Chained Multi-stream Networks Exploiting Pose, Motion, and Appearance for Action Classification and Detection

Mohammadreza Zolfaghari , Gabriel L. Oliveira, Nima Sedaghat, and Thomas Brox
University of Freiburg
Freiburg im Breisgau, Germany
{zolfagha,oliveira,nima,brox}@cs.uni-freiburg.de

Abstract

General human action recognition requires understanding of various visual cues. In this paper, we propose a network architecture that computes and integrates the most important visual cues for action recognition: pose, motion, and the raw images. For the integration, we introduce a Markov chain model which adds cues successively. The resulting approach is efficient and applicable to action classification as well as to spatial and temporal action localization. The two contributions clearly improve the performance over respective baselines. The overall approach achieves state-of-the-art action classification performance on HMDB51, J-HMDB and NTU RGB+D datasets. Moreover, it yields state-of-the-art spatio-temporal action localization results on UCF101 and J-HMDB.

1. Introduction

Human action recognition is a complex task in computer vision, due to the variety of possible actions is large and there are multiple visual cues that play an important role. In contrast to object recognition, action recognition involves not only the detection of one or multiple persons, but also the awareness of other objects, potentially involved in the action, such as the pose of the person, and their motion. Actions can span various time intervals, making good use of videos and their temporal context is a prerequisite for solving the task to its full extent [42, 41].

The success of convolutional networks in recognition has also influenced action recognition. Due to the importance of multiple visual cues, as shown by Jhuang et al. [13], multi-stream architectures have been most popular. This trend was initiated by Simonyan and Zisserman [36], who proposed a simple fusion of the action class scores obtained with two separate convolutional networks, where one was trained on raw images and the other on optical flow. The relative success of this strategy shows that deep networks for action recognition cannot directly infer the relevant motion cues from the raw images, although, in principle, the network could learn to compute such cues.

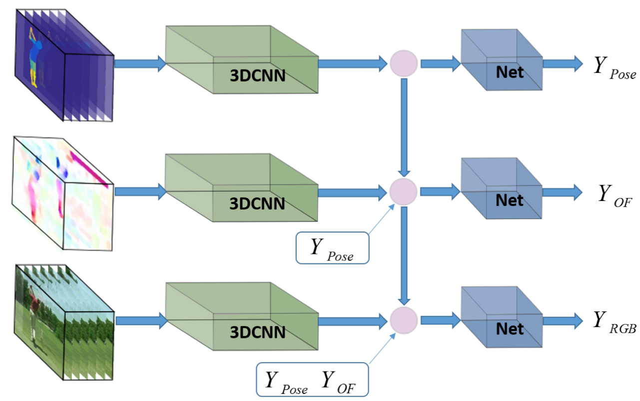

Figure 1. The chained multi-stream 3D-CNN sequentially refines action class labels by analyzing motion and pose cues. Pose is represented by human body parts detected by a deep network. The spatio-temporal CNN can capture the temporal dynamics of pose. Additional losses on Y_{Pose} and Y_{OF} are used for training. The final output of the network Y_{RGB} is provided at the end of the chain.

In this paper, we propose a three-stream architecture[1] that also includes pose, see Figure 1. Existing approaches model the temporal dynamics of human postures with hand-crafted features. We rather propose to compute the position of human body parts with a fast convolutional network. Moreover, we use a network architecture with spatio-temporal convolutions [41]. This combination can capture temporal dynamics of body parts over time, which is valuable to improve action recognition performance, as we show in dedicated experiments. The pose network also yields the spatial localization of the persons, which allows us to apply the approach to spatial action localization in a straightforward manner.

[1] https://lmb.informatik.uni-freiburg.de/projects/action_chain/

The second contribution is on the combination of the multiple streams, as also illustrated in Figure 1. The combination is typically done by summation of scores, by a linear classifier, or by early or late concatenation of features within the network. In this paper, we propose the integration of different modalities via a Markov chain, which leads to a sequential refinement of action labels. We show that such sequential refinement is beneficial over independent training of streams. At the same time, the sequential chain imposes an implicit regularization. This makes the architecture more robust to over-fitting – a major concern when jointly training very large networks. Experiments on multiple benchmarks consistently show the benefit of the sequential refinement approach over alternative fusion strategies.

Since actions may span different temporal resolutions, we analyze videos at multiple temporal scales. We demonstrate that combining multiple temporal granularity levels improves the capability of recognizing different actions. In contrast to some other state-of-the-art strategies to analyze videos over longer time spans, e.g., temporal segmentation networks [47], the architecture still allows the temporal localization of actions by providing actionness scores of frames using a sliding window over video. We demonstrate this flexibility by applying the approach also to temporal and spatio-temporal action detection. Compared to previous spatio-temporal action localization methods, which are typically based on region proposals and action tubes, the pose network in our approach directly provides an accurate person localization at no additional computational costs. Therefore, it consistently outperforms the previous methods in terms of speed and mean average precision.

2. Related Work

Feature based approaches. Many traditional works in the field of action recognition focused on designing features to discriminate action classes [19, 44, 5, 18, 17].

These features were encoded with high order encodings, e.g., bag of words (BoW) [38] or Fisher vector based encodings [33], to produce a global representation for video and to train a classifier on the action labels. Recent research showed that most of these approaches are not only computationally expensive, but they also fail on capturing context and high-level information.

CNN based approaches. Deep learning has enabled the replacement of hand-crafted features by learned features, and the learning of whole tasks end-to-end. Several works employed deep architectures for classification [26, 41, 45, 34]. Thanks to their hiearchical feature representation, deep networks learn to capture localized features as well as context cues and can exploit high-level information from large scale video datasets. Baccouche et al. [2] firstly used a 3D CNN to learn spatio-temporal features from video and in the next step they employed an LSTM to classify video sequences. More recently, several CNN based works presented efficient deep models for action recognition [7, 31, 41]. Tran et al. [41] employed a 3D architecture to learn spatio-temporal features from videos.

Fusion of multiple modalities. Zisserman et al. [36] proposed a two-stream CNN to capture the complementary information from appearance and motion, each modality in an independent stream. Feichtenhofer et al. [9] investigated the optimal position within a convolution network in detail to combine the separate streams. Park et al. [30] proposed a gated fusion approach. In a similar spirit, Wang et al. [50] presented an adaptive fusion approach, which uses two regularization terms to learn fusion weights. In addition to optical flow, some works made use of other modalities like audio [50], warped flow [47], and object information [12] to capture complementary information for video classification. In the present work, we introduce a new, flexible fusion technique for early or late fusion via a Markov chain and show that it outperforms previous fusion methods.

Pose feature based methods. Temporal dynamics of body parts over time provides strong information on the performing action. Thus, this information has been employed for action recognition and localization in several works [4, 21, 43, 39]. Cheron et al. [4] used pose information to extract high-level features from appearance and optical flow. They showed that using pose information for video classification is highly effective. Wang et al. [43] used data mining techniques to obtain a representation for each video and finally, by using a bag-of-words model to classify videos. In the present work, we compute the human body layout efficiently with a deep network and learn the relevant spatio-temporal pose features within one of the streams of our action classification network.

3. Inputs to the Network

We rely on three input cues: the raw RGB images, optical flow, and human pose in the form of human body part segmentation. All inputs are provided as spatio-temporal inputs covering multiple frames.

3.1. Optical Flow

We compute the optical flow with the method from Zach et al. [52], which is a reliable variational method that runs sufficiently fast. We convert the x-component and y-component of the optical flow to a 3 channel RGB image by stacking components and magnitude of them [31]. The flow and magnitude values in the image are multiplied by 16 and quantized into the [0,255] interval [20, 31, 46, 47].

3.2. Body Part Segmentation

Encoder-decoder architectures with an up-convolutional part have been used successfully for semantic segmentation tasks [25, 24, 32, 3, 29], depth estimation [22] and optical

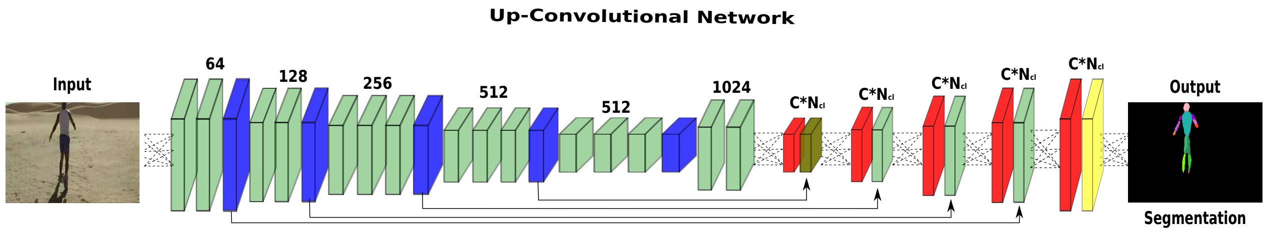

Figure 2. Human body part segmentation architecture. Convolutions are shown in green, pooling in blue, feature map dropout in brown, up-convolutional layers in red and softmax in yellow.

flow estimation [8]. For this work, we make use of Fast-Net [29], a network for human body part segmentation, which will provide our action recognition network with body pose information. Figure 2 illustrates the architecture of Fast-Net. The encoder part of the network is initialized with the VGG network [37]. Skip connections from the encoder to the decoder part ensure the reconstruction of details in the output up to the original input resolution.

We trained the Fast-Net architecture on the J-HMDB [13] and the MPII [1] action recognition datasets. J-HMDB provides body part segmentation masks and joint locations, while MPII provides only joint locations. To make body part masks compatible across datasets, we apply the following methodology, which only requires annotation for the joint locations. First, we derive a polygon for the torso from the joint locations around that area. Secondly, we approximate the other parts by ellipses scaled consistently based on the torso area and the distance between the respective joints; see second column of Fig. 3. We convert the body part segmentation into a 3 channel RGB image, mapping each label to a correspondent pre-defined RGB value.

To the best of our knowledge, we are the first who trained a convolutional network on body part segmentation for the purpose of action recognition. Figure 3 shows exemplary results of the body part segmentation technique on J-HMDB and MPII datasets. Clearly, the network provides good accuracy on part segmentation and is capable of handling images with multiple instances. The pose estimation network has a resolution of 150×150 and runs at 33 fps.

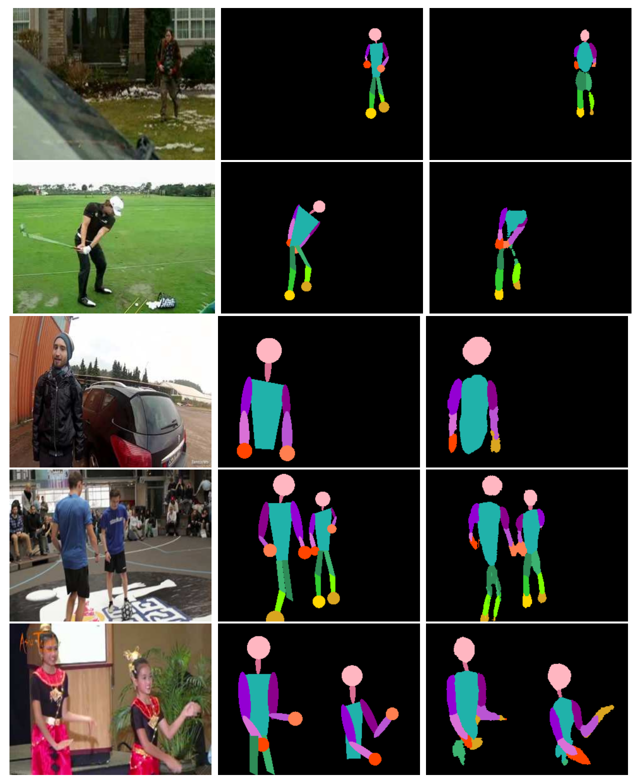

Figure 3. Qualitative results on J-HMDB and MPII datasets (task with 15 body parts). **First column:** Input image. **Second column:** Ground truth. **Third column:** Result predicted with Fast-Net. First two rows correspond to results on J-HMDB and the last ones on MPII.

4. Action Recognition Network

4.1. Multi-stream Fusion with a Markov Chain

To integrate information from the different inputs we rely on the model of a multi-stream architecture [36], i.e., each input cue is fed to a separate convolutional network stream that is trained on action classification. The innovation in our approach is the way we combine these streams. In contrast to the previous works, we combine features from the different streams sequentially. Starting with the human body part stream, we refine the evidence for an action class with the optical flow stream, and finally apply a refinement by the RGB stream.

We use the assumption that the class predictions are conditionally independent due to the different input modalities. Consequently, the joint probability over all input streams factorizes into the conditional probabilities over the separate input streams.

In a Markov chain, given a sequence of inputs $X = \{X_1, X_2, ..., X_S\}$, we wish to predict the output sequence $Y = \{Y_1, Y_2, ..., Y_S\}$ such that $P(Y|X)$ is maximized. Due to the Markov property, $P(Y|X)$ can be decomposed:

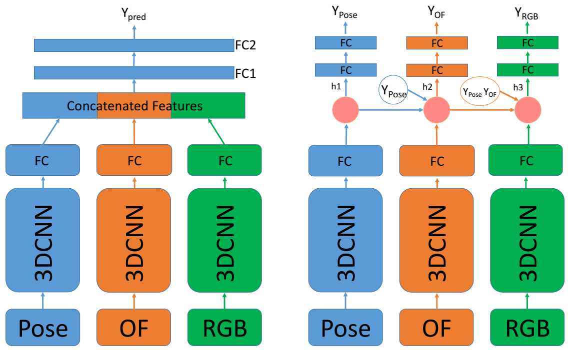

Figure 4. Baseline fusion architecture (left) and the proposed approach (right). In the chained architecture, there is a separate loss function for each stream. The final class label is obtained at the end of the chain (rightmost prediction).

$$P(Y|X) = P(Y_1|X) \prod_{s=2}^{S} P(Y_s|X, Y_1, \ldots, Y_{s-1}) \quad (1)$$

For the state $s \in \{1, \ldots, S\}$, we denote by h_s the hidden state of that stream. We use deep networks to model the likelihood in (1):

$$\begin{aligned} h_s &= f([h_{s-1}, \text{3DCNN}(X_s), (Y_1, \ldots, Y_{s-1})]) \\ P(Y_s|X, Y_{<s}) &= \text{softmax}(\text{Net}_s(h_s)), \end{aligned} \quad (2)$$

where f is a non-linearity unit (ReLU), h_{s-1} denotes the hidden state from the previous stream, and y_s is the prediction of stream s. For the $\text{3DCNN}(\cdot)$, we use the convolutional part and first fully connected layer of the network presented in Figure 5 to encapsulate the information in the input modality, and Net_s is the fully connected part in Figure 5.

At each fusion stage, we concatenate the output of the function $\text{3DCNN}(\cdot)$ with the hidden state and the outputs from the previous stream and apply the non-linearity f before feeding them to Net_s. Finally, at the output part, we use Net_s to predict action labels from h_s. With the $\text{softmax}(\cdot)$ function we convert these scores into (pseudo-)probabilities.

Using the above notation, we consider input modalities as $X = \{X_{pose}, X_{OF}, X_{RGB}\}$, and $X_s = \{x_t\}_{t=1}^{T}$, where x_t is the t-th frame in X_s, and T is the total number of frames in X_s. At the stage $s = 1$, by considering $X_1 = X_{pose}$ we start with an initial hidden state and obtain an initial prediction (see Figure 4-right):

$$\begin{aligned} h_1 &= \text{3DCNN}(X_{pose}) \\ P(Y_1|X) &= \text{softmax}(\text{Net}_1(h_1)) \end{aligned} \quad (3)$$

At each subsequent stage $s \geqslant 2$, we obtain a refined prediction y_s by combining the hidden state and the predictions from the previous stage.

$$\begin{aligned} h_2 &= f([h_1, \text{3DCNN}(X_{OF}), (Y_1)]) \\ P(Y_2|X, Y_{<2}) &= \text{softmax}(\text{Net}_2(h_2)) \\ h_3 &= f([h_2, \text{3DCNN}(X_{RGB}), (Y_1, Y_2)]) \\ P(Y_3|X, Y_{<3}) &= \text{softmax}(\text{Net}_3(h_3)) \end{aligned} \quad (4)$$

In the proposed model, at each stage, the next prediction is made conditioned on all previous predictions and the new input. Therefore, when training the network, the prediction of the output class label does not only depend on the input, but also on the previous state. Thus, the network in that stream will learn complementary features to refine the class labels from the previous streams. With this chaining and joint training, the information at the previous stages serve as the present belief for the predictions at the current stage, as shown in Figure 4-right. This sequential improvement of the class label enables the combination of multiple cues within a large network, while keeping the risk of over-fitting low.

This is in contrast to the fusion approaches that combine features from different, independently trained streams. In such a case, the different streams are not enforced to learn complementary features. In the other extreme, approaches that train all streams jointly but not sequentially, are more prone to over-fitting, because the network is very large, and, in such case, lacks the regularization via the separate streams and their additional losses.

It should be expected that the ordering of the sequence plays a role for the final performance. We compared different ordering options in our experiments and report them in the following section. The ordering that starts with the pose as input and ends with the RGB image yielded the best results.

It is worth noting that the concept of sequential fusion could be applied to any layer of the network. Here we placed the fusion after the first fully-connected layer, but the fusion could also be applied to the earlier convolutional layers.

4.2. Network Configuration

In all streams, we use the C3D architecture [41] as the base architecture, which has $17.5M$ parameters. The network has 8 three-dimensional convolution layers with kernel size of $3 \times 3 \times 3$ and stride 1, 5 three-dimensional pooling layers with kernel size of $2 \times 2 \times 2$ and stride 2 and two fully connected layers followed by a softmax; see Figure 5. Each stream is connected with the next stream via layer FC6; see Figure 4-right. Each stream takes 16 frames as input.

4.3. Training

The network weights are learned using mini-batch stochastic gradient descent (SGD) with a momentum of

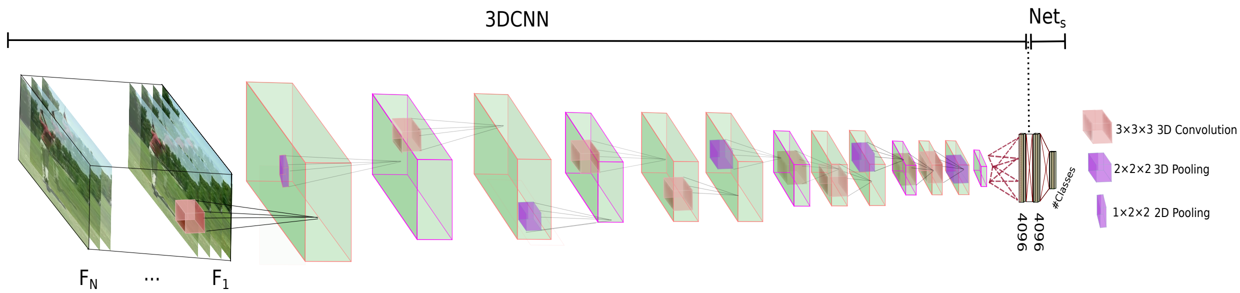

Figure 5. Base architecture used in each stream of the action recognition network. We define the convolutional part and the first fully connected layer as a 3DCNN and the remaining fully connected layers as Net_s.

0.9 and weight decay of $5e^{-4}$. We jointly optimize the whole network without truncating gradients and update the weights of each stream based on the full gradient including the contribution from the following stream. We initialize the learning rate with $1e^{-4}$ and decrease it by a factor of 10 every $2k$ for J-HMDB, $20k$ for UCF101 and NTU, and at multiple steps for HMDB51. The maximum number of iterations was $20k$ for J-HMDB, $40k$ for HMDB51 and $60k$ for the UCF101 and NTU datasets. We initialize the weights of all streams with an RGB network pre-trained on the large-scale Sports-1M dataset [15] except flow streams of J-HMDB and HMDB51 datasets which we initialize with the weights of UCF101 flow stream finetuned on Sports-1M model.

We split each video into clips of 16 frames with an overlap of 8 frames and feed each clip individually into the network stream with size of $16 \times 112 \times 112$. We apply corner cropping as a form of data augmentation to the training data. Corner cropping extracts regions from the corners and the center of the image. It helps to prevent the network from bias towards the center area of the input. Finally, we resize these cropped regions to the size of 112×112. In each iteration, all streams take the same clip from the video with the same augmentation but with different modalities as input.

We used Caffe [14] and an NVIDIA Titan X GPU to run our experiments. The training time for the J-HMDB dataset was ~ 10 hours for the full network.

4.4. Temporal Processing of the Whole Video

At test time, we feed the architecture with a temporal window of 16 frames. The stride over the video is 8. Each set of inputs is randomly selected for cropping operations, which are 4 corners and 1 center crop for the original image and their horizontal flipping counterpart. We extract scores before the softmax normalization in the last stream (Y_RGB).

In case of action classification, the final score of a video is calculated by taking the average of scores over all temporal windows across a video and 10 crop scores per clip. Apart from averaging, we also tested a multi-resolution approach, which we call **multi-granular (MG)**, where we trained separate networks for three different temporal resolutions. These are assembled as (1) 16 consecutive frames, (2) 16 frames from a temporal window of 32 frames by a sample rate of 2, and (3) 16 frames sampled randomly from the entire video. For the final score, we take the average over the scores produced by these temporal resolution networks. This approach extends the temporal context that the network can see, which can be useful for more complex actions with longer duration.

In case of temporal action detection, we localize the action in time by thresholding the score provided for each frame. Clearly, the MG approach is not applicable here. In addition to the action score, also the human body part network helps in temporal localization: we do not detect an action as long as no human is detected. More details on the spatio-temporal action detection are provided in the experimental section and in the supplemental material.

5. Experiments

5.1. Datasets

UCF-101 [40] contains more than 2 million frames in more than $13,000$ videos, which are divided into 101 human action classes. The dataset is split into three folds and each split contains about 8000 videos for training. The UCF101 dataset also comes with a subset for spatio-temporal action detection.

HMDB51 [16] contains 6766 videos divided into 51 action classes, each with at least 101 samples. The evaluation follows the same protocol used for UCF-101.

J-HMDB contains a subset of videos from the HMDB dataset, for which it provides additional annotation, in particular optical flow and joint localization [13]. Thus, it is well-suited for evaluating the contribution of optical flow, body part segmentation, and the fusion of all cues via a Markov chain. The dataset comprises 21 human actions. The complete dataset has 928 clips and 31838 frames. There are 3 folds for training and testing for this dataset. The videos in J-HMDB are trimmed and come with bounding boxes. Thus, it can be used also as a benchmark for spatial action localization.

Streams	Variant	UCF101	HMDB	J-HMDB
1	RGB	84.2%	53.3%	60.8%
	OF	79.6%	45.2%	61.9%
	Pose	56.9%	36.0%	45.5%
	Pose (GT)	-	-	56.8%
RGB+OF	baseline	87.1%	55.6%	62.7%
	chained	88.9%	61.7%	72.8%
	chained+MG	-	66.0%	-
3 w/o GT	baseline	89.1%	57.5%	70.2%
	chained	90.4%	62.1%	79.1%
	chained+MG	**91.3%**	**71.1%**	-
3 with GT	baseline	-	-	72.0%
	chained	-	-	**83.2%**

Table 1. The value of different cues and their integration for action recognition on the UCF101, HMDB51, and J-HMDB datasets (split 1). Adding optical flow and pose is always beneficial. Integration via the proposed Markov chain clearly outperforms the baseline fusion approach. In all cases, the accuracy achieved with estimated optical flow and body parts almost reaches the upper bound performance when providing ground truth values for those inputs.

NTU RGB+D is a recent action recognition dataset that is quite large and provides depth and pose ground truth [35]. It contains more than 56000 sequences and 4 million frames. NTU provides 60 action classes and $3D$ coordinates for 25 joints. Additionally, the high intra-class variations make NTU one of the most challenging datasets.

5.2. Action Classification

Table 1 shows that fusion with the sequential Markov chain model outperforms the baseline fusion consistently across all datasets. The baseline fusion is shown in Figure 4 and can be considered a strong baseline. It consists of fusing the multiple modalities through feature concatenation followed by a set of fully connected layers. The network is trained jointly.

Adding pose leads to a substantial improvement over the two-stream version. This confirms that pose plays an important role as complementary modality for action recognition tasks. Again, the Markov chain fusion is advantageous with a large margin.

For the J-HMDB dataset, ground truth for optical flow and pose is available and can be provided to the method. While not being relevant in practice, running the recognition with this ground truth shows on how much performance is lost due to erroneous optical flow and pose estimates. Surprisingly, the difference between the results is rather small, showing that the network does not suffer much from imperfect estimates. This conclusion can be drawn independently of the fusion method.

Finally, the temporal multi-granularity fusion (MG) further improves results. Especially on HMDB51, there is a large benefit.

	Datasets		
Methods	UCF101	HMDB51	J-HMDB
TS Fusion [9]	92.5%	65.4%	-
LTC [42]	91.7%	64.8%	-
Two-stream [36]	88.0%	59.4%	-
TSN [47]	**94.2%**	69.4%	-
CPD [28]	92.3%	66.2%	-
Sym. FV+SVM [6]	90.6%	67.8%	-
Multi-Granular [20]	90.8%	63.6%	-
M-fusion [30]	89.1%	54.9%	-
KVMF [53]	93.1%	63.3%	-
P-CNN [4]	-	-	61.1%
Action tubes [10]	-	-	62.5%
TS R-CNN [31]	-	-	70.5%
MR-TS R-CNN [31]	-	-	71.1%
Ours (chained)	91.1%	**69.7%**	**76.1%**

Table 2. Comparison to the state of the art on UCF101, HMDB51, and J-HMDB datasets (over all three splits).

Method	Cross Subject %
Deep LSTM [35]	60.7%
P-LSTM [35]	62.93%
HOG^2 [27]	32.2%
FTP DS [11]	60.23%
ST-LSTM [23]	69.2%
Ours (Pose)	67.8%
Ours (RGB+OF+Pose - Baseline)	76.9%
Ours (RGB+OF+Pose - Chained)	**80.8%**

Table 3. Comparison to literature on the NTU RGB+D benchmark.

5.2.1 Comparison with the state-of-the-art

Table 2 compares the proposed network to the state of the art in action classification. In contrast to Table 1, the comparison does not show the direct influence of single contributions anymore, since this table compares whole systems that are based on quite different components. Many of these systems also use other features extraction approaches, such as improved dense trajectories (IDT), which generally have a positive influence on the results, but also make the system more complicated and harder to control. Our network outperforms the state of the art on J-HMDB, NTU, and HMDB51. Also, on UCF101 dataset our approach is on par with the current state of the art while it does not rely on any additional hand-crafted features. In two stream case (RGB+OF), if we replace the 3DCNN network by the TSN approach [47], we obtain a classification accuracy of 94.05% on UCF101 (over 3 splits), which is the state of the art also on this dataset. However, the TSN approach does not allow for action detection anymore.

Finally, we ran the network on the recent NTU RGB+D dataset, which is larger and more challenging than the previous datasets. The dataset is popular for the evaluation of

Dataset	OPR	ORP	RPO	ROP	PRO	POR
HMDB51	59.8%	57.3%	54.8%	54.1%	56.4%	60.0%
UCF101	86.8%	86.2%	84.3%	84.7%	85.1%	87.1%

Table 4. Impact of chain order on the performance (clip accuracy) on UCF101 and HMDB51 datasets (split1). "O" = Optical flow, "P" = Pose and "R" = RGB.

Dataset	Y_Pose	Y_OF	Y_RGB
UCF101	55.7%	83.0%	90.4%
HMDB51	40.9%	56.4%	62.1%
J-HMDB	47.1%	65.3%	79.1%

Table 5. Sequential improvement of classification accuracy on UCF101, HMDB51 and J-HMDB datasets (Split1) by adding modalities to the chained network.

methods that are based on human body pose. Clearly, the result of our network, shown in Table 3, compares favorably to the existing methods. As a result, the used pose estimation network is competitive with pose estimates using depth images and that our way to integrate this information with the raw images and optical flow is advantageous.

5.2.2 Ordering of modalities in the Markov chain.

Table 4 shows an analysis on how the order of the modalities affects the final classification accuracy. Clearly, the ordering has an effect. The proposed ordering starting with the pose and then adding the optical flow and the RGB images performed best, but there are alternative orders that do not perform much worse.

Table 5 quantifies the improvement in accuracy when adding a modality. Clearly, each additional modality improves the results.

5.2.3 Fusion location

In principle the chained fusion can be applied to any layer in the network. We studied the effect of this choice. In contrast to the large scale evaluation in Feichtenhofer et al. [9], we tested only two locations: FC6 and FC7. Table 6 shows a clear difference only on the J-HMDB dataset. There it seems that an earlier fusion, at a level where the features are not too abstract yet, is advantageous. This is similar to the outcome of the study by Feichtenhofer et al. [9], where the last convolutional layer worked best.

5.2.4 Effect of clip length

We analyzed the effect of the size of the temporal window on the action recognition performance. Larger windows clearly improve the accuracy on all datasets; see Table 7. For the J-HMDB dataset (RGB modality) we use a temporal window ranging from 4 to 16 frames every 4 frames. The highest accuracy is obtained with a 16 frames clip size. Based on the J-HMDB minimum video size, 16 is the highest possible time frame to be explored. We also tested multiple temporal resolutions for the NTU dataset (pose modality). Again, we obtained the best results for the network with the larger clip length as input.

Fusion Location	UCF101	HMDB51	J-HMDB
FC7	89.8%	61.3%	73.9%
FC6	89.6%	62.1%	79.1%

Table 6. Classification performance for different fusion locations on UCF101, HMDB51 and J-HMDB datasets (split1).

Dataset	Clip length	Accuracy
J-HMDB (RGB)	4	44.8%
	8	49.6%
	12	58.7%
	16	60.8%
NTU RGB+D (Pose)	16	61.6%
	32	67.8%

Table 7. Effect of the temporal window size. Using more frames as input to the network consistently increases classification performance.

The conducted experiments confirm that increasing the length of the clip, we decrease the chance of getting unrelated parts of an action in a video. In addition, with longer sequences, 3D convolutions can better exploit their ability to capture abstract spatio-temporal features for recognizing actions.

5.3. Action Detection

To demonstrate the generality of our approach, we show also results on action detection on UCF101 and J-HMDB. Many of the top performing methods for action classification are not applicable to action detection, because they integrate information over time in a complex manner, are too slow, or are unable to spatially localize the action.

This is different for our approach, which is efficient and can be run in a sliding window manner over time and provides good spatial localization via the human body part segmentation. In order to create temporally consistent spatial detections, we link action bounding boxes over time to produce action tube [10]; see the supplemental material for details. We use the frame level action classification scores to make predictions at the tube level. Figure 6 schematically outlines the detection procedure.

We also present a set of qualitative action detection experiments for the UCF and J-HMDB datasets. Figure 7 shows several examples where we can robustly localize the action, even when unusual pose, illumination, viewpoints and motion blur are presented. Additional results exploring failure cases are provided in supplementary material.

Following recent works on action detection [10, 48, 31], we report video-AP. A detection is considered correct if the intersection over union (IoU) with the ground-truth is above a threshold δ and the action label is predicted correctly. The IoU between two tubes is defined as the IoU over the temporal domain, multiplied by the average of the IoU be-

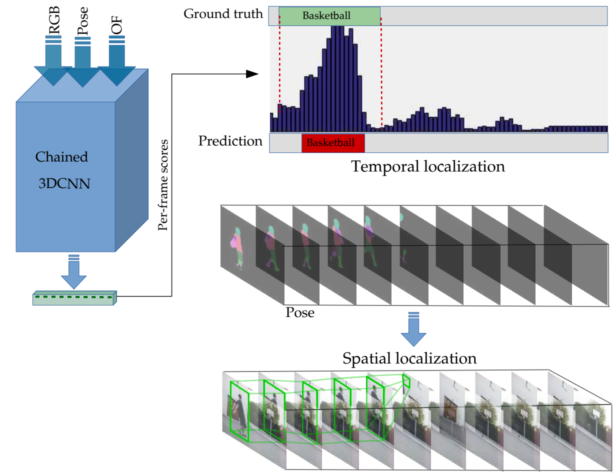

Figure 6. Scheme for spatio-temporal action detection. The chained network provides action class scores and body part segmentations per frame. From these we compute action tubes and their actionness scores; see the supplemental material for details.

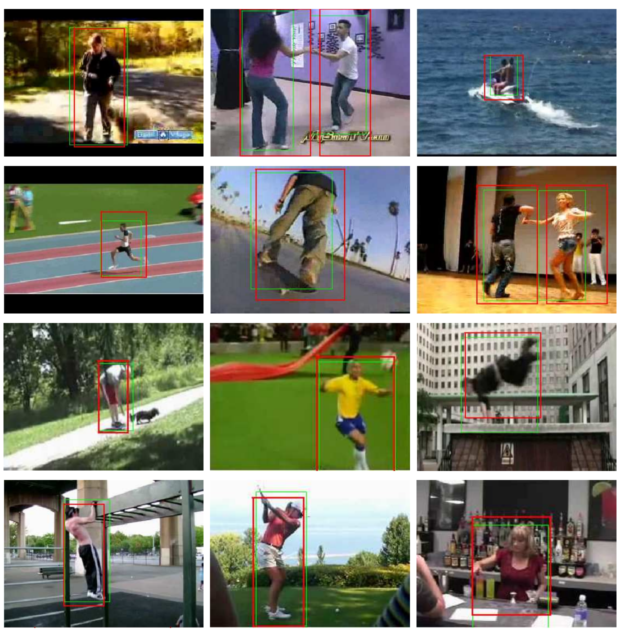

Figure 7. Qualitative results on the action detection task. The first two rows correspond to detections on UCF101, the last ones on J-HMDB. Ground truth bounding boxes are shown in green and detections in red. Our spatial localization is accurate and robust to unusual pose.

tween boxes averaged over all overlapping frames. Video-AP measures the area under the precision-recall curve of the action tube predictions.

Table 8 and Table 9 show the video mAP results on spatial and spatio-temporal action detection with different IoU thresholds on J-HMDB and UCF101 (split1) datasets respectively. Although we did not optimize our approach for action detection, we obtain state-of-the-art results on both datasets. Moreover, the approach is fast: spatial detection runs at a rate of 31 fps and spatio-temporal detection with 10 fps. Compared to the recent works [10, 49, 31, 51], our detection framework has two desirable properties: (1) the pose network directly provides a single detection box per person, which causes a large speed-up; (2) the classification takes advantage of three modalities and the chained fusion, which yields highly accurate per-frame scores.

	J-HMDB				
IoU threshold (δ)	0.1	0.2	0.3	0.4	0.5
Actionness [46]	-	-	-	-	56.4
ActionTubes [10]	-	-	-	-	53.3
Weinzaepfel *et al.* [48]	-	63.1	63.5	62.2	60.7
Peng *et al.* [31]	-	74.3	-	-	73.1
Ours	**78.81**	**78.20**	**77.12**	**75.05**	**73.47**

Table 8. Spatial action detection results (Video mAP) on the J-HMDB dataset. Across all IoU thresholds, our model outperforms the state of the art.

	UCF101			
IoU threshold (δ)	0.05	0.1	0.2	0.3
Weinzaepfel *et al.* [48]	54.28	51.68	46.77	37.82
Yu *et al.* [51]	42.80	-	-	-
Peng *et al.* [31]	54.46	50.39	42.27	32.70
Weinzaepfel *et al.* [49]	62.8	-	45.4	-
Ours	**65.22**	**59.52**	**47.61**	**38.00**

Table 9. Spatio-temporal action detection results (Video mAP) on UCF101 dataset (split1). Across all IoU thresholds, our model outperforms the state of the art.

6. Conclusions

We have proposed a network architecture that integrates multiple cues sequentially via a Markov chain model. We have shown that this sequential fusion clearly outperforms other ways of fusion, because it can consider the mutual dependencies of cues during training while avoiding overfitting due to very large network models. Our approach provides state-of-the-art performance on all four challenging action classification datasets UCF101, HMDB51, J-HMDB and NTU RGB+D while not using any additional handcrafted features. Moreover, we have demonstrated the value of a reliable pose representation estimated via a fast convolutional network. Finally, we have shown that the approach generalizes also to spatial and spatio-temporal action detection, where we obtained state-of-the-art results as well.

7. Acknowledgements

We acknowledge funding by the ERC Starting Grant VideoLearn and the Freiburg Graduate School of Robotics.

References

[1] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele. 2d human pose estimation: New benchmark and state of the art analysis. *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014. 3

[2] M. Baccouche, F. Mamalet, C. Wolf, C. Garcia, and A. Baskurt. Sequential deep learning for human action recognition. In *Proceedings of the Second International Conference on Human Behavior Unterstanding*, HBU'11, pages 29–39, 2011. 2

[3] V. Badrinarayanan, A. Kendall, and R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *arXiv preprint arXiv: 1511.00561*, 2015. 2

[4] G. Chéron, I. Laptev, and C. Schmid. P-CNN: Pose-based CNN Features for Action Recognition. In *ICCV*, 2015. 2, 6

[5] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1 - Volume 01*, CVPR '05, pages 886–893, Washington, DC, USA, 2005. IEEE Computer Society. 2

[6] C. R. de Souza, A. Gaidon, E. Vig, and A. M. L. Peña. Sympathy for the details: Dense trajectories and hybrid classification architectures for action recognition. *CoRR*, abs/1608.07138, 2016. 6

[7] A. Diba, A. M. Pazandeh, and L. V. Gool. Efficient two-stream motion and appearance 3d cnns for video classification. *CoRR*, abs/1608.08851, 2016. 2

[8] A. Dosovitskiy, P. Fischer, E. Ilg, P. Häusser, C. Hazrba, V. Golkov, P. v.d. Smagt, D. Cremers, and T. Brox. Flownet: Learning optical flow with convolutional networks. In *IEEE International Conference on Computer Vision (ICCV)*, Dec 2015. 3

[9] C. Feichtenhofer, A. Pinz, and A. Zisserman. Convolutional two-stream network fusion for video action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 2, 6, 7

[10] G. Gkioxari and J. Malik. Finding action tubes. 2015. 6, 7, 8

[11] J. F. Hu, W. shi Zheng, J. Lai, and J. Zhang. Jointly learning heterogeneous features for rgb-d activity recognition. In *Computer Vision and Pattern Recognition (CVPR) (In press)*, 2015. 6

[12] M. Jain, J. C. van Gemert, and C. G. M. Snoek. What do 15,000 object categories tell us about classifying and localizing actions? In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 46–55, June 2015. 2

[13] H. Jhuang, J. Gall, S. Zuffi, C. Schmid, and M. J. Black. Towards understanding action recognition. In *International Conf. on Computer Vision (ICCV)*, pages 3192–3199, 2013. 1, 3, 5

[14] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014. 5

[15] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei. Large-scale video classification with convolutional neural networks. In *CVPR*, 2014. 5

[16] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre. HMDB: a large video database for human motion recognition. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2011. 5

[17] Z. Lan, M. Lin, X. Li, A. G. Hauptmann, and B. Raj. Beyond gaussian pyramid: Multi-skip feature stacking for action recognition. *CoRR*, abs/1411.6660, 2014. 2

[18] I. Laptev. On space-time interest points. *Int. J. Comput. Vision*, 64(2-3):107–123, Sept. 2005. 2

[19] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. In *Proceedings of the IEEE*, pages 2278–2324, 1998. 2

[20] Q. Li, Z. Qiu, T. Yao, T. Mei, Y. Rui, and J. Luo. Action recognition by learning deep multi-granular spatio-temporal video representation. In *Proceedings of the 2016 ACM on International Conference on Multimedia Retrieval*, ICMR '16, pages 159–166, New York, NY, USA, 2016. ACM. 2, 6

[21] I. Lillo, J. C. Niebles, and A. Soto. A hierarchical pose-based approach to complex action understanding using dictionaries of actionlets and motion poselets. *CoRR*, abs/1606.04992, 2016. 2

[22] F. Liu, C. Shen, G. Lin, and I. D. Reid. Learning depth from single monocular images using deep convolutional neural fields. *IEEE Trans. Pattern Anal. Mach. Intell.*, 38(10):2024–2039, 2016. 2

[23] J. Liu, A. Shahroudy, D. Xu, e. B. Wang, Gang", J. Matas, N. Sebe, and M. Welling. *Spatio-Temporal LSTM with Trust Gates for 3D Human Action Recognition*, pages 816–833. Springer International Publishing, 2016. 6

[24] W. Liu, A. Rabinovich, and A. C. Berg. Parsenet: Looking wider to see better. *arXiv preprint arXiv: 1506.04579*, 2015. 2

[25] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. *CVPR*, Nov. 2015. 2

[26] B. Mahasseni and S. Todorovic. Regularizing long short term memory with 3d human-skeleton sequences for action recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 2

[27] E. Ohn-Bar and M. M. Trivedi. Joint angles similarities and hog2 for action recognition. In *Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, CVPRW '13, pages 465–470, Washington, DC, USA, 2013. IEEE Computer Society. 6

[28] K. Ohnishi, M. Hidaka, and T. Harada. Improved dense trajectory with cross streams. In *Proceedings of the 2016 ACM on Multimedia Conference*, MM '16, pages 257–261, New York, NY, USA, 2016. ACM. 6

[29] G. L. Oliveira, W. Burgard, and T. Brox. Efficient deep models for monocular road segmentation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016. 2, 3

[30] E. Park, X. Han, T. L. Berg, and A. C. Berg. Combining multiple sources of knowledge in deep cnns for action recognition. In *WACV*, pages 1–8. IEEE Computer Society, 2016. 2, 6

[31] X. Peng and C. Schmid. Multi-region two-stream R-CNN for action detection. In *ECCV 2016 - European Conference on Computer Vision*, Amsterdam, Netherlands, Oct. 2016. 2, 6, 7, 8

[32] O. Ronneberger, P.Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, volume 9351 of *LNCS*, pages 234–241. Springer, 2015. 2

[33] J. Sanchez, F. Perronnin, T. E. J. Mensink, and J. Verbeek. Image classification with the fisher vector: Theory and practice. *International Journal of Computer Vision*, 2013. 2

[34] N. Sedaghat, M. Zolfaghari, and T. Brox. Orientation-boosted voxel nets for 3d object recognition. *CoRR*, abs/1604.03351, 2016. 2

[35] A. Shahroudy, J. Liu, T.-T. Ng, and G. Wang. Ntu rgb+d: A large scale dataset for 3d human activity analysis. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 6

[36] K. Simonyan and A. Zisserman. Two-stream convolutional networks for action recognition in videos. In *Advances in Neural Information Processing Systems*, 2014. 1, 2, 3, 6

[37] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *ICLR*, 2015. 3

[38] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *Proceedings of the Ninth IEEE International Conference on Computer Vision - Volume 2*, ICCV '03, pages 1470–, Washington, DC, USA, 2003. IEEE Computer Society. 2

[39] K. Soomro, H. Idrees, and M. Shah. Predicting the where and what of actors and actions through online action localization. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 2

[40] k. Soomro, A. Roshan Zamir, and M. Shah. UCF101: A dataset of 101 human actions classes from videos in the wild. In *CRCV-TR-12-01*, 2012. 5

[41] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, pages 4489–4497, 2015. 1, 2, 4

[42] G. Varol, I. Laptev, and C. Schmid. Long-term temporal convolutions for action recognition. *CoRR*, abs/1604.04494, 2016. 1, 6

[43] C. Wang, Y. Wang, and A. L. Yuille. An approach to pose-based action recognition. In *Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition*, CVPR '13, pages 915–922, Washington, DC, USA, 2013. IEEE Computer Society. 2

[44] H. Wang and C. Schmid. Action recognition with improved trajectories. In *Proceedings of the 2013 IEEE International Conference on Computer Vision*, ICCV '13, pages 3551–3558, Washington, DC, USA, 2013. IEEE Computer Society. 2

[45] L. Wang, Y. Qiao, and X. Tang. Action recognition with trajectory-pooled deep-convolutional descriptors. In *CVPR*, pages 4305–4314, 2015. 2

[46] L. Wang, Y. Qiao, X. Tang, and L. Van Gool. Actionness estimation using hybrid fully convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2708–2717, 2016. 2, 8

[47] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. Val Gool. Temporal segment networks: Towards good practices for deep action recognition. In *ECCV*, 2016. 2, 6

[48] P. Weinzaepfel, Z. Harchaoui, and C. Schmid. Learning to track for spatio-temporal action localization. In *ICCV 2015 - IEEE International Conference on Computer Vision*, pages 3164–3172, Santiago, Chile, Dec. 2015. IEEE. 7, 8

[49] P. Weinzaepfel, X. Martin, and C. Schmid. Towards weakly-supervised action localization. *CoRR*, abs/1605.05197, 2016. 8

[50] Z. Wu, Y. Jiang, X. Wang, H. Ye, X. Xue, and J. Wang. Fusing multi-stream deep networks for video classification. *CoRR*, abs/1509.06086, 2015. 2

[51] G. Yu and J. Yuan. Fast action proposals for human action detection and search. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1302–1311, June 2015. 8

[52] C. Zach, T. Pock, and H. Bischof. A duality based approach for realtime tv-l1 optical flow. In *Proceedings of the 29th DAGM Conference on Pattern Recognition*, pages 214–223, Berlin, Heidelberg, 2007. Springer-Verlag. 2

[53] W. Zhu, J. Hu, G. Sun, X. Cao, and Y. Qiao. A key volume mining deep framework for action recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1991–1999, June 2016. 6