IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received September 15, 2020, accepted November 25, 2020, date of publication December 9, 2020,
date of current version December 28, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3043386

Temporal Memory Network Towards Real-Time Video Understanding

ZIMING LIU[1,*], JINYANG LI[1,*], GUANGYU GAO[1], (Member, IEEE), AND ALEX K. QIN[2], (Senior Member, IEEE)

[1]Beijing Institute of Technology, Beijing 100081, China
[2]Data Science Research Institute, Swinburne University of Technology, Melbourne, VIC 3122, Australia

Corresponding author: Guangyu Gao (guangyugao@bit.edu.cn)

This work was supported mainly by the National Natural Science Foundation of China under Grant 61972036, in part by grants under Grant U1736117 and Grant 91746210, and in part by the Australian Research Council (ARC) under Grant LP180100114 and Grant DP200102611.

*Ziming Liu and Jinyang Li are co-first authors.

ABSTRACT Action recognition is the basic task for video understanding. Although the action recognition has achieved impressive performance in the static image-based task (e.g. Stanford40) with deep learning, real-time video-based action recognition is still a challenging task due to video's high complexity and computation cost. Motivated by human's recognition ability with only a short glance, we propose the fast light-weighted Temporal Memory Network (TMNet) to achieve real-time video action recognition. The TMNet has a self-supervised structure for exploring both spatial and temporal information with a single video frame. TMNet has three main parts, the base backbone, the regression branch, and the classification branch. Specifically, the base backbone network is a shallow 2D CNN network to obtain the video's initial feature sequences. The classification branch is based on existing successful video recognition models(e.g. TSN, I3D). To make the TMNet learn the spatial-temporal information at a lower cost, we add a self-supervised regression branch. This branch is based on light-weighted 2D CNN and only uses one frame as input. In the training stage, the input of TMNet is a video sequence, the classification branch combined with the base backbone is responsible for learning the video sequence's spatial-temporal feature. Meanwhile, the self-supervised regression branch aims to learn the same spatial-temporal feature under the supervision of the classification branch's output. And the regression branch's input is a single-frame feature sampled from the encoded video sequence. In this way, the regression branch is forced to learn temporal information of adjacent frames with one frame. Therefore, TMNet only needs one frame to predict each video's spatial-temporal information in the inference stage. Finally, TMNet can achieve real-time action recognition and better accuracy by extracting temporal information from a static image. Abundant ablation experiments demonstrate TMNet has a good trade-off between accuracy and speed.

INDEX TERMS Video action recognition, spatial-temporal feature, real-time video understanding.

I. INTRODUCTION

There have been a lot of significant works concentrating on action recognition. Early researchers try to solve the problem from single still images due to the limit of computation sources [1]–[4]. These works have achieved satisfying performance in small image-based data sets, such as PASCAL VOC action [5] and Stanford 40 [6]. These achievements demonstrate the feasibility of recognizing action with only one glance (single video frame). Although significant achievements are obtained in still-image-based action recognition task, the understanding of actions in video data is more complex, there is abundant spatial and temporal information existing in video data. Recently, most successful action recognition (video understanding) studies use larger-scale video data sets and complex deep learning models. Two-stream architecture [7], C3D [8] and LSTM based models [9] are the representations of the most successful methods for video action recognition. Although these methods have good performance in the recognition accuracy, their inference speed is much slower than simple 2D CNN. Therefore, we choose TSN [10], a 2D CNN based model, as the baseline network of

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

the proposed TMNet. We aim to improve both the accuracy and speed performance of the TSN further to realize the real-time video understanding.

As mentioned above, Video Action Recognition methods have achieved significant performance on large scale video dataset. However, most of these methods, including the two-stream network, C3D, LSTM based model, suffer the problem of high computation cost, resulting in terrible inference speed. Video Action Recognition is unlikely to be widely applied in real-time scenes because of the problems mentioned above. Some researchers are trying to reduce the parameters and computation cost of such large models. For example, to deal with the problem of large computation cost and parameters, the I3D [11] and R(1+2)D [12] are proposed based on C3D [8]. However, as we know, these modified models always perform worse than C3D in the real application. And these variants are still slower than a simple 2D CNN. Therefore, we choose the most popular 2D CNN-based method, TSN [10], as the baseline network. Although the 2D CNNs have fewer parameters and faster inference speed, The current 2D CNN based methods can't capture motion or temporal information with only a single image as input. And their excellent results are obtained by inference with a dense video sequence, which increases the computation cost and decreases their inference speed. Thus, we propose a new 2D CNN-based model which can recognize actions with the sparse video sequence, even one frame. To this end, we design the new 2D CNN-based action recognition method, Temporal Memory Network (TMNet),[*] which can learn temporal information from a single frame and recognize action better with fewer frames. With TMNet, we can not only keep the advantage of spatial-temporal information but also realize the real-time video understanding.

Several situations are hard to deploy large action recognition models, especially for computation source limited situations. For example, due to the limited storage and computing capacity of small edge computing devices, it is painful and uneconomical to deploy large models such as C3D for Video Action Recognition. While our TMNet is based on one-frame action recognition and has few parameters, it is possible to implement such a light-weighted model on small devices, such as the dashcam.

Meanwhile, the input of most action recognition models is a sampled sparse video sequence. Thus, the performance of Video Action Recognition is not just related to the model, but also dependent on the input data, i.e., how to sample the original video [13]. In some specific applications, such as stream media or live video applications, there are requirements of both real-time and accuracy. While the stream media data is not stored in the storage, those large recognition models will have difficulty in sampling video and feeding data. In this situation, our TMNet does not suffer such limitation because TMNet is a one-glance action recognition model, which only needs a single frame for video understanding.

TMNet aims to use one-glance input and obtain a better feature containing both temporal and spatial features. To achieve that, we have an underlying assumption: *motion information or temporal information can be learned by the network with only one frame as input.* Figure 1 shows the motivation of the TMNet. Mapping A is the classification branch of the TMNet, while Mapping B is the self-supervised regression branch. According to our assumption, Mapping B aims to extract the spatial-temporal feature which is the same as the output of mapping A, with only one frame. There have been some methods that proved the feasibility of predicting temporal information with only a single image. Wang *et al.* [3] propose Hybrid Video Memory (HVM) machine to hallucinate temporal features of still images with optical flow, which depend on high computation cost. There also have been several works about predicting optical flow which contains temporal information from a single image [14], [15]. These works proved that temporal information could be learned from a single frame.

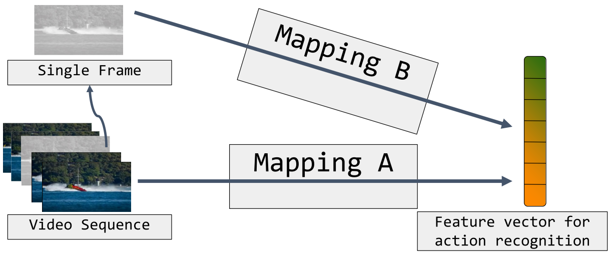

FIGURE 1. The motivation and basic idea of the TMNet.

Specifically, we designed a light-weighted model containing two branches, called Temporal Memory Network (TMNet). The TMNet can remember both the temporal and spatial information of a video with a self-supervised training way. In this way, it can predict the temporal information with only one frame in the inference stage. The TMNet has three parts, *the base backbone, the self-supervised regression branch, and the classification branch*, as shown in Figure 2.

In the training stage, we feed a video sequence to the TMNet to make sure the classification branch combined with the base backbone can provide enough spatial-temporal information. Firstly, the original video sequences are encoded as the initial feature sequences by the base backbone. Then, there are two different pathways. One keyframe is sampled from the feature sequence and fed into the regression branch. On the other hand, the classification branch extracts the video's feature with the whole feature sequence. This process follows the common practice of most CNN based action recognition models. Therefore, the output of the classification branch is a spatial-temporal feature, which is also the supervised signal for the self-supervised regression branch. The regression branch is optimized with the classification branch's output as supervised signal.

In the inference stage, only one sampled frame is needed to extract spatial-temporal information, as the self-supervised

[*] The code will be released in https://github.com/ziming-liu/TMNet.

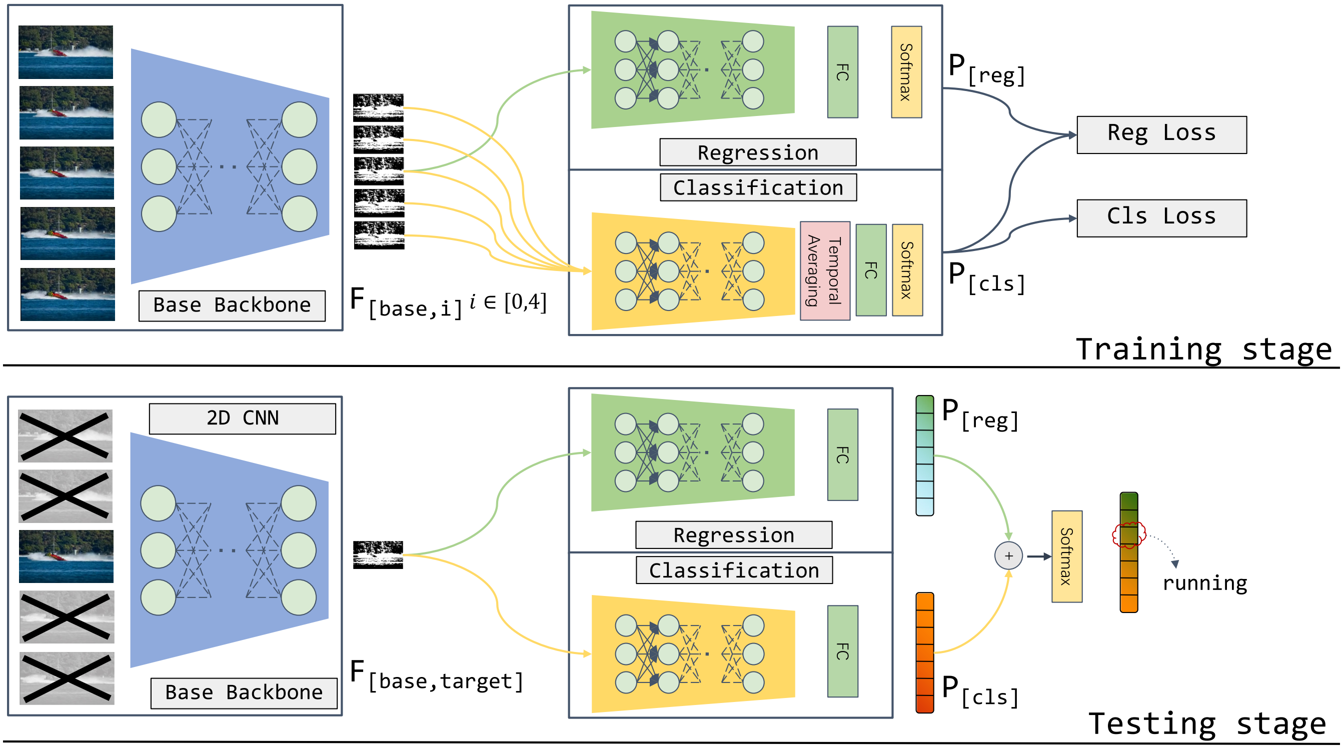

FIGURE 2. In the training stage, we feed one frame to the regression branch (RB), while the classification branch (CB) learns spatial-temporal features from the whole video sequence. In the inference stage, the regression branch can predict spatial-temporal information from a single frame. Finally, the best result is obtained by fusing the outputs of CB and RB.

regression branch has already learned how to predict the spatial-temporal feature with a single frame. In addition, the classification branch can predict the appearance information of a single image. Finally, we can obtain the best result by combining these two kinds of features.

All in all, the contribution of TMNet can be summarized as follows.

- Unlike most previous work that focuses on designing more sophisticated and robust models, we concentrate on realizing real-time video understanding with competitive results by proposing a new one-glance simple 2D CNN-based model, TMNet.
- We design a self-supervised feature consistency mechanism, which promises the TMNet can learn spatial-temporal information from a single frame.
- The architecture of the TMNet is flexible. The base backbone and classification branch can be replaced with most successful video networks. We propose three variants in this paper, i.e. TMNet-2D, TMNet-3D-A, TMNet-3D-B.
- TMNet keeps the advantages of 2D CNN (i.e. fewer parameters, faster speed, lower computation cost), and achieves significant performance for the one-glance action recognition. There is a good speed-accuracy trade-off for the TMNet.

II. RELATED WORKS

Action recognition has been an attractive and challenging research direction in the past few years. Many works are solving the problem from different directions, from still images to videos, from RGB images to optical flow. Usually, we achieve image recognition with apparent features, but more and more methods achieve action recognition with both appearance information and temporal information. In the past, temporal information is learned from the video directly. Current works, such as predicting optical flow from images (i.e. predicting temporal information from still images), have been proved to be feasible. Therefore, we summarise the related works about action recognition into three categories: (1) Action recognition on still images, (2) Action recognition on videos, and (3) Predicting optical flow from still-image, (4) Real-time action recognition.

A. ACTION RECOGNITION ON STILL IMAGES

After deep learning was proposed in 2006 [16], the feature learning ability of the convolution neural network has been drawn more attention and emerged gradually with the update of numerical calculation equipment(graphics processor units) and annotated data [17]. CNN can achieve reasonable performance on a visual recognition task and even match or exceed human performance in some areas. The CNN-based method is also used for the action recognition of still images. Gkioxari *et al.* [1] proposed R*CNN to classify images and return object proposals, which combines the object and environment factors to realize the action recognition of still images. Zhang *et al.* [4] performed recognition in the presence of only image-level action labels in the training stage, with a systematic approach. To deal with single image action recognition, the authors in [2] divided the human body into seven parts: head, torso, arms, hands, and lower body, and defined a few semantic part actions for each of them. However, most of these methods are designed and trained only on a small image dataset, and the number of action categories is also limited. Action is closely related to temporal information. But these still images based works all rely on the apparent features of annotated images without considering temporal information. Therefore, Wang *et al.* proposed a novel HVM machine to address action recognition with few images, via hallucinating motion cues of still images from videos. But the HVM machine hallucinates the motion cues on the existence of the optical flow, which need to be densely annotated and not always available.

B. ACTION RECOGNITION ON VIDEOS

Video-based action recognition is considered relatively well-established. Traditionally, the best-performing method is iDT (Improve dense trajectory) [18], and the follow-up work improves on this basis. Recently, many new methods driven by deep learning have been proposed to solve the action recognition problem. We summarise them into three categories, including (1) the two-stream model, (2) 3D CNN model, and (3) the LSTM-based model.

The two-stream model is built with two streams containing an RGB stream and an optical flow stream. Both spatial information and temporal information are well-considered. Simonyan *et al.* firstly proposed two-stream ConvNets combining the spatial RGB feature and the temporal optical flow feature. Inspired by this architecture, TSN (Temporal Segment Networks) [10] uses an efficient sparse temporal sampling strategy to learn the two-stream model, showing better performance on long videos' recognition. TRN (Temporal Relation Network) [19] is another novel model based on two-stream architecture, which revealed temporal relations existing in videos.

Another category is 3D CNN models, such as C3D [8], which learns both spatial information and temporal information with a 3D kernel. Varol *et al.* [20] demonstrated the advantages of using long-term temporal convolutions with increased temporal input and high-quality optical flow features. I3D [11] proposed to use 3D convolution operation to learn the spatial-temporal information. Pseudo-3D Residual Net (P3D ResNet) [21] R(2+1)D [12] etc. are all the modified version of I3D. They tried to use fewer parameters to achieve better performance. Although 3D CNNs attained high accuracy, these methods consume large computing resources and need clean video clips. While our model uses 2D CNN to learn the temporal information unsupervised, this design can achieve similar performance and use fewer parameters and time.

There are also some works based on LSTM [22]. Two-stream LSTM [23] is proposed to learn spatial features from CNNs and temporal features from LSTM models. These sophisticated models, in some cases, are difficult to deploy in the terminal. Meanwhile, the accuracy will decrease when only pictures are provided.

C. LEARNING TEMPORAL INFORMATION FROM STILL-IMAGE

One of the underlying assumptions of our model is that temporal information can be learned from still images, and there have been a lot of works proving that. For example, several studies started to directly predict the optical flow (a kind of temporal information) from still images. P-CNN [24] was designed to predict motion without human labeling effort. Im2Flow [15] proposed an encoder-decoder CNN model that converted a still image into an accurate flow chart. It put the original static image and the generated optical flow image into the two-stream network. TVNet [14] is an end-to-end trainable network to learn optical flow features, which was modified from the traditional TV-L1 algorithm. These methods aiming to predict the optical flow demonstrate that learning temporal information from still images is possible. Therefore, we claim that learning temporal information from still images is feasible.

D. REAL-TIME ACTION RECOGNITION

As described in Section II-B, various action recognition methods have different architectures and parameters, which affect the inference time of the action recognition models. Commonly, the methods based on 2D CNN have faster speed and less running time, while those methods based on 3D CNN or optical flow inputs suffer slower inference speed. In this subsection, we discussed the current real-time action recognition models to make an overall comparison. Firstly, for the action recognition task, TSN is the most popular and light-weight model based on 2D CNN [10]. It can achieve faster speed than most action recognition models and keep advanced accuracy performance. But TSN is also weak in learning temporal information. TSM [25] is a modified model based on TSN, having a low-cost Temporal Shift Module, which shifts parts of the channel vector along the temporal dimension. In this way, TSM can learn the temporal information of adjacent frames. Besides these models based on 2D CNN, Eco [26] is another method towards the online video understanding. It is composed of 2D CNN and 3D CNN, achieving fast speed by a new sampling strategy. They achieved faster speed compared with TSN under some conditions. T-C3D [27] is another 3D CNN based method for real-time action recognition, it used 3D convolution operation to replace the traditional heavy calculation features (e.g., optical flow and IDT) to achieve faster speed. Due to the 3D CNN, T-C3D is slower than the above methods. Here, we propose a different way to learn temporal information in a very low-cost and self-supervised way. The main architecture is based on 2D CNN to promise faster speed in the inference stage.

III. TEMPORAL MEMORY NETWORK

We first explain the motivation and intuition of the Temporal Memory Network (TMNet): *learning video information with* *only one frame*. Then, we introduce the three parts of the TMNet separately. In this paper, we only use a simple design to validate our assumption, and the final results support this hypothesis.

A. MOTIVATION

The previous action recognition model mostly concentrates on modifying the backbone of the action classification network. These models take more frames as input, but there is only slight performance improvement. Different from these approaches, we argue that few frames are enough to recognize human actions in videos, and even a single frame can predict the information of adjacent frames. Therefore, we propose the TMNet to improve the 2D CNN based model with a self-supervised strategy. Specifically, the TMNet learns motion prior information from a video sequence in the training stage, but it recognizes actions by predicting temporal information from a single frame (one-glance) in the inference stage.

B. TMNet-2D

The TMNet is a novel action recognition framework, which is entirely different from most CNN networks for Video Action Recognition. Firstly, it contains three parts: base backbone (BB), self-supervised regression branch (RB), and classification branch (CB). The BB and CB are based on existing action recognition models (e.g. TSN), the RB is a light-weighted 2D CNN. Secondly, there are different pipelines for the TMNet in the training stage and inference stage.

Given the input of base backbone as $\{I_i\}$, ($i = \{0, 1, \ldots, T\}$, $T > 1$ if training stage, $T = 1$ if inference stage). The base backbone is a shallow 2D CNN which is used to extract the original features sequences $\{F_{base,0}, \ldots F_{base,i}, \ldots F_{base,T-1}\}$ where T is same as the original input video sequence. The outputs of the base backbone are the input to the regression branch and classification branch. The details can be found in the next two subsections III-B1 and III-B2.

For the training strategy, in the training stage, the output of the classification branch is used for action classification with traditional cross-entropy loss, this is a supervised learning process. The regression branch is trained in a self-supervised way to make its feature consistent with the video's feature from the classification branch. In the inference stage, the outputs of both two branches are combined together for final classification.

1) SELF-SUPERVISED REGRESSION BRANCH (RB)

Most researches have proved that video-based action recognition relies on both temporal information and spatial information. Generally, these models used more video frames as input to achieve higher performance. However, the improvement is slight even with a longer sequence. Actually, there is no need to use too many video frames since there is heavy redundancy in video frames. Meanwhile, more video frames also introduce large computation cost and make the model slower.

Inspired by the ideas of motion prediction in still image [3], [14], [15], we propose the self-supervised regression branch (RB) to learn the spatial-temporal feature of the whole video with only one frame. Given the output feature sequence of the base backbone network, i.e. $\{F_{base,0}, \ldots F_{base,i}, \ldots F_{base,T-1}\}$, where T is the same as the original input video sequence. We sample one frame $F_{base,i}$ from the output feature sequence. The $F_{base,i}$ is the input of the regression branch (RB). We will discuss different sampling strategies in the experiments section. As shown in Table 3, we found that sampling the center frame of the sequence is more suitable for predicting the temporal information of adjacent frames.

The regression branch is optimized with a self-supervised way, there is a lot different ways to achieve this. In this paper, we choose the simplest but useful strategy. Specifically, the output feature of the classification branch is regarded as a pseudo supervised signal. The feature of the regression branch needs to fit it with a regression loss $loss_{Reg}$, i.e. the Reg Loss in Figure 2.

Following the common practice, we use the smooth L1 loss as the regression loss $loss_{Reg}$. The other options, such as Cross-Entropy loss, MSE loss, also could achieve similar results. But the difference between these losses is not the main point of this paper. The formulation of the $loss_{Reg}$ is shown as following.

$$Loss_{Reg}(x) = \begin{cases} 0.5x^2 & if\ |x| < 1 \\ |x| - 0.5 & otherwise \end{cases} \tag{1}$$

where $x = F_{Reg} - F_{Cls}$. The F_{Cls} and F_{Reg} are the output feature of two branches of the TMNet.

2) CLASSIFICATION BRANCH (CB)

The classification branch is another important part of the TMNet, whose role is to provide a supervised signal for the regression branch. Because the goal of the regression branch is to learn the temporal information of the target video, we should make sure the classification branch can provide powerful and reliable video features as pseudo supervised signal. Therefore, there are two ways to make sure that: 1) The classification branch and base backbone are based on the existing successful action recognition models. 2) And the input of the classification branch is the whole video sequence to provide enough temporal information.

Here, the classification branch can be most CNN-based action recognition models, such as 2D ResNet [28], SlowNet [29], I3D [11]. Therefore, the TMNet is a flexible architecture for video understanding, and it can be used together with most action recognition networks. For the TMNet-2D, we only use a modified simple 2D ResNet as a classification branch to verify our assumption. This CB network is composed of 2D CNN, temporal average pooling, and a fully connected layer.

Following most action recognition models, the CB encodes the video feature sequence $\{F_{base,0}, \ldots F_{base,i}, \ldots F_{base,T-1}\}$ into a single video feature F_{Cls}, this feature contains both spatial and temporal information of videos.

C. INFERENCES WITH ONE GLANCE

The TMNet has learned the spatial-temporal feature of the whole video in the training stage, with the feature consistency between two branches. Thus, the TMNet can predict the spatial-temporal feature of the video by the regression branch with only one frame, i.e. inferences with one glance.

For the one-glance action recognition, as shown in Figure 2, there is only one input frame F_i in the inference stage. The base backbone maps F_i to $F_{base,i}$. After that, $F_{base,i}$ is the input to both RB and CB. In our basic setting, RB and CB are all 2D CNNs, therefore, we can obtain two feature vectors F_{reg} and F_{cls} with two branches. Finally, we combine them to obtain better accuracy performance. More related experiments about one-glance action recognition can be found in the experiments section.

When we perform one glance action recognition with TMNet-2D, F_{reg} and F_{cls} generate different types of features, referring to different semantic information. Except for spatial features, F_{reg} also contains temporal or motion features of a video. However, the F_{cls} contains only spatial information when the input is a single frame. Therefore, it's essential to fuse these two different types of features F_{reg} and F_{cls} to obtain better accuracy performance in the one-glance action recognition, although two branches may decrease the inference speed of TMNet slightly.

IV. EXPERIMENTS

To demonstrate the advantages of the TMNet model, we conducted experiments from three aspects: (1) The speed and accuracy trade-off of the TMNet. (2) Abundant ablation studies about the details of the TMNet. (3) The comparison with the state-of-the-art action recognition methods to prove that TMNet can not only run faster but also achieve state-of-the-art accuracy.

A. THE SPEED AND RUNNING TIME OF THE TMNet

To prove the TMNet is suitable for real-time action recognition but keep the state-of-the-art accuracy performance. We compare TMNet with several state-of-the-art action recognition models. These experiments are conducted on the Server with Intel(R) Xeon(R) Silver 4210 CPU(2.20GHz), 256G memory, and 8 Nvidia 2080Ti GPUs. Figure 3 visualizes performance comparison of these models, including Resnet3D-18 (3 × 8 Frames), Res-I3D (3 × 8 Frames), TSN (8 Frames), TSM (8 Frames) and TMNet. We reported the inference speed and top1 accuracy, as well as the FLOPs in Figure 3.

This figure shows that TMNet outperforms other methods in top1 accuracy when the input is video sequences of 8 frames but keeps fast inference speed similar to the light-weighted 2D CNN (TSN, TSM). Moreover, we use fewer video frames to speed up the inference of TMNet. The results suggest that TMNet achieves almost 2× faster speed compared with TSN or TSM. Also, TMNet has a good balance in both accuracy and speed when the input is video sequences of 2 frames.

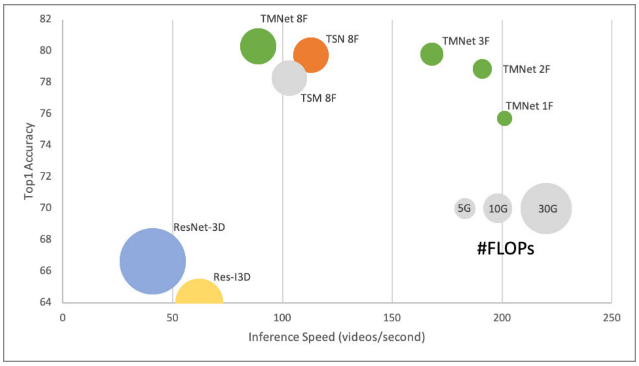

FIGURE 3. TMNet-2D has better accuracy speed trade-off on the UCF dataset. All of the results are recorded in the same environment. *F* in this Figure denotes the number of frames for each video sequence input.

For the FLOPs, similar to the TSN, TMNet has lower FLOPs because of the 2D CNN architecture. And the FLOPs slows down sharply with fewer input frames. This promise TMNet can achieve real-time video understanding with one-glance (one frame).

Furthermore, we compare the difference of those TMNets with different inputs. Figure 4 shows that the inference speed increases faster and the FLOPs decrease sharply with fewer input frames, especially when the input only contains 1-2 frames. This indicates that inference with one frame is important for real-time video understanding.

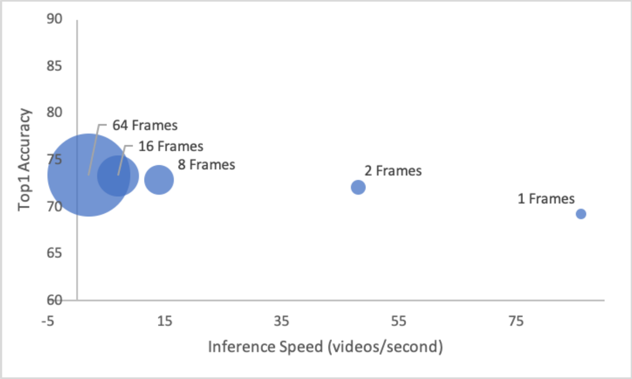

FIGURE 4. The comparison of the performance of TMNet (based on ResNet-18) with different numbers of frames as inputs. "Ten crop" data augment operation is used for inference. The size of the balls means the FLOPs of the model, same as Figure 3.

Figure 4 also suggests that TMNet has the ability to run faster and keep the recognition performance stable. All of the above results demonstrate that TMNet has a good accuracy-speed trade-off.

TABLE 1. The one-glance action recognition results with different temporal length in training stage. "RB" and "CB" denote that the results are predicted by output features of regression branch or classification branch.

	RB			CB			$RB+CB$		
num of frames	mAP	acc@1	acc@5	mAP	acc@1	acc@5	mAP	acc@1	acc@5
3 Frames (TSN)							73.63	73.96	93.44
1 Frame	67.21	68.09	91.28	68.67	69.52	91.59	67.98	68.83	91.30
8 Frames	73.23	73.46	92.76	74.43	74.68	92.28	74.78	75.05	92.84
16 Frames	73.34	73.62	92.99	74.37	74.62	92.49	74.95	75.26	93.42
32 Frames	73.66	74.02	92.65	74.26	74.54	92.33	74.89	75.23	93.31
64 Frames	**73.85**	**74.15**	92.78	74.32	74.54	92.84	**75.13**	**75.39**	93.13

TABLE 2. The ablation study of the network depth for the base backbone, regression and classification branches. The input of the TMNet is 16 frames' video sequence.

base backbone	RB and CB	mAP	acc@1	acc@5	mAP	acc@1	acc@5	mAP	acc@1	acc@5
		RB			CB			$RB+CB$		
0 stage	4 stages	72.5	73.01	92.49	73.22	73.67	91.75	74.18	74.70	93.13
1 stage	3 stages	72.17	72.67	92.20	74	74.31	92.28	74.56	74.97	92.94
2 stages	2 stages	73.34	73.62	92.99	74.37	74.62	92.49	74.95	75.26	93.42
3 stages	1 stage	73.07	73.46	92.92	74.91	75.20	92.99	75.04	75.34	93.44

B. ABLATION STUDIES

1) THE LENGTH OF THE VIDEO SEQUENCE

As mention in Section III, the classification branch learns the target temporal-spatial feature from temporal feature sequence. The regression branch aims to learn the same spatial-temporal features by feature consistency loss. Therefore, the temporal length of the temporal feature sequence which comes from the base backbone is one of the crucial factors for the training. To prove the TMNet can extract temporal-spatial information with one-frame and figure out the effect of different sequence lengths, we conduct ablation experiments by sampling a video sequence with $T = \{1, 8, 16, 32, 64\}$. We sampled these sequences with equal intervals.

Firstly, we prove that TMNet can learn better video features than simple 2D CNN architecture under the condition of one-frame inference. Here, we compare TMNet with a standard TSN [10] model, which used three frames in 3 segments to train a 2D CNN. The results suggest that TMNet significantly outperforms TSN when there is more than 1 frame each sequence used for TMNet training.

Secondly, to prove the regression branch can learn temporal-spatial information, we conducted more ablation studies. As shown in Table 1, better performance can be obtained by fusing the features of the regression branch and classification branch. Therefore, the RB feature is different from the CB feature. Moreover, both mAP and top-1 accuracy of the TMNet increase by using more frames for training. More frames mean that better temporal information is encoded by the classification branch. This indicates that the regression branch is learning both spatial and temporal information.

Besides, when the video sequence only has one frame in the training stage, the performance of the TMNet decreases sharply, and the fusing result of RB and CB is not better than the result of CB. This also proves that the regression branch can learn the temporal information because the TMNet's performance heavily depends on if there is no temporal feature provided for RB in the training stage. And this result also suggests the importance of the temporal information for action recognition, especially for one-frame inference.

All of the above discussions demonstrate the motivation of this paper, TMNet can extract spatial-temporal information with one-glance action recognition (one-frame inference). And better video features can be obtained with longer video sequences used in the training stage.

2) THE NETWORK DEPTHS

To keep a fair comparison with other action recognition models, we make the depth of the TMNet same as most CNN based models, i.e. four stages, same as the ResNet [28]. For the three parts of the TMNet network, we compared four different designs, as shown in Figure 2.

We found that the best results are obtained when we use 3 stages base backbone and 1 stage branches. But the result of RB is better with 2 stages / 2 stages architecture. This means that a shallow network is not suitable for the self-supervised regression branch (RB) to learn powerful temporal information.

TABLE 3. The keyframe sampling strategy for regression branch in the training stage.

	RB			CB			$RB+CB$		
segment	mAP	acc@1	acc@5	mAP	acc@1	acc@5	mAP	acc@1	acc@5
beggining	73.31	73.67	92.84	74.15	74.31	92.36	74.80	75.05	93.02
middle	73.77	74.12	93.02	74.80	75.10	92.55	75.38	75.65	92.99
ending	73.38	73.70	92.65	75.25	75.52	92.86	75.50	75.73	93.34

TABLE 4. The key-frame sampling strategy of the regression branch in the training stage. We recorded the results of RB + CB fused features, and there is no data augment operation adapted. The row is the sampling strategy for the regression branch's training, while the column is the sampling strategy for the one-glance inference.

	mAP	acc@1	acc@5	mAP	acc@1	acc@5	mAP	acc@1	acc@5
training inference	beginning			middle			ending		
beginning	71.59	71.93	91.57	72.28	72.59	91.33	71.93	72.16	90.93
middle	73.75	73.96	92.39	73.36	73.57	92.25	74.18	74.33	92.18
ending	71.59	71.93	91.57	72.37	72.75	91.67	71.87	72.27	92.28

3) THE WEIGHT α OF THE REGRESSION LOSS

As mentioned in section III, the self-supervised regression loss of TMNet is re-weighted with weight α. Here, we conducted a series of experiments to find the best balance between regression loss and classification loss. These ablation experiments are conducted with a 2 stages/2 stages TMNet using 16 frames video sequence. We change the α coefficient of $loss_{reg}$ from 0.05 to 1.0. mAP and top-1 accuracy of RB, CB and RB+CB are recorded in Figure 5. Apparently, the mAP and top-1 accuracy have the same trend. We can find that the TMNet has the best performance when α is 0.8. The performance begins to decrease when α is too high or too low.

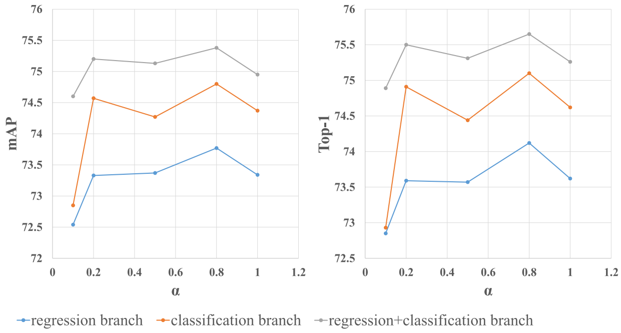

FIGURE 5. This figure shows the results with the changing of α, the coefficient of the regression loss. Different color refer to the results of different branches. We present both mAP and top-1 accuracy to find out the best setting.

4) FRAME SAMPLING STRATEGIES FOR TRAINING

Previous works [10] have proposed that the temporal position of the sampled frame has a critical effect on the model's performance. As shown in Figure 2, there is a frame sampling operation before the RB and the CB. Following the experiments above, we conducted three experiments about how to sample the keyframe for the regression branch (RB).

Following previous practices, we divide the whole video clip into three segments and then sample from one of them. The comparison results are shown in Table 3. Finally, the best RB results are obtained when we use the center frame as RB's input, the same as human's intuition. The center frame can inference and predict the information of both forward frames and backward frames in a short-range. However, we also noticed that the best results of the TMNet are obtained for the last-frame case. This is because the classification branch's performance also affects the final results.

5) FRAME SAMPLING STRATEGIES FOR INFERENCE

In addition to the sampling operation in the training stage, we also conduct abundant ablation experiments to figure out the effect of sampling strategies in the inference stage. Same as the last experiment, we sampled the keyframe from the beginning segment, middle segment, or ending segment. The inference model is the pre-trained model in the last experiment. And we don't use any data augment operation to make sure the results are stable.

As shown in Table 4, the best inference results are obtained when we sample the input frames from the middle segment, no matter which sampling strategy we choose in the training stage. It suggests that TMNet is more sensitive to the sampling strategy of the inference stage. Same as the subsection IV-B4, the frame from the middle segment is more suitable for extracting temporal information.

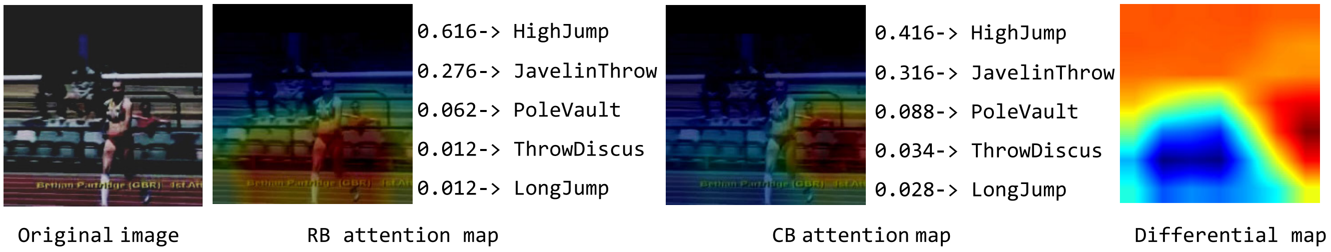

FIGURE 6. The visualization of the output feature maps of regression branch and classification branch with CAM method [30].

6) TMNet-3D

The main idea of the TMNet is to learn the temporal-spatial feature from a single frame. Besides the basic version based on TSN [10], we also explore to use a 3D CNN structure as the classification branch (CB) to guide the training of the regression branch (RB).

Here, we propose two variants of the 3D TMNet, TMNet-3D-A and TMNet-3D-B, as shown in Figure 7. TMNet-3D-A and TMNet-3D-B have a more powerful ability to generate the supervised signal for the regression branch.

TMNet-3D-A replaces the classification branch with 3D CNN architecture (e.g. SlowFastNet [29], I3D [11]). Commonly, 3D CNN can learn better temporal information than TSN architecture. In the training stage, TMNet-3D-A keeps the same setting as TMNet-2D. In the inference stage, it can also achieve faster inference with only the base backbone and regression branch, as the blue dashed line box in Figure 7 shown.

TMNet-3D-B further replaces the base backbone with 3D CNN. Under this configure, we can introduce most state-of-the-art action recognition models into the TMNet architecture to achieve better results. In the inference stage, the light-weighted regression branch can replace parts of the heavy 3D CNN.

The variants of the TMNet-3D not only achieve better accuracy performance but also prove the flexibility of the TMNet architecture, easily fused with most action recognition networks.

We compared the performance of the TMNet-3D with TMNet-2D and other methods in Table 5. The results prove that TMNet-3D indeed has better accuracy performance. And the TMNet-3D-A can also keep the fast inference ability.

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

After the abundant ablation studies which demonstrate the efficiency of TMNet, in this subsection, we compare the TMNet with most state-of-the-art video action recognition methods, as shown in Figure 5, to prove that TMNet can also achieve state-of-the-art accuracy performance. We conducted experiments on the two most popular benchmark data sets: UCF101, HMDB51. We also showed the results of both the 2D version and the 3D version of TMNet. The final results demonstrate that TMNet not only has fast inference speed but also has good accuracy.

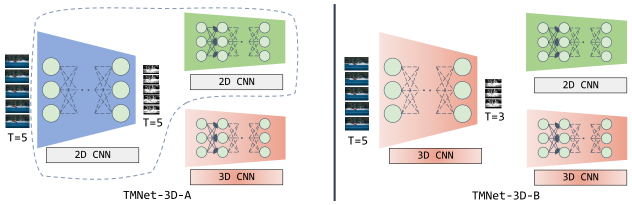

FIGURE 7. The variants of the TMNet-3D.

TABLE 5. The comparison with the state-of-the-art action recognition methods on UCF101 and HMDB51 data sets. The Top-1 accuracy of these methods on UCF and HMDB is recorded.

Methods	UCF	HMDB
DT+MVSM [31]	83.5	55.9
iDT+FV [32]	85.9	57.2
TDD+FV	90.3	63.2
C3D+iDT [8]	90.4	-
LTC+iDT [8]	92.4	67.2
Two Stream [7]	88.6	-
TSN-BN-Inception [10]	94.0	68.5
C3D [8]	82.3	56.8
Conv Fusion	82.6	56.8
ST-ResNet [33]	93.5	66.4
Inception3D [34]	87.2	56.9
3D ResNet101 [35] (16 frames)	88.9	61.7
3D ResNeXt101 [35] (16 frames)	90.7	63.8
STC [34] (ResNet101, 16 frames)	90.1	62.6
DynamoNet [36] (ResNeXt101,16 frames)	91.6	66.2
RGB-I3D [11] (ResNet50,32 frames)	94.5	69.0
TMNet-2D(ResNet50, 16 frames)	89.6	59.5
TMNet-3D-A(SlowNet50 [29], 16 frames)	90.3	62.5
TMNet-3D-B(SlowNet50 [29], 16 frames)	94.5	72.3

D. DATASETS

We conduct abundant ablation experiments on popular action recognition data sets: *HMDB51*, *UCF101* [37].

HMDB51 is a large-scale human action dataset, which contains 51 action categories and 7,000 manually annotated clips. These video clips were extracted from different sources, including digitized movies and YouTube. HMDB51 has been widely used as a baseline for action recognition models, and it's still a challenging dataset, the state-of-the-art art results are still not good enough.

UCF101 is another popular data set of action recognition, whose video clips were collected from YouTube. There are 101 action categories and 13320 videos in UCF101. Besides, the 101 action categories are divided into 25 groups, and each group contains 4-7 videos of an action. The same group's videos have a similar background or viewpoint.

E. TRAINING AND INFERENCE DETAILS

We give the training and inference details of our experiments in this subsection. All of the ablation experiments were conducted on a server with three Nvidia 1080Ti GPUs. We use PyTorch framework to establish models.

For the ablation studies, in the training stage, we sample 16 frames from original videos as input, and the interval of frames is 2. The backbone of the TMNet is the ResNet or ResNeXt. The input frame size is 224 × 224. Several data augment methods are used on the original data. For example, the image size is resized to 256 × 256. Then, we used the multi-scale crop to obtain multi-images with different scales, and the scale factors are random sampled from [1, 0.875, 0.75, 0.66, 0.5]. We also use random flip on images with 0.5 flip ratios. In addition, we use a stepped learning rate, i.e. the initial learning rate is 0.01, which is decreased to 0.001 and 0.0001 at iterator 10k and iterator 16k. The total iterator is 24k finally.

In the inference stage, we also use the image size of 224 × 224. To achieve more stable performance, we use a random crop strategy. But for the ''one-glance action recognition'' configure, a single frame is used for inference, there is no data augment operation used.

V. VISUALIZATION

To figure out what the TMNet learned, we visualized the feature map of the last layer of both the self-regression and classification branches of the TMNet. Because these features are CNN output features, shaped like $C \times H \times W$, we use the class active map (CAM) [30] to obtain the attention map.

Figure 6 shows the example of the action *HighJump*. For the ''RB attention map'', there is apparently a larger motion area is captured, similar to the linear superposition of a video feature sequence. And the regression branch recognizes this action more confidently (Probability: 0.616).

For the ''CB attention map'', the activation area is smaller, which indicates that only spatial information is learned in the classification branch. The recognition confidence (Probability: 0.416) of the CB is also lower than that of the RB. This suggests that the regression branch's output feature is a kind of spatial-temporal feature, not just the spatial feature of the single frame.

VI. CONCLUSION

In this article, we aim to train a light-weight model that can learn spatial-temporal information from a single one frame. We use a self-supervised way to achieve that. Specifically, We proposed a novel model, called temporal memory network (TMNet). The TMNet contains three parts and two losses. Except for the base backbone, TMNet has two branches, a self-supervised regression branch, and a classification branch. In the training stage, the regression branch uses a self-supervised loss to learn spatial-temporal features with 2D CNN architecture. The supervised signal is provided by the classification branch. In the inference stage, the TMNet is a still-image based action recognition model, the input is the single frame of videos. In this way, we can achieve lower computation cost and faster inference speed. In addition, the architecture of the TMNet is also flexible, three variants (TMNet-2D, TMNet-3D-A, TMNet-3D-B) is proposed. In this paper, we use the most simple architecture to realize our motivation, i.e. learning spatial-temporal information from a single frame. In the future, more details can be investigated to improve TMNet further.

REFERENCES

[1] G. Gkioxari, R. Girshick, and J. Malik, ''Contextual action recognition with R*CNN,'' in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1080–1088.

[2] Z. Zhao, H. Ma, and S. You, ''Single image action recognition using semantic body part actions,'' in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 3391–3399.

[3] Y. Wang, L. Zhou, and Y. Qiao, ''Temporal hallucinating for action recognition with few still images,'' in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5314–5322.

[4] Y. Zhang, L. Cheng, J. Wu, J. Cai, M. N. Do, and J. Lu, ''Action recognition in still images with minimum annotation efforts,'' *IEEE Trans. Image Process.*, vol. 25, no. 11, pp. 5479–5490, Nov. 2016.

[5] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, ''The Pascal visual object classes (VOC) challenge,'' *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, Jun. 2010.

[6] B. Yao, X. Jiang, A. Khosla, A. L. Lin, L. Guibas, and L. Fei-Fei, ''Human action recognition by learning bases of action attributes and parts,'' in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 1331–1338.

[7] K. Simonyan and A. Zisserman, ''Two-stream convolutional networks for action recognition in videos,'' in *Proc. NeurIPS*, 2014, pp. 568–576.

[8] 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.

[9] A. Ullah, J. Ahmad, K. Muhammad, M. Sajjad, and S. W. Baik, ''Action recognition in video sequences using deep bi-directional LSTM with CNN features,'' *IEEE Access*, vol. 6, pp. 1155–1166, 2018.

[10] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. Van Gool, ''Temporal segment networks: Towards good practices for deep action recognition,'' in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Springer: Amsterdam, The Netherlands, 2016, pp. 20–36.

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

[12] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri, ''A closer look at spatiotemporal convolutions for action recognition,'' in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6450–6459.

[13] Z. Liu, G. Gao, A. K. Qin, T. Wu, and C. H. Liu, ''Action recognition with bootstrapping based long-range temporal context attention,'' in *Proc. 27th ACM Int. Conf. Multimedia*, Oct. 2019, pp. 583–591.

[14] L. Fan, W. Huang, C. Gan, S. Ermon, B. Gong, and J. Huang, ''End-to-end learning of motion representation for video understanding,'' in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6016–6025.

[15] R. Gao, B. Xiong, and K. Grauman, "Im2Flow: Motion hallucination from static images for action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5937–5947.

[16] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, Jul. 2006.

[17] J. Gu, Z. Wang, J. Kuen, L. Ma, A. Shahroudy, B. Shuai, T. Liu, X. Wang, G. Wang, J. Cai, and T. Chen, "Recent advances in convolutional neural networks," *Pattern Recognit.*, vol. 77, pp. 354–377, May 2018.

[18] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu, "Dense trajectories and motion boundary descriptors for action recognition," *Int. J. Comput. Vis.*, vol. 103, pp. 60–79, Mar. 2013.

[19] 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.

[20] 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.

[21] Z. Qiu, T. Yao, and T. Mei, "Learning spatio-temporal representation with pseudo-3D residual networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 5533–5541.

[22] S. Hochreiter and J. J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[23] H. Gammulle, S. Denman, S. Sridharan, and C. Fookes, "Two stream LSTM: A deep fusion framework for human action recognition," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2017, pp. 177–186.

[24] J. Walker, A. Gupta, and M. Hebert, "Dense optical flow prediction from a static image," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 2443–2451.

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

[26] M. Zolfaghari, K. Singh, and T. Brox, "Eco: Efficient convolutional network for online video understanding," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 695–712.

[27] K. Liu, W. Liu, C. Gan, M. Tan, and H. Ma, "T-c3D: Temporal convolutional 3D network for real-time action recognition," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 7138–7145.

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

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

[30] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2921–2929.

[31] Z. Cai, L. Wang, X. Peng, and Y. Qiao, "Multi-view super vector for action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 596–603.

[32] H. Wang and C. Schmid, "Action recognition with improved trajectories," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 3551–3558.

[33] R. Christoph and F. A. Pinz, "Spatiotemporal residual networks for video action recognition," in *Proc. NeurIPS*, 2016, pp. 3468–3476.

[34] A. Diba, M. Fayyaz, V. Sharma, M. M. Arzani, R. Yousefzadeh, J. Gall, and L. Van Gool, "Spatio-temporal channel correlation networks for action classification," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 284–299.

[35] K. Hara, H. Kataoka, and Y. Satoh, "Learning spatio-temporal features with 3D residual networks for action recognition," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2017, pp. 3154–3160.

[36] A. Diba, V. Sharma, L. Van Gool, and R. Stiefelhagen, "DynamoNet: Dynamic action and motion network," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6192–6201.

[37] 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*. [Online]. Available: http://arxiv.org/abs/1212.0402

ZIMING LIU received the M.E. degree from the Beijing Institute of Technology, in 2020. He is currently pursuing the Ph.D. degree with INRIA, France. His current research interests include computer vision and deep learning, specifically, including action recognition, object detection, and self-supervised learning. He was awarded the "outstanding graduates" title in 2020. He has published two articles during his master's studies in the ACM MM and IEEE CVPR workshops. He and his team won the 2nd place in "Vision Meets Drone Challenge Task-2: object detection in videos" of ICCV 2019.

JINYANG LI was born in Harbin, Heilongjiang, China, in 1998. She is currently pursuing the B.S. degree in software engineering with the School of Computer Science, Beijing Institute of Technology, Beijing. Her research interests include action recognition and deep learning.

GUANGYU GAO (Member, IEEE) received the M.S. degree in computer science and technology from Zhengzhou University, Zhengzhou, China, in 2007, and the Ph.D. degree in computer science and technology from the Beijing University of Posts and Telecommunications, Beijing, China, in 2013. He studied with the National University of Singapore, Singapore, as a Government-Sponsored Joint-Ph.D. Student from 2012 to 2013. Since 2013, he has been joining the Beijing Institute of Technology (BIT), as an Associate Professor with the School of Computer Science and Technology. His current research interests include computer vision, and applications of multimedia and unsupervised learning. He was awarded as the IBM Faculty Award in 2016, 2017, and 2019, which highly competitive and recognizes the quality of his research and program as well as its importance to the industry. His work won the IEEE Nominate Video Award at the 12th IEEE International Conference on Ubiquitous Intelligence and Computing (UIC 2015). He also directed his students to win the Honourable Mention in "Vision Meets Drone: Challenge-Task-2: object detection in videos" of ICCV 2019.

ALEX K. QIN (Senior Member, IEEE) received the B.Eng. degree from Southeast University, Nanjing, China, in 2001, and the Ph.D. degree from Nanyang Technology University, Singapore, in 2007. From 2007 to 2017, he was with the University of Waterloo, Waterloo, ON, Canada, with INRIA, Grenoble-Rhône-Alpes, France, and with RMIT University, Melbourne, VIC, Australia. In 2017, he joined the Swinburne University of Technology, Melbourne, as an Associate Professor. He is currently the Director of the Swinburne Intelligent Data Analytics Laboratory, the Program Lead of the Swinburne Data Science Research Institute, and the Leader of Machine Learning and Intelligent Optimisation (MLIO) Research Group. His research interests include machine learning, evolutionary computation, computer vision, remote sensing, services computing, and pervasive computing. He was a recipient of the 2012 IEEE Transactions on Evolutionary Computation Outstanding Paper Award and the Overall Best Paper Award at the 18th Asia Pacific Symposium on Intelligent and Evolutionary Systems in 2014. He is also the Vice-Chair of the IEEE Neural Networks Technical Committee and the IEEE Emergent Technologies Task Forces on "Collaborative Learning and Optimisation" and "Multitask Learning and Multitask Optimisation".