# Spatiotemporal Multi-Task Network for Human Activity Understanding

Yao Liu
Alibaba Group
xuanyao0111@gmail.com

Jianqiang Huang
Alibaba Group
jianqiang.hjq@alibaba-inc.com

Chang Zhou
Alibaba Group
zhouchang.zc@alibaba-inc.com

Deng Cai
Zhejiang University
dengcai@gmail.com

Xian-Sheng Hua
Alibaba Group
huaxiansheng@gmail.com

## ABSTRACT

Recently, remarkable progress has been achieved in human action recognition and detection by using deep learning techniques. However, for action detection in real-world untrimmed videos, the accuracies of most existing approaches are still far from satisfactory, due to the difficulties in temporal action localization. On the other hand, the spatiotempoal features are not well utilized in recent work for video analysis. To tackle these problems, we propose a spatiotemporal, multi-task, 3D deep convolutional neural network to detect (including temporally localize and recognition) actions in untrimmed videos. First, we introduce a fusion framework which aims to extract video-level spatiotemporal features in the training phase. And we demonstrate the effectiveness of video-level features by evaluating our model on human action recognition task. Then, under the fusion framework, we propose a spatiotemporal multi-task network, which has two sibling output layers for action classification and temporal localization, respectively. To obtain precise temporal locations, we present a novel temporal regression method to revise the proposal window which contains an action. Meanwhile, in order to better utilize the rich motion information in videos, we introduce a novel video representation, interlaced images, as an additional network input stream. As a result, our model outperforms state-of-the-art methods for both action recognition and detection on standard benchmarks.

## CCS CONCEPTS

• **Computing methodologies → Activity recognition and understanding**;

## KEYWORDS

Spatiotemporal Features, Multi-Task, Action Recognition, Action Detection


*ThematicWorkshops'17, October 23–27, 2017, Mountain View, CA, USA*

ACM ISBN 978-1-4503-5416-5/17/10…$15.00
https://doi.org/https://doi.org/10.1145/3126686.3126705

## 1 INTRODUCTION

Recently, action recognition and localization in videos receive extensive research interests because of its wide applications on real-world video analyses. Early methods [21, 42, 43], utilized engineering features or spatiotemporal local descriptors as video representations, based on which the actions are detected and classified. In recent years, deep convolutional neural networks have shown the remarkable progress in many computer vision related areas, such as image classification [14, 20, 34, 38, 39], objection detection [11, 13, 31]. Deep learning based action classification approaches [7, 33, 46, 47] also have been proposed but mainly for trimmed short videos (one video only contains one action). However, most videos in real-world application are long, untrimmed, and contain multiple action instances, which means action detection in these videos actually involves another challenge task: temporal localization of the actions. In this paper, we propose a novel spatiotemporal multi-task network for action detection from real-world untrimmed videos, based on a 3D deep convolutional neural network architechture. Additionally, we also explore a better representation of videos to capture motion information.

Unlike images, videos are 3D signals in nature and have motion information. Therefore how to extract effective features that reveal both spatial and motion characteristics from videos is a challenge topic. Recent works [33, 47] tackle this issue by sampling several frames and extracting CNN features as the representation of the videos. Obviously, this representation mainly contains appearance information thus not sufficient for action detection. [16, 40, 41] extract spatiotemporal features of fixed-size volumes. However it is unreasonable to apply the video-level labels to all sampled volumes. What's far more important is that the spatiotempoal features are not well utilized in these work. Inspired by the idea of long-range temporal structure modeling [8, 10, 24, 45] and descriptor aggregation structure [57], we propose a fusion framework to extract video-level spatiotemporal features.

For temporal localization, we argue that the spatiotemporal features are effective for modeling actions and locating their temporal boundaries. Most state-of-the-art approaches [18, 44] rely on handcraft features, such as improved Dense Trajectory (iDT) with Fisher Vector [25, 43], but the performances are still far from satisfactory. Recent works [23, 28, 52] extract features by 2D deep convolutional neural networks. However, only appearance information is considered in these methods. To explicitly take temporal localization into consideration, we present a multi-task 3D convolutional network based the proposed fusion framework. Specifically, this network

has two sibling output layers, one outputs a discrete probability distribution and another outputs temporal window regression offsets.

In addition, we also study the impact of different representations of videos. Recently, several papers [33, 49] employ multi-stream structure for action recognition, which aims to incorporate the appearance and motion information simultaneously. In work [47], the authors proposed using RGB difference images and warped optical flow fields as input modality. However, the computation of optical flow is very expensive. And the movement information is not well expressed in these approaches. To tackle this problem, we introduce the concept of interlaced images, which extended from a traditional technique for "painting" videos on an electronic display screen. This method is simple, but effective in capturing motion perception from videos. In our work, the interlaced images will be used as an additional input stream to capture motion information, which further improves the performance of our model.

In summary, the main contributions of this paper are as follows. First, we present a fusion framework to extract video-level spatiotemporal features in the training phase. Our experiments show the effectiveness of video-level spatiotemporal features and this method achieves significantly superior performances compared with most recent methods on action recognition task. Second, based on the proposed fusion framework, we design a multi-task 3D convolutional neural network for action detection in an end-to-end fashion. To the best of our knowledge, this work is the first end-to-end framework for action detection based on 3D convolutional neural networks. Third, with the help of our study on representation of videos, we propose to use interlaced images as inputs for capturing more motion information. In our experiments, the model based on interlaced images outperforms traditional optical flow fields based approaches.

## 2 RELATED WORK

In this section, we introduce related work along four directions: action recognition, temporal action localization, multi-task learning and cross-modality representation.

**Action recognition.** In early works [21, 42], hand-craft features have proved effective for video classification and the method [3, 43] based on the improved dense trajectories (iDTs) achieve competitive performance on standard benchmarks. In recent years, deep learning techiques, especially the convolutional neural networks [14, 20, 34], have show their powerful ability in many computer vision areas. The works [14, 39] have achieved human-level performance in the challenging ImageNet [4] classification task. To exploit deep neural networks for video analysis, deep neural network models [37, 46, 47, 49, 59] have been proposed and achieve better performance than traditional methods. A few attempts [48, 50, 55] discover new approaches for pooling frame-level features for better performance, and [7, 37] explore novel fusion structure on training stage for better model learning. Recent methods [6, 51, 53] apply RNN to understand a video by feeding a sequence of frame-level features, and achieve competitive results for both action recognition and video caption task. However, these approaches are not possible to learn from entire videos due to their limited temporal coverage. Our method tackle this issue by its fusion framework.

To learn spatiotemporal features of videos, an early work [16] extended deep convolutional neural network to three-dimensional, Tran [40] trained deep 3D Convolutional Networks (C3D) on a large-scale video dataset [19] and achieved state-of-the-art performance. Varol [41] extend the 3D convolutional networks to long-term temporal convolution structures and achieve significant improvement over original models. Other works [7, 22, 37, 59] propose their models with spatiotemporal features for better performance. These methods directly operated on volumes clipped from videos. However, it is problematic to apply video-level labels to all sampled volumes. We follow this line of work with the addition of fusion mechanism in the training phase to improve original models.

**Temporal localization.** Recent work [44] treat this topic as a classification problem and apply a temporal sliding window approach, where each window is regarded as an action candidate subject to classification. To specifically tackle the temporal precision of action detection, Gaidon [9, 10] modeled the structure of action sequence with atomic action units (actoms). Xu [50] introduce a latent concept descriptor of convolutional feature map, and achieve great results on action detection with VLAD encoding. In [26] the approximately normalized Fisher Vectors were proposed to reduce the high dimensionality of FV. Another line of work focus on localizing action in space and times at the same time. Jain [15] and Soomro [35] merge super-voxels in their works for action localization. Gkioxari [12] and Raptis [30] automatically localize a spatio-temporal tube in a video. Ramanathan [52] use RNN with the addition of an attention mechanism to attend to the action participants for action detection within longer untrimmed videos. To exploit deep learning for temporal localization, Shou [32] proposed Segment-CNN which is a deep network framework with multi-stage processes for temporal action localization. The most of these works focus on learning from data when the temporal boundaries have been annotated for action instances in untrimmed videos. This paper introduced here aims to solve the problem of precise action detection in untrimmed long videos in the wild.

**Multiple task learning.** Recent works have demonstrated that deep learning approaches with multi-task structure can boost up the performance of each task. For object detection task, Fast R-CNN [11] proposed a multi-task framework to take object recognition and localization into consideration simultaneously, and later approaches [27, 31] have followed this idea. For face detection, [27, 58] detection and alignment tasks are jointly modeled in their proposed deep multi-task framework, which exploit the inherent correlation between these two tasks to improve overall performance. In [29], more than two tasks have been integrated into a deep framework, such as face detection, landmark localization, pose estimation, gender recognition and etc. For video analysis, Diba [5] proposed a multi-task framework which joint action recognition and motion estimation, and achieved impressive performance for action recognition. As an analogy of object and recognition in still images, in which instance recognition and bounding-box regression are unified in one framework by using CNN features, we propose an end-to-end deep framework, which consider action classification and temporal localization simultaneously by using spatiotemporal features.

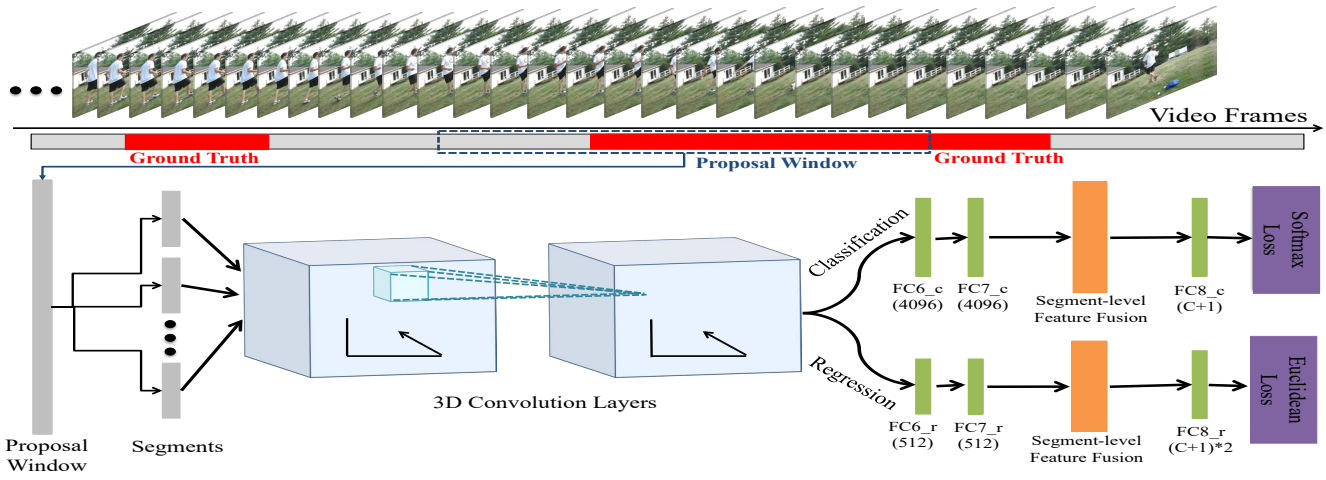


**Figure 1: Overview of our model. The input to the model is a untrimmed video, and the output is a set of action predictions. For the entire video, we first generate dozens to hundreds of proposals of various durations via temporal sliding window. Each proposal window will be fed into the spatiotemporal multi-task network by dividving into $K$ segments. After fusion layer in each branch, the first sibling layer outputs a discrete probability distribution over $C + 1$ categories, the second sibling layer outputs temporal window regression offsets. During prediction, some post-processing methods (e.g. temporal NMS etc.) are used.**

**Cross-modality representation.** To incorporate the appearance and motion information, [33] proposed the two-stream architecture which consists of optical flow and RGB streams. Following this direction, Wang [47] introduced two new type of input modalities, namely stacked RGB difference and stacked warped optical flow field. This work also explored a number of pretty practices to solve the difficulties caused by different input modalities, such as cross-modality pre-training. The work [41] show that the impact of different representation and demonstrate the importance of high-quality optical flow map for learning precise models. [49] extend the two-stream framework to a multi-stream architecture, and introduced audio spectrogram to improve the performance for video classification. However, the most computationally expensive step in these approaches comes from the calculation of optical flow. In [56], optical flow was replaced by motion vector, which can be obtained directly from compressed videos. To exploit more effective representations, the concept of dynamic image has been proposed in [1], which is a compact representation of videos and obtained by directly applying rank pooling on the raw image pixels. To capture rich motion information from videos, we introduce the concept of interlaced images to capture motion information.

## 3 OUR METHOD

As observed in the Section 1, the action detection task contains two sub-tasks: action instance recognition and temporal action localization. Temporal localization is a challenge topic, due to the difficulties in temporal feature extraction. For object detection in still images, the couvolutional neural networks provide better features compare with traditional hand-craft features, and achieve impressive performance. Howeve, there is no remarkable improvment when extend the deep networks from images to videos for action recognition and detection. This motivates us to build a framework which can extract effective features for action detection in videos.

In this section, we will detail the proposed approach for action detection in long untrimmed videos. Specifically, we firstly present the proposed fusion framework together with feature aggregation mechanism and sampling strategy to extract video-level spatiotemporal features. And then we will illustrate the architectures of the proposed multi-task 3D convolutional neural network in detail, as shown in Figure 1. Finally, we describe the training process as well as the testing process of the learned model.

### 3.1 Video-level feature extraction

As we discussed in Sec.1, it is problematic to apply video-level labels to all sampled volumes. Our experiments show that a volume can only cover very limited frames (e.g. 16), due to the limited of storage and computational resources. However, these small volumes are more likely to be irrelevant to or less relevant to the action categories at video level, especially for complex sports actions, such as dunk. It would be extremely a loss and hurt to the final performance. To tackle this issue, we propose a fusion framework to extract real video-level features by aggregating segmental spatiotemporal features, as shown in Figure 2.

**Segment-level fusion.** This fusion framework aims to capture the appearance and temporal information of the entire video for video-level predictions. Instead of fusing the inferred results of different volumes, our framework perform aggregating operation directly in the training phase. Specifically, given a video, we divide it into several segments of the same durations, and then sample a sequence of short snippets from these segments. Note that a snippet contains a certain number of frames sampled from the

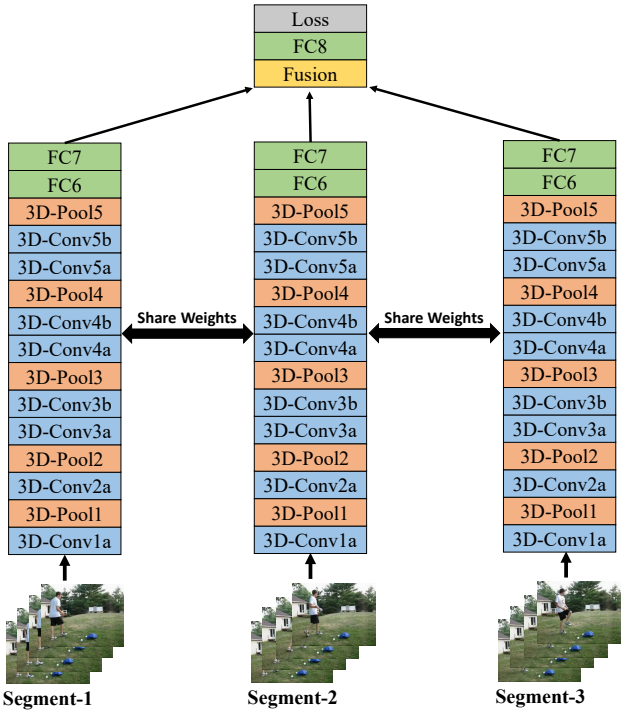


**Figure 2: Overview of our fusion architecture. Given a video, we divide it into $K$ segments, and we illustrate an example of $K = 3$ in this figure. Then these segments will be fed into the 3D convolutional neural networks respectively, and the weights of networks before fusion layer are shared. By applying concatenation fusion method, we can extract video-level spatiotemporal features. Finally, the obtained features can be used for classification and detection task.**

corresponding segment. In our framework, each snippet in this sequence will be fed into the 3D convolutional neural network and produce the corresponding spatiotemporal features. These features will be aggregated in the learning process, and derived as the video-level spatiotemporal features for optimizing the loss value of video-level rather than the segment or volume levels.

To be clearer, our intention here is to fuse the spatiotemporal features of all segments (in particular fully connected layer), such as concatenating these features. The idea is inspired by the recent work [47] of long-range temporal modeling, which produces the final video-level prediction by fuse the preliminary prediction of snippets. However, according to our experiments, performance is improved when perform the fusion operation on the feature layer compared with the prediction layer. The final architecture is shown in Figure 2.

Formally, given a video $V_v$, where $v = 1, ..., N$ and $N$ is the number of video samples, we firstly divide the video into $K$ segments $\{S_v^1, S_v^2, ..., S_v^K\}$ in equal duration. Then we perform the sampling operation on each segment $S_v^k$ to produce a sequence snippets $\{T_v^1, T_v^2, ..., T_v^K\}$. This operation is denoted as:

$$T_v^k = R(S_v^k), k = 1, ..., K, \tag{1}$$

where $K$ is the number of segments and $R(S_v^k)$ represent the sampling function. This sequence of snippets will be fed into the 3D convolution neural network as input stream, and a set of spatiotemporal features $\{f_v^1, f_v^2, ..., f_v^K\}$ will be obtained by performing convolutional network parameters $W$ on each snippet. Note that $f_v^k \in \mathbb{R}^D$, where $D$ is the length of feature vector (e.g. 4096). These feature vectors will be fused to produce an output feature vector $F_v$, where $F_v \in \mathbb{R}^{D'}$, and $D'$ is the length of the final feature vector to learn our model. Specifically, the fusion function is $F_v = H(f_v^1, f_v^2, .., f_v^K)$.

In summary, the process of video-level feature extraction can be formed as:

$$F_v = H(G(R(S_v^1); W), ..., G(R(S_v^K); W)), \tag{2}$$

where $G(R(S_v^k); W)$ is the function representing the operations of the 3D deep network before fusion layer. In this work, the fusion operation is implemented by inserting a fusion layer before the final fully connected layer.

So far, there are three open questions remained for our fusion framework about aggregation function $H$, sampling function $R$ and the number of segments $K$. In this paper we evaluated several different forms of the aggregation function $H$, including maximum, stochastic, evenly averaging and concatenation. Our experiments show that the model with concatenation fusion achieves the best performance, thus the concatenation fusion is used in our final framework. We also assessed three different forms of the sampling function $R$, including random sampling $n$ frames, random a volume and isometric sampling, as shown in Figure 4. Among these sampling strategies, the performance for random sampling volumes and sampling isometric achieve similar level, and sampling isometric has better performance than the other two. The last question is how to select an appropriate value of $K$. The main purpose of dividing a video into $K$ segments is to extract richer features from the videos constrained by the limited of memory and computation resources. In the extreme case, $K = 1$, the model will degenerate to the simply original 3D convolutional neural network (C3D[40]), and a larger $K$ will lead to more parameters. Finally, in our framework $K$ is set to 3.

**Interlaced images.** As we discussed in Sec.1, motion information is pretty useful for video analysis, and optical flow is a widely used method to capture motion information in videos. However, the range of optical flow fields is different with RGB images. Most approaches will discretize optical flow fields into the interval of 0 to 255 by a linear transformation. Though the results of these methods are pretty resonably by using transformed optical flow fields as inputs, but the cost of generating optical flow fields is very large. We introduce the interlaced images to capture motion information from RGB images more efficiently. Traditionally, one interlaced image is generated from two frames in raw videos by filling the value of odd lines with corresponding line in the first raw frames and operating the same rules on even lines. We extend this method with a parameter $L$ to assemble $L$ raw frames for one interlaced image, the method is shown in Figure 3. In our experiments, we generate one interlaced image from 3 raw frames.

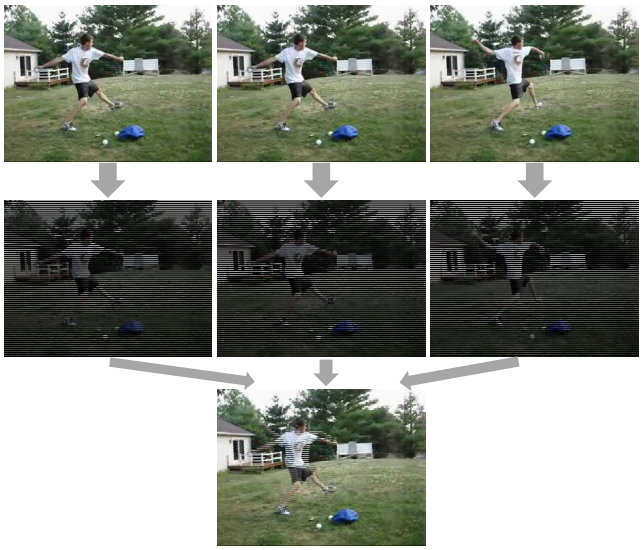

**Figure 3: Examples of interlaced images. We show an example of $M = 3$ to generate a interlaced image. Top row: 3 raw frames in a video. Center row: Intermediate result images. Bottom row: The resulting interlaced image.**

## 3.2 Multi-task framework

As aforementioned, multi-task framework is adopted for action detection in end-to-end fashion from untrimmed long videos, as shown in Figure 1. In order to achieve optimal performance, a few practical concerns should be taken care of, such as the strategy to generate temporal proposal windows. To this end, we adopt a series of good practices in training multi-task 3D convolutional neural networks.

**Network architecture.** For spatiotemporal feature extraction, [16, 40] proposed 3D convolutional neural networks to perform convolution/pooling in spatial and temporal dimensions simultaneously. The network, the network in [40] (C3D) is trained on a large-scale video dataset, Sports-1M [19], and achieve the state-of-the-art performance for action recognition. Therefore, we follow the network architecture in [40], in which we keep all 3D convolution layers, all 3D pooling layers and two fully connected layers (fc6 and fc7). After the second fully connected layer (fc7), we insert a fusion layer followed by a dropout layer, which is described in Sec.3.1, to boost up the performance of original C3D models and extract video-level spatiotemporal features. The number of filters for the last fully connected layer (fc8) is correspond with the number of categories $C$.

For multi-task learning, we add a new branch after the last convolution layer (conv5b) based on the proposed fusion framework. Specifically, five layers are added: global pooling layer - fully connected layer - fully connected layer - fusion layer - fully connected layer. Follow the parameters in [40], all 3D convolution layers have kernel size 3 and stride 1 in all three dimensions. All 3D pooling layers use max pooling and have kernel size of 2ÃŮ2 in spatial with stride 2, while vary in temporal. The input for this network is a

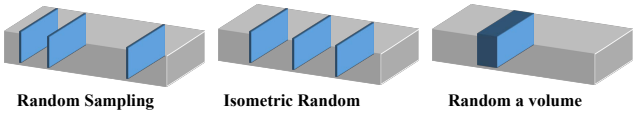


**Figure 4: Three examples of sampling strategy in segments. The left example shows that random sampling $L$ frames. The center example shows that sampling $L$ frames with the isometric rule. The right example shows that random a volume of length $L$. Note that a sampled volume is a spatial-temporal video clip from consecutive frames.**

sequence snippets of a video of dimension $S * 16 * 128 * 171$, where $S$ is the number of segments.

**Loss function.** Our proposed multi-task 3D convolutional neural network has two sibling output layers. The first outputs a discrete probability distribution (per proposal window), $p = (p_0, ..., p_C)$, over $C + 1$ categories. As usual, p is computed by a softmax over the $C + 1$ outputs of the final fully connected layer. The second sibling layer outputs temporal window regression offsets, $t^c = (t^c_{start}, t^c_{center})$, each of the $C$ action classes, indexed by $c$. Especially, $t^k$ specifies a time-point shift relative to a temporal proposal window. Each proposal window is labeled with a ground-truth class $u$ and a ground-truth temporal window regression target $v$.

We define the multi-task loss $L$ to combine loss for classification and temporal window regression:

$$L(p, u, t^u, v) = L_{cls}(p, u) + \lambda[u \geq 1]L_{reg}(t^u, v), \tag{3}$$

in which the first loss, $L_{cls}$ is conventional log loss for true class $u$, which is effective for training deep networks for classification and formed as:

$$L_{cls}(p, u) = -logp_u \tag{4}$$

The second loss, $L_{reg}$, is defined over a tuple of true temporal window regression targets $v = (v_{start}, v_{center})$, and a predicted tuple $t^u = (t^u_{start}, t^u_{center})$, for class $u$. Specially, we employ the Euclidean loss for each proposal temporal window and the catch-all background class is labeled $u = 0$ by convention. For background proposal windows there is no notion of a ground-truth temporal regression target and hence $L_{reg}$ can be ignored, the loss formed as:

$$L_{reg}(t^u, v) = ||t^u - v||^2_2. \tag{5}$$

In Eq.3, the Iverson bracket indicator function $[u \geq 1]$ evaluates to 1 when $u \geq 1$ and 0 otherwise, and the hyper-parameter $\lambda$ balances the contribution from each loss. We find that $\lambda = 10$ works well in practice, through empirical validation. For temporal regression, we normalize the ground-truth regression targets $v$ to have zero mean and unit variance.

## 3.3 Training

**Initialization.** Most deep frameworks for image tasks initialize network weights from a pre-trained model, such as pre-trained ImageNet [4] networks. Follow this idea, we initialize our network weights with a pre-trained C3D model [40] on Sports-1M [19] and perform the transformation. Specifically, the last fully connected layer and softmax (which were trained for 200-way Sports-1M

classification) of the original network are replaced with two sibling layers described in Section 3.2. In this work, we use the stochastic gradient descent (SGD) to learn the network weights.

**Mini-batch sampling.** Unlike classification task, example sampling is important for detection task both in still images and videos. During fine-tuning, each SGD mini-batch is constructed from $N$ videos, chosen uniformly at random (as is common practice). With the limited of memory resources (e.g. 12GB for NVIDIA M40), we construct a mini-batch of size 12 by random 4 temporal proposal windows from each chosen video. And we keep one proposal window from these 4 selected proposal windows is positive example, while others are negative examples. The positive examples are proposal windows that have temporal intersection over union (IoU) overlap with a groundtruth temporal window of at least 0.5. Note that the temporal windows labeled with a foreground action class are also positive examples. The remaining proposal windows that have a maximum temporal IoU with groundtruth in the interval $[0, 0.5)$ are defined as negative samples and are labeled as $u = 0$.

To generate proposal windows, we slide temporal windows with varied durations. In our experiments, the length of proposal windows vary from 16 frames to 1600 frames. During training, examples are horizontally flipped with probability 0.5, and random cropping mechanism will be applied. Note that the examples are snippets sampled from videos, these operations have to be careful to maintain consistency of the video. No other data augmentation is used in this work.

**SGD hyper-parameters.** According to previous work on image detection [11], the fully connected layers used for softmax classification and temporal window regression are initialized from zero-mean Gaussian distributions with standard deviations 0.01 and 0.001, respectively. Biases are initialized to 0. For all experiments, all layers use a per-layer learning rate of 1 for weights and 2 for biases and a global learning rate of 0.0001. A momentum of 0.9 and parameter decay of 0.0005 (on weights and biases) are used. For action recognition task, the learning rate decreases to its $\frac{1}{10}$ every 12000 iterations. The maximum iteration is set as 30000. For action detection task, the learning rate decreases to its $\frac{1}{10}$ every 20000 iterations. The maximum iteration is set as 45000. Specifically, our method is implemented based on Caffe [17].

## 3.4 Prediction

During prediction, we process the entire video directly to generate temporal proposal windows by using the strategy described in Section 3.3. Then we feed these proposal windows into the multi-task 3D convolutional neural network to obtain action category predictions and temporal window regression offsets. The windows predicted as the background will be removed. Finally, we apply temporal Non-maximum Suppression (NMS) on remaining temporal windows to remove redundant detections.

# 4 EXPERIMENTS

In this section, we firstly introduce the evaluation datasets. Then we evaluate the effectiveness of the proposed fusion framework for learning video-level features and the novel representation of videos for capturing motion information. Finally, we also compare the performance of our method with the-state-of-the-art methods. In addition, we also discussed the results of each experiments together with discussions about the performance differences.

**Table 1: Performance comparision of different fusion function in UCF101.**

| Fusion Function | Accuracy | #layer |
|---|---|---|
| Maximum | 80.45% | 18 |
| Average | 82.89% | 18 |
| Stochastic | 82.20% | 18 |
| Concatenation | **86.38**% | 17 |

**Table 2: Performance comparision of different sampling strategies (Sec.3.1) in UCF101.**

| Sampling Strategy | Random frames | Random a volume | Isometric sampling |
|---|---|---|---|
| Accuracy | 85.47% | 86.38% | **86.44**% |

## 4.1 Datasets and setup

We conduct our experiments on two large video datasets, namely UCF101 [36] and ActivityNet [2]. UCF101 [36], contains 101 action classes and 13,320 trimmed videos. This dataset was built for action recognition, we follow the provided evaluation protocol and adopt the three training/testing splits for evaluation. The second dataset is ActivityNet [2], which contains 68.8 hours of temporal annotations in 849 hours of untrimmed, unconstrained video. There are 1.41 action instances per video and 193 instances per class. ActivityNet validation is applied to evaluate the action detection accuracy.

## 4.2 Evaluation of the fusion framework

In this subsection we focus on the study of the proposed fusion framework in UCF101 action recognition datasets [36]. All experiments in this subsection are conducted on the split 1 of UCF101 dataset [36]. We first study the impact of the number of segments, different fusion functions and different sampling strategies for our framework. Next, we study the effect of using interlaced images as inputs and compare it with optical flow fields. We also compare our final model with the state-of-the-art methods.

**Fusion function.** Here we evaluate four candidate functions: (1) maximum, (2) average, (3) stochastic, (4) concatenation for the form of fusion function. In this experiment, the number of segments is 3, and the sampling strategy is random a volume in each segment. The experimental results are summarized in Table 1, and concatenation fusion function achieves the best performance. Compare these functions, we argue that the concatenation function maintains temporal characteristics among segments and enhances the temporal features. Following the results of this experiments, we choose concatenation as the default aggregation function.

**Sampling strategy.** As discussed in Section 3.1, the sampling is useful for analysis. Here, we evaluate three candidate strategies: (1)

**Table 3: Comparison with state-of-the-art methods (RBG inputs only) in UCF101.**

| Methods | Acc. (RGB only) |
|---|---|
| Two Stream [33] | 72.70% |
| C3D (1 net) [40] | 82.30% |
| C3D (3 nets) [40] | 85.20% |
| TDD+FV [46] | 82.80% |
| LTC [41] | 81.50% |
| KVMF [59] | 85.30% |
| Two Stream (VGG16) [47] | 84.50% |
| Two Stream (BN-Inception v3) [47] | 84.50% |
| TSN [47] | 85.70% |
| Wu, multi-stream [49] | 84.00% |
| Ours | **86.77**% |

random sampling frames, (2) random a volume, (3) isometric sampling frames. In this experiment, the fusion function is concatenation, and the number of segments is 3. The results are summarized in Table 2. We see that the model with isometric sampling in segments achieves the best performance. Compare with other strategies, we believe that isometric sampling can cover more frames and capture richer information in videos. So we choose isometric sampling as the default sampling strategy.

**RGB images based model.** We compare our framework with recent state-of-the art methods, especially deep learning based methods, in UCF101 dataset (split 1)[36]. Specifically, our method only use RGB images and processed RGB images (interlaced images) as inputs. So the results reported in this paper are RGB input only. The results are summarized in Table 3.

**Analysis.** Compare with the original 3D convolutional networks based model (C3D) [40], our method outperforms by 4.14% and 1.24% for 1-net model and 3-net model respectively. Compare with two stream architecture based methods, our model also achieve better performance, where Two Stream (VGG16) and Two Stream (BN-Inception v3) are implemented in [47] based on original two-stream method [33]. Wu [49] extend the two stream architecture to multi-stream architecture. We also compare our result with other recent deep learning based methods, such as trajectory-pooled deep-convolutional descriptors (TDD) [46], long term convolution networks (LTC) [41], and key volume mining framework (KVMF) [59] and Temporal Segment Networks (TSN) [47]. Our best result outperforms other methods by 1.07% on the UCF101 dataset [36] (split 1) by using a small 3D convolutional neural networks. The superior performance of our methods demonstrates the effectiveness of proposed spatiotemporal fusion framework and justifies the importance of video-level spatiotemporal features.

## 4.3 Action detection

In this subsection, we evaluate our multi-task framework in ActivityNet [2] for action detection. Specifically, evaluation is applied on the ActivityNet validation set, because the online evaluation system has been closed. We compare our method with existing work [2], which provides a baseline result for comparsion. The method of this work [2] is based on combination of dense trajectories, SIFT, and ImageNet-pretrained CNN features. This work investigate the performance of the different feature types, individually and collectively. Specifically, the value of alpha ($\alpha$) is the overlap threshold. The results are summarized in Table 4. And the performance of proposed method outperforms these methods.

**Table 4: Summary of action detection results. We report the mAP score for all activity classes.**

| Method | $\alpha = 0.1$ | $\alpha = 0.2$ | $\alpha = 0.3$ | $\alpha = 0.4$ | $\alpha = 0.5$ |
|---|---|---|---|---|---|
| [2] with MF | 11.7% | 11.4% | 10.6% | 9.7% | 8.9% |
| [2] with DF | 7.2% | 6.8% | 4.9% | 4.1% | 3.7% |
| [2] with SF | 4.2% | 3.9% | 3.1% | 2.1% | 1.9% |
| [2] Fusion | 12.5% | 11.9% | 11.1% | 10.4% | 9.7% |
| Ours | 28.89% | 24.85% | 20.17% | 17.88% | 14.21% |

**Table 5: Comparison with the optical flow based methods in UCF101. Optical Flow Fields* represent the result by using optical flow fields as inputs in our framework.**

| Input Modality | Accuracy |
|---|---|
| Optical Flow Fields* | 75.80% |
| $LTC_{Flow}$(16f) [41] | 78.7% |
| $LTC_{Flow}$(100f) [41] | 82.60% |
| $LTC_{Flow}$(60f+100f) [41] | 83.80% |
| Optical Flow [Clarifai] [33] | 81.0% |
| Optical Flow [GoogLeNet] [47] | 83.9% |
| Optical Flow [VGGNet-16] [47] | 85.7% |
| Optical Flow [BN-Inception] [47] | 87.2% |
| Interlaced Images | 84.32% |
| Interlaced Images+static-rgb | **87.36**% |

This experiments demonstrates the effectiveness of our video-level spatiotemporal features based multi-task deep network for action detection in untrimmed long videos. For comparsion, we report the results of baseline method based different features, such as hand-craft features, CNN features and etc. The results show that the features have a great impact on the performance. And compared to other features, spatiotemporal features are more effective for video analysis. In addition, the multi-task network architecture is also valuable for video analysis.

## 4.4 Evaluation of interlaced images

As described in Section 3.1, we propose a novel representation of videos to capture motion information from RGB images, namely interlaced images. To evaluate it, we feed the interlaced images into our framework with the default settings. The settings have been discussed above. And we setup two experiments for comparison.

In the first experiment, we compare proposed method with optical flow fields based methods. And the results are summarized in Table 5. Firstly, we compare our method with recent work [41]

**Table 6: Performance comparision of different input modalities in UCF101. For comparison, we also report the accuracy of fusing interlaced images based model and rgb images based model.**

| Input Modality | Accuracy |
|---|---|
| RGB Image [47] | 84.5% |
| RGB Difference [47] | 83.8% |
| RGB Image + RGB Difference [47] | 87.3% |
| MDI-end-to-end [1] | 70.90% |
| MDI-end-to-end+static-rgb [1] | 76.90% |
| EMV-CNN with ST+TI [56] | 79.3% |
| static-rgb | 86.77% |
| Interlaced Images | 84.32% |
| Interlaced Images+static-rgb | **87.36**% |

which feeds optical flow fields into long-term 3D convolutional neural networks for action recognition. For fair comparison, we also investigate the performance of our 3D Convolution based framework, which is named Optical Flow Fields* in the table. Obviously, the performance of our method outperforms all the 3D Convolution based models. Except 3D Convolution based mehthods, we also compare the results with normal convolutional network based models, for example two-stream ConvNets. The performance have been reported in work [33, 47]. Our result also outperforms these CNN based methods, even the very deep model, such as BN-Inception. Specifically, we generate optical flow fields following method [54] by using Opencv.

In the second experiment, we compare proposed method with other input modalities based methods. The accuracies of recent works for video representation [1, 56] are also reported in Table 6. The authors designe new video representation as input modalities, and the performance outperforms most optical flow fields based methods. For comparison, we also report the result of the processed RBG images based method in recent work [1, 56]. And for fair comparsion, we aslo report result of one model and multiple models. Specifically, we report the result of two models, which fuses the result of interlaced images based model and RGB images baed model. The experiment settings have discussed above. Finally, the performance of proposed method outperforms these works.

## 4.5 State-of-the-art comparisons

Finally, we compare with the state-of-the-art methods in UCF101 in Table 7. We make comparsion with recent work, such as two-stream CNNs, multi-stream CNNs, 3D Convolution based models, handcraft based methods and etc. For comparsion, we also report the result of multi-stream model base on proposed methods. The multi-stream includes RGB images, optical flow and interlaced images as input modality. And we use two kinds of model architectures, proposed network architecture and normal convolutional neural network. The final report of the performance is the result of multi-stream of fusion. Obviously, the proposed method outperforms almost all the approaches and approaches the best method.

**Table 7: Comparison with state-of-the-art methods in UCF101.**

| Methods | Acc. (RGB only) |
|---|---|
| Two Stream [33] | 88.0% |
| C3D (1 net) [40] | 82.30% |
| C3D (3 nets) [40] | 85.20% |
| TDD+FV [46] | 90.30% |
| LTC [41] | 91.70% |
| KVMF [59] | 93.10% |
| Two Stream (VGG16) [47] | 90.90% |
| Two Stream (BN-Inception v3) [47] | 92.0% |
| Wu, multi-stream [49] | 92.60% |
| Two Stream Fusion [7] | 92.5% |
| EMV+RGB-CNN [56] | 86.40% |
| MDI-end-to-end + static-rgb+trj [1] | 89.10% |
| Ours + CNN | **93.36**% |

## 5 CONCLUSION

In this paper, we have presented a spatiotemporal multi-task network for action detection from untrimmed real-world videos. Specifically, a fusion framework based on 3D convolutional neural networks is proposed to extract video-level spatiotemporal features. Meanwhile, a novel video representation, namely interlaced images, is introduced to capture motion information from RGB images. As demonstrated in UCF101[36] and ActivityNet datasets [2], our model outperforms state-of-the-art methods and the effectiveness of video-level spatiotemporal features has been proven. A direction for future work is to explore the deeper 3D convolution neural network architectures and extend our framework to learn more powerful model.

## REFERENCES


[1] Hakan Bilen, Basura Fernando, Efstratios Gavves, Andrea Vedaldi, and Stephen Gould. 2016. Dynamic image networks for action recognition. In *IEEE International Conference on Computer Vision and Pattern Recognition CVPR*.

[2] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. 2015. Activitynet: A large-scale video benchmark for human activity understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 961–970.

[3] César Roberto de Souza, Adrien Gaidon, Eleonora Vig, and Antonio Manuel López. 2016. Sympathy for the details: Dense trajectories and hybrid classification architectures for action recognition. In *European Conference on Computer Vision*. Springer, 697–716.

[4] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 248–255.

[5] Ali Diba, Ali Mohammad Pazandeh, and Luc Van Gool. 2016. Efficient Two-Stream Motion and Appearance 3D CNNs for Video Classification. *arXiv preprint arXiv:1608.08851* (2016).

[6] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. 2015. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2625–2634.

[7] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. 2016. Convolutional Two-Stream Network Fusion for Video Action Recognition. *arXiv preprint arXiv:1604.06573* (2016).

[8] Basura Fernando, Efstratios Gavves, Jose M Oramas, Amir Ghodrati, and Tinne Tuytelaars. 2015. Modeling video evolution for action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 5378–5387.

[9] Adrien Gaidon, Zaid Harchaoui, and Cordelia Schmid. 2011. Actom sequence models for efficient action detection. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 3201–3208.

[10] Adrien Gaidon, Zaid Harchaoui, and Cordelia Schmid. 2013. Temporal localization of actions with actoms. *IEEE transactions on pattern analysis and machine intelligence* 35, 11 (2013), 2782–2795.

[11] Ross Girshick. 2015. Fast r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision*. 1440–1448.

[12] Georgia Gkioxari and Jitendra Malik. 2015. Finding action tubes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 759–768.

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2014. Spatial pyramid pooling in deep convolutional networks for visual recognition. In *European Conference on Computer Vision*. Springer, 346–361.

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385* (2015).

[15] Mihir Jain, Jan Van Gemert, Hervé Jégou, Patrick Bouthemy, and Cees GM Snoek. 2014. Action localization with tubelets from motion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 740–747.

[16] Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu. 2013. 3D convolutional neural networks for human action recognition. *IEEE transactions on pattern analysis and machine intelligence* 35, 1 (2013), 221–231.

[17] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. 2014. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the 22nd ACM international conference on Multimedia*. ACM, 675–678.

[18] Svebor Karaman, Lorenzo Seidenari, and Alberto Del Bimbo. 2014. Fast saliency based pooling of Fisher encoded dense trajectories. In *ECCV THUMOS Workshop*, Vol. 1. 6.

[19] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. 2014. Large-scale video classification with convolutional neural networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. 1725–1732.

[20] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*. 1097–1105.

[21] Ivan Laptev. 2005. On space-time interest points. *International Journal of Computer Vision* 64, 2-3 (2005), 107–123.

[22] Qing Li, Zhaofan Qiu, Ting Yao, Tao Mei, Yong Rui, and Jiebo Luo. 2016. Action Recognition by Learning Deep Multi-Granular Spatio-Temporal Video Representation. In *Proceedings of the 2016 ACM on International Conference on Multimedia Retrieval*. ACM, 159–166.

[23] Yanghao Li, Cuiling Lan, Junliang Xing, Wenjun Zeng, Chunfeng Yuan, and Jiaying Liu. 2016. Online Human Action Detection using Joint Classification-Regression Recurrent Neural Networks. *arXiv preprint arXiv:1604.05633* (2016).

[24] Juan Carlos Niebles, Chih-Wei Chen, and Li Fei-Fei. 2010. Modeling temporal structure of decomposable motion segments for activity classification. In *European conference on computer vision*. Springer, 392–405.

[25] Dan Oneata, Jakob Verbeek, and Cordelia Schmid. 2013. Action and event recognition with fisher vectors on a compact feature set. In *Proceedings of the IEEE International Conference on Computer Vision*. 1817–1824.

[26] Dan Oneata, Jakob Verbeek, and Cordelia Schmid. 2014. Efficient action localization with approximately normalized fisher vectors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2545–2552.

[27] Hongwei Qin, Junjie Yan, Xiu Li, and Xiaolin Hu. 2016. Joint Training of Cascaded CNN for Face Detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3456–3465.

[28] Vignesh Ramanathan, Jonathan Huang, Sami Abu-El-Haija, Alexander Gorban, Kevin Murphy, and Li Fei-Fei. 2015. Detecting events and key actors in multi-person videos. *arXiv preprint arXiv:1511.02917* (2015).

[29] Rajeev Ranjan, Vishal M Patel, and Rama Chellappa. 2016. Hyperface: A deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition. *arXiv preprint arXiv:1603.01249* (2016).

[30] Michalis Raptis, Iasonas Kokkinos, and Stefano Soatto. 2012. Discovering discriminative action parts from mid-level video representations. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 1242–1249.

[31] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*. 91–99.

[32] Zheng Shou, Dongang Wang, and Shih-Fu Chang. Temporal Action Localization in Untrimmed Videos via Multi-stage CNNs. (????).

[33] Karen Simonyan and Andrew Zisserman. 2014. Two-stream convolutional networks for action recognition in videos. In *Advances in Neural Information Processing Systems*. 568–576.

[34] Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014).

[35] Khurram Soomro, Haroon Idrees, and Mubarak Shah. 2015. Action localization in videos through context walk. In *Proceedings of the IEEE International Conference on Computer Vision*. 3280–3288.

[36] K. Soomro, A. Roshan Zamir, and M. Shah. 2012. UCF101: A Dataset of 101 Human Actions Classes From Videos in The Wild. In *CRCV-TR-12-01*.

[37] Lin Sun, Kui Jia, Dit-Yan Yeung, and Bertram E Shi. 2015. Human action recognition using factorized spatio-temporal convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision*. 4597–4605.

[38] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. 2015. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1–9.

[39] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. 2015. Rethinking the inception architecture for computer vision. *arXiv preprint arXiv:1512.00567* (2015).

[40] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. 2015. Learning spatiotemporal features with 3d convolutional networks. In *2015 IEEE International Conference on Computer Vision (ICCV)*. IEEE, 4489–4497.

[41] Gül Varol, Ivan Laptev, and Cordelia Schmid. 2016. Long-term Temporal Convolutions for Action Recognition. *arXiv preprint arXiv:1604.04494* (2016).

[42] Heng Wang, Alexander Kläser, Cordelia Schmid, and Cheng-Lin Liu. 2013. Dense trajectories and motion boundary descriptors for action recognition. *International journal of computer vision* 103, 1 (2013), 60–79.

[43] Heng Wang and Cordelia Schmid. 2013. Action recognition with improved trajectories. In *Proceedings of the IEEE International Conference on Computer Vision*. 3551–3558.

[44] Limin Wang, Yu Qiao, and Xiaoou Tang. 2014. Action recognition and detection by combining motion and appearance features. *THUMOS14 Action Recognition Challenge* 1 (2014), 2.

[45] Limin Wang, Yu Qiao, and Xiaoou Tang. 2014. Latent hierarchical model of temporal structure for complex activity classification. *IEEE Transactions on Image Processing* 23, 2 (2014), 810–822.

[46] Limin Wang, Yu Qiao, and Xiaoou Tang. 2015. Action recognition with trajectory-pooled deep-convolutional descriptors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 4305–4314.

[47] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. 2016. Temporal segment networks: towards good practices for deep action recognition. In *European Conference on Computer Vision*. Springer, 20–36.

[48] Peng Wang, Yuanzhouhan Cao, Chunhua Shen, Lingqiao Liu, and Heng Tao Shen. 2015. Temporal pyramid pooling based convolutional neural networks for action recognition. *arXiv preprint arXiv:1503.01224* (2015).

[49] Zuxuan Wu, Yu-Gang Jiang, Xi Wang, Hao Ye, Xiangyang Xue, and Jun Wang. 2015. Fusing Multi-Stream Deep Networks for Video Classification. *arXiv preprint arXiv:1509.06086* (2015).

[50] Zhongwen Xu, Yi Yang, and Alex G Hauptmann. 2015. A discriminative CNN video representation for event detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1798–1807.

[51] Li Yao, Atousa Torabi, Kyunghyun Cho, Nicolas Ballas, Christopher Pal, Hugo Larochelle, and Aaron Courville. 2015. Describing videos by exploiting temporal structure. In *Proceedings of the IEEE International Conference on Computer Vision*. 4507–4515.

[52] Serena Yeung, Olga Russakovsky, Greg Mori, and Li Fei-Fei. 2015. End-to-end Learning of Action Detection from Frame Glimpses in Videos. *arXiv preprint arXiv:1511.06984* (2015).

[53] Joe Yue-Hei Ng, Matthew Hausknecht, Sudheendra Vijayanarasimhan, Oriol Vinyals, Rajat Monga, and George Toderici. 2015. Beyond short snippets: Deep networks for video classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 4694–4702.

[54] Christopher Zach, Thomas Pock, and Horst Bischof. 2007. A duality based approach for realtime TV-L 1 optical flow. In *Joint Pattern Recognition Symposium*. Springer, 214–223.

[55] Shengxin Zha, Florian Luisier, Walter Andrews, Nitish Srivastava, and Ruslan Salakhutdinov. 2015. Exploiting image-trained cnn architectures for unconstrained video classification. *arXiv preprint arXiv:1503.04144* (2015).

[56] Bowen Zhang, Limin Wang, Zhe Wang, Yu Qiao, and Hanli Wang. 2016. Real-time Action Recognition with Enhanced Motion Vector CNNs. *arXiv preprint arXiv:1604.07669* (2016).

[57] Chen-Lin Zhang, Hao Zhang, Xiu-Shen Wei, and Jianxin Wu. 2016. Deep Bimodal Regression for Apparent Personality Analysis?. In *ChaLearn Looking at People Workshop on Apparent Personality Analysis, ECCV Workshop proceedings, p. in press. Springer Science+ Business Media.*

[58] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. 2016. Joint Face Detection and Alignment using Multi-task Cascaded Convolutional Networks. *arXiv preprint arXiv:1604.02878* (2016).

[59] Wangjiang Zhu, Jie Hu, Gang Sun, Xudong Cao, and Yu Qiao. 2016. A key volume mining deep framework for action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1991–1999.