature and the single-channel confidence map are connected to generate the Displacement Map. Finally, the temporal dimension is transformed to unify the temporal dimensions.

2) TA MODULE

The goal of the Temporal Attention (TA) module is to utilize the temporal dynamic information generated by the Displacement Map to enhance the distinguishability of unimportant temporal dynamic interaction features. First, four convolutional layers are used to transform the input Displacement Map. This transformation can enhance the visual speed features. After global pooling, the spatial information is summarized to obtain attention weights, which can activate channels sensitive to motion information. Cross-temporal interaction ensures that while motion information is enhanced, direct interactions between temporal channels are also established. After temporal convolution operations, weights for the temporal information are generated. Finally, the generated features are fused with the original features through element-wise multiplication, creating features with a temporal attention mechanism. Processing through this module can effectively enhance fast and slow visual speed information while suppressing unimportant information, achieving the purpose of extracting features of visual speed for different short-term actions.

IV. EXPERIMENTS

A. DATASETS

Something-Something V1 & V2 [15]. The dataset is a collection of large numbers of annotated video clips that demonstrate people performing predefined basic actions with everyday items. This dataset was created by a large number of people, and these actions occur in the real world, allowing machine learning models to develop a fine-grained understanding of basic actions. Since the dataset mainly involves interactions between people and everyday items, it requires the model to pay more attention to the details of temporal information. Our method mainly focuses on experimentation and evaluation on Something-Something V1&V2, and we only report our results on the Charades dataset to demonstrate the generalization ability of our model.

Charades [14]. The dataset consists mainly of indoor activities from daily life, with an average video length of 30 seconds. It involves interactions with 46 object classes in 15 indoor scenes and includes 157 action classes. Due to the continuous changes in time and different actions, it becomes a very challenging dataset. The videos contain temporary repetitive actions, requiring the model to predict multiple categories within the videos and to predict the importance of these categories.

B. IMPLEMENTATION

We employ the same training strategy as TSM [35]. We initialize our model using the ImageNet pretrained weights of ResNet50 [36]. For the Something-Something V1 & V2 datasets, the training parameters are as follows: the number of iterations is 50, the batch size is 32, the initial learning rate is 0.01 (decreasing by 0.1 at iterations 30, 40, and 45), the weight decay is 5e-4, and the dropout rate is 0.5. All experimental implementations were using Pytorch on ubuntu system with a single RTX 3090 GPU.

C. COMPARISON WITH STATE-OF-THE ARTS

In this section, we compare the performance of our model with the best-performing model methods on the Something-Something V1 & V2 and Charades datasets. The experimental results include accuracy (Top-1, Top-5, mAP),

TABLE 1. Compares the best performance on the Something-Something V1 & V2 datasets. Most results are copied from the corresponding references. A dash "-" indicates that the result is not provided in the reference.

Method	Frame	Params	FLOPs×clips	Sth-Sth V1		Sth-Sth V2	
				Top-1(%)	Top-5(%)	Top-1(%)	Top-5(%)
I3D from[25]	32	28.0M	153G×2	41.6	72.2	-	-
NL-I3D from[25]	32	35.3M	168G×2	44.4	76.0	-	-
NL-I3D+GCN[25]	32	62.2M	303G×2	46.1	76.8	-	-
S3D-G[26]	64	11.6M	71G×2	48.2	78.7	-	-
CorrNet-101[37]	32	-	224G×30	51.7	-	-	-
CIDC[38]	32	87M	92G×30	-	-	56.3	83.7
RubiksNet[39]	8	-	33G×1	46.4	74.5	58.8	85.6
3D DenseNet121[19]	16	21.4M	31G×1	50.2	78.9	62.9	88.0
TSM[35]	8	24.3M	33G×1	45.6	74.2	58.8	85.4
TSM[35]	16	24.3M	65G×1	47.3	77.1	61.2	86.9
TSM$_{en}$[35]	16+8	48.6M	98G×1	49.7	78.5	62.9	88.1
MSNet[32]	8	24.6M	34G×1	50.9	80.3	63.0	88.4
MSNet[32]	16	24.6M	67G×1	52.1	82.3	64.7	89.4
MSNet$_{en}$[32]	16+8	49.2M	101G×1	54.4	83.8	66.6	90.6
MSNet$_{en}$[32]	16+8	49.2M	101G×10	55.1	84.0	67.1	91.0
TDN-R50[33]	8	-	36G×1	52.3	80.6	64.0	88.8
TDN-R50[33]	16	-	72G×1	53.9	82.1	65.3	89.5
TDN-R50[33]	16+8	-	108G×1	55.1	82.9	67.0	90.3
TDN-R101[33]	16+8	-	198G×1	56.8	84.1	68.2	91.6
AGPN[9]	8	27.6M	-×1	51.6	80.9	63.1	88.6
AGPN[9]	16	27.6M	-×1	54.1	82.7	65.5	90.1
AGPN[9]	16	27.6M	-×10	55.0	83.9	67.0	90.9
TCM[3]	8	24.5M	35G×1	52.2	80.4	63.5	88.7
TCM[3]	16	24.5M	77G×1	53.1	81.2	65.1	89.6
TCM[3]	16+8	49.0M	105G×1	54.7	82.6	66.7	90.7
TCM[3]	16+8	49.0M	105G×10	57.2	85.2	67.8	92.2
STASTA(Ours)	8	24.8M	40G×1	53.8	81.3	64.8	89.6
STASTA(Ours)	16	24.8M	80G×1	54.5	82.9	66.1	90.2
STASTA(Ours)	16+8	49.3M	118G×1	55.6	83.4	67.2	91.3
STASTA(Ours)	16+8	49.3M	118G×10	**57.9**	**86.4**	**68.5**	**92.8**

the number of frames (Frame), floating-point operations per clip (FLOPs×clips), and the number of parameters (Params). As shown in Table.1 and Table.2. Table.1 shows the performance of the latest 26 action recognition methods on the Something-Something V1 & V2 datasets. The table is divided into three parts: the top section represents the 3D convolutional neural network model, the middle section represents the 2D convolutional neural network model, and the bottom section represents our proposed neural network model. Under no additional conditions, the accuracy of our method on the Something-Something V1 dataset reached 57.9%, exceeding the TSM, TDN, and TCM methods by at least 0.7%, 1.5%, 2.2%, and 7% respectively when the input frame is 8. Furthermore, compared to the basic method of 3D models, such as 3D DenseNet121 with 16 frames input, our method improves Top-1 accuracy by 4.3% (54.5% VS. 50.2%) on the Something-Something V1 dataset and by 3.2% (66.1% vs. 62.9%) on the Something-Something V2 dataset. Compared to RubiksNet, the performance is enhanced by 7.4% (53.8% vs. 46.4%) and 6% (64.8% vs. 58.8%). In comparison with the I3D series models on the Something-Something V1 dataset, our model achieves a higher Top-1 accuracy under the highest floating-point operation scenario, which is 1.8% higher than the highest Top-1 value of NL-I3D+GCN, while requiring fewer floating-point operations (118G vs. 303G).

TABLE 2. Compares the results with the best methods on the Charades dataset.

Method	Frame	FLOPs×clips	Charades mAP(%)
Two-Stream CNN[31]	-	-	18.6
TGM[40]	-	1.2G×1	20.6
Coarse-Fine[18]	-	-	25.1
MultiScale TRN[17]	-	-	25.2
Co Slow[16]	8	6.9G×1	21.5
Slow[16]	8	54.9G×1	24.1
I3D[34]	-	-	32.9
SlowFast-R50[12]	8	50.6G×30	37.2
SlowFast-R101[12]	8	96.8G×30	38.9
STASTA(ours)	8	40G×1	39.5
STASTA(ours)	16	40G×1	**41.3**

Table.2 presents a comparison of the performance of our model and the latest state-of-the-arts action recognition models on the Charades dataset. It can be clearly observed that our model achieves outstanding performance. First, the evaluation accuracy of our 8-frame input network surpasses that of the 64-frame I3D method (mAP: 39.5% vs. 32.9%);

When compared to the SlowFast-R101 model, we find that with a 0.6% improvement in accuracy (mAP: 39.5% vs. 38.9%), the GFLOPs are reduced by more than two times (40G vs. 96.8G). In addition, the results of Two-stream CNN and TGM demonstrate that they are not effective in handling videos with multiple action labels. These results show that our model is more precise and effective in modeling the temporal information for classifying multiple short-term actions. When the input frame is 16, the average accuracy of our model reaches 41.3%. This is because the additional input frames enrich the temporal information. Temporal information plays a crucial role in the dataset.

To further analyze the performance of our model, we conducted a comparative analysis based on the model's pathway categories. First, we compared methods that aggregate short-term and long-term temporal features along a single pathway, such as TDN. The results in Table.1 show that our method outperforms TDN on both the Something-Something V1 & V2 datasets. Then, we compared our method with those that use visual speed, such as SlowFast and TCM. The results indicate that our model exhibits a greater performance improvement over these methods. These experimental results suggest that the proposed method can effectively extract temporal feature information of short-term movements and distinguish between the speed and importance of features.

D. ABLATION STUDY

In this section, we study the effectiveness of different modules by conducting a series of ablation experiments. The model uses 8 frames as input and is tested on the Something-Something V1 dataset, with evaluation performed using the validation set.

1) THE EFFECTIVENESS OF EACH MODULE

As shown in Table.3, we present the contribution of the two modules introduced in this paper. From the table, it can be observed that each of the modules we designed can improve the performance of the baseline, but with different levels of contribution. For the first module, Short-Term Action Feature Attention and Fast-Slow Feature Fusion (ST-AFS-FF), the addition of this module increases the Top-1 value from 52.2% to 53.4%, a gain of 1.2%; and the Top-5 value from 62.7% to 64.1%, a gain of 1.4%. This indicates that the module can focus on regions related to short-term actions and make full use of spatiotemporal information to distinguish between different action actions. When the second module, Displacement Map Temporal Attention (DM-TA), is added to the network alone, the Top-1 and Top-5 values improve by 0.4% (52.6% vs. 52.2%) and 0.7% (63.4% vs. 62.7%), respectively. The experimental results show that this module enhances the performance of action recognition, suggesting that it can focus on regions related to visual speed information in short-term actions and obtain visual speed features of different actions. Finally, by observing the last row of the table, we find that the model achieves the maximum performance improvement after adding both modules (Top-1: 53.8% vs. 52.2%, Top-5: 64.8% vs. 62.7%). This result effectively demonstrates the effectiveness of the model we propose.

TABLE 3. Performance comparison of different modules in the STASTA model.

ST-AFS -FF	DM-TA	Top-1 (%)	Top-5 (%)	Δ Top-1 (%)	Δ Top-5 (%)
-	-	52.2	62.7	baseline	baseline
√	-	53.4	64.1	+1.2	+1.4
-	√	52.6	63.4	+0.4	+0.7
√	√	53.8	64.8	+1.6	+2.1

2) THE ORDER OF THE MODULES

We conducted ablation experiments by swapping the order of the two modules, as shown in Table.4. We adjusted the sequence of the modules and found that when the model's order is DM-TA followed by ST-AFS-FF, the Top-1 and Top-5 values improve by 1.4% (53.6% vs. 52.2%) and 1.6% (64.3% vs. 62.7%), respectively. When the model's order is ST-AFS-FF followed by DM-TA, the Top-1 and Top-5 values show an even greater improvement, reaching 1.6% (53.8% vs. 52.2%) and 2.1% (64.8% vs. 62.7%), respectively. The experimental results indicate that the model primarily focuses on temporal feature information; the acquisition of fine-grained temporal information can better enhance performance. Placing DM-TA after the temporal features yields the best performance.

TABLE 4. Impact of the order of modules ST-AFS-FF and DM-TA on performance.

Sequence	DM-TA	Top-1 (%)	Top-5 (%)	Δ Top-1 (%)	Δ Top-5 (%)
-	-	52.2	62.7	baseline	baseline
DM-TA	ST-AFS-FF	53.6	64.3	+1.4	+1.6
ST-AFS-FF	DM-TA	53.8	64.8	+1.6	+2.1

E. DISCUSSION

This section reviews a series of dialogues exploring the method. Experimental findings reveal that incorporating diverse types of datasets enhances accuracy. These results suggest that the proposed approach exhibits a degree of generalizability. However, current experiments have been conducted exclusively on two-stream 2D Convolutional Neural Networks (CNNs) and have not been extended to 3D CNNs. It is imperative to experiment with 3D CNNs to evaluate the generalizability and performance of the method comprehensively.

F. VISUALIZATION

We used Grad-CAM [41] to perform visualization analysis on a video of the action "a person opens a fridge, drinks all the milk from a pitcher, and then closes the fridge" in the Charades dataset. We compared our results with three models: MultiScale TRN, I3D, and SlowFAST-R50 (8-frame input). As shown in Fig.5, the visualization results indicate that the baseline model MultiScale TRN cannot

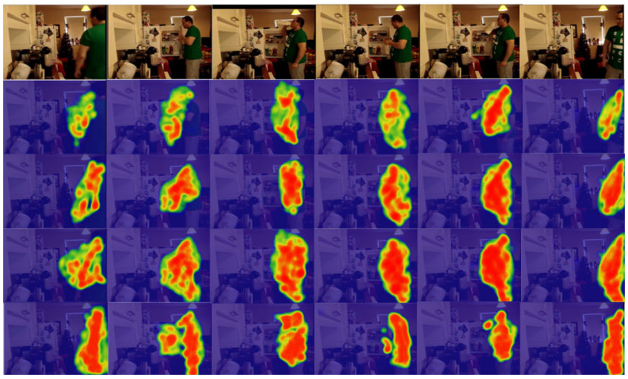

FIGURE 5. **Visualization of activation layer maps using Grad-CAM on the charades dataset. The first row is the original video frames, the second row is MultiScale TRN, the third row is I3D, the fourth row is SlowFAST-R50-8, and the fifth row is our STASTA. We have only visualized specific frames.**

completely focus on the motion area. I3D can capture the content of the action, but the motion location is not accurate. SlowFAST-R50 can focus on the action content, but the motion location and detail acquisition are not precise enough. Our model can more accurately identify the entire motion cycle and the specific location of actions, with a more precise focus on the motion location. In this example, we found that our model can track the moving hand, fridge door, and milk throughout the video. However, other methods cannot effectively extract motion information and moving objects. This example shows that our method can identify the entire motion cycle, represent coarse-grained motion feature information, and pay more attention to multiple short-term actions. At the same time, the specific location of the motion can identify fine-grained action information more accurately than other methods. This fully demonstrates that the model can effectively obtain multi-granular action information.

Through the visualization of the activation maps in Fig.5, an example from the Charades dataset is shown. It can be seen that our model performs better in recognizing multiple short-term actions. The model is able to extract multi-granular action information and accurately recognize even with slight changes in movement.

V. CONCLUSION

We have proposed an innovative spatiotemporal attention model for short-term actions to handle multiple short-term actions in videos. The model includes the ST-AFS-FF and DM-TA modules. The ST-AFS-FF module effectively captures fine-grained temporal dynamics of different short-term movements in temporal features, enabling discrimination between action features of varying importance. The DM-TA module effectively extracts visual velocity features of movements at different speeds from temporal features, enhancing effective visual velocity information by considering cross-temporal dynamic interactions. Extensive experiments on two representative datasets have verified the effectiveness of this method. In the future, we will integrate our modules into other outstanding models to further improve the performance of action recognition, and explore how to enhance the precision of human localization and delve deeper into the intricacies of human interactions.

ACKNOWLEDGMENT

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

REFERENCES

[1] S. Xie, C. Sun, J. Huang, Z. Tu, and K. Murphy, "Rethinking spatiotemporal feature learning for video understanding," 2017, *arXiv:1712.04851*.

[2] J. Stroud, D. Ross, C. Sun, J. Deng, and R. Sukthankar, "D3D: Distilled 3D networks for video action recognition," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis.*, Mar. 2020, pp. 625–634.

[3] Y. Liu, J. Yuan, and Z. Tu, "Motion-driven visual tempo learning for video-based action recognition," *IEEE Trans. Image Process.*, vol. 31, pp. 4104–4116, 2022.

[4] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7794–7803.

[5] M. Sun, W. Wang, X. Zhu, and J. Liu, "MOSO: Decomposing motion, scene and object for video prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2023, pp. 18727–18737.

[6] C. Yang, Y. Xu, B. Dai, and B. Zhou, "Video representation learning with visual tempo consistency," 2020, *arXiv:2006.15489*.

[7] X. Liu, H. Xu, and M. Wang, "Learning joints relation graphs for video action recognition," *Frontiers Neurorobotics*, vol. 16, Oct. 2022, Art. no. 918434, doi: 10.3389/fnbot.2022.918434.

[8] Y. Kong, Y. Wang, and A. Li, "Spatiotemporal saliency representation learning for video action recognition," *IEEE Trans. Multimedia*, vol. 24, pp. 1515–1528, 2022, doi: 10.1109/TMM.2021.3066775.

[9] Y. Chen, H. Ge, Y. Liu, X. Cai, and L. Sun, "AGPN: Action granularity pyramid network for video action recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 8, pp. 3912–3923, Aug. 2023.

[10] W. Wu, D. He, T. Lin, F. Li, C. Gan, and E. Ding, "MVFNet: Multi-view fusion network for efficient video recognition," in *Proc. AAAI Conf. Artif. Intell.*, 2021, vol. 35, no. 4, pp. 2943–2951.

[11] C. Yang, Y. Xu, J. Shi, B. Dai, and B. Zhou, "Temporal pyramid network for action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2020, pp. 591–600.

[12] C. Feichtenhofer, H. Fan, J. Malik, and K. He, "SlowFast networks for video recognition," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Oct. 2019, pp. 6202–6211.

[13] S. Li, Z. Wang, Y. Liu, Y. Zhang, J. Zhu, X. Cui, and J. Liu, "FSformer: Fast-slow transformer for video action recognition," *Image Vis. Comput.*, vol. 137, Sep. 2023, Art. no. 104740, doi: 10.1016/j.imavis.2023.104740.

[14] G. A. Sigurdsson, G. Varol, X. Wang, A. Farhadi, I. Laptev, and A. Gupta, "Hollywood in homes: Crowdsourcing data collection for activity understanding," in *Computer Vision—ECCV*, Amsterdam, The Netherlands. Springer, 2016, pp. 510–526.

[15] R. Goyal, S. E. Kahou, V. Michalski, J. Materzynska, S. Westphal, H. Kim, V. Haenel, I. Fruend, P. Yianilos, M. Mueller-Freitag, and F. Hoppe, "The 'something something' video database for learning and evaluating visual common sense," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 5842–5850.

[16] L. Hedegaard and A. Iosifidis, "Continual 3D convolutional neural networks for real-time processing of videos," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2022, pp. 369–385.

[17] B. Zhou, A. Andonian, A. Oliva, and A. Torralba, "Temporal relational reasoning in videos," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 803–818.

[18] K. Kahatapitiya and M. S. Ryoo, "Coarse-fine networks for temporal activity detection in videos," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 8381–8390.

[19] Y. Zhou, X. Sun, C. Luo, Z.-J. Zha, and W. Zeng, "Spatiotemporal fusion in 3D CNNs: A probabilistic view," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2020, pp. 9829–9838.

[20] R. Christoph and F. A. Pinz, ''Spatiotemporal residual networks for video action recognition,'' in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 2, 2016.

[21] L. Sun, K. Jia, D.-Y. Yeung, and B. E. Shi, ''Human action recognition using factorized spatio-temporal convolutional networks,'' in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 4597–4605.

[22] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, ''Large-scale video classification with convolutional neural networks,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1725–1732.

[23] W. Jian-Chao, W. Li-Min, and W. Gang-Shan, ''Group activity recognition in videos: A survery,'' *Ruan Jian Xue Bao*, vol. 34, no. 2, pp. 964–984, 2023, doi: 10.13328/j.cnki.jos.006693.

[24] 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.

[25] X. Wang and A. Gupta, ''Videos as space-time region graphs,'' in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 399–417.

[26] S. Xie, C. Sun, J. Huang, Z. Tu, and K. Murphy, ''Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification,'' in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 305–321.

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

[28] S.-H. Gao, M.-M. Cheng, K. Zhao, X.-Y. Zhang, M.-H. Yang, and P. Torr, ''Res2Net: A new multi-scale backbone architecture,'' *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 2, pp. 652–662, Feb. 2021.

[29] M. Lee, S. Lee, S. Son, G. Park, and N. Kwak, ''Motion feature network: Fixed motion filter for action recognition,'' in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 387–403.

[30] Y. Li, B. Ji, X. Shi, J. Zhang, B. Kang, and L. Wang, ''TEA: Temporal excitation and aggregation for action recognition,'' in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2020, pp. 909–918.

[31] K. Simonyan and A. Zisserman, ''Two-stream convolutional networks for action recognition in videos,'' in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 27, 2014, pp. 1–9.

[32] H. Kwon, M. Kim, S. Kwak, and M. Cho, ''MotionSqueeze: Neural motion feature learning for video understanding,'' in *Computer Vision—ECCV*, Glasgow, U.K. Springer, 2020, pp. 345–362.

[33] L. Wang, Z. Tong, B. Ji, and G. Wu, ''TDN: Temporal difference networks for efficient action recognition,'' in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 1895–1904.

[34] J. Carreira and A. Zisserman, ''Quo vadis, action recognition? A new model and the kinetics dataset,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 6299–6308.

[35] J. Lin, C. Gan, and S. Han, ''TSM: Temporal shift module for efficient video understanding,'' in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Oct. 2019, pp. 7083–7093.

[36] K. He, X. Zhang, S. Ren, and J. Sun, ''Deep residual learning for image recognition,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778.

[37] H. Wang, D. Tran, L. Torresani, and M. Feiszli, ''Video modeling with correlation networks,'' in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 349–358.

[38] X. Li, B. Shuai, and J. Tighe, ''Directional temporal modeling for action recognition,'' in *Computer Vision—ECCV*, Glasgow, U.K. Springer, 2020, pp. 275–291.

[39] L. Fan, S. Buch, G. Wang, R. Cao, Y. Zhu, J. C. Niebles, and L. Fei-Fei, ''RubiksNet: Learnable 3D-shift for efficient video action recognition,'' in *Proc. Eur. Conf. Comput. Vis.* Springer, 2020, pp. 505–521.

[40] A. Piergiovanni and M. Ryoo, ''Temporal Gaussian mixture layer for videos,'' in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 5152–5161.

[41] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, ''Grad-CAM: Visual explanations from deep networks via gradient-based localization,'' in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 618–626.

LIU TING-LONG was born in 1988. He received the master's degree. He is currently a Technician with Dalian Polytechnic University, specializing in the areas of machine vision and pattern recognition. His research interests include developing novel methods for image and video understanding, as well as their applications in robotic and human–computer interaction.