IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 4, 2019, accepted December 13, 2019, date of publication December 25, 2019, date of current version January 6, 2020.

Digital Object Identifier 10.1109/ACCESS.2019.2962284

Learning Long-Term Temporal Features With Deep Neural Networks for Human Action Recognition

SHENG YU[1], LI XIE[1], LIN LIU[1], AND DAOXUN XIA[2]
[1]School of Information Science and Engineering and Provincial Demonstration Software Institute, Shaoguan University, Shaoguan 512005, China
[2]School of Big Data and Computer Science, Guizhou Normal University, Guiyang 550001, China

Corresponding author: Sheng Yu (yu_sheng4105@126.com)

This work was supported in part by the National Nature Science Foundation of China under Grant 61762023, in part by the Shaoguan Science and Technology Plan Project under Grant 2019sn064, in part by the Shaoguan University Research Project under Grant SY2018KJ03 and Grant SZ2016KJ09, and in part by the Shaoguan University Talent Introduction Research Project.

ABSTRACT One of challenging tasks in the field of artificial intelligence is the human action recognition. In this paper, we propose a novel long-term temporal feature learning architecture for recognizing human action in video, named Pseudo Recurrent Residual Neural Networks (P-RRNNs), which exploits the recurrent architecture and composes each in different connection among units. Two-stream CNNs model (GoogLeNet) is employed for extracting local temporal and spatial features respectively. The local spatial and temporal features are then integrated into global long-term temporal features by using our proposed two-stream P-RRNNs. Finally, the Softmax layer fuses the outputs of two-stream P-RRNNs for action recognition. The experimental results on two standard databases UCF101 and HMDB51 demonstrate the outstanding performance of proposed method based on architectures for human action recognition.

INDEX TERMS Action recognition, residual learning, recurrent neural networks, long short-term memory (LSTM).

I. INTRODUCTION

Human action recognition in video is an important and focused research topic with various useful applications, such as intelligent video surveillance, video retrieval, human-computer interaction and smart home appliance [1]–[3]. Due to background clutter, lighting conditions, partial occlusion and viewpoint change, action recognition is limited [4]. Similar to other vision problems, effective visual features of human action in video are crucial for action recognition [5], [6]. For example, Figure 1 shows two examples from the HMDB51 dataset [7]. The appearance feature in each frame is not sufficient to differentiate the class.

The feature representation of human action recognition can be roughly divided into two types. One is based on hand-crafted features, such as Histogram of Gradient (HOG) [8], Histogram of Optical Flow (HOF) [9] and Improved Dense Trajectories (IDTs) [10]. Because of strong performance on the image recognition, IDTs features with fisher Vector (FV) [11] or Vector of Locally Aggregated Descriptors (VLAD) [12] have been applied onto action recognition in video [10], [13], [14], which have achieved the state-of-the-art results. Richard and Gall [15] proposed RNNs-based encoding method [16] to aggregate local feature descriptors for action recognition. Another is deep learning-based technique. Similar with many computer vision tasks, the progress on human action recognition is significantly advanced by deep learning techniques.

FIGURE 1. Exaples from the HMDB51 dataset [7]. The groundtruth for the two samples are "sit" and "stand". Temporal information is crucial to correctly determine these two classes.

The associate editor coordinating the review of this manuscript and approving it for publication was Mehul S. Raval.

Compared with hand-crafted features, deep neural networks, such as deep belief network [17], Convolutional Neural Networks (CNNs) [18]–[21] and RNNs with Long Short-Term Memory (LSTM) [22] are much more suitable for various vision tasks such as object detection [23], [24] and image recognition [18].

CNNs are powerful frameworks that achieve impressive performance for human action recognition [4], [25]–[30]. Currently, most of the works of human action recognition make use of 2-dimensional convolutional kernels [29], [31]–[33], expanded from the mainframe of CNNs for image classification [18]. The two-stream architectures [29], [31] consist of spatial stream and temporal stream, which take RGB images and optical flow images as inputs respectively. Therefore, the two-stream CNNs architectures are often applied to describe spatio-temporal information in video. Each stream focuses on different type of feature learning of video clip. The temporal stream mainly learns motion features of videos. The spatial stream processes appearance contents of videos.

Recently, CNNs with 3-dimensional (3D) convolutional kernels are proposed to directly learn spatio-temporal features for action recognition [34]–[36]. Tran *et al.* [37] experimentally found $3 \times 3 \times 3$ convolutional kernel obtained the highest accuracy. In [38], a deep 3-dimensional (3D) Residual ConvNet to extract spatio-temporal feature. Carreira and Zisserman [26] proposed a new Two-Stream Inflated 3D ConvNet (I3D) for action recognition in video. More recently, handcrafted features with BoW/FV representation are simple to integrate with the I3D model [39]. However, most of 3D convolutional models failed to exploit long-term motion features of the video, and the performance of these 3D models is limited [40]. The primary reason for this failure that the number of parameters of 3D convolutional kernels is much larger than 2D kernels. Moreover, 3D CNNs cannot be pretrained on ImageNet [41]. In addition, 3D models only preserve short-term temporal features [42] whereas long-term motion features are crucial for representation of human action in video.

RNNs are effective architecture to model contextual information. But for standard RNNs, the long range of context that in practical accessed is quite limited due to gradient vanishing problem [43]. In RNNs with LSTM, a set of memory blocks with gating architecture are used to process the information flow such that gradient vanishing problem is tackled, and long-term features are better extracted. To capture stronger and longer spatio-temporal representations, hybrid neural networks are proposed by using CNNs in combination with LSTM [44]–[46]. Simultaneous training of CNNs and LSTM models is prone to overfitting on challenging benchmark database HMDB51 [7] and UCF101 [47], and recognition accuracy is lower than hand-crafted feature methods. In order to tackle overfitting problem, Yu *et al.* [42] proposed single layer pi-LSTM architecture to learn long-term information for action recognition. However, shallow LSTM has difficulty in learning rich semantic features. Meanwhile, unsupervised learning architecture for capturing video representation is also proposed by using LSTM model [48].

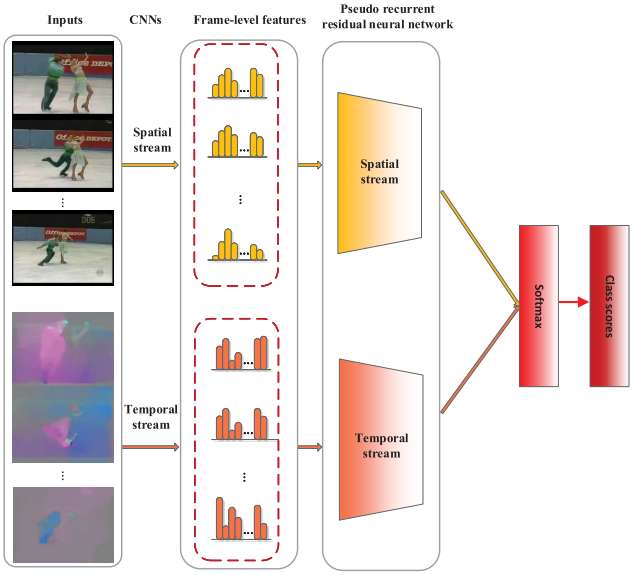


FIGURE 2. Framework of our proposed pseudo recurrent residual neural network for action recognition. The main steps include as follows: (1) extracting the frame-level features of the RGB and flow images by two-stream CNNs, (2) learning long-term temporal features by two-stream P-RRNNs, (3) Softmax layer to fuse spatial and temporal stream, and output class scores.

In this paper, we experimentally evaluate the proposed method to learn richer semantic features and model longer-term temporal information in video for action recognition. Our main contributions can be summarized as follows: First, we introduce residual learning into the recurrent structure and propose Pseudo Recurrent Residual Neural Networks (P-RRNNs) to model long-term temporal features. Our model outperforms other RNNs based architectures in action recognition tasks. Meanwhile, the number of model parameters is greatly reduced. Second, different from most of approaches that extract temporal features from sample video frames, we directly learn spatial and long-term temporal features from holistic video clip to depict the human action information, which is able to obtain more robust visual features. Third, we combine our P-RRNNs features with IDT features for action recognition. The experimental results prove the complimentary of both features. Figure 2 summarizes the pseudo recurrent residual neural network architecture in the study.

The paper is organized as follows: in next section, we review literatures related with the presented works. In section III, we elaborate on the details of the proposed P-RRNNs for action recognition in video. In section IV, we analyze the performance of the approach. Finally, the conclusion of the work is in Section V.

II. RELATED WORKS

Human action recognition has been studied by researchers for decades. Early studies [49], [50] mainly focus their research

on simple actions with flat background, such as hand clapping, boxing and walking. With rapid developments in local feature extracting and encoding, action recognition is gradually towards practical applications [10], [13], [51], [52]. According to deep neural networks, various reported methods are categorized into hand-crafted feature-based methods and deep learning-based methods.

A. HAND-CRAFTED FEATURES FOR ACTION RECOGNITION

Early methods [53]–[55] for action recognition in videos primarily focus on hand-crafted features, which described appearance and motion information by using a number of local features. Local features are effective tool in image recognition, which represent image through feature descriptors, such as Scale-Invariant Feature Transform (SIFT) [56], Speeded Up Robust Features (SURF) [57]. Inspired by success of image recognition, the researchers directly extend the image classification methods to learn spatio-temporal information of video for action recognition. In [54], HOG descriptor is extended to Histograms of Oriented 3D spatio-temporal Gradients (HOG3D) for action recognition in video. Inspired by the Harris corner descriptor [58], Harris3D [55] is proposed to encode the region of the interest (ROI). Extended from SIFT, SIFT-3D [53] is proposed to represent the spatio-temporal motion features for action recognition. Recently, one effective approach is Dense Trajectories (DT) [14], which consist of HOG, HOF, and Motion Boundary Histogram (MBH). IDTs features [10] take camera motion into account to improve performance of action recognition based on DT approach. Furthermore, Crmona and Climent [59] combine IDTs and subtensor projections to depict the human action. Whereas, to extract IDTs features has much higher computational complexity and intractable on large scale data sets.

By adopting encoding methods, such as Bag of Visual Words (BoVW) [60], FV, or VLAD, local descriptors can be embedded into a global video-level feature vector for action recognition. Comparing with BoVW, the FV and VLAD can process statistics analysis of high order local features, and obtain noticeable higher accuracy. But these encoding methods obviously lead to the loss of temporal order of local features in the video. Meanwhile, graphical models, such as conditional random fields [61], are well-known approaches to extract the long-term temporal features for action recognition.

B. DEEP LEARNING FOR ACTION RECOGNITION

Many advances of action recognition in video are inspired by success on image classification [62], [63]. The breakthrough of image domain also rekindled the focus on deep learning for video recognition. The CNNs models play significant roles in the image classification and achieves state-of-the-art results. Extending the 2D convolutional filter, Ji *et al.* [35] proposed 3D CNNs to address video features from both the spatial and temporal dimensions. Karpathy *et al.* [32] built a large-scale dataset of action recognition, which consists of 1 million video clips belonging to 487 classes, namely Sport1M, and proposed various ways to fuse motion information into the current CNNs model. Tran *et al.* [37] applied convolutional 3D (C3D) on 16 consecutive frame to learn motion and appearance information for action recognition, and experimentally found $3 \times 3 \times 3$ convolutional kernel obtained the highest accuracy. Varol *et al.* [64] proposed Long-term Temporal Convolution (LTC) models to expand temporal length of inputs. Simonyan and Zisserman [29] first proposed a two-stream CNNs architecture, which applies two networks for extracting appearance and motion features from two information streams, and fuse them by using average pooling or a linear SVM. Yang *et al.* [65] and Shi *et al.* [66] proposed additional information sources to learn richer appearance and motion features for action recognition based on two-stream framework. Duta *et al.* [67] proposed Spatio-Temporal VLAD (ST-VLAD) to integrate spatio-temporal features. Kar *et al.* [25] found that only a small part of frames play a crucial role to discriminate an human action class, and they proposed a temporally pooling frames method to filtrate spatio-temporal action attention components. Wang *et al.* [28] proposed a temporal segment network (TSN) architecture that incorporates sparse temporal sampling and video-level supervision to learn more proper long-term temporal features. An [68] used restricted Boltzmann machine as feature encoder to encode the spatial and temporal features and a SVM classifier is applied to recognize human action in video. Cherian *et al.* [69] explored generalized rank pooling (GRP) to preserve video frames temporal order information for improve action performance. Choutas *et al.* [70] proposed pose motion (PoTion) forms to recognize video action. However, the PoTion method needs to be combined with I3D [26] to obtain high recognition accuracy. Meantime, it is affected by the human pose estimator.

Recently, with the improvement of ResNet [21], [71] for image classification, Feichtenhofer *et al.* [72] proposed a spatio-temporal ResNet (ST-ResNet) that associates ResNet with the two-stream CNNs. To effectively learn spatio-temporal features, they apply a residual connection from the spatial stream to the temporal stream. Meantime, inspired by the success of recurrent neural networks in sequential information modeling [73]–[76], many researchers [42], [44], [45], [48], [77], [78] propose LSTM model for action recognition. Ng *et al.* [44] and Donahue *et al.* [45] extracted frame-level features of video by using CNNs model, and train LSTM with the frame-level feature for direct video-level prediction. Srivastava *et al.* [48] proposed an approach for learning the sequence information in unsupervised settings by using LSTM architecture. To mitigate the overfitting problem, Yu *et al.* [42] proposed a single-layer LSTM frameworks for learning long-term motion features. To learn spatio-temporal information, Zhang *et al.* [79] proposed multi-level recurrent residual networks to produce complementary representations for action recognition. The recurrent residual model is also use of temporal skip connection.

Among these RNN-based approaches, the recurrent residual networks are [44], [45], [79] closely related to us. In [44] and [45], the deep learning model is built by CNNs

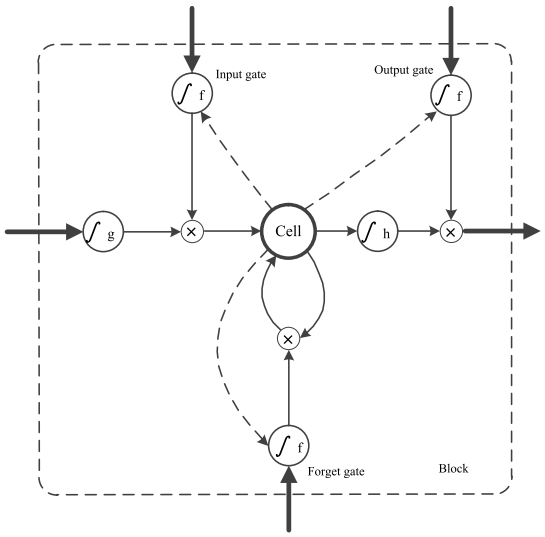


FIGURE 3. LSTM memory block with one cell [43].

and LSTMs, which feeding the output of CNNs into LSTMs. In contrast, we introduce pseudo residual learning into RNNs, which can design deeper model to learn richer semantic features for action recognition. In [79], three stream multilevel recurrent residual networks are proposed for action recognition. Each stream consist of ResNets and a recurrent model. However, the proposed network mainframe fuses pseudo residual learning into recurrent neural networks to learn long-term temporal features.

Motivated by above analysis, we propose a pseudo recurrent residual neural network, which considers skip one or more hidden layers in RNNs. The model can learn richer action semantic features and long-term motion information to classify action recognition in video.

III. METHODS

In this section, we describe the key components of the P-RRNNs, including LSTM and GRU architectures, and pseudo recurrent residual network architectures.

A. LONG SHORT-TERM MEMORY

Figure 3 is a LSTM memory block with one cell, where $\otimes$ represents multiplication, and dashed lines represent weight between the cell to the gates. All other connection weights within the block are fixed to one. The LSTM architecture uses a set of memory blocks which each block contains one or more self-connected memory cells, and three multiplicative units: the input, output and forget gates. Three gates provide continuous analogues of write, read and reset operations for the cells. The gate activation function usually is the logistic sigmoid. The cell input and output activation functions are hyperbolic tangent or logistic sigmoid. Based on the specialized memory architecture, LSTM is able to effectively tackle the vanishing and exploding gradient problem. Assuming that $x = (x_1, x_2, \cdots, x_T)$ is a length T input

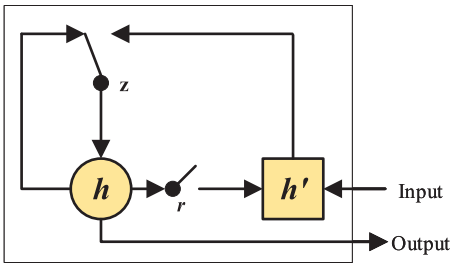


FIGURE 4. GRU architecture [80].

feature, at time step t, for all LSTM neurons in some layer, activations are computed as follows:

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i) \quad (1)$$
$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f) \quad (2)$$
$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o) \quad (3)$$
$$g_t = \sigma(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \quad (4)$$
$$c_t = f_t \odot c_{t-1} + i_t \odot g_t \quad (5)$$
$$h_t = o_t \odot c_{t-1} + \tanh(c_t) \quad (6)$$

where σ is the logistic sigmoid function, $\odot$ denotes element-wise multiplication, i_t, f_t, o_t and c_t are the input gate, forget gate, output gate and memory cell activation vectors, respectively, b_i, b_f, b_o and b_c denote the bias terms, $W_{\alpha\beta}$ is the weighted matrix between α and β, such as W_{xi} is the weighted matrix from the inputs x_t to the input gates i_t.

B. GATED RECURRENT UNIT (GRU)

The GRU architecture is a simplified variant of the LSTM architecture [80], in which coupled the input and the forget gate into an update gate [81].

Compared to three units of LSTM architecture, the GRU reduce the gating signals to two. The GRU architecture is showed in Figure 4, which consists of an update gate z and a reset gate r. The update gate moderates the rate at which the information at the previous moment is allowed to enter the current state. Oppositely, the reset gate is applied onto controlling how much status information of the previous moment can be ignored. At time step t, the information of the forward propagation can be computed as follows:

$$h_t = (1 - z_t)h_{t-1} + z_t h'_t \quad (7)$$
$$z_t = \sigma(W_z x_t + U_z h_{t-1}) \quad (8)$$
$$h'_t = \tanh(Wx_t + U(r_t h_{t-1})) \quad (9)$$
$$r_t = \sigma(W_r x_t + U_r h_{t-1}) \quad (10)$$
$$y_t = \sigma(W_o h_t) \quad (11)$$

where $\odot$ is represents the multiplication of the corresponding elements of two matrices. $W_o \in \mathbb{R}^{h\times y}$ is the weight matrix between input and output layer, h and y is number of nodes in hidden layer and output layer, respectively, $W \in \mathbb{R}^{x\times h}$ is represents the connection weight matrix of the input layer to the update gate, and x is the feature dimension of the input feature vector. $U_z \in \mathbb{R}^{h\times h}$ is the weight matrix between the

hidden layer and the updated gate of the previous moment. $W_r \in \mathbb{R}^{x\times h}$ and $U_r \in \mathbb{R}^{h\times h}$ are the connection weight matrixes of the input layer and hidden layer of the previous time to the reset gate, respectively. $W \in \mathbb{R}^{x\times h}$ and $U \in \mathbb{R}^{h\times h}$are also the connection weight matrixes of the input layer and hidden layer of the previous time to the candidate state h'.

C. RESIDUAL NETWORKS

He *et al.* [21] proposed residual learning in CNNs architecture for image classification. Let $H(x)$ indicate the desired mapping. The theory of ResNet is to account for the mapping of the learned function from one layer to another as $H(x) = F(x) + x$, where x is original input, and $F(x)$ is residual function. By using the spatial skip connection, the input feature x is directly forwarded and added to the next layer, it only remains to approximate the residual functions $F(x) = H(x) - x$, where the input x and output $H(x)$ need have the same dimensions. To learn long-term temporal features with RNNs for action recognition, we approximate the desired mapping functions $H(x) = F(x) \oplus x$, where $\oplus$ represents the concatenation of two feature vectors. Such skip connection in RNNs can be regarded as a Pseudo Recurrent Residual Neural Networks (P-RRNNs).

D. PSEUDO RECURRENT RESIDUAL NEURAL NETWORKS (P-RRNNs) ARCHITECTURE

Inspired by the success of ResNets [21] in image recognition tasks, and RNNs are good at processing sequential information, we design a novel pseudo recurrent residual neural networks to pursue spatio-temporal feature for action recognition. More specifically, we integrate the residual learning into the recurrent neural network and propose pseudo recurrent residual recurrent neural network architectures. In our studies, we design three P-RRNNs architecture variants, as show in Figure 5.

Figure 5(a) illustrates an unfold two-stream P-RRNNs with skip connections over time. Therefore, the l^{th} hidden layer receives the feature-maps of the input and upper preceding layers, $\boldsymbol{x}_0, \boldsymbol{x}_{l-1}$, as input:

$$\boldsymbol{x}_l = F_l([\boldsymbol{x}_0, \boldsymbol{x}_{l-1}]) \tag{12}$$

$[\boldsymbol{x}_0, \boldsymbol{x}_{l-1}]$ refers to the concatenation of the feature-maps produced in layer $0, l-1$, and $l > 1$. Figure 5(a) depicts the video frame feature stream and optical flow feature stream. Blue circle indicates input feature vectors of the two-stream RRNNs at time t. Orange circles indicates hidden layers, the number of hidden layers of each stream is set to 3. Meantime, the number of hidden units is set to 512 [46]. C represents the concatenation of two vectors. The action recognition is achieved by merging the outputs of the two-stream with the Softmax layer. Such concatenation of two vectors can be regarded a pseudo residual connection, and the network architecture regarded as an pseudo recurrent residual neural networks. We named this recognition network structure as an Input P-RRNNs (IP-RRNNs).

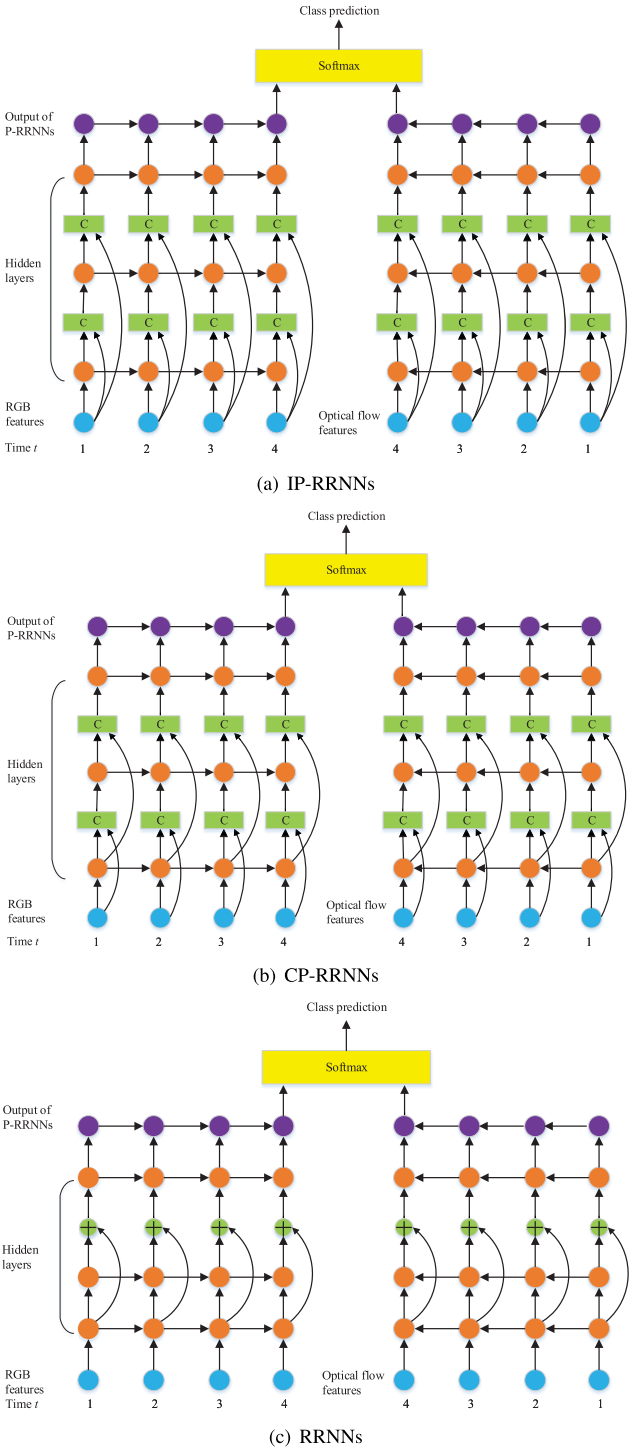


FIGURE 5. Three P-RRNNs architectures for action recognition. (a) In the IP-RRNNs, except for the first hidden layer, the input of other hidden layers is the concatenation of the output of the previous hidden layer and the input feature-maps of the network. (b) In addition to the first hidden layer, the input of other hidden layers is the concatenation of the output of the last two layers. (c) residual connection on recurrent neural networks.

Figure 5(b) is a Cross-layer P-RRNNs (CP-RRNNs), the l^{th} hidden layer receives the feature-maps of the last two preceding layers, $\boldsymbol{x}_{l-2}, \boldsymbol{x}_{l-1}$, as input:

$$\boldsymbol{x}_l = F_l([\boldsymbol{x}_{l-2}, \boldsymbol{x}_{l-1}]) \tag{13}$$

where$[\boldsymbol{x}_{l-1}, \boldsymbol{x}_{l-2}]$ refers to the concatenation of the feature-maps produced in layer $l-1$, $l-2$, and $l \geq 2$.

Figure 5(c) are Recurrent Residual Neural Networks (RRNNs). The residual connection is similar to the deep residual CNNs, and $\oplus$ represents the addition of two feature vectors.

IV. EXPERIMENTS

In this section, we demonstrate extensive experimental effectiveness of pseudo recurrent residual neural network on the HMDB51 and UCF101 datasets. Firstly, we introduce two challenging human action benchmark datasets and network training in Section IV-A. We study the recurrent units of P-RRNNs in Section IV-B and the RNN architectures in Section IV-C. Next, we evaluate in Section IV-D the impact of different residual architectures. Then, we show effect of multi-stream fusion architecture in Section IV-E and computational costs in Section IV-F. Finally, we compare our method with the state-of-the-art in Section IV-G.

We conduct experiments on Ubuntu 14.04 with Intel Core i7-7700, 32GB Memory, and a NVIDIA GTX Titan Graphics Card.

A. DATABASES AND IMPLEMENTATION DETAILS

1) DATABASES

UCF101 database [47] contains 13320 video clips that are downloaded from YouTube, and has 101 human action classes. The video clips were temporally trimmed and fixed frame rate of 25 FPS, and resolution of 320×240 respectively. Each action class is divided into 25 groups which contain 4 to 7 video clips. Following the literature, to ensure that video clips from the same video were not used for both training and testing, three train and test splits are used for action recognition on UCF101.

Moreover, HMDB51 database [7] contains 6766 videos divided into 51 human action categories, which is collected from a wide range of sources from digitized movies to online videos such as YouTube. For evaluation purposes, we follow the constraint that video clips in the training and testing set could come from different video file. Specifically, three distinct training and testing splits were generated from the database that 3570 clips in the training set and 1530 clips in the testing set.

2) IMPLEMENTATION DETAILS

Following TDD [82], we set the label between the video frames and video snippets from the video to be the same. We implement our network in Caffe. To alleviate the overfitting issue, we use two-stage training tactic to train our proposed networks. Firstly, we train the two-stream GoogLeNet models, and initialize GoogLeNet parameters of both streams with pre-trained models from ImageNet [41]. We build fine-tuned network by using Stochastic Gradient Descent (SGD) with a batch size of 128, and the momentum is set to 0.9. For spatial stream, the input is a single RGB frame image of size $224 \times 224 \times 3$, and learning rate starts from 0.001, and decreases to its 0.1 every 2,000 iterations, stops at 10,000 iterations. For temporal stream, the input is a $224 \times 224 \times 2L$ volume, where L is the number of stacking optical flows. Meanwhile, we initialize the learning rate as 0.005, get reduced by a factor of 10 after 10,000 and 15,000 iterations, stops at 20,000 iterations. The dropout ratios are set to 0.5 [83] for both streams. We select softmax loss for GoogLeNet training.

TABLE 1. Performance with different standard deviation of IP-LSTM and IP-GRU architectures on the split1 of UCF101.

	$\sigma = 0.1$	$\sigma = 0.01$	$\sigma = 0.001$	$\sigma = 0.0001$
IP-GRU	82.90%	83.50%	84.10%	83.20%
IP-LSTM	86.30%	86.60%	87.80%	87.10%

At the second stage, we train the P-RRNNs with LSTM and GRU units from scratch for the temporal stream and spatial stream. The initialization of the P-RRNNs is important. To ensure that good hyper-parameters are used, we experiment with Gaussian distribution with mean of zero and several standard deviation $\sigma = \{0.1, 0.01, 0.001, 0.0001\}$ on the split1 of UCF101. Table 1 reports results with different standard deviation of IP-LSTM and IP-GRU architectures. We can observer that via setting $\sigma = 0.001$ we achieve better performance. Therefore, we initialized weights of the P-RRNNs with LSTM and GRU from a Gaussian distribution with mean of zero and 10^{-2} variance.The training parameters of both streams are the same. Specifically, the initial learning rate is set to 0.01 and get reduced by a factor of 10 after 2,000 and 5,000 iterations. The whole training procedure stops at 10,000 iterations. We use dropout of 0.6 for both streams [84]. Cross entropy loss is used for P-RRNNs. During training of P-RRNNs, the two-stream GoogLeNet parameters are fixed.

At test time, given a video, we input the video frame by frame to P-RRNNs, and the class scores for the whole video are then obtained. The maximum class score is the classification of the input video. For both databases, we select the same evaluation protocol. Three distinct training and testing splits are provided by the organizers. The performance is estimated by mean recognition accuracy across three splits.

B. EVALUATION ON RECURRENT UNITS IN P-RRNNs

Different types of recurrent units may significantly influence the complexity and performance of RNNs. Recently, GRU is proposed and become one of the most commonly used recurrent units in RNNs. Therefore, in this section we focus on two types of recurrent units: LSTM units and GRUs. We measure these recurrent units on the task of action recognition on the UCF101. More specifically, we employ IP-RRNNs architecture with 512 hidden units [46]. Similarly, the number of layers is set to three. Table 2 compares the accuracy of our method with GRU and LSTM architectures. The results demonstrate the LSTM units that obtaining higher performance, in which gains 2.8% compare to GUR units on the

TABLE 2. Performance of the GRU and LSTM architectures on the UCF101.

	LSTM	GRU
Split1	87.8%	84.1%
Split2	89.0%	86.4%
Split3	88.8%	86.5%
Average	88.5%	85.7%

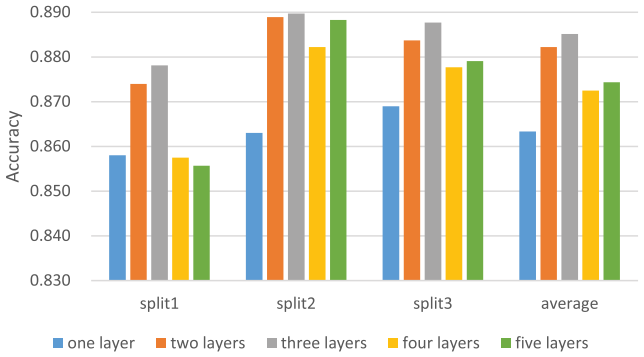


FIGURE 6. Performance of the IP-RRNNs with 1 to 5 hidden layers on the UCF101.

UCF101 dataset. The reason that the GRU model simplifies the network architecture, and reduces feature learning ability for action recognition. Consequently, we choose LSTM architecture to learn long-term action feature in the remainder if there is no special explanation.

C. EVALUATION ON DIFFERENT NUMBER OF HIDDEN LAYERS OF P-RRNNs

In deep neural network architecture, each layer can carry different recognition information. But it is easy to appear of overfitting with the number of layers increased. In this section, we investigate the networks with number of 1 to 5 hidden layers, and each layer with 512 hidden units. Figure 6 reports action recognition results with different number of hidden layers of IP-RRNNs. The discrimination increases with the increase of hidden layer when the number of hidden layers is less than 4. It indicates that in these experiments, IP-RRNNs can gain recognition accuracy from increased depth, and number of 3-layers architecture obtains the most discriminative performance than others. The recognition accuracy is degraded when the layer of the network more than three. The reason may be that the model overfitting the training dataset. Therefore, we use 3-layers IP-RRNNs architecture in the remainder of this paper.

D. EVALUATION ON DIFFERENT P-RRNNs MODELS

In these experiments, we study the effect of different P-RRNNs models. We already investigated the network with LSTM units better than GRU architecture for action recognition. In this set of experiments, we also further explore advantage of the LSTM architecture compare to GRU. Table 3 shows the accuracy on the UCF101. The IP-RRNNs with LSTM (IP-LSTM) model obtains the highest average accuracy 88.5%. The IP-LSTM model strengthen feature propagation to increase representational power for action recognition. Meanwhile, the results show that the recognition accuracy of LSTM is also higher than that of GRU architecture, which again proves that LSTM architecture is more suitable for human action recognition in video. Therefore, considering the performance of the model, we select IP-LSTM architecture for action recognition in the remainder of this paper.

TABLE 3. Performance of three type P-RRNNs on the UCF101.

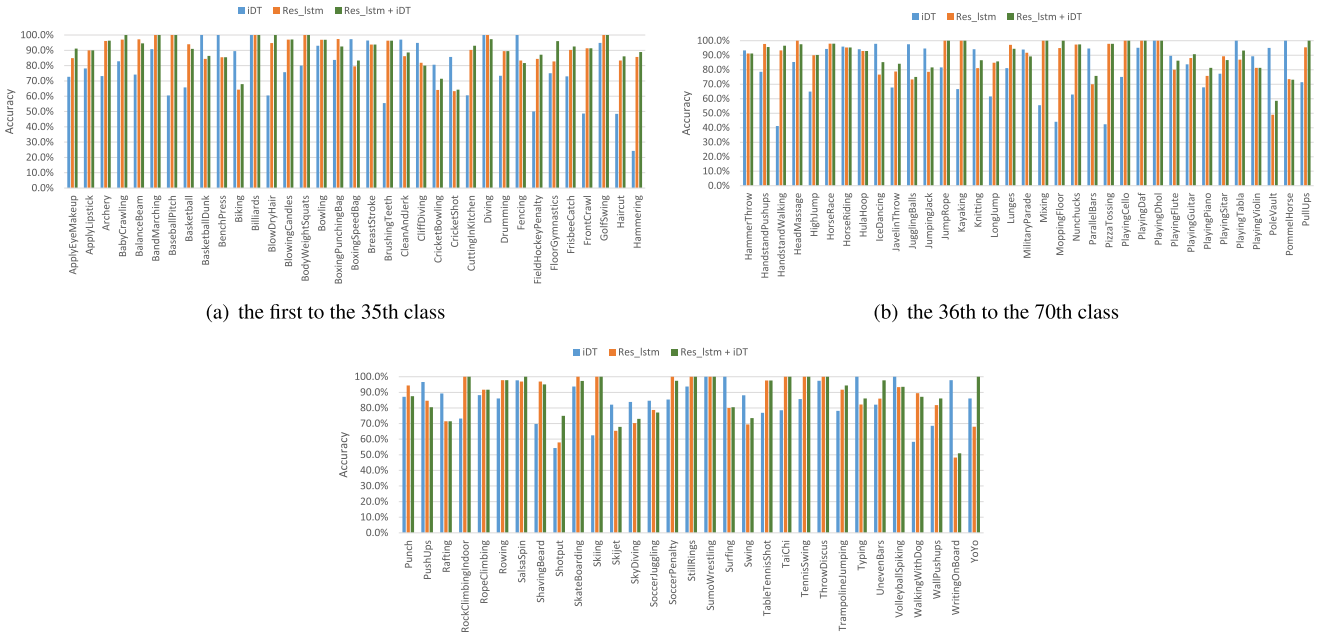
	Split1	Split2	Split3	Average
LSTM	82.3%	83.1%	82.8%	82.7%
GRU	80.4%	82.0%	81.3%	81.2%
IP-LSTM	87.8%	89.0%	88.8%	88.5%
IP-GRU	84.1%	86.4%	86.5%	85.7%
CP-LSTM	87.2%	87.8%	88.1%	87.7%
CP-GRU	83.9%	87.1%	87.1%	86.0%
RRNN-LSTM	86.6%	88.0%	88.2%	87.6%
RRNN-GRU	85.2%	86.5%	87.3%	86.3%

E. EVALUATION ON EFFECT OF HYBRID NETWORK MODELS

In this subsection, we analyze the benefit of two different networks' fusion combining the IDT model and IP-LSTM model. The hybrid network is used to learn hand-crafted features and deep learning features respectively. Specifically, IDT features extended local features such as HOF, HOG, and MBH, which depict spatial and short-term motion related features in video. The IP-LSTM extract long-term temporal feature. Possible reason could be both handcrafted and deep learning features that play important role in action recognition in video. Therefore, we choose average fusion method to obtain the probabilities of each video.

Firstly, we illustrate the split accuracy of each action category from UCF101 in Figure 7. The accuracy of each category is applied to analyze the performance of IP-LSTM, and then its impact on the recognition accuracy after fusion with the IDT features. On the UCF101 dataset, our IP-LSTM perform perfectly on most categories such as ''Baby Crawling'' and ''Golf Swing''. The results also show that the improvement is obviously for most action classes, like ''Apply Eye Makeup'', ''Archery'' and ''Baby Crawling''. On the other hand, there are some categories in the IP-LSTM model with a slightly lower accuracy than the IDT stream, such as ''Hammer throw'', ''Juggling balls'' and ''Boxing speed bag''. We find these action categories hold the characteristics of fast behavioral movements, which facilitate the extraction of IDT features and effectively describe behavioral actions.

The Table 4 shows the average accuracy of all action categories on the UCF101 dataset by fusing outputs of different streams. We can observe that the fusion model of IP-LSTM and IDT is able to significantly improve the performance and obtain a recognition accuracy of 91.4%. The results show that the IP-LSTM stream and IDT feature stream are complementary. Therefore, it is crucial to merger two types of features into successful action recognition system.



(a) the first to the 35th class

(b) the 36th to the 70th class

(c) the 71th to the 101-th class

FIGURE 7. The split1 results of IP-LSTM of all the action categories from UCF-101 dataset.

TABLE 4. Performance of the multi-stream feature fusion model on the UCF101.

	Split1	Split2	Split3	Average
IDT	82.0%	85.5%	84.0%	83.8%
IP-LSTM	87.8%	89.0%	88.8%	88.5%
IP-LSTM + IDT	90.2%	92.2%	91.9%	91.4%

TABLE 5. Accuracy of the with and without pseudo recurrent residual architecture on the UCF101.

	Without pseudo recurrent residual architecture	With pseudo recurrent residual architecture
LRCN [45]	82.9%	86.1%
Our method	85.6%	88.5%

Finally, the superiority of pseudo recurrent residual architecture is illustrated. The LRCN [45] and the proposed method are evaluated with using UCF101 dataset. The results are listed in Table 5. The results show that the pseudo recurrent residual architecture performs better than LRCN and our without pseudo recurrent residual architecture separately. The pseudo recurrent residual model is useful for action recognition in video.

F. COMPUTATIONAL COSTS

In this part, we analyze the computational cost of our method. Our IP-RRNNs model consists of two-stream CNNs and two-stream pseudo recurrent residual neural networks. The calculation of optical flow takes up time. It is around 60 millisecond for optical flow calculation of one frame image by GPU acceleration. For the training time, the two-stream CNNs need around one day to train, and two-stream IP-LSTMs need several hours to train by a Titan X GPU. In Table 6, we compare the runtime and accuracy of IP-LSTMs with other RNN-like action recognition methods. Our IP-LSTM can reach a speed of 23.2fps. In UCF101, our method is not state-of-the-art in terms of accuracy, but it is computationally efficient.

TABLE 6. Speed and accuracy results of RNNs based methods on UCF101 and HMDB51.

Method(Year)	UCF101	HMDB51	Number of parameters	Average Speed
LRCN(2015) [45]	82.9%	-	33.4M	19.6fps
Two-stream LSTM(2015) [44]	88.6%	-	36.1M	14.3fps
RLSTM-g3(2016) [85]	86.9%	55.3%	126.7M	6.3fps
MRNN(2018) [79]	81.9%	51.3%	78.5M	8.6fps
VideoLSTM(2018) [46]	89.2%	56.4%	50.3M	10.6fps
Our IP-LSTM	88.5%	58.6%	27.6M	23.2fps

G. COMPARISON WITH STATE-OF-THE-ART RESULTS

In this subsection, we compare our method against the recently proposed and relevant state-of-the-art methods. Our proposed IP-LSTM model achieves a better performance than the most of previous methods on the two datasets. The comparative results are summarized in Table 7 for the UCF101 and HMDB51 datasets. Currently, the state-of-the-art of 98.2% on UCF101 and 80.9% on HMDB51 are obtained in literature [6] and [62], which are CNNs-based models and pretrained on Kinetics [86]. Our method is pre-trained on ImageNet [41], and somewhat inferior in respect to recognition accuracy. But compared to RNNs based methods

TABLE 7. Comparison with previous state-of-the-art methods.

Feature	Method	HMDB51	UCF101
Hand-crafted feature based	IDT + SVM [13]	52.1%	83.8%
	IDT + TT [59]	65.3%	89.3%
CNN-based	Two-stream CNNs [29]	59.4%	88.0%
	I3D [26]	80.7%	98.0%
	PoTion [6]	43.7%	65.2%
	I3D + PoTion [6]	**80.9%**	**98.2%**
	TSN [28]	69.4%	94.2%
	Spatiotemporal networks [4]	68.9%	94.2%
	MARS + RGB + Flow [62]	**80.9%**	98.1%
	Asymmetric 3D-CNN (RGB + RGBF) [63]	63.5%	89.5%
	STDDCN [33]	66.9%	93.8%
RNN-based	cLSTM [48]	-	84.3%
	LRCN [45]	-	82.9%
	Two-stream LSTM [44]	-	88.6%
	RNN as encoding [15]	54.9%	81.9%
	High RNN [79]	50.8%	81.4%
	MRNN [79]	51.3%	81.9%
	VideoLSTM [46]	56.4%	88.9%
	IP-LSTM (Ours)	58.6%	88.5%
Hand-crafted feature + deep learning based	C3D + IDT [37]	-	90.4%
	Asymmetric 3D-CNN (RGB + RGBF + IDT) [63]	65.4%	95.6%
	Convolutional Pooling + IDT [2]	64.1%	89.6%
	ST-VLAD (CNN + IDT) [67]	67.6%	91.5%
	VideoLSTM + IDT [46]	63.0%	91.5%
	VideoLSTM + IDT + Objects [46]	73.7%	92.2%
	TDD + IDT [82]	65.9%	91.5%
	Spatiotemporal networks + IDT [4]	72.2%	94.9%
	IP-LSTM + IDT (Ours)	68.2%	91.4%

such as VideoLSTM [46], LRCN [45], MRNN [79], our accuracy outperforms or basically equals to previous methods on two datasets. But, as show in Table 6, our IP-LSTM model with the number of parameters is much less than other RNNs based architectures. In addition, the RNNs architecture is good at dealing with temporal features, but it may lose spatial features, which is equally critical to action recognition. Therefore, the accuracy of the RNNs-based method is lower than CNNs-based method.

We also notice that our IP-LSTM combines with IDT features can achieve 91.4% and 68.2% accuracy on UCF101 and HMDB51 dataset. Comparing with other deep model fuses IDT features, our method outperforms the VideoLSTM with [46] by 5.2% and outperforms the CNN with IDT model [67] 0.6% on HMDB51 dataset. This implies that our deep learning features are highly complimentary to hand-crafted features for action recognition.

V. CONCLUSION

In this study, we propose a pseudo recurrent residual neural network to learn long-term temporal for improving the performance in action recognition. The P-RRNNs architecture consist of two-stream pseudo recurrent residual neural networks for learn video feature from spatial stream and temporal stream. We show that LSTM with pseudo residual learning significantly improve the performance of the network. In addition, P-RRNNs can obtain more robust visual features by using holistic video clip to depict the human action feature compares to the usage of sample frames. Experimental results show that our approach achieves promising performance on UCF101 and HMDB51 datasets, and also obtain further improvements by fusing IDTs features.

For the future works, we will carry out additional studies on modelling deeper P-RRNNs to increase recognition accuracy for action recognition. We will also learn the effective fusing of deep learning features and hand-crafted features.

REFERENCES

[1] G. A. Sigurdsson, O. Russakovsky, and A. Gupta, "What Actions are Needed for Understanding Human Actions in Videos?" in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2137–2146.
[2] P. Wang, L. Liu, C. Shen, and H. T. Shen, "Order-aware convolutional pooling for video based action recognition," *Pattern Recognit.*, vol. 91, pp. 357–365, Jul. 2019.
[3] D. Wang, Y. Yuan, and Q. Wang, "Early Action Prediction With Generative Adversarial Networks," *IEEE Access*, vol. 7, pp. 35795–35804, 2019.
[4] C. Feichtenhofer, A. Pinz, and R. P. Wildes, "Spatiotemporal multiplier networks for video action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 4768–4777.
[5] A. Cherian, S. Sra, S. Gould, and R. Hartley, "Non-linear temporal subspace representations for activity recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2197–2206.
[6] W. Zhu, J. Hu, G. Sun, X. Cao, and Y. Qiao, "A key volume mining deep framework for action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 1991–1999.
[7] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre, "HMDB: A large video database for human motion recognition," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 2556–2563.
[8] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2005, pp. 886–893.
[9] R. Chaudhry, A. Ravichandran, G. Hager, and R. Vidal, "Histograms of oriented optical flow and Binet–Cauchy kernels on nonlinear dynamical systems for the recognition of human actions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 1932–1939.
[10] H. Wang and C. Schmid, "Action recognition with improved trajectories," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 3551–3558.
[11] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the Fisher Kernel for large-scale image classification," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2010, pp. 143–156.
[12] H. Jegou, F. Perronnin, M. Douze, J. Sanchez, P. Perez, and C. Schmid, "Aggregating Local Image Descriptors into Compact Codes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 9, pp. 1704–1716, Sep. 2012.
[13] H. Wang, A. Klaser, C. Schmid, and C.-L. Liu, "Action recognition by dense trajectories," in *Proc. CVPR*, Jun. 2011, pp. 3169–3176.
[14] 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, no. 1, pp. 60–79, May 2013.
[15] A. Richard and J. Gall, "A bag-of-words equivalent recurrent neural network for action recognition," *Comput. Vis. Image Understand.*, vol. 156, pp. 79–91, Mar. 2017.
[16] A. Graves, A.-R. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, May 2013, pp. 6645–6649.
[17] B. Leng, X. Zhang, M. Yao, and Z. Xiong, "A 3D model recognition mechanism based on deep Boltzmann machines," *Neurocomputing*, vol. 151, pp. 593–602, Mar. 2015.
[18] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.
[19] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.
[20] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: https://arxiv.org/abs/1409.1556

[21] 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.

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

[23] R. Girshick, "Fast R–CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.

[24] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 91–99.

[25] A. Kar, N. Rai, K. Sikka, and G. Sharma, "AdaScan: Adaptive scan pooling in deep convolutional neural networks for human action recognition in videos," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3376–3385.

[26] 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.

[27] L. Wang, Y. Xiong, Z. Wang, and Y. Qiao, "Towards good practices for very deep two-stream convnets," 2015, *arXiv:1507.02159*. [Online]. Available: https://arxiv.org/abs/1507.02159

[28] 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.* Cham, Switzerland: Springer, 2016, pp. 20–36.

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

[30] F. Osayamwen and J.-R. Tapamo, "Deep learning class discrimination based on prior probability for human activity recognition," *IEEE Access*, vol. 7, pp. 14747–14756, 2019.

[31] J. Zhu, W. Zou, and Z. Zhu, "Learning gating convnet for two-stream based methods in action recognition," 2017, *arXiv:1709.03655*. [Online]. Available: https://arxiv.org/abs/1709.03655

[32] 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.

[33] W. Hao and Z. Zhang, "Spatiotemporal distilled dense-connectivity network for video action recognition," *Pattern Recognit.*, vol. 92, pp. 13–24, Aug. 2019.

[34] A. Diba, M. Fayyaz, V. Sharma, A. H. Karami, M. M. Arzani, R. Yousefzadeh, and L. Van Gool, "Temporal 3D convnets: New architecture and transfer learning for video classification," 2017, *arXiv:1711.08200*. [Online]. Available: https://arxiv.org/abs/1711.08200

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

[36] H. Xu, A. Das, and K. Saenko, "R–C3D: Region convolutional 3D network for temporal activity detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 5783–5792.

[37] 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.

[38] D. Tran, J. Ray, Z. Shou, S.-F. Chang, and M. Paluri, "Convnet architecture search for spatiotemporal feature learning," 2017, *arXiv:1708.05038*. [Online]. Available: https://arxiv.org/abs/1708.05038

[39] L. Wang, P. Koniusz, and D. Q. Huynh, "Hallucinating IDT descriptors and I3D optical flow features for action recognition with CNNs," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2019, pp. 8698–8708.

[40] K. Hara, H. Kataoka, and Y. Satoh, "Can spatiotemporal 3D CNNs retrace the history of 2D CNNs and imagenet?" in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6546–6555.

[41] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.

[42] S. Yu, Y. Cheng, L. Xie, Z. Luo, M. Huang, and S. Li, "A novel recurrent hybrid network for feature fusion in action recognition," *J. Vis. Commun. Image Represent.*, vol. 49, pp. 192–203, Nov. 2017.

[43] K. Kawakami, "Supervised sequence labelling with recurrent neural networks," Ph.D. dissertation, Dept. Comput. Sci., Univ. Toronto, Toronto, ON, Canada, 2008.

[44] J. Yue-Hei Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici, "Beyond short snippets: Deep networks for video classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 4694–4702.

[45] J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell, "Long-term recurrent convolutional networks for visual recognition and description," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 2625–2634.

[46] Z. Li, K. Gavrilyuk, E. Gavves, M. Jain, and C. G. Snoek, "VideoLSTM convolves, attends and flows for action recognition," *Comput. Vis. Image Understand.*, vol. 166, pp. 41–50, Jan. 2018.

[47] 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: https://arxiv.org/abs/1212.0402

[48] N. Srivastava, E. Mansimov, and R. Salakhudinov, "Unsupervised learning of video representations using lstms," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 843–852.

[49] C. Schuldt, I. Laptev, and B. Caputo, "Recognizing human actions: A local SVM approach," in *Proc. 17th Int. Conf. Pattern Recognit.*, Aug. 2004, pp. 32–36.

[50] D. Weinland, R. Ronfard, and E. Boyer, "A survey of vision-based methods for action representation, segmentation and recognition," *Comput. Vis. Image Understand.*, vol. 115, no. 2, pp. 224–241, Feb. 2011.

[51] A. Jain, A. Gupta, M. Rodriguez, and L. S. Davis, "Representing videos using mid-level discriminative patches," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 2571–2578.

[52] Z. Lan and A. G. Hauptmann, "Beyond spatial pyramid matching: Space-time extended descriptor for action recognition," 2015, *arXiv:1510.04565*. [Online]. Available: https://arxiv.org/abs/1510.04565

[53] P. Scovanner, S. Ali, and M. Shah, "A 3-dimensional sift descriptor and its application to action recognition," in *Proc. 15th ACM Int. Conf. Multimedia*, 2007, pp. 357–360.

[54] A. Klaser, M. Marszałek, and C. Schmid, "A Spatio-temporal descriptor based on 3D-gradients," in *Proc. 19th Brit. Mach. Vis. Conf.*, 2008, pp. 1–275.

[55] I. Laptev, "On space–time interest points," *Int. J. Comput. Vis.*, vol. 64, nos. 2–3, pp. 107–123, Sep. 2005.

[56] D. G. Lowe, "Distinctive image features from scale–invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, Nov. 2004.

[57] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2006, pp. 404–417.

[58] C. G. Harris and M. Stephens, "A combined corner and edge detector," in *Proc. Alvey Vis. Conf.*, 1988, vol. 15, no. 50, pp. 10–5244.

[59] J. M. Carmona and J. Climent, "Human action recognition by means of subtensor projections and dense trajectories," *Pattern Recognit.*, vol. 81, pp. 443–455, Sep. 2018.

[60] F.-F. Li and P. Perona, "A Bayesian hierarchical model for learning natural scene categories," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2005.

[61] Y. Wang and G. Mori, "Max-margin hidden conditional random fields for human action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 872–879.

[62] N. Crasto, P. Weinzaepfel, K. Alahari, and C. Schmid, "MARS: Motion-augmented RGB stream for action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019.

[63] H. Yang, C. Yuan, B. Li, Y. Du, J. Xing, W. Hu, and S. J. Maybank, "Asymmetric 3D Convolutional Neural Networks for action recognition," *Pattern Recognit.*, vol. 85, pp. 1–12, Jan. 2019.

[64] 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.

[65] X. Yang, P. Molchanov, and J. Kautz, "Multilayer and multimodal fusion of deep neural networks for video classification," in *Proc. 24th ACM Int. Conf. Multimedia*, 2016, pp. 978–987.

[66] Y. Shi, Y. Tian, Y. Wang, and T. Huang, "Sequential deep trajectory descriptor for action recognition with three–stream CNN," *IEEE Trans. Multimedia*, vol. 19, no. 7, pp. 1510–1520, Jul. 2017.

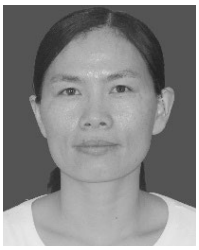
[67] I. C. Duta, B. Ionescu, K. Aizawa, and N. Sebe, "Spatio-temporal vlad encoding for human action recognition in videos," in *Proc. Int. Conf. Multimedia Modeling*. Cham, Switzerland: Springer, 2017, pp. 365–378.

[68] F.-P. An, "Human action recognition algorithm based on adaptive initialization of deep learning model parameters and support vector machine," *IEEE Access*, vol. 6, pp. 59405–59421, 2018.

[69] A. Cherian, B. Fernando, M. Harandi, and S. Gould, "Generalized rank pooling for activity recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3222–3231.

[70] V. Choutas, P. Weinzaepfel, J. Revaud, and C. Schmid, "PoTion: Pose motion representation for action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7024–7033.
[71] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Proc. 31st AAAI Conf. Artif. Intell.*, 2017.
[72] C. Feichtenhofer, A. Pinz, and R. Wildes, "Spatiotemporal residual networks for video action recognition," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 3468–3476.
[73] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 3104–3112.
[74] S. Merity, N. S. Keskar, and R. Socher, "Regularizing and optimizing LSTM language models," 2017, *arXiv:1708.02182*. [Online]. Available: https://arxiv.org/abs/1708.02182
[75] R. Trianto, T.-C. Tai, and J.-C. Wang, "Fast-LSTM acoustic model for distant speech recognition," in *Proc. IEEE Int. Conf. Consum. Electron. (ICCE)*, Jan. 2018, pp. 1–4.
[76] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: A neural image caption generator," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3156–3164.
[77] C.-Y. Ma, M.-H. Chen, Z. Kira, and G. Alregib, "TS–LSTM and temporal-inception: Exploiting spatiotemporal dynamics for activity recognition," *Signal Process., Image Commun.*, vol. 71, pp. 76–87, Feb. 2019.
[78] L. Wang, Y. Xu, J. Cheng, H. Xia, J. Yin, and J. Wu, "Human action recognition by learning Spatio–temporal features with deep neural networks," *IEEE Access*, vol. 6, pp. 17913–17922, 2018.
[79] Z. Zheng, G. An, and Q. Ruan, "Multi-level recurrent residual networks for action recognition," 2017, *arXiv:1711.08238*. [Online]. Available: https://arxiv.org/abs/1711.08238
[80] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," 2014, *arXiv:1406.1078*. [Online]. Available: https://arxiv.org/abs/1406.1078
[81] K. Greff, R. K. Srivastava, J. Koutnik, B. R. Steunebrink, and J. Schmidhuber, "LSTM: A search space odyssey," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 10, pp. 2222–2232, Oct. 2017.
[82] L. Wang, Y. Qiao, and X. Tang, "Action recognition with trajectory-pooled deep-convolutional descriptors," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 4305–4314.
[83] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.
[84] Y. Gal and Z. Ghahramani, "A theoretically grounded application of dropout in recurrent neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 1019–1027.
[85] B. Mahasseni and S. Todorovic, "Regularizing long short term memory with 3D human–skeleton sequences for action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 3054–3062.
[86] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev, M. Suleyman, and A. Zisserman, "The kinetics human action video dataset," 2017, *arXiv preprint arXiv:1705.06950*.

SHENG YU received the B.S. degree from Jinggangshan University, China, in 2007, and the M.S. degree from Nanchang Hang Kong University, China, in 2010, and the Ph.D. degree in computer science and technology from Xiamen University, China, in 2018. He is currently a Lecturer with the Shaoguan University, in 2013. His research interests include computer vision and pattern recognition.



LI XIE received the B.S. degree from the Hunan University of Science and Technology, China, in 2006, and the M.S. degree from Hunan University, China, in 2009. She is currently a Lecturer with the Shaoguan University, in 2011. Her research interests include multimedia computing and deep learning.

LIN LIU received the Ph.D. degree from the School of Computer Science and Engineering, South China University of Technology, in 2013. He is currently an Associate Professor with the Shaoguan University. His research interests include mass storage technology, information system energy saving, and cloud computing.

DAOXUN XIA received the B.S. and M.S. degrees from the School of Mathematics and Computer Science, Guizhou Normal University, China, in 2004, the M.S. degree from the College of Computer Science and Technology, Guizhou University, China, in 2010, and the Ph.D. degree from the College of Information Science and Engineering, Xiamen University, in 2016. He is currently an Associate Professor with the Guizhou Normal University. His main research interests include object detection, object recognition, computer vision, and big data technology.