Detecting events and key actors in multi-person videos

Vignesh Ramanathan[1], Jonathan Huang[2], Sami Abu-El-Haija[2], Alexander Gorban[2], Kevin Murphy[2], and Li Fei-Fei[1]

[1]Stanford University, [2]Google

vigneshr@cs.stanford.edu*, jonathanhuang@google.com, haija@google.com, gorban@google.com, kpmurphy@google.com, feifeili@cs.stanford.edu

Abstract

Multi-person event recognition is a challenging task, often with many people active in the scene but only a small subset contributing to an actual event. In this paper, we propose a model which learns to detect events in such videos while automatically "attending" to the people responsible for the event. Our model does not use explicit annotations regarding who or where those people are during training and testing. In particular, we track people in videos and use a recurrent neural network (RNN) to represent the track features. We learn time-varying attention weights to combine these features at each time-instant. The attended features are then processed using another RNN for event detection/classification. Since most video datasets with multiple people are restricted to a small number of videos, we also collected a new basketball dataset comprising 257 basketball games with 14K event annotations corresponding to 11 event classes. Our model outperforms state-of-the-art methods for both event classification and detection on this new dataset. Additionally, we show that the attention mechanism is able to consistently localize the relevant players.

1. Introduction

Event recognition and detection in videos has hugely benefited from the introduction of recent large-scale datasets [22, 55, 23, 41, 14] and models. However, this is mainly confined to the domain of single-person actions where the videos contain one actor performing a primary activity. Another equally important problem is event recognition in videos with multiple people. In our work, we present a new model and dataset for this specific setting.

Videos captured in sports arenas, market places or other outdoor areas typically contain multiple people interacting with each other. Most people are doing "something", but not all of them are involved in the main event. The main event is dominated by a smaller subset of people. For instance, a "shot" in basketball is determined by one or two people (see Figure 1). In addition to recognizing the event, it is also important to isolate these key actors. This is a significant challenge which differentiates multi-person videos from single-person videos.

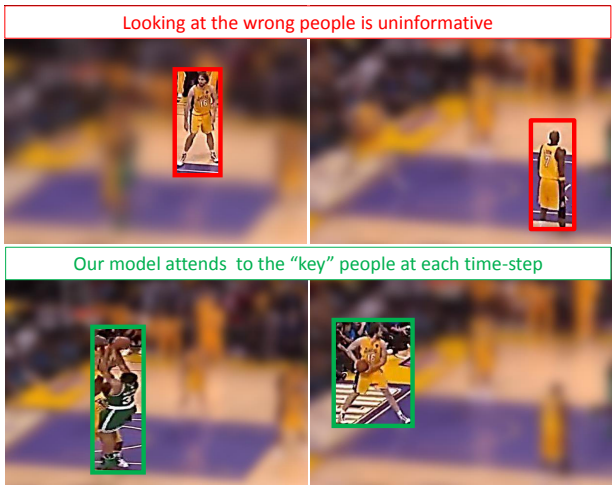

Figure 1. Looking at the wrong people in a multi-person event can be very uninformative as seen in the basketball video in the first row. However, by observing the correct people in the *same video*, we can easily identify the event as a "2-pointer success" based on the shooter and the player throwing the ball into play. We use the same intuition to recognize the key players for event recognition.

Identifying the people responsible for an event is thus an interesting task in its own right. However acquiring such annotations is expensive and it is therefore desirable to use models that do not require annotations for identifying these key actors during training. This can also be viewed as a problem of weakly supervised key person identification. In this paper, we propose a method to classify events by using

*This work was done while Vignesh Ramanathan was an intern at Google

a model that is able to "attend" to this subset of key actors. We do this without ever explicitly telling the model who or where the key actors are.

Recently, several papers have proposed to use "attention" models for aligning elements from a fixed input to a fixed output. For example, [3] translate sentences in one language to another language, attending to different words in the input; [70] generate an image-caption, attending to different regions in the image; and [72] generate a video-caption, attending to different frames within the video.

In our work, we use attention to decide which of several people is most relevant to the action being performed; this attention mask can change over time. Thus we are combining spatial and temporal attention. Note that while the person detections vary from one frame to another, they can be associated across frames through tracking. We show how to use a recurrent neural network (RNN) to represent information from each track; the attention model is tasked with selecting the most relevant track in each frame. In addition to being able to isolate the key actors, we show that our attention model results in better event recognition.

In order to evaluate our method, we need a large number of videos illustrating events involving multiple people. Most prior activity and event recognition datasets focus on actions involving just one or two people. Multi-person datasets like [47, 39, 7] are usually restricted to fewer videos. Therefore we collected our own dataset. In particular we propose a new dataset of basketball events with time-stamp annotations for all occurrences of 11 different events across 257 videos each 1.5 hours long in length. This dataset is comparable to the THUMOS [22] detection dataset in terms of number of annotations, but contains longer videos in a multi-person setting.

In summary, the contributions of our paper are as follows. First, we introduce a new large-scale basketball event dataset with 14K dense temporal annotations for long video sequences. Second, we show that our method outperforms state-of-the-art methods for the standard tasks of classifying isolated clips and of temporally localizing events within longer, untrimmed videos. Third, we show that our method learns to attend to the relevant players, despite never being told which players are relevant in the training set.

2. Related Work

Action recognition in videos Traditionally, well engineered features have proved quite effective for video classification and retrieval tasks [8, 18, 21, 30, 37, 38, 31, 40, 42, 49, 50, 65, 66]. The improved dense trajectory (IDT) features [66] achieve competitive results on standard video datasets. In the last few years, end-to-end trained deep network models [20, 23, 54, 53, 62] were shown to be comparable and at times better than these features for various video tasks. Other works like [68, 71, 74] explore methods for pooling such features for better performance. Recent works using RNN(s) have achieved state-of-the-art results for both event recognition and caption-generation tasks [9, 36, 56, 72]. We follow this line of work with the addition of attention to attend to the event participants.

Another related line of work jointly identifies the region of interest in a video while recognizing the action. Gkioxari et al. [11] and Raptis et al. [45] automatically localize a spatio-temporal tube in a video. Jain et al. [19] merge super-voxels for action localization. Other works like [4, 44] learn to localize actors based on weak annotations from partially aligned movie scripts. While these methods perform weakly-supervised action localization, they target single actor videos in short clips where the action is centered around the actor. Methods like [28, 43, 60, 67] require annotations during training to localize the action.

Multi-person video analysis Activity recognition models for events with well defined group structures such as parades have been presented in [63, 15, 34, 24]. These models utilize the structured layout of participants to identify group events. More recently, [29, 7, 25] use context as a cue for recognizing interaction-based group activities. However, these methods are restricted to smaller datasets [48, 7, 29].

Attention models Itti et al. [17] explored the idea of saliency-based attention in images, with other works like [51] using eye-gaze data as a means for learning attention. Mnih et al. [33] attend to image regions of varying resolutions through an RNN. Attention has also been used for image classification [6, 13, 69] and detection [2, 5, 73].

Bahdanau et al. [3] showed that attention-based RNN models can effectively align input words to output words for machine translation. Following this, Xu et al. [70] and Yao et al. [72] used attention for image-captioning and video-captioning respectively. In all these methods, attention aligns a sequence of input features with words of an output sentence. However, we use attention to identify the most relevant person during different phases of the event.

Action recognition datasets Action recognition in videos has evolved with the introduction of more sophisticated datasets starting from smaller KTH [50], HMDB [27] to larger , UCF101 [55], TRECVID-MED [41] and Sports-1M [23] datasets. More recently, THUMOS [22] and ActivityNet [14] also provide a detection setting with temporal annotations for actions in untrimmed videos. There are also fine-grained datasets in specific domains such as MPII cooking [46] and breakfast [26]. However, most of these datasets focus on single-person activities with hardly any need for recognizing the people responsible for the event. On the other hand, publicly available multi-person activity datasets like [48, 7, 39] are restricted to a very small number of videos. One of the contributions of our work is a multi-player basketball dataset with dense temporal event annotations in long videos.

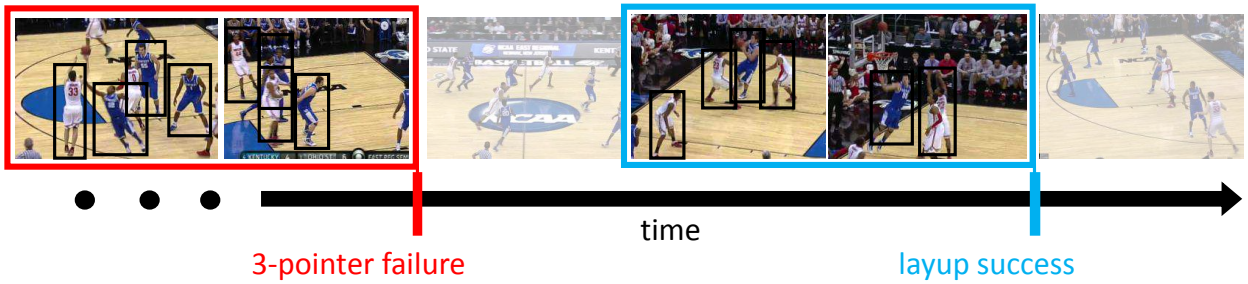

Figure 2. We densely annotate every instance of 11 different basketball events in long basketball videos. As shown here, we collected both event time-stamps and event labels through an AMT task.

Person detection and tracking. There is a very large literature on person detection and tracking. There are also specific methods for tracking players in sports videos [52]. Here we just mention a few key methods. For person detection, we use the CNN-based multibox detector from [59]. For person tracking, we use the KLT tracker from [64]. There is also work on player identification (e.g., [32]), but in this work, we do not attempt to distinguish players.

Event	# Videos Train (Test)	Avg. # people
3-point succ.	895 (188)	8.35
3-point fail.	1934 (401)	8.42
free-throw succ.	552 (94)	7.21
free-throw fail.	344 (41)	7.85
layup succ.	1212 (233)	6.89
layup fail.	1286 (254)	6.97
2-point succ.	1039 (148)	7.74
2-point fail.	2014 (421)	7.97
slam dunk succ.	286 (54)	6.59
slam dunk fail.	47 (5)	6.35
steal	1827 (417)	7.05

Table 1. The number of videos per event in our dataset along with the average number of people per video corresponding to each of the events. The number of people is higher than existing datasets for multi-person event recognition.

3. NCAA Basketball Dataset

A natural choice for collecting multi-person action videos is team sports. In this paper, we focus on basketball games, although our techniques are general purpose. In particular, we use a subset of the 296 NCAA games available from YouTube.[1] These games are played in different venues over different periods of time. We only consider the most recent 257 games, since older games used slightly different rules than modern basketball. The videos are typically 1.5 hours long. We manually identified 11 key event types listed in Tab. 1. In particular, we considered 5 types of shots, each of which could be successful or failed, plus a steal event.

[1]https://www.youtube.com/user/ncaaondemand

Next we launched an Amazon Mechanical Turk task, where the annotators were asked to annotate the "end-point" of these events if and when they occur in the videos; end-points are usually well-defined (e.g., the ball leaves the shooter's hands and lands somewhere else, such as in the basket). To determine the starting time, we assumed that each event was 4 seconds long, since it is hard to get raters to agree on when an event started. This gives us enough temporal context to classify each event, while still being fairly well localized in time.

The videos were randomly split into 212 training, 12 validation and 33 test videos. We split each of these videos into 4 second clips (using the annotation boundaries), and subsampled these to 6fps. We filter out clips which are not profile shots (such as those shown in Figure 3) using a separately trained classifier; this excludes close-up shots of players, as well as shots of the viewers and instant replays. This resulted in a total of 11436 training, 856 validation and 2256 test clips, each of which has one of 11 labels. Note that this is comparable in size to the THUMOS'15 detection challenge (150 trimmed training instances for each of the 20 classes and 6553 untrimmed validation instances). The distribution of annotations across all the different events is shown in Tab. 1. To the best of our knowledge, this is the first dataset with dense temporal annotations for such long video sequences.

In addition to annotating the event label and start/end time, we collected AMT annotations on 850 video clips in the test set, where the annotators were asked to mark the position of the ball on the frame where the shooter attempts a shot.

We also used AMT to annotate the bounding boxes of all the players in a subset of 9000 frames from the training videos. We then trained a Multibox detector [58] with these annotations, and ran the trained detector on all the videos in our dataset. We retained all detections above a confidence of 0.5 per frame; this resulted in 6–8 person detections per clip, as listed in Tab. 1. The multibox model achieves an average overlap of 0.7 at a recall of 0.8 with ground-truth bounding boxes in the validation videos. The dataset is available at: http://basketballattention.appspot.com/.

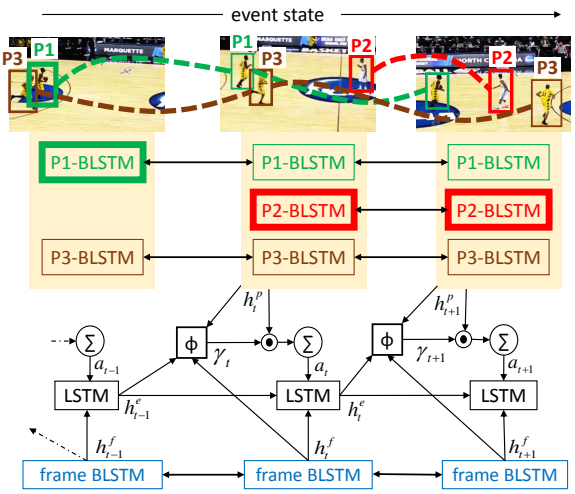

Figure 3. Our model, where each player track is first processed by the corresponding BLSTM network (shown in different colors). Pi-BLSTM corresponds to the i'th player. The BLSTM hidden-states are then used by an attention model to identify the "key" player at each instant. The thickness of the BLSTM boxes shows the attention weights, and the attended person can change with time. The variables in the model are explained in Sec. 4. BLSTM stands for "bidirectional long short term memory".

4. Our Method

All events in a team sport are performed in the same scene by the same set of players. The only basis for differentiating these events is the action performed by a small subset of people at a given time. For instance, a "steal" event in basketball is completely defined by the action of the player attempting to pass the ball and the player stealing from him. To understand such an event, it is sufficient to observe only the players participating in the event.

This motivates us to build a model (overview in Fig. 3) which can reason about an event by focusing on specific people during the different phases of the event. In this section, we describe our unified model for classifying events and simultaneously identifying the key players.

4.1. Feature extraction

Each video-frame is represented by a 1024 dimensional feature vector f_t, which is the activation of the last fully connected layer of the Inception7 network [16, 57]. In addition, we compute spatially localized features for each person in the frame. In particular, we compute a 2805 dimensional feature vector p_{ti} which contains both appearance (1365 dimensional) and spatial information (1440 dimensional) for the i'th player bounding box in frame t. Similar to the RCNN object detector[10], the appearance features were extracted by feeding the cropped and resized player region from the frame through the Inception7 network and spatially pooling the response from a lower layer. The spatial feature corresponds to a 32×32 spatial histogram, combined with a spatial pyramid, to indicate the bounding box location at multiple scales. While we have only used static CNN representations in our work, these features can also be easily extended with flow information as suggested in [53].

4.2. Event classification

Given f_t and p_{ti} for each frame t, our goal is to train the model to classify the clip into one of 11 categories. As a side effect of the way we construct our model, we will also be able to identify the key player in each frame.

First we compute a global context feature for each frame, h^f_t, derived from a bidirectional LSTM applied to the frame-level feature as shown by the blue boxes in Fig. 3. This is a concatenation of the hidden states from the forward and reverse LSTM components of a BLSTM and can be compactly represented as:

$$h^f_t = \text{BLSTM}_{frame}(h^f_{t-1}, h^f_{t+1}, f_t). \quad (1)$$

Please refer to Graves et al. [12].

Next we use a unidirectional LSTM to represent the state of the event at time t:

$$h^e_t = \text{LSTM}(h^e_{t-1}, h^f_t, a_t), \quad (2)$$

where a_t is a feature vector derived from the players, as we describe below. From this, we can predict the class label for the clip using $w_k^\intercal h^e_t$, where the weight vector corresponding to class k is denoted by w_k. We measure the squared-hinge loss as follows:

$$L = \frac{1}{2} \sum_{t=1}^{T} \sum_{k=1}^{K} \max(0, 1 - y_k w_k^\intercal h^e_t)^2, \quad (3)$$

where y_k is 1 if the video belongs to class k, and is -1 otherwise.

4.3. Attention models

Unlike past attention models [3, 70, 72] we need to attend to a different set of features at each time-step. There are two key issues to address in this setting.

First, although we have different detections in each frame, they can be connected across the frames through an object tracking method. This could lead to better feature representation of the players.

Second, player attention depends on the state of the event and needs to evolve with the event. For instance, during the start of a "free-throw" it is important to attend to the player making the shot. However, towards the end of the event the success or failure of the shot can be judged by observing the person in possession of the ball.

With these issues in mind, we first present our model which uses player tracks and learns a BLSTM based representation for each player track. We then also present a simple tracking-free baseline model.

Attention model with tracking. We first associate the detections belonging to the same player into tracks using a standard method. We use a KLT tracker combined with bipartite graph matching [35] to perform the data association.

The player tracks can now be used to incorporate context from adjacent frames while computing their representation. We do this through a separate BLSTM which learns a latent representation for each player at a given time-step. The latent representation of player i in frame t is given by the hidden state h^p_{ti} of the BLSTM across the player-track:

$$h^p_{ti} = \text{BLSTM}_{track}(h^p_{t-1,i}, h^p_{t+1,i}, p_{ti}). \tag{4}$$

At every time-step we want the most relevant player at that instant to be chosen. We achieve this by computing a_t as a convex combination of the player representations at that time-step:

$$\begin{aligned} a^{track}_t &= \sum_{i=1}^{N_t} \gamma^{track}_{ti} h^p_{ti}, \\ \gamma^{track}_{ti} &= \text{softmax}\left(\phi\left(h^f_t, h^p_{ti}, h^e_{t-1}\right); \tau\right), \end{aligned} \tag{5}$$

where N_t is the number of detections in frame t, and $\phi()$ is a multi layer perceptron, similar to [3]. τ is the softmax temperature parameter. This attended player representation is input to the unidirectional event recognition LSTM in Eq. 2. This model is illustrated in Figure 3.

Attention model without tracking. Often, tracking people in a crowded scene can be very difficult due to occlusions and fast movements. In such settings, it is beneficial to have a tracking-free model. This could also allow the model to be more flexible in switching attention between players as the event progresses. Motivated by this, we present a model where the detections in each frame are considered to be independent from other frames.

We compute the (no track) attention based player feature as shown below:

$$\begin{aligned} a^{notrack}_t &= \sum_{i=1}^{N_t} \gamma^{notrack}_{ti} p_{ti}, \\ \gamma^{notrack}_{ti} &= \text{softmax}\left(\phi\left(h^f_t, p_{ti}, h^e_{t-1}\right); \tau\right), \end{aligned} \tag{6}$$

Note that this is similar to the tracking based attention equations except for the direct use of the player detection feature p_{ti} in place of the BLSTM representation h^p_{ti}.

5. Experimental evaluation

In this section, we present three sets of experiments on the NCAA basketball dataset: 1. event classification, 2. event detection and 3. evaluation of attention.

5.1. Implementation details

We used a hidden state dimension of 256 for all the LSTM and BLSTM RNNs, an embedding layer with ReLU non-linearity and 256 dimensions for embedding the player features and frame features before feeding to the RNNs. We used 32×32 bins with spatial pyramid pooling for the player location feature. All the event video clips were four seconds long and subsampled to 6fps. The τ value was set to 0.25 for the attention softmax weighting. We used a batch size of 128, and a learning rate of 0.005 which was reduced by a factor of 0.1 every 10000 iterations with RMSProp [61]. The models were trained on a cluster of 20 GPUs for $100k$ iterations over one day. The hyperparameters were chosen by cross-validating on the validation set.

5.2. Event classification

In this section, we compare the ability of methods to classify isolated video clips into 11 classes. We do not use any additional negatives from other parts of the basketball videos. We compare our results against different control settings and baseline models explained below:

- *IDT[66]*: We use the publicly available implementation of dense trajectories with Fisher encoding.
- *IDT[66] player*: We use IDT along with averaged features extracted from the player bounding boxes.
- *C3D [62]*: We use the publicly available pre-trained model for feature extraction with an SVM classifier.
- *LRCN [9]*: We use an LRCN model with frame-level features. However, we use a BLSTM in place of an LSTM. We found this to improve performance. Also, we do not back-propagate into the CNN extracting the frame-level features to be consistent with our model.
- *MIL [1]*: We use a multi-instance learning method to learn bag (frame) labels from the set of player features.
- *Only player*: We only use our player features from Sec. 4.1 in our model without frame-level features.
- *Avg. player*: We combine the player features by simple averaging, without using attention.
- *Attention no track*: Our model without tracks (Eq. 6).
- *Attention with track*: Our model with tracking (Eq. 5).

The mean average precision (mAP) for each setting is shown in Tab. 2. We see that the method that uses both global information and local player information outperforms the model only using local player information ("Only player") and only using global information ("LRCN"). We also show that combining the player information using a weighted sum (i.e., an attention model) is better than uniform averaging ("Avg. player"), with the tracking based version of attention slightly better than the track-free version. Also, a standard weakly-supervised approach such as

Event	IDT[66]	IDT[66] player	C3D [62]	MIL[1]	LRCN [9]	Only player	Avg. player	Our no track	Our track
3-point succ.	0.370	0.428	0.117	0.237	0.462	0.469	0.545	0.583	**0.600**
3-point fail.	0.501	0.481	0.282	0.335	0.564	0.614	0.702	0.668	**0.738**
fr-throw succ.	0.778	0.703	0.642	0.597	0.876	0.885	0.809	**0.892**	0.882
fr-throw fail.	0.365	0.623	0.319	0.318	0.584	**0.700**	0.641	0.671	0.516
layup succ.	0.283	0.300	0.195	0.257	0.463	0.416	0.472	0.489	**0.500**
layup fail.	0.278	0.311	0.185	0.247	0.386	0.305	0.388	0.426	**0.445**
2-point succ.	0.136	0.233	0.078	0.224	0.257	0.228	0.255	0.281	**0.341**
2-point fail.	0.303	0.285	0.254	0.299	0.378	0.391	**0.473**	0.442	0.471
sl. dunk succ.	0.197	0.171	0.047	0.112	0.285	0.107	0.186	0.210	**0.291**
sl. dunk fail.	0.004	0.010	0.004	0.005	**0.027**	0.006	0.010	0.006	0.004
steal	0.555	0.473	0.303	0.843	0.876	0.843	**0.894**	0.886	0.893
Mean	0.343	0.365	0.221	0.316	0.469	0.452	0.489	0.505	**0.516**

Table 2. Mean average precision for event *classification* given isolated clips.

Event	IDT[66]	IDT player[66]	C3D [62]	LRCN [9]	Only player	Avg. player	Attn no track	Attn track
3-point succ.	0.194	0.203	0.123	0.230	0.251	**0.268**	0.263	0.239
3-point fail.	0.393	0.376	0.311	0.505	0.526	0.521	0.556	**0.600**
free-throw succ.	0.585	0.621	0.542	0.741	0.777	**0.811**	0.788	0.810
free-throw fail.	0.231	0.277	0.458	0.434	**0.470**	0.444	0.468	0.405
layup succ.	0.258	0.290	0.175	0.492	0.402	0.489	0.494	**0.512**
layup fail.	0.141	0.200	0.151	0.187	0.142	0.139	0.207	**0.208**
2-point succ.	0.161	0.170	0.126	0.352	0.371	**0.417**	0.366	0.400
2-point fail.	0.358	0.339	0.226	0.544	0.578	**0.684**	0.619	0.674
slam dunk succ.	0.137	0.275	0.114	0.428	0.566	0.457	0.576	**0.555**
slam dunk fail.	0.007	0.006	0.003	**0.122**	0.059	0.009	0.005	0.045
steal	0.242	0.255	0.187	**0.359**	0.348	0.313	0.340	0.339
Mean	0.246	0.273	0.219	0.400	0.408	0.414	0.426	**0.435**

Table 3. Mean average precision for event *detection* given untrimmed videos.

MIL seems to be less effective than any of our modeling variants.

The performance varies by class. In particular, performance is much poorer (for all methods) for classes such as "slam dunk fail" for which we have very little data. However, performance is better for shot-based events like "free-throw", "layups" and "3-pointers"where attending to the shot making person or defenders can be useful.

5.3. Event detection

In this section, we evaluate the ability of methods to temporally localize events in untrimmed videos. We use a sliding window approach, where we slide a 4 second window through all the basketball videos and try to classify the window into a negative class or one of the 11 event classes. We use a stride length of 2 seconds. We treat all windows which do not overlap more than 1 second with any of the 11 annotated events as negatives. We use the same setting for training, test and validation. This leads to 90200 negative examples across all the videos. We compare with the same baselines as before. However, we were unable to train the MIL model due to computational limitations.

The detection results are presented in Tab. 3. We see that, as before, the attention models beat previous state-of-the-art methods. Not surprisingly, all methods are slightly worse at temporal localization than for classifying isolated clips. We also note a significant difference in classification and detection performance for "steal" in all methods. This can be explained by the large number of negative instances introduced in the detection setting. These negatives often correspond to players passing the ball to each other. The "steal" event is quite similar to a "pass" except that the ball is passed to a player of the opposing team. This makes the "steal" detection task considerably more challenging.

5.4. Analyzing attention

We have seen above that attention can improve the performance of the model at tasks such as classification and detection. Now, we evaluate how accurate the attention models are at identifying the key players. (Note that the models were never explicitly trained to identify key players).

To evaluate the attention models, we labeled the player who was closest (in image space) to the ball as the "shooter". (The ball location is annotated in 850 test clips.)

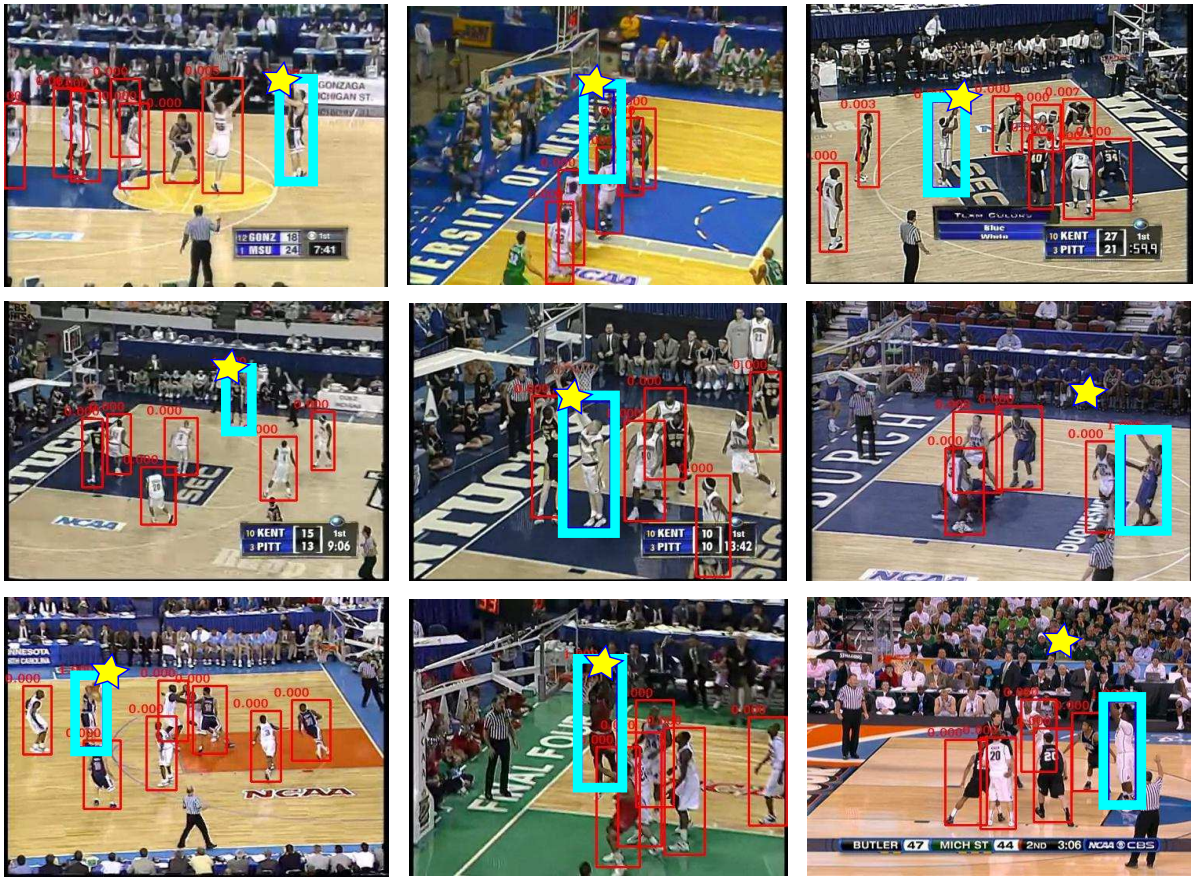

Figure 4. We highlight (in cyan) the "attended" player at the beginning of different events. The position of the ball in each frame is shown in yellow. Each column shows a different event. In these videos, the model attends to the person making the shot at the start of the event.

We used these annotations to evaluate if our "attention" scores were capable of classifying the "shooter" correctly.

Event	Chance	Attn. with track	Attn. no track
3-point succ.	0.333	0.445	0.519
3-point fail.	0.334	0.391	0.545
free-throw succ.	0.376	0.416	0.772
free-throw fail.	0.346	0.387	0.685
layup succ.	0.386	0.605	0.627
layup fail.	0.382	0.508	0.605
2-point succ.	0.355	0.459	0.554
2-point fail.	0.346	0.475	0.542
slam dunk succ.	0.413	0.347	0.686
slam dunk fail.	0.499	0.349	0.645
Mean	0.377	0.438	0.618

Table 4. Mean average precision for attention evaluation.

The mean AP for this "shooter" classification is listed in Tab. 4. The results show that the track-free attention model is quite consistent in picking the shooter for several classes like "free-throw succ./fail", "layup succ./fail." and "slam dunk succ.". This is a promising result which shows that attention on player detections is capable of localizing the player making the shot. This could be useful for providing more detailed descriptions including the shooter identity.

We also visualize the attention masks visually for sample videos in Figure 4. In order to make results comparable across frames, we annotated 5 points on the court and aligned all the attended boxes for an event to one canonical image. Fig. 5 shows the resulting heatmap of spatial distributions of the attended players with respect to the court. It is interesting to note that our model consistently focuses under the basket for a layup, at the free-throw line for free-throws and outside the 3-point ring for 3-pointers.

Another interesting observation is that the attention for the tracking based model is less selective in focusing on the shooter. We observed that the tracking model is often reluctant to switch attention between frames and focuses on a single player throughout the event. This biases it towards players present throughout the video. For instance, in free-throws (Fig. 6) it attends to the defender at a specific position, who is visible throughout the event unlike the shooter.

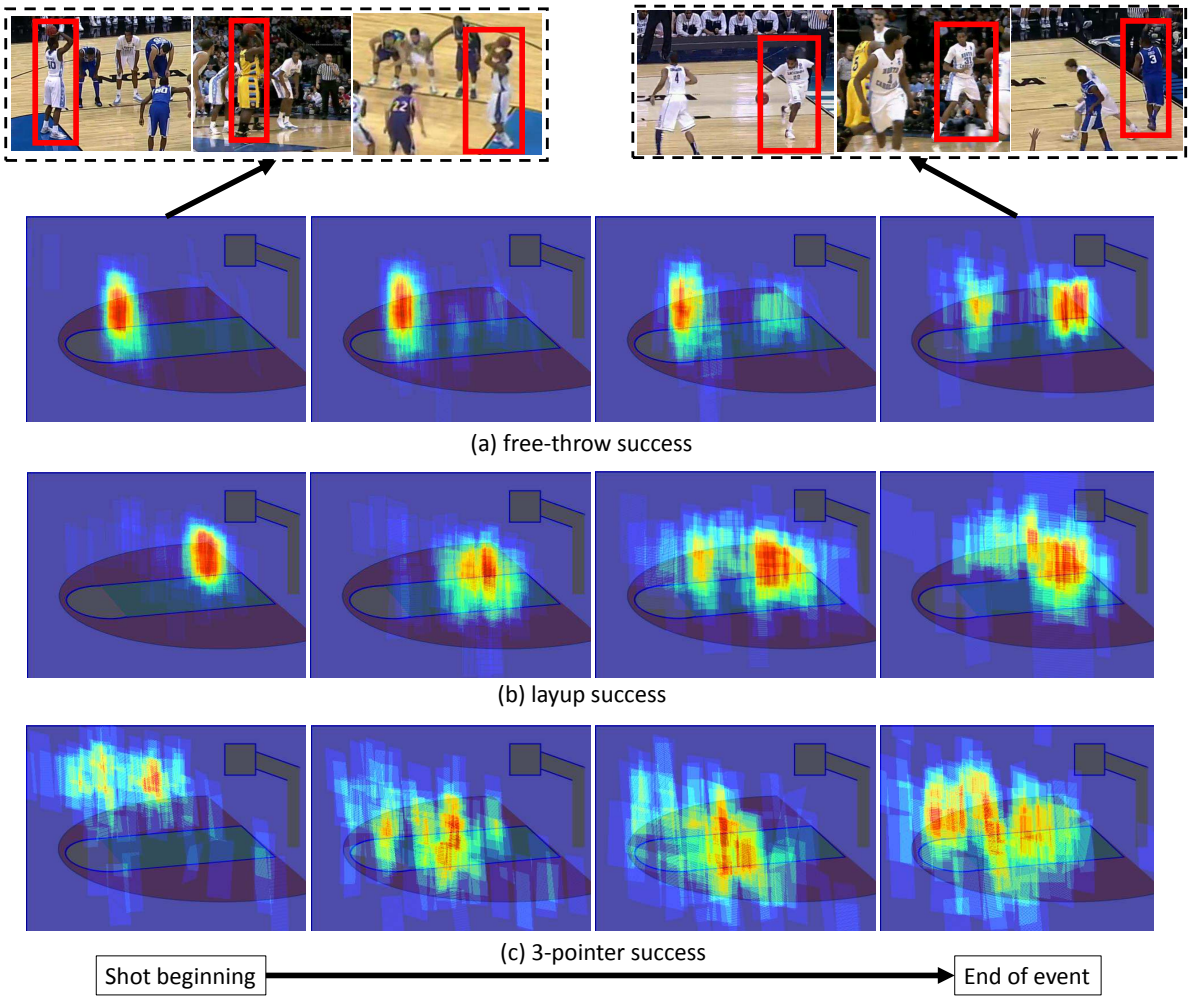

Figure 5. We visualize the distribution of attention (from model without tracks) over different positions of a basketball court as the event progresses. This is shown for 3 different events. These heatmaps were obtained by first transforming all videos to a canonical view of the court (shown in the background of each heatmap). The top row shows the sample frames which contributed to the "free-throw" success heatmaps. The model focuses on the location of the shooter at the beginning of an event and later the attention disperses to other locations.

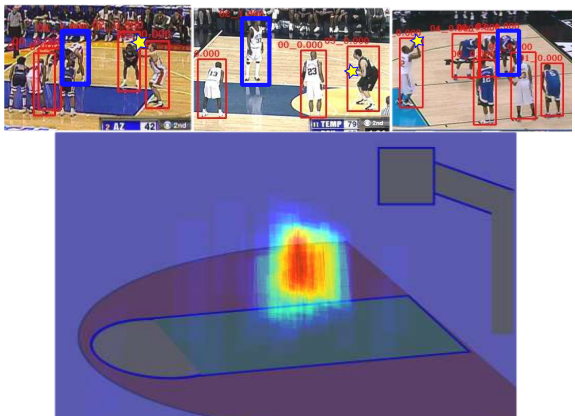

Figure 6. The distribution of attention for our model with tracking, at the beginning of "free-throw success". Unlike Fig. 5, the attention is concentrated at a specific defender's position. Free-throws have a distinctive defense formation, and observing the defenders can be helpful as shown in the sample images in the top row.

6. Conclusion

We have introduced a new attention based model for event classification and detection in multi-person videos. Apart from recognizing the event, our model can identify the key people responsible for the event without being explicitly trained with such annotations. Our method can generalize to any multi-person setting. However, for the purpose of this paper we introduced a new dataset of basketball videos with dense event annotations and compared our performance with state-of-the-art methods. We also evaluated our model's ability to recognize the "shooter" in the events and visualized the spatial locations attended by our model.

Acknowledgements

We thank A. Korattikara, V. Rathod and K. Tang for useful comments. We also thank O. Camburu and N. Johnston for helping with the GPU implementation. This research is partly supported by ONR MURI and Intel ISTC-PC.

References

[1] S. Andrews, I. Tsochantaridis, and T. Hofmann. Support vector machines for multiple-instance learning. In *Advances in neural information processing systems*, pages 561–568, 2002. 5, 6

[2] J. Ba, V. Mnih, and K. Kavukcuoglu. Multiple object recognition with visual attention. *arXiv preprint arXiv:1412.7755*, 2014. 2

[3] D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014. 2, 4, 5

[4] P. Bojanowski, F. Bach, I. Laptev, J. Ponce, C. Schmid, and J. Sivic. Finding actors and actions in movies. In *ICCV*, 2013. 2

[5] J. C. Caicedo, F. U. K. Lorenz, and S. Lazebnik. Active object localization with deep reinforcement learning. *ICCV*, 2015. 2

[6] C. Cao, X. Liu, Y. Yang, Y. Yu, J. Wang, Z. Wang, Y. Huang, L. Wang, C. Huang, W. Xu, et al. Look and think twice: Capturing top-down visual attention with feedback convolutional neural networks. *ICCV*, 2015. 2

[7] W. Choi, K. Shahid, and S. Savarese. What are they doing?: Collective activity classification using spatio-temporal relationship among people. In *Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on*, pages 1282–1289. IEEE, 2009. 2

[8] N. Dalal et al. Human detection using oriented histograms of flow and appearance. In *ECCV*, 2006. 2

[9] 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. *arXiv preprint arXiv:1411.4389*, 2014. 2, 5, 6

[10] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 580–587. IEEE, 2014. 4

[11] G. Gkioxari and J. Malik. Finding action tubes. *arXiv preprint arXiv:1411.6031*, 2014. 2

[12] A. Graves, N. Jaitly, and A.-r. Mohamed. Hybrid speech recognition with deep bidirectional lstm. In *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*, pages 273–278. IEEE, 2013. 4

[13] K. Gregor, I. Danihelka, A. Graves, and D. Wierstra. Draw: A recurrent neural network for image generation. *arXiv preprint arXiv:1502.04623*, 2015. 2

[14] F. C. Heilbron, V. Escorcia, B. Ghanem, and J. C. Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *CVPR*, 2015. 1, 2

[15] S. S. Intille and A. F. Bobick. Recognizing planned, multiperson action. *Computer Vision and Image Understanding*, 81(3):414–445, 2001. 2

[16] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015. 4

[17] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, pages 1254–1259, 1998. 2

[18] M. Jain, H. Jégou, and P. Bouthemy. Better exploiting motion for better action recognition. In *CVPR*, 2013. 2

[19] M. Jain, J. Van Gemert, H. Jégou, P. Bouthemy, and C. G. Snoek. Action localization with tubelets from motion. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 740–747. IEEE, 2014. 2

[20] S. Ji, W. Xu, M. Yang, and K. Yu. 3d convolutional neural networks for human action recognition. *T-PAMI*, 2013. 2

[21] Y.-G. Jiang et al. Trajectory-based modeling of human actions with motion reference points. In *ECCV*, 2012. 2

[22] Y.-G. Jiang, J. Liu, A. Roshan Zamir, G. Toderici, I. Laptev, M. Shah, and R. Sukthankar. THUMOS challenge: Action recognition with a large number of classes, 2014. 1, 2

[23] A. Karpathy et al. Large-scale video classification with convolutional neural networks. In *CVPR*, 2014. 1, 2

[24] S. M. Khan and M. Shah. Detecting group activities using rigidity of formation. In *Proceedings of the 13th annual ACM international conference on Multimedia*, pages 403–406. ACM, 2005. 2

[25] M. Khodabandeh, A. Vahdat, G.-T. Zhou, H. Hajimirsadeghi, M. J. Roshtkhari, G. Mori, and S. Se. Discovering human interactions in videos with limited data labeling. *arXiv preprint arXiv:1502.03851*, 2015. 2

[26] H. Kuehne, A. Arslan, and T. Serre. The language of actions: Recovering the syntax and semantics of goal-directed human activities. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 780–787. IEEE, 2014. 2

[27] H. Kuehne et al. Hmdb: a large video database for human motion recognition. In *ICCV*, 2011. 2

[28] T. Lan, Y. Wang, and G. Mori. Discriminative figure-centric models for joint action localization and recognition. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 2003–2010. IEEE, 2011. 2

[29] T. Lan, Y. Wang, W. Yang, S. N. Robinovitch, and G. Mori. Discriminative latent models for recognizing contextual group activities. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 34(8):1549–1562, 2012. 2

[30] I. Laptev et al. Learning realistic human actions from movies. In *CVPR*, 2008. 2

[31] B. Laxton, J. Lim, and D. Kriegman. Leveraging temporal, contextual and ordering constraints for recognizing complex activities in video. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8. IEEE, 2007. 2

[32] W.-L. Lu, J.-A. Ting, J. J. Little, and K. P. Murphy. Learning to track and identify players from broadcast sports videos. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(7):1704–1716, July 2013. 3

[33] V. Mnih, N. Heess, A. Graves, et al. Recurrent models of visual attention. In *Advances in Neural Information Processing Systems*, pages 2204–2212, 2014. 2

[34] D. Moore and I. Essa. Recognizing multitasked activities from video using stochastic context-free grammar. In *AAAI*, 2002. 2

[35] J. Munkres. Algorithms for the assignment and transportation problems. *Journal of the Society of Industrial and Applied Mathematics*, 5(1):32–38, March 1957. 5

[36] J. Y.-H. Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici. Beyond short snippets: Deep networks for video classification. *arXiv preprint arXiv:1503.08909*, 2015. 2

[37] J. C. Niebles, C.-W. Chen, and L. Fei-Fei. Modeling temporal structure of decomposable motion segments for activity classification. In *ECCV*, 2010. 2

[38] S. Oh et al. Multimedia event detection with multimodal feature fusion and temporal concept localization. *Machine Vision and Applications*, 25, 2014. 2

[39] S. Oh, A. Hoogs, A. Perera, N. Cuntoor, C.-C. Chen, J. T. Lee, S. Mukherjee, J. Aggarwal, H. Lee, L. Davis, et al. A large-scale benchmark dataset for event recognition in surveillance video. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 3153–3160. IEEE, 2011. 2

[40] D. Oneata, J. Verbeek, and C. Schmid. Action and event recognition with fisher vectors on a compact feature set. In *ICCV*, 2013. 2

[41] P. Over et al. An overview of the goals, tasks, data, evaluation mechanisms and metrics. In *TRECVID*, 2014. 1, 2

[42] X. Peng, C. Zou, Y. Qiao, and Q. Peng. Action recognition with stacked fisher vectors. In *ECCV*, 2014. 2

[43] A. Prest, V. Ferrari, and C. Schmid. Explicit modeling of human-object interactions in realistic videos. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(4):835–848, 2013. 2

[44] V. Ramanathan, A. Joulin, P. Liang, and L. Fei-Fei. Linking people with "their" names using coreference resolution. In *European Conference on Computer Vision (ECCV)*, 2014. 2

[45] M. Raptis, I. Kokkinos, and S. Soatto. Discovering discriminative action parts from mid-level video representations. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1242–1249. IEEE, 2012. 2

[46] M. Rohrbach, S. Amin, M. Andriluka, and B. Schiele. A database for fine grained activity detection of cooking activities. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1194–1201. IEEE, 2012. 2

[47] M. S. Ryoo and J. K. Aggarwal. Spatio-temporal relationship match: Video structure comparison for recognition of complex human activities. In *Computer vision, 2009 ieee 12th international conference on*, pages 1593–1600. IEEE, 2009. 2

[48] M. S. Ryoo and J. K. Aggarwal. UT-Interaction Dataset, ICPR contest on Semantic Description of Human Activities (SDHA). http://cvrc.ece.utexas.edu/SDHA2010/Human_Interaction.html, 2010. 2

[49] S. Sadanand and J. J. Corso. Action bank: A high-level representation of activity in video. In *CVPR*, 2012. 2

[50] C. Schüldt, I. Laptev, and B. Caputo. Recognizing human actions: a local svm approach. In *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, volume 3, pages 32–36. IEEE, 2004. 2

[51] N. Shapovalova, M. Raptis, L. Sigal, and G. Mori. Action is in the eye of the beholder: eye-gaze driven model for spatio-temporal action localization. In *Advances in Neural Information Processing Systems*, pages 2409–2417, 2013. 2

[52] H. B. Shitrit, J. Berclaz, F. Fleuret, , and P. Fua. Tracking Multiple People under Global Appearance Constraints. *International Conference on Computer Vision*, 2011. 3

[53] K. Simonyan and A. Zisserman. Two-stream convolutional networks for action recognition in videos. In *Advances in Neural Information Processing Systems*, pages 568–576, 2014. 2, 4

[54] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv:1409.1556*, 2014. 2

[55] 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. 1, 2

[56] N. Srivastava, E. Mansimov, and R. Salakhutdinov. Unsupervised learning of video representations using lstms. *arXiv:1502.04681*, 2015. 2

[57] C. Szegedy et al. Scene classification with inception-7. http://lsun.cs.princeton.edu/slides/Christian.pdf, 2015. 4

[58] C. Szegedy, S. Reed, D. Erhan, and D. Anguelov. Scalable, high-quality object detection. *arXiv preprint arXiv:1412.1441*, 2014. 3

[59] C. Szegedy, A. Toshev, and D. Erhan. Deep neural networks for object detection. In *NIPS*, 2013. 3

[60] Y. Tian, R. Sukthankar, and M. Shah. Spatiotemporal deformable part models for action detection. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 2642–2649. IEEE, 2013. 2

[61] T. Tieleman and G. Hinton. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. coursera: Neural networks for machine learning, 2012. 5

[62] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri. C3d: generic features for video analysis. *arXiv preprint arXiv:1412.0767*, 2014. 2, 5, 6

[63] N. Vaswani, A. R. Chowdhury, and R. Chellappa. Activity recognition using the dynamics of the configuration of interacting objects. In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, volume 2, pages II–633. IEEE, 2003. 2

[64] C. J. Veenman, M. J. Reinders, and E. Backer. Resolving motion correspondence for densely moving points. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 23(1):54–72, 2001. 3

[65] H. Wang et al. Evaluation of local spatio-temporal features for action recognition. In *BMVC*, 2009. 2

[66] H. Wang et al. Action Recognition by Dense Trajectories. In *CVPR*, 2011. 2, 5, 6

[67] L. Wang, Y. Qiao, and X. Tang. Video action detection with relational dynamic-poselets. In *Computer Vision–ECCV 2014*, pages 565–580. Springer, 2014. 2

[68] P. Wang, Y. Cao, C. Shen, L. Liu, and H. T. Shen. Temporal pyramid pooling based convolutional neural networks for action recognition. *arXiv preprint arXiv:1503.01224*, 2015. 2

[69] T. Xiao, Y. Xu, K. Yang, J. Zhang, Y. Peng, and Z. Zhang. The application of two-level attention models in deep convolutional neural network for fine-grained image classification. *arXiv preprint arXiv:1411.6447*, 2014. 2

[70] K. Xu, J. Ba, R. Kiros, A. Courville, R. Salakhutdinov, R. Zemel, and Y. Bengio. Show, attend and tell: Neural image caption generation with visual attention. *arXiv preprint arXiv:1502.03044*, 2015. 2, 4

[71] Z. Xu, Y. Yang, and A. G. Hauptmann. A discriminative cnn video representation for event detection. *arXiv:1411.4006v1*, 2015. 2

[72] L. Yao, A. Torabi, K. Cho, N. Ballas, C. Pal, H. Larochelle, and A. Courville. Describing videos by exploiting temporal structure. *stat*, 1050:25, 2015. 2, 4

[73] D. Yoo, S. Park, J.-Y. Lee, A. Paek, and I. S. Kweon. Attentionnet: Aggregating weak directions for accurate object detection. *arXiv preprint arXiv:1506.07704*, 2015. 2

[74] S. Zha, F. Luisier, W. Andrews, N. Srivastava, and R. Salakhutdinov. Exploiting image-trained cnn architectures for unconstrained video classification. *arXiv:1503.04144v2*, 2015. 2