data

MDPI

Article

Distinct Two-Stream Convolutional Networks for Human Action Recognition in Videos Using Segment-Based Temporal Modeling

Ashok Sarabu * and Ajit Kumar Santra *

SITE, VIT University, Vellore, Tamil Nadu 632014, India

* Correspondence: sarabu.ashok@gmail.com (A.S.); ajitkumar@vit.ac.in (A.K.S.)

Received: 16 October 2020; Accepted: 8 November 2020; Published: 11 November 2020

Abstract: The Two-stream convolution neural network (CNN) has proven a great success in action recognition in videos. The main idea is to train the two CNNs in order to learn spatial and temporal features separately, and two scores are combined to obtain final scores. In the literature, we observed that most of the methods use similar CNNs for two streams. In this paper, we design a two-stream CNN architecture with different CNNs for the two streams to learn spatial and temporal features. Temporal Segment Networks (TSN) is applied in order to retrieve long-range temporal features, and to differentiate the similar type of sub-action in videos. Data augmentation techniques are employed to prevent over-fitting. Advanced cross-modal pre-training is discussed and introduced to the proposed architecture in order to enhance the accuracy of action recognition. The proposed two-stream model is evaluated on two challenging action recognition datasets: HMDB-51 and UCF-101. The findings of the proposed architecture shows the significant performance increase and it outperforms the existing methods.

Keywords: segment-based temporal modeling; two-stream network; action recognition

1. Introduction

Human Action Recognition is an emerging research area that has gained prominent attention in computer vision. One of the reason for which researchers are interested in the action recognition in video is the wide range of its applications in human-computer communication, video retrieval, management of web videos, surveillance [1], medicine, etc. When ompared to the still image recognition, temporal content in the video provides supplemental data for action recognition, as the number of actions can be accurately recognized using motion information. Action recognition in videos is a strenuous job because of the similarity in visual contents (frames) [2], view-point variation, camera motion, scale and pose of actor, and lighting conditions. Recently, the introduction of deep CNNs has made a major breakthrough performance in speech and image recognition tasks. Since then, computer vision researchers have started to apply the deep CNNs to action recognition in videos [3,4].

Deep learning in video action recognition is relatively slow when compared to image recognition. There are two reasons; first, scale and diversity of the video action recognition datasets are relatively small as compared to the image recognition datasets. Thus, small datasets will lead to overfitting, and the model will not be generalized for recognition. It is hard to create large-scale video datasets and train them on depth networks. Second, when compared to the image datasets, video data will contain an additional cue, called temporal information, which needs complex data analysis. Recently, many researchers have made attempts to solve these challenges and proposed solutions. Karpathy et al. [3], studied the performance of video action recognition on SPORTS-1M video classification dataset, compared the different CNN models. Du et al. proposed the C3D model,

a three-dimensional convolution network for video action recognition. Later, Simonyan et al. [5] presented a two-stream architecture for the first time for action recognition in videos, that works on two CNNs which showed good performance improvement. The researchers for the aforementioned methods are able to utilize the temporal component, but work only for a short time; in lengthy videos, information cannot persist for a long time. To solve this problem, Wang et al. [6] designed a video level segmental architecture, called Temporal Segment Networks that can efficiently learns the features and retrieve the long-range time-varying features from the videos.

In this paper, we propose a two-stream CNN model for identifying actions in videos built on a two-stream network model. The proposed architecture is inspired from a two-stream idea [5], a two-stream model with the similar two-stream structures for human action recognition in videos. Specifically, the RGB image is the input to the spatial stream. Furthermore, the stack of consecutive optical flow images is the input to the temporal stream. Each stream is implemented while using identical two-stream, and the final results of both streams are combined with the late fusion technique. The other methods proposed in [5–11], by researchers utilized similar network models for two streams for human action recognition in videos. However, in human visual cortex systems, recognizing an object and its action are entirely two different processes. Inspired by the human visual cortex process, we proposed similar two-stream CNN architecture for action recognition in videos. Because of the variable length of videos, we attempt to add a video segmentation technique [6], to retrieve the long term temporal features. The proposed model of two-stream convolutional network is shown in Figure 1. Data augmentation and advanced cross-modal pretraining are employed because of the small size of datasets and to avoid labeled noise. The first step in our model is segmenting video in three parts. In the next step, the three snippets are randomly sampled and fed into the proposed two-stream network. Subsequently, the final category score is captured at the end of each stream of the network and fused for the final video level prediction.

With experimental results of our proposed model using the two most popular action recognition datasets, HMDB-51, and UCF-101, the contribution to this papers is three-fold. First, two-stream with multiple networks produces better performance than the two-stream with similar network models. In our experimentation, we found that ResNets and Inception-V2 produced better feature extraction and performance than other network models. Second, data augmentation and advanced cross-modal pre-training techniques are employed because of existing small datasets and noisy labels. Finally, the segment based temporal modeling technique for long-term temporal information better captures long-range information.

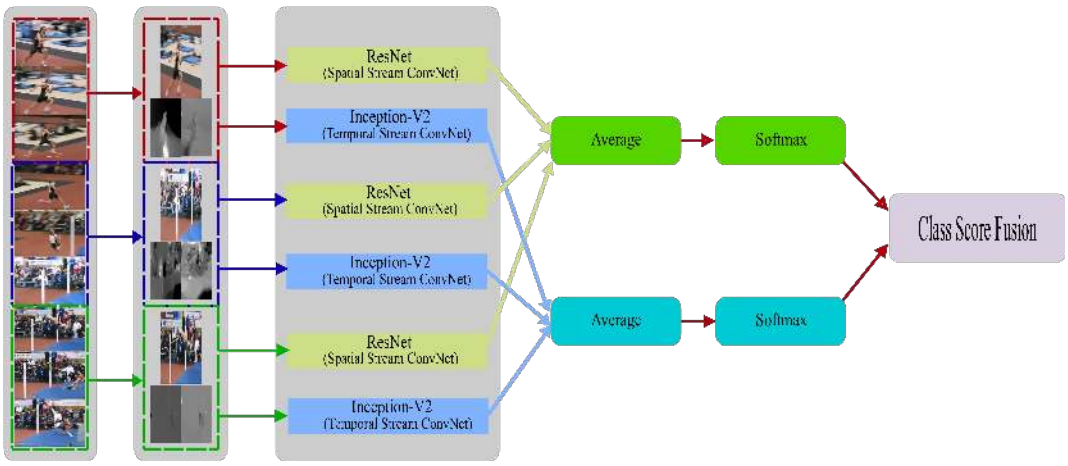

Figure 1. Distinct Two-Stream Convolutional Networks for Human Action Recognition in Videos while using Segment-Based Temporal Modeling.

2. Related Works

Recently, deep CNNs attained tremendous success in image recognition. Driven with the success of CNNs in image recognition, computer vision researchers transferred to videos. Action recognition in videos in deep learning is categorised into three categories that are based on the network architectures (1) Space-time networks. (2) Hybrid networks. (3) Two-stream networks.

2.1. Space-Time Networks

Space-time networks are the two-dimensional convolution networks with an additional convolution operation for temporal information. Ji et al. [12] presented a model that is one of the seminal works, recognizes actions in videos applying convolution neural networks. Ji et al. [12] extract spatial and temporal information by applying three-dimensional convolutions on adjacent frames. The networks repeat the same three-dimensional (3D) convolutions and sampling. Finally, a 128-dimensional feature vector is generated and it is used for action classification.

The 3D CNN in [12] was later extended to three-dimensional convolution networks [4], a deep network architecture trained on large scale datasets. The three-dimensional convolution networks contain five convolution layers, five max-pooling layers, two fully connected layers, and a softmax loss function layers. Even though information of the two streams is considered in training, the overall cost of computing and model storage is remarkably high. Liu et al. [13] proposed SSNET, stack of convolutional layers are added to temporal data and showed the best performance on skeleton data. Diba et al. [14] presented a three-dimensional temporal architecture, a new temporal layer called "Temporal Transition Layer" applied in the 3D DenseNet-based network. This method ignored the temporal information and only evaluated the RGB frames. Qui et al. [15], proposed Pseudo-3D Residual Network, (3*3*3) convolution filters are replaced with (1*3*3) in the spatial stream and (3*1*1) convolution filters in the temporal stream. (3*1*1) convolution filter is used in order to extract spatial information, and (1*3*3) is used to retrieve temporal features. When compared to the 3D convolution networks, this architecture is successful in terms of performance in video action recognition.

2.2. Hybrid Networks

Hybrid networks work on the principle of aggregating temporal information [16,17]. The aggregation of temporal information is done by adding the recurrent layers on the top layers of CNN's. These networks take advantage of both CNNs and LSTMs, and shows the positive results in capturing the spatial information, temporal information, and long-range dependencies [8,18–20]. Wang et al. [16], presented Long-term Recurrent Convolution Network (LRCN), in which frames were processed with CNN, and the output of CNN is fed into a stack of LSTMs. Veeriah et al. [21], designed a different gating for LSTM, called differential Recurrent Neural Network (dRNN). This method is good at learning significant spatio-temporal structures. Ng et al. [17] proposed two methods that can handle full-length videos. Two methods aggregates frame-level outputs of CNN to video level prediction. They discussed six different methods that showed, adding of LSTM layers after CNN outperforms temporal pooling information. Wu et al. [22] presented a hybrid network that uses both CNN and LSTM. They first extract spatial, temporal features with CNN and later fed as input to LSTM network for long term temporal features. However, because of the additional parameter of LSTM, LSTM has not shown the acceleration in performance in action recognition in videos.

2.3. Two-Stream Networks

Two stream networks use two CNNs for spatial and temporal information in videos. Simonyan et al. [5], presented a two-stream architecture for action recognition in videos. In this architecture, RGB images are fed into a spatial stream, optical flow frames are fed into temporal stream. Finally, softmax scores are obtained by fusing outputs of two-stream outputs. Wang et al. [8] presented Trajectory-pooled Deep-convolution Descriptor and integrated trajectory features and deep

network learned features. This method shows superior performance by combining deep networks and shallow local features. Feichtenhofer et al. [9] proposed a new spatiotemporal architecture and explored various fusing schemes. They found fusing network spatially at last convolution layers boosts accuracy. Wang et al. [6] introduced Temporal Segment Networks, in which they improved the performance by training on the whole video by modeling long-range temporal structure. Yunbo et al. [11] introduced an end-to-end learning neural network, which combinedly performs pixel level action recognition and segmentation. They solved the action recognition by two-stream network along with temporal aggregation. Christoph et al. [7], designed a spatio-temporal ResNet, which allows the learning of the spatio-temporal feature by connecting static and optical flow channel streams, which increases the interactions between both streams.

3. Technical Approach

In this section, we provide a comprehensive overview of our proposed network architecture for action recognition in videos. First, we discuss the proposed distinct two-stream convolutional networks for human action recognition in videos. Subsequently, ResNet and Inception-V2 used as CNN model for spatial stream and temporal stream are discussed. After that, Temporal segmentation Network is introduced to capture the long-range temporal features. Finally, optimizing the network training strategies are presented.

3.1. Distinct Two-Stream Convolution Networks

Video is a collection of spatial and temporal information. The human visual cortex system that is mentioned in [23] processes information with two streams called spatial stream and temporal stream. Information is static image appearance in spatial stream; it only depicts scenes and objects. In the temporal stream, information is the movement of objects between consecutive frames, conveys the orientation of camera and objects. Inspired from [5,23], designed a two-stream CNN to retrieve the spatial and temporal features with two similar CNN models from videos. For spatial information, RGB frames are used. For temporal information, dense optical flow frames are used in order to extract the motion of objects across the video. Each of these streams is processed identical and independent deep convolutional neural network models. Specifically, the RGB image is fed to the spatial stream CNN. For temporal stream, stack of optical flow images are fed as input. Optical flow images are a combination of horizontal and vertical convoluted images. The number of optical flow images (L) is set to ten. Because the optical flow images consist of both horizontal and vertical convoluted images, the total number of flow images is set to 2L = 20 [5]. Finally, spatial and temporal streams are individually trained end-to-end, and the output of two streams are combined to get the final classification decision. Averaging and SVM are two fusing methods used in [5], in order to fuse scores of two streams.

In this subsection, we present our distinct two-stream CNN for action recognition based on the architecture presented in [5]. In this network architecture, the spatial and temporal streams are trained with different CNN models, as shown in Figure 2. The reasons for modeling our proposed architecture is, when two-streams with similar CNNs are trained and fused together, generates a large number of redundant features. Because optical flow frames are horizontal and vertical components that are derived from the RGB image, and when trained with a similar CNN generates redundant features. The second reason is, in human action recognition, object recognition and motion recognition are two different processes. Similarly, here. two-stream action recognition can be trained with two different CNN models. With many experiments, we observed that the performance of distinct two-stream action recognition is better than the two-stream model with similar CNNs.

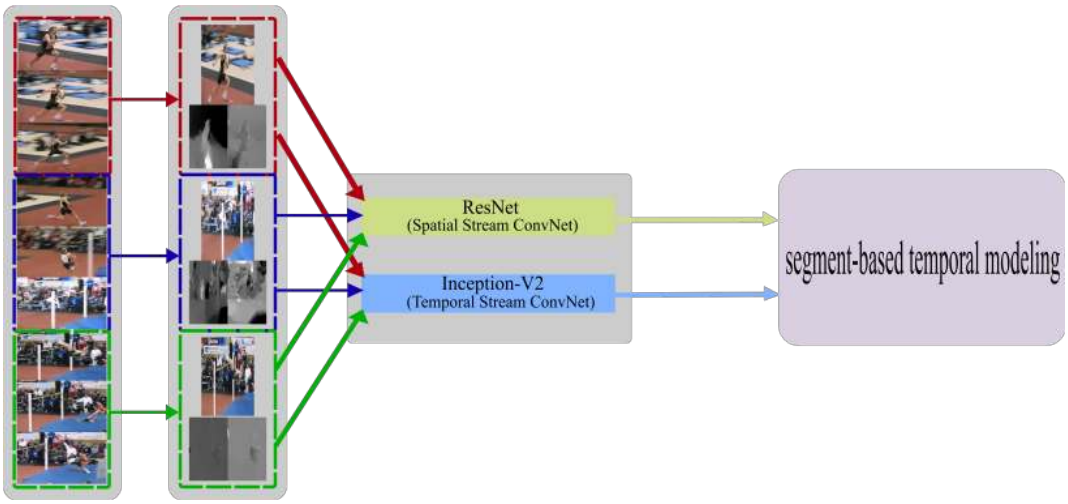

Figure 2. Distinct Two-Stream Convolutional Network.

3.2. Base Networks

In the above section, we discussed a two-stream spatio-temporal action network. A good action recognition model will retrieve more discrete spatial and temporal features. Previous studies [24,25] have shown that deeper CNN models can extract discrete features. In [25], the features of hidden layers and its mechanism of the CNN model were visualized. Moreover, when compared to the different CNN models with different depths and going deeper layer, the CNN model is better in extracting discriminant features, which can increase the prediction rate of the model. Another set of recent studies [26,27], showed that with the increase in network depth can learn more features in lengthy videos. Residual Networks (ResNet) in [24,28], addressed the issue of degradation [29], which is caused by deep layers of the CNNs. ResNets and Inception-V2 are the underlying networks in our models. ResNet is used to retrieve spatial and temporal features and Inception-V2 is utilized to increase the performance of model. We introduce two models ResNet and Inception-V2, in order to investigate them further and explore the potential of the distinct two-stream CNN.

3.2.1. Residual Network

ResNet is used as one of the CNN models in our proposed network. The primary reason to use ResNet with deep layers is that it extracts discriminant features from frames. Network degradation arises as to the number of layers increases. To solve this issue, He et al. [24] presented a deep ResNet. Instead of the original underlying fitting function, they use a trained residual network by residual unit is

$$x_{i+1} = \sigma\left(x_i + F\left(x_i; W_i\right)\right) \tag{1}$$

where x_i and x_{i+1} are the input and output of the ith layer of the network. $F\left(x_i; W_i\right)$ is non-linear residual mapping of the weight of CNN filters $W_l = \left\{ W_{l,k} \middle|_{1<k<K} \right\}$, and sigma is the ReLU function [30]. The benefit of using the residual block is that it acts as the shortcut connection that connects the first layer to any layer in the network, which breaks the conventional form of connecting one layer to the next layer. With this, the gradient loss may skip some layers and pass from the loss layer to any layer that it is connected, and this will avoid the gradient explosion problem. This shortcut connection does not increase the computational cost and an increase in the number of parameters. In ResNet, after every convolution operation and before the activation layer, batch normalization (BN) [31], is performed. This will solve the covariate shift problem and, also, the convergence of the network will be fast [24]. Finally, the global average pooling and softmax layer are employed combinedly in the place of a single fully connected layer. This effectively decreases the number of

parameters. Besides, the bottleneck structure will decrease the computational overhead, and the network efficiency is guaranteed.

3.2.2. Inception-V2

Inception-V2 [31] is used as another CNN model in our proposed network. Inception-V2 is the module used in order to reduce the CNN complexity. This CNN model is the advanced version of the GoogleLeNet [26], which solves the saturation and vanishing gradient problem. The distribution of X is to be unchanged, because even a minor change will be change the value of X when the network goes deep. A higher learning rate can be used for faster optimization. The primary concept of the Inception-V2 is the replacement of 5*5 convolution with two 3*3 convolutions. This replacement of convolution will not only decrease the parameter number, but also increase more non-linear transformations, enhances the model to learn more features. Other advantages of adding the batch normalization reduce the internal covariate shift by normalizing the output of each layer to N(0,1).

In original two-stream CNN for video action recognition [5], VGG-M-2048 [32], is used to train the model, and both of the streams use the same network structure. Feichtenhofer et al. [9], improved the performance using VGG-16 [25] instead of VGG-M-2048. ResNet and Inception-V2 are used as the CNN models in our proposed architecture. ResNets with an increase in the number of layers can extract more features [24]. Furthermore, Inception-V2, with an increase in the network depth and width, can improve performance [31]. Looking at the benefits of the ResNet and Inception-V2, we used these as base models in our two-stream architecture. ResNet has less computational complexity and filters when compared to the VGG-M-2048. In terms of computational complexity, VGG-16 uses 15.3 B FLOPs and VGG-19 uses 19.6 B FLOPs, whereas ResNet-152 only uses 11.3 B FLOPs. Similarly, the computational complexity of ResNet-50 is 3.8 B FLOPs and ResNet-101 is 7.6 B FLOPs. Finally, the total number of parameters of both streams are 182 M in our model.

3.3. Segment-Based Temporal Modeling

Problem with the original two-stream CNN [5] architecture is its inability to maintain temporal information in deep CNN networks. The cause for this problem is, it only works on one frame in the spatial stream or a stack of optical flow frames for the temporal stream. Therefore, the network is unable to retrieve long-range temporal information effectively. Segment based long-range temporal information plays an important role in finding action recognition in videos. For example [22,33], in some complex video actions, comprises multiple stages are required in order to classify the action and subject. And, action is important from the beginning to the final point of the video (basketball dunk and shooting similar for some short time, so start to the endpoint to be considered to classify correct action). Therefore, there may be misclassification of action if a video is considered only for some part of the time, which leads to unsatisfied performance. To improve the performance, we implement the long-range temporal model proposed in [6] to extract long-term temporal information in our proposed distinct two-stream convolutional networks for human action recognition in videos.

In order to model the long-range segment based temporal modeling [6], we divided the video into K segments (K = 3) in equal duration, expressed as {S1, S2, S3}. For short snippets, modeling is done while using,

$$\mathrm{TSN}\left(T_1, T_2, \cdots, T_j\right) = \mathcal{H}\left(\mathcal{G}\left(\mathcal{F}\left(T_1; \mathbf{W}\right), \mathcal{F}\left(T_2; \mathbf{W}\right), \cdots, \mathcal{F}\left(T_j; \mathbf{W}\right)\right)\right) \quad (2)$$

where $\mathcal{F}(T_1;\ \mathbf{W})$ represents Convolutional function with parameters, $\mathcal{G}$ represents an averaging function, H represents softmax function. Subsequently, we sample each segment (S_j) into short snippets {T_1, T_2, T_3}. These short snippets are fed as an input to the proposed two-stream architecture to get an initial action classification score. Afterwards, this score is fused with average function to obtain a final decision among snippets. Based on this consensus, the final prediction scores are calculated

while using the widely used softmax function. Additionally, the final loss function of segmental consensus is calculated while using the equation,

$$\mathcal{L}(y, \mathbf{G}) = -\sum_{i=1}^{C} y_i \left(G_i - \log \sum_{j=1}^{C} \exp G_j \right) \tag{3}$$

where 'n' is the total number of the acting categories, y_i is the ground-truth label, **G** is classification score of *i*. The value of **G** is average result of the short snippets of same categories. For the proposed architecture, all segmental frames are utilized together to optimize the network parameter **W**. In the backpropagation, the gradient of W to the loss value L can be derived as,

$$\frac{\partial \mathcal{L}(y, \mathbf{G})}{\partial \mathbf{W}} = \frac{\partial \mathcal{L}}{\partial \mathbf{G}} \sum_{k=1}^{K} \frac{\partial \mathcal{G}}{\partial \mathcal{F}(T_k)} \frac{\partial \mathcal{F}(T_k)}{\partial \mathbf{W}} \tag{4}$$

Subsequently, we use stochastic gradient descent (SGD) to train the model parameters. As in Equation (4), guarantees that the model parameters are updated using the segmental consensus category **G** for three short snippets. Thus, with this optimization technique, long-range segment based temporal information is preserved, and model parameters are learned from the entire video. Aggregation function (**G**) is the main concept in segment-based temporal modeling. The averaging function is an aggregation function used to predict by averaging final results at the snippet level for every class with $g_i = \frac{1}{N} \sum_{n=1}^{N} f_i^n$. Final result of this function, g_i with respect to $f_i^k$is,

$$\frac{\partial g_i}{\partial f_i^n} = \frac{1}{N} \tag{5}$$

4. Network Training Strategies

4.1. Data Augmentation

Data augmentation is used in our model in order to create manifold training samples. Data augmentation strategies are used when there are fewer training samples to avoid overfitting problems. In our proposed model, horizontal flipping, random cropping, and scale-jittering [27] are employed to augment the training data. In corner cropping, the extracted regions are only selected from the corner or center of the image to avoid focus of the information only on the center of the image. We use the multi-scale jittering technique [34], which is applied in ImageNet classification. We set the input size of the image or optical flow field to 256 × 340. Height and width of cropped region are selected randomly from {256,224,192,168} and resized to 224 × 224 for model training.

4.2. Advanced Cross-Modal Pre-Training

Pre-training is a great way to initialize the deep convolution neural network model when the dataset size is small, i.e., when the training samples are less [5]. Input to the Spatial stream network is RGB images, so a pre-trained model can be used, such as ImageNet [30], in order to set the initial weights of CNN. However, the input to the temporal stream network is optical flow frames, and, Optical flow frames and RGB difference contain the distinct features of video, and their data distribution is not the same as RGB images. Therefore, it is not possible to use pre-trained networks for temporal stream networks. Accordingly, we propose an advanced cross-modal pre-training technique. First, we apply the linear transformation operation [6] on optical flow frames to get the values in an interval of [0, 255]. Now, the values of optical flow frames will be in the same range of RGB images. Then, the first layers of CNN weights of RGB models are modified to fit the weights of the optical flow fields (because the RGB image has three channels and temporal stream input has 10 inputs, including horizontal and vertical images, we average the weight of the three channels weights of RGB

to replicate the channel number of temporal network input (output kernel size = (64,10,7,7))). We do this process from scratch, and then we replace the values of the first layer of CNN model with the values of same layers of the RGB pre-trained model.

5. Experiments

In this section, we discuss the implementation details of proposed architecture and the datasets. Subsequently, we evaluate the performance of two-stream networks with similar and distinct network architectures. Afterwards, the performance of the proposed advance cross-modal pre-training is presented. Finally, we present the experimental results and analysis in the last section.

5.1. Datasets and Implementation Detials

We perform experiments on large-scale action recognition datasets, namely UCF101 [35] and HMDB51 [36]. The UCF101 dataset consists of 101 action classes with 13,320 videos in total. Each video consists of an average of 100–300 frames with a duration of 3–10 s. The HMDB51 dataset consists of video clips from different online sources, such as YouTube and Google. The dataset consists of 51 action categories with 6766 videos in total. We evaluate the proposed two-stream architecture by following the standard evaluation scheme while using three training and testing splits of the UCF-101 dataset. Affitionally, the results compared with state-of-art methods. The evaluation of average accuracy is made on three splits of UCF-101 and HMDB-51.

The mini-batch gradient descent method is implemented to train the network parameters. We initialize the network parameters with pre-training models from ImageNet [30]. We initialize the values of batch size, weight decay, and momentum to 256, 0.0005, and 0.9, respectively. Initially, both stream's learning rate is initialized to 10^{-4}. When training the spatial stream network learning rate is decreased to 10^{-1} for every 15 K iterations and the entire network training halts at 36 K iterations. Similarly, when training the temporal stream network, the learning rate is decreased to 1/10 at 20 K and 32 K iterations and the entire network training halts at 40 K iterations. TVL1 optical flow algorithm [37] is used to extract optical flow frames from videos. To speed-up the training process, we apply data-parallelization with multiple GPUs on the Caffe platform [38] and related code is released on GitHub (https://github.com/ashoksarabu/Distinct-Two-Stream).

5.2. Testing

We evaluate our proposed model with the parameters of the original two-stream convolution network [5]. We sample a fixed number of RGB images or optical flow stacks (25 in our experiment) with an equal interval of times between them. For each of the frames, we crop four corners, one center, and horizontal flipping to evaluate the CNNs. We use weighted averaging to fuse two stream's results. When the network is trained, the performance gap between two streams is smaller than the original two-stream convolution network [5]. Because of this small gap, we initialize the weights of spatial stream to 1 and temporal stream to 1.5.

5.3. Exploration Study

In this section, we examine and evaluate the efficiency of the proposed network with the two-stream identical networks. We propose an improved cross-modal pre-training approach in Section 3 is evaluated in the experiments and its effectiveness of the proposed network model.

The experimental tests are performed on the proposed CNN architecture using the same CNN model with different depths, and with different CNN models. Inception-V2 [31] and ResNet with different depths are used to evaluate and test the model. ResNet-50, ResNet-101, and ResNet-152 [24] are ResNets with different depths used as CNN for both streams. The experimental results of the proposed model are evaluated and compared in Table 1. The comparisons of experimental results are made based on 1. Two streams with the identical CNN model, 2. Two streams with non-identical network models and depths. From Table 1, we found out that ResNet-101 performed better for spatial

stream network. When compared to the two streams CNN with similar networks, two-stream CNNs with different networks performed better.

Table 1. Performance of Distincttwo-stream Convolutional Network on UCF-101.

Network Architectures for Two-Streams	Spatial	Temporal	Two-Stream
Spatial_ResNet-101 + Temporal_ResNet-50	84.1%	85.3%	94.3%
Spatial_ResNet-152 + Temporal_ResNet-50	86.2%	85.3%	94.3%
Spatial_ResNet-101 + Temporal_Inception-V2	84.1%	88.8%	95.0%

Moreover, similar networks with different depths performed well as compared to similar networks with similar depths. ResNet-50 performed better for the temporal stream and ResNet-101 for the spatial stream network. We achieved the best performance with an accuracy of 95.00 percent when ResNet-101 is used as the spatial stream network model, and Inception-V2 is used as the temporal stream network model.

We evaluate the experiments with ResNet-50 and Inception-V2 models to verify the efficiency of advanced cross-modal pre-training technique discussed in the previous section, as mentioned above. Specifically, three case-studies are used. First, training the temporal stream network from scratch. Second, training the temporal stream network with the technique proposed in [6]. Third, training the temporal stream network with our proposed method. The experimental results of the three case studies mentioned earlier are performed on UCF-101 dataset and tabulated in Table 2. From the results that are tabulated in Table 2, we summarize that method used for pre-train the temporal stream network and initializing a deep convolution network achieved great accuracy when compared to training from scratch. Moreover, the proposed advanced cross-modal pre-trained has an increase of 0.3% with CNN models ResNet-50 and Inception-V2 when compared to the method proposed in [6].

Table 2. Performance evaluation of temporal stream CNN on UCF-101dataset.

Training Strategy	ResNet-50	Inception-V2
From scratch	78.5%	82.4%
Pre-Training [6]	84.1%	87.3%
Proposed - Advanced cross-modal pre-training	85.3%	88.8%

5.4. Comparison with State-of-the-Art

After investigating the different models for two-stream models for recognizing human action in videos, we found the optimal accuracy. We evaluation of the proposed model are based on the UCF-101 and HMDB-51 action recognition datasets on all splits and reported. The empirical results are presented in Table 3. When compared to state-of-the-art results, our proposed architecture with ResNet-101 for the spatial stream network and the Inception-V2 model for the temporal stream network, has performed better. Compared to the original two-stream convolution neural network [5] and ST-ResNet model [7], the accuracy for the UCF-101 dataset has been improved by 7.10% and 1.7%. Similarly, for the HMDB-51 dataset, compared to the original two-stream convolution neural network [5] and the ST-ResNet model [7], we got optimal accuracy has been increased by 8.5% and 1.5%, respectively. From the experimental findings, we conclude that effectiveness of our distinct two-stream convolutional network for human action recognition in videos based on segment based temporal modeling. Furthermore, spatiotemporal heterogenous network accuracy has been improved compared to the two-stream action recognition methods with similar network models.

Table 3. Comparison of our proposed method distinct two-stream convolutional network with state-of-art methods on UCF-101 andHMDB-51 datasets.

Methodology	UCF-101	HMDB-51
Two-stream network [5]	88.0%	59.4%
Two-stream network fusion [9]	92.5%	65.4%
Spatio-Temporal 3D CNNs [4]	85.2%	–
Factorized Spatio-Temporal CNNs [36]	88.1%	59.1%
Pseudo-3D residual networks [14]	93.7%	–
Temporal Segment Networks [6]	94.0%	68.5%
Temporal 3D CNNs [13]	93.2%	63.5%
SpatioTemporal residual networks [7]	93.4%	66.4%
(Proposed) Distinct two-stream CNN	95.0%	67.9%

6. Conclusions

In this paper, we presented a distinct two-stream convolutional networks for recognizing human action in videos using segment based temporal modeling. Human action recognition is two individual processes, which is, two different independent streams processes appearance and motion. Inspired by this, we attempted to experiment with the two-stream convolution neural network with two different network models for two streams. Additionally, we achieved the best performance when compared to existing two-stream networks. With all the experiments, it is found that the distinct two-stream convolution networks for recognizing action in videos perform better than two-stream convolution networks with similar network models. In our experiments, we found that ResNet-101 and Inception-V2 models, when employed as network models for a two-stream network with segment based temporal modeling, yield the best performance. Finally, data augment techniques and advanced cross-modal pretraining are applied in order to increase the performance.

Author Contributions: Conceptualization, A.S.; Investigation, A.S. and A.K.S.; Methodology, A.S.; Resources, A.S.; Software, A.S.; Supervision, A.K.S.; Validation, A.K.S.; Writing—original draft, A.S.; Writing—review & editing, A.S. and A.K.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Nanda, A.; Sa, P.K.; Choudhury, S.K.; Bakshi, S.; Majhi, B. A neuromorphic person re-identification framework for video surveillance. *IEEE Access* **2017**, *5*, 6471–6482. [CrossRef]
2. Nanda, A.; Chauhan, D.S.; Sa, P.K.; Bakshi, S. Illumination and scale invariant relevant visual features with hypergraph-based learning for multi-shot person re-identification. *Multimed. Tools Appl.* **2019**, *78*, 3885–3910. [CrossRef]
3. Karpathy, A.; Toderici, G.; Shetty, S.; Leung, T.; Sukthankar, R.; Fei-Fei, L. Large-scale video classification with convolutional neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 24–27 June 2014; pp. 1725–1732
4. Tran, D.; Bourdev, L.; Fergus, R.; Torresani, L.; Paluri, M. Learning spatiotemporal features with 3d convolutional networks. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 4489–4497.
5. Simonyan, K.; Zisserman, A. Two-stream convolutional networks for action recognition in videos. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 8–13 December 2014; pp. 568–576.
6. Wang, L.; Xiong, Y.; Wang, Z.; Qiao, Y.; Lin, D.; Tang, X.; Gool, L.V. Temporal segment networks: Towards good practices for deep action recognition. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2016; pp. 20–36.

7. Christoph, R.P.W.; Pinz, F.A. Spatiotemporal residual networks for video action recognition. In Proceedings of the Advances in Neural Information Processing Systems, Barcelona, Spain, 5–10 December 2016; pp. 3468–3476.
8. Wang, L.; Qiao, Y.; Tang, X. Action recognition with trajectory-pooled deep-convolutional descriptors. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 4305–4314.
9. Feichtenhofer, C.; Pinz, A.; Zisserman, A. Convolutional two-stream network fusion for video action recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 1933–1941.
10. Ji, J.; Buch, S.; Soto, A.; Niebles, J.C. End-to-end joint semantic segmentation of actors and actions in video. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 702–717.
11. Wang, Y.; Long, M.; Wang, J.; Yu, P.S. Spatiotemporal pyramid network for video action recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1529–1538.
12. Ji, S.; Xu, W.; Yang, M.; Yu, K. 3D convolutional neural networks for human action recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *35*, 221–231. [CrossRef] [PubMed]
13. Liu, J.; Shahroudy, A.; Wang, G.; Duan, L.Y.; Kot, A.C. Skeleton-based online action prediction using scale selection network. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *42*, 1453–1467. [CrossRef] [PubMed]
14. Diba, A.; Fayyaz, M.; Sharma, V.; Karami, A.H.; Arzani, M.M.; Yousefzadeh, R.; Gool, L.V. Temporal 3d convnets: New architecture and transfer learning for video classification. *arXiv* **2017**, arXiv:1711.08200.
15. Qiu, Z.; Yao, T.; Mei, T. Learning spatio-temporal representation with pseudo-3d residual networks. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 5533–5541.
16. Wang, K.; Wang, X.; Lin, L.; Wang, M.; Zuo, W. 3d human activity recognition with reconfigurable convolutional neural networks. In Proceedings of the 22nd ACM International Conference on Multimedia, Orlando, FL, USA, 3 November 2014; pp. 97–106.
17. Ng, J.Y.H.; Hausknecht, M.; Vijayanarasimhan, S.; Vinyals, O.; Monga, R.; Toderici, G. Beyond short snippets: Deep networks for video classification. In Proceedings of the IEEE Vonference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 4694–4702.
18. Kar, A.; Rai, N.; Sikka, K.; Sharma, G. Adascan: Adaptive scan pooling in deep convolutional neural networks for human action recognition in videos. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–27 July 2017; pp. 3376–3385.
19. Diba, A.; Sharma, V.; Gool, L.V. Deep temporal linear encoding networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2329–2338.
20. Liu, J.; Shahroudy, A.; Perez, M.L.; Wang, G.; Duan, L.Y.; Chichung, A.K. Ntu rgb+ d 120: A large-scale benchmark for 3d human activity understanding. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *42*, 2684–2701. [CrossRef] [PubMed]
21. Veeriah, V.; Zhuang, N.; Qi, G.J. Differential recurrent neural networks for action recognition. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 4041–4049.
22. Wu, Z.; Wang, X.; Jiang, Y.G.; Ye, H.; Xue, X. Modeling spatial-temporal clues in a hybrid deep learning framework for video classification. In Proceedings of the 23rd ACM International Conference on Multimedia, Brisbane, Australia, 13 October 2015; pp. 461–470.
23. Goodale, M.A.; Milner, A.D. *Separate Visual Pathways for Perception and Action*; Psychology Press: London, UK, 1992.
24. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
25. Yu, W.; Yang, K.; Bai, Y.; Xiao, T.; Yao, H.; Rui, Y. Visualizing and comparing AlexNet and VGG using deconvolutional layers. In Proceedings of the 33 rd International Conference on Machine Learning, New York, NY, USA, 19–24 June 2016.

26. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
27. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
28. He, K.; Zhang, X.; Ren, S.; Sun, J. Identity mappings in deep residual networks. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2016; pp. 630–645.
29. Srivastava, R.K.; Greff, K.; Schmidhuber, J. Highway networks. *arXiv* **2015**, arXiv:1505.00387.
30. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst*. **2017**, *60*, 84–90.
31. Ioffe, S.; Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv* **2015**, arXiv:1502.03167.
32. Chatfield, K.; Simonyan, K.; Vedaldi, A.; Zisserman, A. Return of the devil in the details: Delving deep into convolutional nets. *arXiv* **2014**, arXiv:1405.3531.
33. Wang, L.; Qiao, Y.; Tang, X. Latent hierarchical model of temporal structure for complex activity classification. *IEEE Trans. Image Process* **2013**, *23*, 810–822. [CrossRef] [PubMed]
34. Vinyals, O.; Toshev, A.; Bengio, S.; Erhan, D. Show and tell: A neural image caption generator. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3156–3164.
35. Soomro, K.; Zamir, A.R.; Shah, M. UCF101: A dataset of 101 human actions classes from videos in the wild. *arXiv* **2012**, arXiv:1212.0402.
36. Kuehne, H.; Jhuang, H.; Garrote, E.; Poggio, T.; Serre, T. HMDB: A large video database for human motion recognition. In Proceedings of the 2011 International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; pp. 2556–2563.
37. Zach, C.; Pock, T.; Bischof, H. A duality based approach for realtime tv-l 1 optical flow. In *Joint Pattern Recognition Symposium*; Springer: Berlin/Heidelberg, Germany, 2007; pp. 214–223.
38. Jia, Y.; Shelhamer, E.; Donahue, J.; Karayev, S.; Long, J.; Girshick, R.; Guadarrama, S.; Darrell, T. Caffe: Convolutional architecture for fast feature embedding. In Proceedings of the 22nd ACM International Conference on Multimedia, Orlando, FL, USA, 3 November 2014; pp. 675–678.

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).