Two-stream Flow-guided Convolutional Attention Networks for Action Recognition

An Tran Loong-Fah Cheong
Department of Electrical & Computer Engineering, National University of Singapore
an.tran@u.nus.edu eleclf@nus.edu.sg

Abstract

This paper proposes a two-stream flow-guided convolutional attention networks for action recognition in videos. The central idea is that optical flows, when properly compensated for the camera motion, can be used to guide attention to the human foreground. We thus develop cross-link layers from the temporal network (trained on flows) to the spatial network (trained on RGB frames). These cross-link layers guide the spatial-stream to pay more attention to the human foreground areas and be less affected by background clutter. We obtain promising performances with our approach on the UCF101, HMDB51 and Hollywood2 datasets.

1. Introduction

Human action recognition in video is an important and challenging problem in computer vision. Like many other computer vision problems, an effective visual representation of actions in video data is vital to deal with these problems.

Over the last decade, there is a great evolution of features for action recognition from short video clips [29, 10, 22, 26]. The research works can be roughly divided into two mainstreams. The first type of representation is *hand-crafted* local features in combination with the *Bag-of-Features* (BoFs) paradigm [14, 11, 29]. Probably the most successful approach of local features representation is to extract improved dense trajectory features [29] and deploy Fisher vector representation [20]. The second approach is to utilize *deep learning algorithms* to learn features automatically from data (*e.g.*, RGB frames or optical flows) [22, 10, 26, 28, 30]. Probably the most successful approach of local features representation is to extract improved dense trajectory features [29] and deploy Fisher vector representation [20]. High performances of neural network architectures have been recently reported on video action recognition, specially those of two-stream convolutional networks [22, 30].

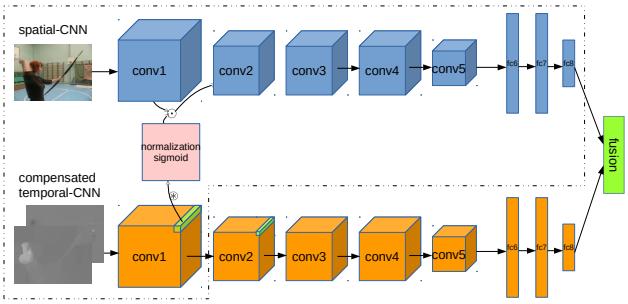


Figure 1: Our proposed *Two-stream Flow-guided Convolutional Attention Networks* (Two-stream FCANs) for action recognition. A video clip is represented by RGB frames and optical flows. Two streams of data are fed into two separate CNNs: *spatial stream* that models scene and object contexts (blue), while (compensated) *temporal-stream* likely provides some motion-based attentions on foreground actions (orange). We leverage attentions provided from *temporal-stream* to assist recognition processes in *spatial-stream* by cross-link layers (pink). The attention weighted feature maps are fused by element-wise multiplication to preserve spatial-temporal structure in videos. The Two-stream FCAN refers to the entire architecture of two streams with the late fusion stage, whereas we call the area inside dashed lines the FCAN model. Best viewed in color.

Due to different network architectures and types of data in these two-stream networks, the learned features should have characteristics that help to deal with the different types of nuances in these specific data streams. RGB frames in video usually provide scene and object contexts in the background together with the human forms in the foreground. However, the spatial area occupied by the human foreground is usually much smaller than the area of the background such that it might not be effectively represented. On the contrary, optical flows in videos, when properly compensated for camera motion, immediately isolate the moving human silhouettes (see Figure 2), and provide motion

cues. In view of the preceding discussion, feature responses of a CNN model (*i.e*., called *spatial-CNN*) on RGB data are likely to be activations more on background contexts rather than on human foreground actions. In contrast, a CNN model (*i.e*., called *temporal-CNN*) trained on flow data would often fire more on the human forms and movements.

In this paper, we propose a novel flow-guided convolutional attention networks (FCANs) for action recognition based on the aforementioned two-stream network architecture (see Figure 1). This attention guiding is partly motivated by the primate visual system, whereby it is known that there is connectivity between the motion pathway and the form pathway [27]. To model the attention guidance, we propose cross-link layers from the temporal stream to the spatial stream. There can be multiple such cross-link layers, but as we shall show later, the optimal number is in the range of one to two layers. Each cross-link layer has three components: (1) a convolution layer to reduce the dimension of the flow feature tensor; (2) a mean-variance normalization layer; (3) a sigmoid function. The final cross-link output is an attention map, which is used to control the level of activation in the corresponding layer in the spatial-stream via an element-wise multiplication.

Our contributions are two-fold. First, we propose a flow-guided convolution based attention mechanism for action recognition task. Second, we perform comprehensive evaluations two-stream FCANs based on 3d-convolution operations; we also explore the effects of different number of cross-link layers to understand where they are most effective. We visualize attentions provided by cross-link layers in our FCAN model (3D version) to show the attentive capacity of the (compensated) flows. Lastly, we achieve promising results on the HMDB51 and the UCF101 datasets. All codes and models [1] are implemented in Caffe framework [9].

2. Related Work

Visual features. Many hand-crafted features have been proposed in the history of action recognition community, such as HOG/HOF [14], HOG3D [11], MBH [29], *etc*.

Inspired by recent successes in image classification [12], there have been extensions of the neural networks to the video action recognition problem [22, 10, 26]. CNN architectures play significant roles in these works, either as an individual module or as an encoder module for a type of recurrent neural networks (RNNs). Karpathy *et al*. [10] propose a large-scale video dataset, namely Sport1M, and investigate different ways to embed temporal information into the current CNN architecture. Two-stream CNN model [22] has demonstrated good performance on the UCF101 dataset [24] by combining predictions from two CNNs: *spatial-CNNs* trained from RGB frames and *temporal-CNNs* trained from optical flows. Recently, Tran *et al*. [26] extend 2D CNNs to 3D CNNs by developing 3d-convolution and 3d-pooling layers. In [31, 4], the output of a CNN's last layer is fed into a recurrent sequence model usually formed by LSTM cells. Interesting works [28, 30] have attempted to model longer temporal information of videos.

[1] https://github.com/antran89/two-stream-fcan

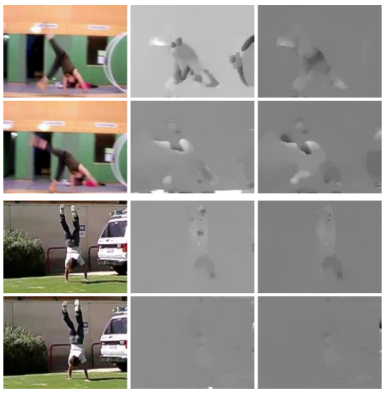


Figure 2: Columns represent examples of three modalities of inputs: RGB frames, optical flows (*x,y*-directions), and compensated optical flows. First two rows consist of two frames of a *cartwheel* video in HMDB51 dataset, and last two ones consist of two frames of a *handstand walking* video in UCF101 dataset.

Attention for action recognition. Attention is a mechanism used to confer more weights on a subset of features. The attention mechanism has also been applied to action recognition [16, 21, 1, 2]. Bazzani *et al*. use additional human fixation data to train mixture density network for saliency prediction and apply it to action recognition with the so called C3D [26] features obtained from the 3D CNNs. On the contrary, our approach does not use any additional data except flows to predict attentions. Furthermore, Sharma *et al*. [21] extract image features in each frame with the VGG16 CNN model [23] and predict visual attention in each frame using a recurrent model with LSTM cell. The work most similar to ours is VideoLSTM [16]. VideoLSTM [16] uses convolutional LSTM trained on optical flow to predict attention for a second convolutional LSTM layer. Our FCAN is a convolution-based network that embeds attention in the process of action classification.

Before the deep learning era, there have been works incorporating saliency into action recognition from videos. Several saliency measures have been proposed for actions in [25, 17] and they show improvements in the recognition accuracy when focusing attention on the foreground.

3. Flow-guided Convolutional Attention Networks (FCANs)

We propose the cross-link layers to model the interactions of the two networks in the two-stream convolutional network [22]. The whole network is differential, so it can be trained end-to-end with the stochastic gradient descent (SGD) and back-propagation algorithm [15]. The overall architecture of the FCAN is shown in Figure 1. In the ensuing discussion, we describe the 3D-FCAN model, which is the 3D version based on 3D convolution (*e.g.*, C3D [26]) building blocks (please refer to the Supplementary Materials for the 2D-FCAN model based on 2D convolution (*e.g.*, Alexnet [12])).

3.1. 3D version of flow-guided convolutional attention networks

With the assumption that the magnitudes of optical flows, when appropriately compensated for camera motions, usually correlate with the foreground regions, we develop the framework of flow-guided convolutional attention networks (FCANs) as shown in Figure 1. The C3D network [26] provides explicit representation of the time dimension in the architecture. Both the 3D-FCAN and 2D-FCAN have similar structures, except for the operations in the convolution and mean-variance normalization layer. Let $x\_rgb^l \in \mathbb{R}^{C_l \times T_l \times H_l \times W_l}$, $x\_flow^l \in \mathbb{R}^{C_l \times T_l \times H_l \times W_l}$ be the feature map of layer $l \in \{0, 1, ..., L\}$ in the *spatial-* and *temporal-C3D* respectively, with C_l, T_l, H_l, W_l being the number of channels, temporal length, height and width of the feature map. Specifically, in the proposed FCANs, x_rgb and x_flow are the feature maps from a pooling layer in the C3D network. We develop attentive cross-link layers between the early pooling layers from the *temporal-C3D* to the *spatial-C3D*. As we shall show later, the optimal number of cross-link layers is between one and two, because the activations from the *temporal-C3D* at these early stages are still largely retinotopic. They directly point to the foreground regions and help the *spatial-C3D* learn distributed feature representation focused around these regions for the label prediction task. In the following, we report results for the case of only one attentive cross-link layer. Such cross-link layer includes the following three steps: reducing dimensions of a flow feature tensor x_flow^l (Equ. 1), mean-variance normalization (Equ. 2), and attention prediction (Equ. 3). We use a 3d-convolutional layer to reduce a flow feature tensor $x\_flow^l \in \mathbb{R}^{C_l \times T_l \times H_l \times W_l}$ to $x\_link^l \in \mathbb{R}^{1 \times T_l \times H_l \times W_l}$:

$$x\_link^l = W_{3D\_link} \circledast x\_flow^l. \tag{1}$$

where $\circledast$ is a 3d-convolution operation along the channel dimension C_l. We initialize the filter weights W_{3D_link} to $\frac{1}{C_l}$ in the training phase. Then, we normalize the feature tensor x_link^l by the mean μ and variance σ of all the spatial-temporal feature activations in x_link^l:

$$\hat{x}^l_{t,h,w} = \frac{x\_link^l_{t,h,w} - \mu}{\sigma}. \tag{2}$$

The mean-variance normalization layer transforms the raw attention scores x_link^l into a normalized range $\hat{x}^l \in [-1, 1]$. Finally, the normalized attention score $\hat{x}^l$ is converted to an attention probability score $a^l \in [0, 1]$ by a sigmoid function:

$$a^l_{t,h,w} = \text{sigmoid}(\hat{x}^l_{t,h,w}). \tag{3}$$

where $a^l_{h,w} \in \mathbb{R}^{1 \times T_l \times H_l \times W_l}$.

We apply the flow-guided attention map on the feature map x_rgb^l of the *spatial-C3D* by multiplicative interaction:

$$x\_rgb^l_{att} = \mathbf{r}(a^l, C_l) \odot x\_rgb^l. \tag{4}$$

where $\mathbf{r}(a^l, C_l)$ is the C_l-times replication of the predictive attention map a^l along the channel dimension, and $\odot$ denotes element-wise multiplication operation.

The attended feature map $x\_rgb^l_{att}$ is forwarded into the next layer $l + 1$ to learn more abstract attended features:

$$x\_rgb^{l+1} = f^{l+1}_{spatial\_C3D}(x\_rgb^l_{att}). \tag{5}$$

where $f^{l+1}_{spatial\_C3D}$ is the operation in the next layer (*e.g.*, convolution layer). Recall that we choose to have only one attentive cross-link layers because the activations in the higher layers of the *temporal-C3D* would be more abstract and not necessarily correspond to the notions of foreground objects.

4. Experiments

4.1. Data sets

We evaluate the two-stream FCANs on two datasets for action recognition.

UCF101 [24]. This dataset is among the largest available action recognition benchmarks. UCF101 has 101 action classes and about 13320 videos (180 frames/video on average). There are three splits of training/testing data, and the performance is measured by mean classification accuracy across the splits.

HMDB51 [13]. HMDB51 has 51 action categories and 6,766 videos. The dataset has two versions (original and motion-stabilized), and we use the original version which is more challenging for action recognition. The dataset has diverse background contexts and variations in motion pattern. It has three train/test splits with 3,570 training and 1,530 test videos.

Hollwood2 [18]. The Hollywood2 [18] dataset has 12 categories with 1,707 videos, which consist of 823 training and 884 test videos. The performance is measured by mean average precision (mAP) over all classes.

4.2. Implementation details

Video preprocessing. For direct comparison with the two-stream CNNs work [22], we sample a fixed number of frames (*i.e.*, 25) per video with equal temporal spacing in both training and testing. Optical flows are computed with TV-L1 [32]. We choose an OpenCV implementation of TV-L1 because it has a good balance of efficiency and accuracy.

Compensated flows. Similar to the Improved Dense Trajectories work [29], we deploy a global motion estimation method based on the assumption that two consecutive frames are related by a homography. Removal of this background motion (induced by the camera motion) renders the optical flow magnitudes more indicative of the locations of the human silhouettes (*e.g.*, Figure 2). After compensation, we extract the $x-$, $y-$ optical flows and convert them into gray-scale images $[0, 255]$ by a linear rescaling. This rescaling has two-fold benefits. First, it will reduce the size of the flow datasets dramatically, as we now save the flow fields as images rather than as floating point numbers (*e.g.*, from few TBs to dozen of GBs in UCF101). Second, by saving the flow fields as images, we are able to fine-tune our CNNs from models pre-trained with large-scale image dataset (*i.e.*, ImageNet).

ConvNet architectures. We utilize the C3D [26] as the main component for our two-stream FCAN network. During our experiments, in the C3D network, we achieve higher performance by setting a high dropout ratio of 0.9 and 0.8 for fully connected layers fc6, fc8 respectively. We need to have more regularization (*i.e.*, higher dropout) in the C3D network due to the higher risk of over-fitting for the high capacity C3D models when dealing with small datasets (*e.g.*, UCF101, HMDB51). We also experiment with 2D-CNN architecture (*e.g.*, AlexNet), but flow-guided attentions does not provide benefits to frame-based representations.

Data augmentation. At training time, we sample 25 (overlapping) clips per videos with a temporal length of 1 frame for the 2D-CNN and 16 frames for the C3D network. We also adopt the corner and multi-scale cropping strategy for training the baseline models [30]. However, for training the FCAN models, we do not use multi-scale cropping because the attention maps already delineate which regions require more resolution and which require more of an overall gist for background context. Note that in our case, each random crop sample should apply to the same location of both the RGB and flow images; without this correspondence, the cross-link layers would be meaningless.

Pre-trained weights. In order not to overfit the CNN models in our experiments, we follow the initialization strategies in [30]. For the *spatial-CNN* and *spatial-C3D* networks, we initialize them with the pre-trained weights obtained from large-scale datasets (*i.e.*, ImageNet [9] and Sports1M [26] respectively).

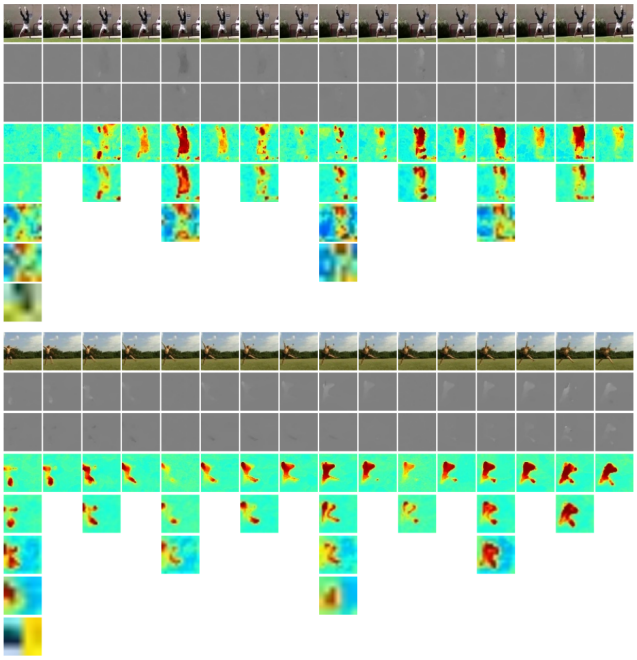


Figure 3: Visualizations of flow-guided attention provided by a temporal-C3D network. The top half shows a handstand walking video in UCF101, while the bottom one shows a cartwheel video in HMDB51 dataset. From top to bottom in each half: 16 RGB frames, flow-x, flow-y, attentions at layers pool1, pool2, pool3, pool4 and pool5. The spatial-temporal resolution of feature maps sequentially decreases with pooling layers, but we upsample the feature maps to have same sizes. Warm color indicates high saliency value. Best viewed in color.

Training. Our attention network is trained end-to-end with the standard back-propagation algorithms. We use the mini-batch stochastic gradient descent (SGD) algorithm to optimize the cross-entropy error function. The initial learning rate is 0.0001. We use mini-batches of 256 samples for the 2D-CNN networks, and 128 samples for the C3D architectures. For UCF101, we optimize the networks for 20K iterations, during which the learning rate is twice decreased with a factor of 0.1 at the 12K and 18K iterations. Due to the smaller dataset size of HMDB51 and Hollywood2, we run the SGD algorithm for 10K iterations and reduce the learning rate with a factor of 0.1 at the 4K and 8K iterations. In contrast to [26], we do not train a SVM on features fc6 extracted from the C3D models and our models are trained and tested in an end-to-end fashion. As can be shown in Section 4.4, the performance of our end-to-end training is

Models	UCF101		HMDB51		Hollywood2
	Clip acc. (%)	Video acc. (%)	Clip acc. (%)	Video acc. (%)	mAP (%)
spatial-C3D	80.5	83.6	51.3	53.9	43.6
temporal-C3D	70.6	83.1	38.5	50.7	53.9
temporal-C3D-comp	72.0	84.6	42.6	55.8	67.7
VideoLSTM RGB[16]	-	79.6	-	43.3	-
VideoLSTM flow[16]	-	82.1	-	52.6	-
FCAN	81.5	85.4	51.6	54.6	46.9
FCAN-comp	82.7	87.2	53.5	56.9	50.3
VideoLSTM two-stream [16]	-	89.2	-	56.4	-
Twostream-C3D	86.8	91.8	54.8	64.4	51.2
Twostream-C3D-comp	86.8	91.4	55.7	67.1	65.9
Twostream-FCAN	**87.2**	91.9	54.8	63.3	56.3
Twostream-FCAN-comp	86.7	**91.9**	**55.9**	**68.2**	**71.1**

Table 1: Results for two-stream FCAN models and their baselines on UCF101 and HMDB51 (both split 1) dataset. 3D convolutional neural networks have inputs with temporal length of 16 frames for both the RGB and optical flow modalities. The two-stream FCAN network has one attentive cross-link layer.

better than the results of fc6+SVM pipeline reported in [26].

Testing. For a fair comparison, we also adopt the same testing scheme in other CNN-based works (*e.g.*, two-stream CNNs [22], temporal segment networks [30]). Given a test video, we sample 25 segments of RGB or flow frames with equal temporal spacing between them. For each segment, we crop the center of a frame to evaluate a model. The final score of the video is computed by averaging the scores across different crops and segments. We find that averaging the last fully connected layer (*i.e.*, fc8) scores always produce better results than the softmax scores.

4.3. Baselines

We compare our two-stream FCAN models with a set of baselines proposed recently [22, 16]. The foremost baselines for our two-stream FCAN models are two-stream C3D. Besides, we also compare our FCAN with the following model:

VideoLSTM [16]. VideoLSTM [16] utilizes a convolutional LSTM to estimate motion-based attention. In contrast, we use convolution layers in a *temporal-CNN* network to provide flow-based attention. The results of VideoLSTM are directly extracted from [16].

4.4. Results and analysis

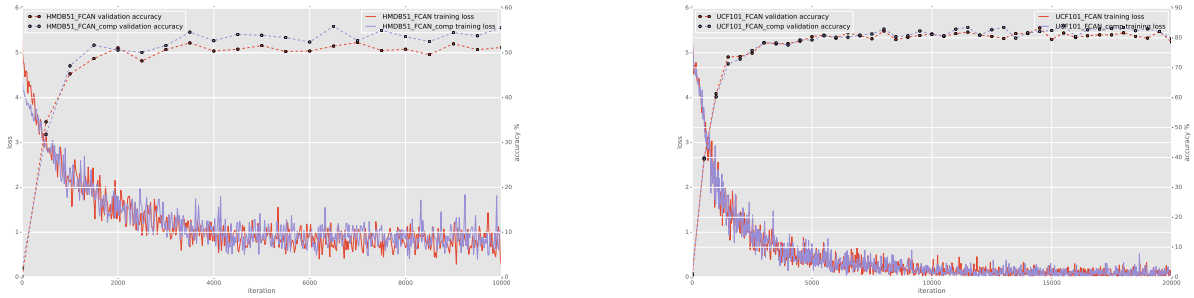
This section reports the performances of our two-stream FCAN models, effects of compensated flows on the FCAN models, and the results of some exploratory studies.

Performance of two-stream FCAN networks. Table 1 shows performance of the two-stream FCAN on three datasets. With compensated flows, our two-stream FCAN demonstrates better performances than two baselines: two-stream C3D and videoLSTM two-stream [16]. In particular, two-stream FCAN-comp (with compensated flows) outperforms two-stream C3D-comp 0.5% on UCF101, 1.1% on HMDB51 and 1.5% on Hollywood2, and the performance gain over the baseline videoLSTM two-stream [16] is much more significant: 2.7% on UCF101 and 11.8% on HMDB51. Focusing just on the FCAN networks (recall from Figure 1 that FCAN is largely the spatial part of our architecture), we too observe consistent improvements over *spatial-C3D* and "videoLSTM RGB" in terms of video-level accuracy. Lastly, it is also evident that motion compensation is important in improving the performance of our FCAN networks. The improvement is more significant in HMDB51 than in UCF101 because many videos in HMDB51 contain more complex camera motions. Only with compensated flows, human foregrounds stand out from the background (*e.g.*, Figure 2). Therefore, attentive effects in FCANs become more substantial. Figure 4 also shows that FCAN models learned on compensated flows have better generalization ability than on normal flows, especially on HMDB51 dataset.

Exploration study. We evaluate the effects of varying the number of cross-link layers from the lower to the higher layers. In particular, we start with just one cross-link layer at layer pool1, and then successively add more cross-link layers until all five layers are connected. Table 2 shows the performance of FCAN when we gradually increase the cross-link layers from the lower to higher layers. In general, the performance of FCAN-comp gradually decreases when the number of cross-link layers increases from one to five

Models	UCF101		HMDB51	
	Clip acc. (%)	Video acc. (%)	Clip acc. (%)	Video acc. (%)
FCAN-comp pool1	82.7	87.2	53.5	56.9
FCAN-comp pool2	82.1	86.3	52.7	57.7
FCAN-comp pool3	81.6	86.1	52.0	57.1
FCAN-comp pool4	79.9	86.2	48.2	52.2
FCAN-comp pool5	78.3	85.0	45.1	49.1

Table 2: Evaluations of FCAN-comp with different numbers of cross-link layers on UCF101 (split 1) and HMDB51 (split 1) dataset. The suffix pool-n means that there are cross-link layers from layer 1 to layer n. 3D convolutional neural networks have inputs with temporal length of 16 frames for both the RGB and optical flow modalities. All flows in this experiment are compensated flows.



(a) HMDB51 (b) UCF101

Figure 4: Training loss and validation accuracy of FCAN and FCAN-comp models on UCF101 and HMDB51 (both split 1) dataset.

layers. FCAN-comp achieves peak performance at the first pooling layer pool1 in UCF101, while its peak performance in the HMDB51 is attained at adding the pooling layer pool2 (*i.e.*, with 87.2% and 57.7% respectively in video accuracy). From the visualizations of the activation maps in Figure 3, it can be seen that those in the higher layers (*e.g.*, pool3, pool4, pool5) are no longer retinotopic, and may not correspond to the human silhouettes. Therefore, creating cross-link at these layers is counter-productive, usually degrading the performance of our classifiers.

Visualization of attention layers. In Figure 3, we provide a visualization of the attention maps provided by the flows in Equ. 3, using two video sequences from the UCF101 and HMDB51 dataset. In the *cartwheel* sequence, the motions of the actor are significant, and our attention maps in the pool1 and pool2 cross-link layers are indeed indicative of the actor's silhouettes. Attention map from the pool3 layer begins to be blurry. At higher layers (*e.g.*, pool4, pool5), the attention maps have more abstract and complex patterns. Similar trends also appear in the *handstand walking* sequence. These observations and the quantitative results in Table 2 corroborate our design choice of having one attentive cross-link layers.

Errors analysis. Now, delving into the performance gain of FCAN over *spatial-C3D*, we find that FCAN has better accuracy in all five action types in UCF101. Specifically, the performance gain is more noticeable on the action types of "human-human interaction", "human-object interaction" and "body-motion only". These action classes mostly tend to be those categories which have significant motions, allowing the compensated optical flows to pick up vividly the human form. In UCF101, some classes gain remarkable performance over *spatial-C3D*, such as JumpingJack (76% vs. 65%), JumpRope (95% vs. 58%), HandstandWalking (44% vs. 29%), HandstandPushups (82% vs. 68%), Lunges (57% vs. 43%), MilitaryParade (94% vs. 82%), WallPushups (77% vs. 60%) and SalsaSpin (98% vs 78%) (see details in Figure 5). FCAN obtains marginal improvements over *spatial-C3D* on "playing musical instruments" because there are not much motions in the video sequences. In some "sports" sequences, FCAN's performance gain over spatial-C3D is also significant, such as Clean&Jerk (97% vs. 85%), CricketBowling (72% vs. 56%), and CricketShot (63% vs. 53%). Scene contexts play important roles in sports sequences, and if the motions are also difficult to be picked up (*e.g.* the swing of a golf club), then the improvement of FCAN over *spatial-C3D* is limited compared to other types of actions.

Models	UCF101		HMDB51	
	Clip acc. (%)	Video acc. (%)	Clip acc. (%)	Video acc. (%)
Twostream-TSN [30] [2]	81.3	91.5	49.1	64.5
Twostream-FCAN-comp	86.7	91.9	55.9	68.2
Twostream-FCAN-comp + Twostream-TSN	88.9	93.4	61.3	70.1

Table 3: Ensemble of TSN-BatchNorm-Inception [30] and FCAN features on UCF101 (split 1) and HMDB51 (split 1) dataset. 3D convolutional neural networks have inputs with temporal length of 16 frames for both RGB and optical flow modalities. All features are combined with equal weights.

Figure 6 presents some examples of flow-guided attention for selected sequences of UCF101 dataset. We show in the top half selected sequences from classes in which our FCAN model outperforms *spatial-C3D*, specifically, JumpingJack, JumpRope, HandstandWalking, HandstandPushups, Lunges, WallPushups, SalsaSpin, Clean&Jerk, CricketBowling, CricketShot. As can be observed, the FCAN model focuses on the spatio-temporally varying human torsos to make predictions. FCAN does eliminate some effects of background context by putting attention values of nearly 0.5 for background regions. We also highlight in the bottom half some cases in which our FCAN does not perform well. In these sequences, the compensated flows are erroneous due to a variety of reasons. For example, in the Front Crawl sequence, there are additional areas of focus in the swimming pool due to wave motions there that are not compensated. Similarly, in the HandstandWalking sequence, there are two distinct planes in the background which causes failure in the homography-based compensation. In the HammerThow and PlayingViolin sequences, the pertinent motions (*e.g.*, hand swing, bow movement) are small and/or elongated and the flow algorithm lacks the quality to clearly delineate these fine motions. Lastly, in the BlowingCandles sequence, 3D-FCAN wrongly focuses on the cake; this is due to the erroneous optical flow estimation caused by the varying candle-light illumination.

Ensemble of FCAN and frame-based CNN models. Table 3 shows the results of our two-stream FCAN-comp and its ensemble with Temporal Segment Networks (TSN) [30]. The latter is essentially a two-stream frame-based CNN model; however, it takes into account frames over longer temporal range (with short snippets randomly sampled from each segment). We re-implement TSN with BatchNorm-Inception architecture for each stream of RGB and compensated flows. With our implementations, our two-stream FCAN-comp out-performs two-stream TSN [30] 0.4% and 3.7% on UCF101 and HMDB51 split 1 respectively. However, when we combine two kinds of features including spatio-temporal models (*i.e.*, two-stream FCAN-comp) and frame-based models (*i.e.*, two-stream TSN), we achieve significantly better performances. Our conjecture is that while our 3D spatio-temporal FCAN model should in principle subsume the TSN (which only randomly samples some snippets), its 3D CNN architectures may have difficulties in learning all the information, and thus there is some complementarity between the two sets of features.

[2]The results are reproduced with our own implementations and data.

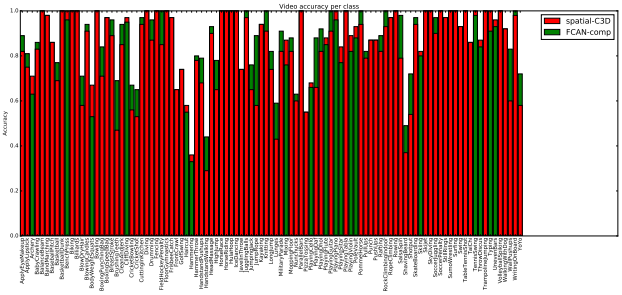


Figure 5: Classwise accuracy of FCAN-comp compared to spatial-C3D model. If a bar has only one color (red), both models have the same performance on the corresponding class. If green is on top of a bar, our FCAN-comp improves the accuracy, and vice versa.

4.5. Comparison with the state of the art

In Table 4, we compare our results to the state-of-the-art on UCF101 and HMDB51 dataset. Different methods are grouped into three categories: hand-crafted features, deep learning approaches, and attention-based networks. Note that most of methods are not directly comparable to our results because of using different network architectures and improvement schemes. Although combining hand-crafted features IDT with Fisher Vector encoding [29] is a strong baseline, our two-stream FCAN comfortably outperforms them by a margin of 6.1% and 9.5% on UCF101 and HMDB51 respectively. We also observe a noticeable improvement over the original two-stream 2D-CNN [22] with 4.0% and 7.3% increase in UCF101 and HMDB51 respectively. We also compare to longer temporal models (*e.g.*, LTC[28], I3D [3]), although they are not directly compara-

Method	UCF101	HMDB	HW2
[29] IDT+FV	85.9	57.2	64.3
[19] IDT+HSV	87.9	61.1	-
[17] IDT+Actionness	-	60.4	-
[8] VideoDarwin	-	63.7	73.7
[7] RankPool + IDT	91.4	66.9	76.7
[22] Two-stream (avg)	86.9	58.0	-
[22] Two-stream (SVM)	88.0	59.4	-
[31] Two-stream LSTM	88.6	-	-
[28] LTC	91.7	64.8	-
[30] TSN (2 modalities)	94.0	68.5	-
[30] TSN (3 modalities)	94.2	69.4	-
[3] Two-stream I3D	**98.0**	**80.7**	-
[6] Two-stream fusion	92.5	65.4	-
[5] ST-ResNet	93.4	66.4	-
[21] Soft attention	-	41.3	-
[16] VideoLSTM	89.2	56.4	-
Two-stream FCAN-comp	92.0	66.7	71.1
Ensemble (4 models)	93.4	68.2	**78.4**

Table 4: Comparison with the state-of-the-art on UCF101, HMDB51 and Hollywood2(HW2) with mean accuracy across 3 splits. We only compare with deep learning approaches with equal length in the temporal models, and not with handcrafted features such as IDT[29]. We would expect our results to be better after combining with IDT features.

ble to our work. Our temporal length is 16 frames, while they are 100 and 64 frames in LTC and I3D respectively. Our results are on par with LTC in the UCF101 dataset, but are better than LTC on HMDB51 (*i.e.*, 66.7% vs. 64.8%). Two-stream I3D [3] achieves astonishing performance since they train a 3D-CNN architecture on big video dataset and fine-tune on UCF101 and HMDB51. We also achieve encouraging results compared to TSN[30] (3 modalities) on UCF101 and HMDB51, although they improve accuracy by using a better 2D-CNN architecture. In the regime of attention-based models, our method shows promising results compared to other related works. First, we outperform VideoLSTM [16] by a margin of 2.8% on UCF101 (*i.e.*, 92.0% vs. 89.2%), of 10.3% on HMDB51 (*i.e.*, 66.7% vs. 56.4%). Furthermore, we also see a large margin of improvement in the performance of our two-stream FCAN model on HMDB51 when compared to that of the soft attention model [21] (*i.e.*, 66.7% vs. 41.3%). We also obtain a new state-of-the-art result on Hollywood2 dataset. We attribute these successes to the explicit temporal modeling in the C3D architectures and the attentive property of the (compensated) flows.

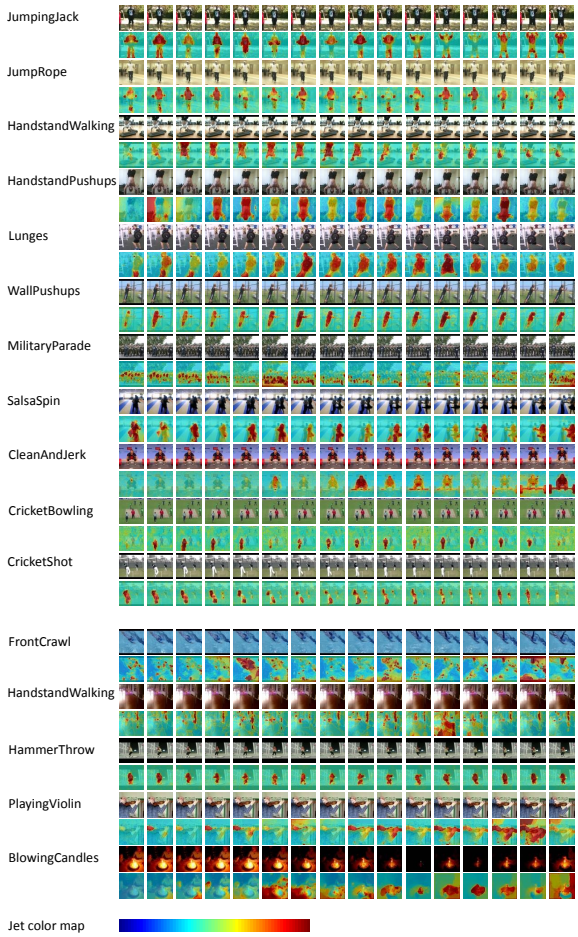


Figure 6: Rows represent examples of attention over time in videos in UCF101 dataset. The top half shows examples from UCF101 of successful classes with large improvements brought about by our FCAN, while the bottom one shows examples of classes with decreases in performance. For each pair of sequences, we show original images and attention maps overlaid on images. The attention map is encoded by jet color map. The intensities are in the range [0,1], and the color scheme looks like the last row. Best viewed in color.

5. Conclusion

This paper introduces two-stream flow-guided convolutional attention network (two-stream FCAN) and shows that it can improve performances of the two-stream C3D. We also show that while compensated optical flows can provide some form of attention guidance, the advantage of this attention is prominent when there is explicit temporal modeling in the CNN model. The attention in our approach is modeled simply, but it shows good performances compared to the recurrent attention models for action recognition.

References

[1] N. Ballas, L. Yao, C. Pal, and A. Courville. Delving Deeper into Convolutional Networks for Learning Video Representations. In *International Conference of Learning Representations (ICLR)*, nov 2016.

[2] L. Bazzani, H. Larochelle, and L. Torresani. Recurrent Mixture Density Network for Spatiotemporal Visual Attention. In *International Conference of Learning Representations (ICLR) 2017*, mar 2017.

[3] J. Carreira and A. Zisserman. Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, may 2017.

[4] J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell. Long-term Recurrent Convolutional Networks for Visual Recognition and Description. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, jun 2015.

[5] C. Feichtenhofer, A. Pinz, and R. P. Wildes. Spatiotemporal Residual Networks for Video Action Recognition. nov 2016.

[6] C. Feichtenhofer, A. Pinz, and A. Zisserman. Convolutional Two-Stream Network Fusion for Video Action Recognition. apr 2016.

[7] B. Fernando, P. Anderson, M. Hutter, and S. Gould. Discriminative Hierarchical Rank Pooling for Activity Recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[8] B. Fernando, E. Gavves, M. Oramas, A. Ghodrati, T. Tuytelaars, K. U. Leuven, J. M. Oramas, A. Ghodrati, and T. Tuytelaars. Modeling Video Evolution for Action Recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, jun 2015.

[9] 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*, jun 2014.

[10] A. Karpathy and T. Leung. Large-scale Video Classification with Convolutional Neural Networks. *Proceedings of 2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1725–1732, 2014.

[11] A. Klaser, M. Marszałek, C. Schmid, A. Kläser, and M. Marszalek. A Spatio-Temporal Descriptor Based on 3D-Gradients. *BMVC*, 2008.

[12] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, pages 1097–1105, 2012.

[13] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre. HMDB: A large video database for human motion recognition. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 2556–2563, 2011.

[14] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8, 2008.

[15] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, nov 1998.

[16] Z. Li, E. Gavves, M. Jain, and C. G. M. Snoek. VideoLSTM Convolves, Attends and Flows for Action Recognition. *ArXive*, jul 2016.

[17] Y. Luo, L.-F. Cheong, and A. Tran. Actionness-assisted Recognition of Actions. In *The IEEE International Conference on Computer Vision (ICCV)*, 2015.

[18] M. Marszalek, I. Laptev, and C. Schmid. Actions in context. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 2929–2936, 2009.

[19] X. Peng, L. Wang, X. Wang, and Y. Qiao. Bag of Visual Words and Fusion Methods for Action Recognition: Comprehensive Study and Good Practice. *arXiv preprint arXiv:1405.4506*, 2014.

[20] F. Perronnin, J. Sánchez, and T. Mensink. Improving the Fisher Kernel for Large-Scale Image Classification. In K. Daniilidis, P. Maragos, and N. Paragios, editors, *Computer Vision ECCV 2010 SE - 11*, volume 6314 of *Lecture Notes in Computer Science*, pages 143–156. Springer Berlin Heidelberg, 2010.

[21] S. Sharma, R. Kiros, and R. Salakhutdinov. Action Recognition using Visual Attention. *arXiv preprint arXiv:1511.04119*, nov 2015.

[22] K. Simonyan and A. Zisserman. Two-Stream Convolutional Networks for Action Recognition in Videos. *arXiv preprint arXiv:1406.2199*, pages 1–11, 2014.

[23] K. Simonyan and A. Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recoginition. In *Intl. Conf. on Learning Representations (ICLR)*, pages 1–14, 2015.

[24] K. Soomro, A. R. Zamir, and M. Shah. UCF101 : A Dataset of 101 Human Actions Classes From Videos in The Wild. Technical Report November, 2012.

[25] W. Sultani and I. Saleemi. Human Action Recognition across Datasets by Foreground-weighted Histogram. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, 2014.

[26] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri. Learning Spatiotemporal Features with 3D Convolutional Networks. In *The IEEE International Conference on Computer Vision (ICCV)*, dec 2015.

[27] D. C. Van Essen and J. H. R. Maunsell. Hierarchical organization and functional streams in the visual cortex. *Trends in Neurosciences*, 6:370–375, 1983.

[28] G. Varol, I. Laptev, and C. Schmid. Long-term Temporal Convolutions for Action Recognition. *arXiv:1604.04494*, apr 2016.

[29] H. Wang and C. Schmid. Action Recognition with Improved Trajectories. In *International Conference on Computer Vision*, Sydney, Australia, oct 2013.

[30] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. Van Gool. Temporal Segment Networks: Towards Good Practices for Deep Action Recognition. In *ECCV 2016 - European Conference on Computer Vision*, aug 2016.

[31] J. Yue-Hei Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici. Beyond Short Snippets: Deep Networks for Video Classification. In *The IEEE*

Conference on Computer Vision and Pattern Recognition (CVPR), jun 2015.

[32] C. Zach, T. Pock, and H. Bischof. A duality based approach for realtime tv-l1 optical flow. In *In Ann. Symp. German Association Patt. Recogn*, pages 214–223, 2007.