Int J Comput Vis (2018) 126:375–389
https://doi.org/10.1007/s11263-017-1013-y

Every Moment Counts: Dense Detailed Labeling of Actions in Complex Videos

Serena Yeung[1] · **Olga Russakovsky**[1,2] · **Ning Jin**[1] · **Mykhaylo Andriluka**[1,3] · **Greg Mori**[4] · **Li Fei-Fei**[1]

Received: 8 March 2016 / Accepted: 17 April 2017 / Published online: 22 May 2017
© Springer Science+Business Media New York 2017

Abstract Every moment counts in action recognition. A comprehensive understanding of human activity in video requires labeling every frame according to the actions occurring, placing multiple labels densely over a video sequence. To study this problem we extend the existing THUMOS dataset and introduce MultiTHUMOS, a new dataset of dense labels over unconstrained internet videos. Modeling multiple, dense labels benefits from temporal relations within and across classes. We define a novel variant of long short-term memory deep networks for modeling these temporal relations via multiple input and output connections. We show that this model improves action labeling accuracy and further enables deeper understanding tasks ranging from structured retrieval to action prediction.

1 Introduction

Humans are great at multi-tasking: they can be walking while talking on the phone while holding a cup of coffee. Further, human action is continual, and every minute is filled with potential labeled actions (Fig. 1). However, most work on human action recognition in video focuses on recognizing discrete instances or single actions at a time: for example, which sport Karpathy et al. (2014) or which single cooking activity Rohrbach et al. (2012) is taking place. We argue this setup is fundamentally limiting. First, a single description is often insufficient to fully describe a person's activity. Second, operating in a single-action regime largely ignores the intuition that actions are intricately connected. A person that is running and then jumping is likely to be simultaneously doing a sport such as basketball or long jump; a nurse that is taking a patient's blood pressure and looking worried is likely to call a doctor as her next action. In this work, we go beyond the standard one-label paradigm to dense, detailed, multilabel understanding of human actions in videos.

There are two key steps on the path to tackling detailed multilabel human action understanding: (1) finding the right dataset and (2) developing an appropriate model. In this paper we present work in both dimensions.

The desiderata for a video dataset include the following: video clips need to be long enough to capture multiple consecutive actions, multiple simultaneous actions need to be annotated, and labeling must be dense with thorough coverage of action extents. Video annotation is very time-consuming and expensive, and to the best of our knowledge no such dataset currently exists. UCF101 (Soomro et al. 2012), HMDB51 (Kuehne et al. 2011), and Sports1M (Karpathy et al. 2014) are common challenging action recognition datasets. However, each video is associated with non-localized labels (Sports1M), and the videos in UCF101 and HMDB51 are further temporally clipped around the action. MPII Cooking (Rohrbach et al. 2012) and Breakfast (Kuehne et al. 2014) datasets contain long untrimmed video sequences with multiple sequential actions but still only one label per frame; further, they are restricted to closed-world kitchen environments. THUMOS (Jiang et al. 2014) contains long untrimmed videos but most videos (85%) only contain a single action class.

Communicated by Ivan Laptev, Cordelia Schmid.

✉ Serena Yeung
serena@cs.stanford.edu

1 Stanford University, Stanford, CA, USA

2 Carnegie Mellon University, Pittsburgh, PA, USA

3 Max Planck Institute for Informatics, Saarbrücken, Germany

4 Simon Fraser University, Burnaby, BC, Canada

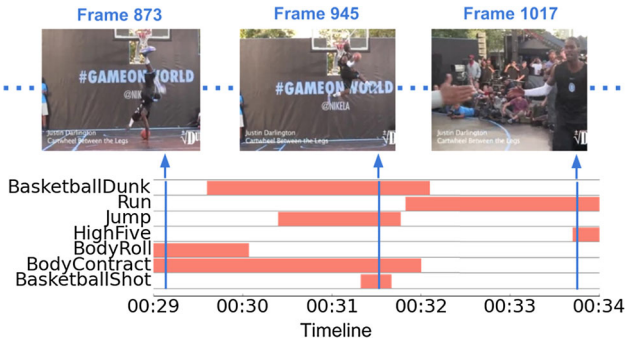

Fig. 1 In most internet videos there are multiple simultaneous human actions. Here, we show a concrete example from a basketball video to illustrate our target problem of dense detailed multi-label action understanding

To overcome these problems, we introduce a new action detection dataset called MultiTHUMOS, significantly extending the annotations on 413 videos (30 h) of THUMOS action detection dataset. First, MultiTHUMOS allows for an in-depth study of simultaneous human action in video: it extends THUMOS from 20 action classes with 0.3 labels per frame to 65 classes and 1.5 labels per frame. Second, MultiTHUMOS allows for a thorough study of the temporal interaction between consecutive actions: the average number of distinct action categories in a video is 10.5 (compared to 1.1 in THUMOS). Going further, MultiTHUMOS lends itself to studying intricate relationships between action labels: the 45 new annotated classes include relationships such as hierarchical (e.g., more general Throw or PoleVault and more specific Basketball Shot or PoleVaultPlantPole) and fine-grained (e.g., Guard vs. Block or Dribble vs. Pass in basketball). Figure 1 shows an example of our dense multilabel annotation.

Reasoning about multiple, dense labels on video requires models capable of incorporating temporal dependencies. A large set of techniques exist for modeling temporal structure, such as hidden Markov models (HMMs), dynamic time warping, and their variants. Recent action recognition literature has used recurrent neural networks known as long short term memory (LSTM) for action recognition in videos (Donahue et al. 2014). We introduce MultiLSTM, a new LSTM-based model targeting dense, multilabel action analysis. Taking advantage of the fact that more than 45% of frames in MultiTHUMOS have 2 or more labels, the model can learn dependencies between actions in nearby frames and between actions in the same frame, which allows it to subsequently perform dense multilabel temporal action detection on unseen videos.

In summary, our contributions are:

1. We introduce MultiTHUMOS, a new large-scale dataset of dense, multilabel action annotations in temporally untrimmed videos, and
2. We introduce MultiLSTM, a new recurrent model based on an LSTM that features temporally-extended input and output connections.

Our experiments demonstrate improved performance of MultiLSTM relative to a plain LSTM baseline on our dense, multilabel action detection benchmark.

2 Related Work

Visual analysis of human activity has a long history in computer vision research. Thorough surveys of the literature include Poppe (2010) and Weinland et al. (2010). Here we review recent work relevant to dense labeling of videos.

2.1 Datasets

Research focus is closely intertwined with dataset creation and availability. The KTH (Schuldt et al. 2004) and Weizmann (Blank et al. 2005) datasets were catalysts for a body of work. This era focused on recognizing individual human actions, based on datasets consisting of an individual human imaged against a generally stationary background. In subsequent years, the attention of the community moved towards more challenging tasks. Benchmarks based on surveillance video were developed for crowded scenes, such as the TRECVID Surveillance Event Detection (Over et al. 2011). Interactions between humans or humans and objects (Ryoo and Aggarwal 2009; Oh et al. 2011) have been studied.

Another line of work has shifted toward analyzing "unconstrained" internet video. Datasets in this line present challenges in the level of background clutter present in the videos. The Hollywood (Marszałek et al. 2009), HMDB (Kuehne et al. 2011), UCF 101 (Soomro et al. 2012), ActivityNet (Fabian Caba Heilbron et al. 2015), and THUMOS (Jiang et al. 2014) datasets exemplify this trend. Task direction has also moved toward a retrieval setting, finding a (small) set of videos from a large background collection, including datasets such as TRECVID MED (Over et al. 2011) and Sports 1M (Karpathy et al. 2014).

While the push toward unconstrained internet video is positive in terms of the difficulty of this task, it has moved focus away from human action toward identifying scene context. Discriminating diving versus gymnastics largely involves determining the scene of the event. The MPII Cooking dataset (Rohrbach et al. 2012) and Breakfast dataset (Kuehne et al. 2014) refocus efforts toward human action within restricted action domains (Table 1). The MultiTHUMOS dataset we propose shares commonalities with this line, but emphasizes generality of video, multiple labels per frame, and a broad set of general to specific actions.

Table 1 Our MultiTHUMOS dataset overcomes many limitations of previous datasets

	Detection	Untrimmed	Open-world	Multilabel
UCF101 (Soomro et al. 2012)	–	–	Yes	–
HMDB51 (Kuehne et al. 2011)	–	–	Yes	–
Sports1M (Karpathy et al. 2014)	–	Yes	Yes	–
Cooking (Rohrbach et al. 2012)	Yes	Yes	–	–
Breakfast (Kuehne et al. 2014)	Yes	Yes	–	–
THUMOS (Jiang et al. 2014)	Yes	Yes	Yes	–
MultiTHUMOS	Yes	Yes	Yes	Yes

2.2 Deep Learning for Video

In common with object recognition, hand-crafted features for video analysis are giving way to deep convolutional feature learning strategies. The best hand-crafted features, the dense trajectories of Wang et al. (2011), achieve excellent results on benchmark action recognition datasets. However, recent work has shown superior results by learning video features (often combined with dense trajectories). Simonyan and Zisserman (2014a) present a two-stream convolutional architecture utilizing both image and optical flow data as input sources. Zha et al. (2015) examine aggregation strategies for combining deep learned image-based features for each frame, obtaining impressive results on TRECVID MED retrieval. Karpathy et al. (2014) and Tran et al. (2015) learn spatio-temporal filters in a deep network and apply them to a variety of human action understanding tasks. Mansimov et al. (2015) consider methods for incorporating ImageNet training data to assist in initializing model parameters for learning spatio-temporal features. Wang et al. (2015) study temporal pooling strategies, specifically focused on classification in variable-length input videos.

2.3 Temporal Models for Video

Constructing models of the temporal evolution of actions has deep roots in the literature. Early work includes Yamato et al. (1992), using hidden Markov models (HMMs) for latent action state spaces. Lv and Nevatia (2007) represented actions as a sequence of synthetic 2D human poses rendered from different view points. Constraints on transitions between key poses are represented using a state diagram called an "Action Net" which is constructed based on the order of key poses of an action. Shi and Cheng (2011) proposes a semi-Markov model to segment a sequence temporally and label segments with an action class. Tang et al. (2012) extend HMMs to model the duration of each hidden state in addition to the transition parameters of hidden states.

Temporal feature aggregation is another common strategy for handling video data. Pooling models include aggregating over space and time, early and late fusion strategies, and temporal localization (Tong et al. 2014; Myers et al. 2014; Oh et al. 2014).

Discriminative models include those based on latent SVMs over key poses and action grammars (Niebles et al. 2010; Vahdat et al. 2011; Pirsiavash and Ramanan 2014). A recent set of papers has deployed deep models using LSTM models (Hochreiter and Schmidhuber 1997) for video analysis (Donahue et al. 2014; Ng et al. 2015; Srivastava et al. 2015; Yao et al. 2015). These papers have shown promising results applying LSTMs for tasks including video classification and sentence generation. In contrast, we develop a novel LSTM that performs spatial input aggregation and output modeling for dense labeling output.

2.4 Action Detection

Beyond assigning a single label to a whole video, the task of action detection localizes this action within the video sequence. An example of canonical work in this vein is Ke et al. (2007). More recent work extended latent SVMs to spatio-temporal action detection and localization (Tian et al. 2013; Lan et al. 2011). Rohrbach et al. (2015) detect cooking actions using hand-centric features accounting for human pose variation. Ni et al. (2014) similarly utilize hand-centric features on the MPII Cooking dataset, but focus on multiple levels of action granularity. Gkioxari and Malik (2014) train SVMs for actions on top of deep learned features, and further link them in time for spatio-temporal action detection. In contrast, we address the task of dense multilabel action detection.

2.5 Attention-Based Models

Seminal work on computational spatial attention models for images was done by Itti et al. (1998). Recent action analysis work utilizing attention includes Shapovalova et al. (2013) who use eye-gaze data to drive action detection and localization. Xu et al. (2015) use visual attention to assist in caption generation. Yao et al. (2015) develop an LSTM for video caption generation with soft temporal attention. Our method builds on these directions, using an attention-based input temporal context for dense action labeling.

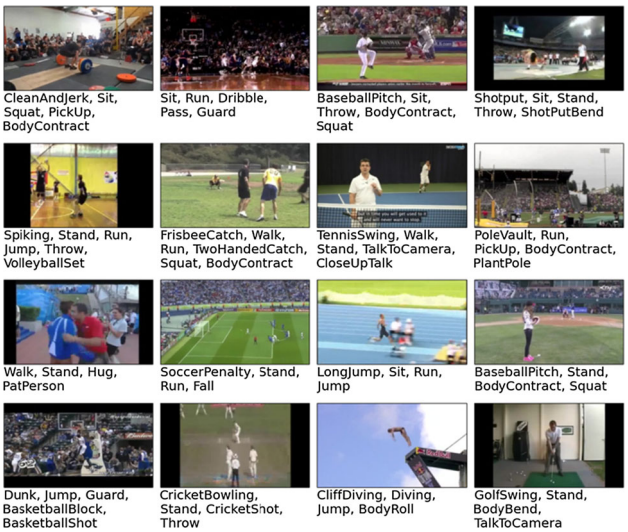

Fig. 2 Our MultiTHUMOS dataset contains multiple action annotations per frame

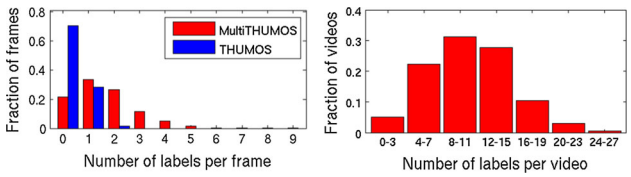

Fig. 3 *Left* MultiTHUMOS has significantly more labels per frame than THUMOS (Jiang et al. 2014) (1.5 in MultiTHUMOS versus 0.3 in THUMOS). *Right* additionally, MultiTHUMOS contains up to 25 action labels per video compared to ≤3 labels in THUMOS

3 The MultiTHUMOS Dataset

Research on detailed, multilabel action understanding requires a dataset of untrimmed, densely labeled videos. However, we are not aware of any existing dataset that fits these requirements. THUMOS (Jiang et al. 2014) is untrimmed but contains on average only a single distinct action labeled per video. MPII Cooking (Rohrbach et al. 2012) and Breakfast (Kuehne et al. 2014) datasets have labels of sequential actions, but contain only a single label per frame and are further captured in closed-world settings of a single or small set of kitchens (Table 1).

To address the limitations of previous datasets, we introduce a new dataset called MultiTHUMOS.[1] MultiTHUMOS contains dense, multilabel, frame-level action annotations (Fig. 2) for 30 h across 400 videos in the THUMOS '14 action detection dataset (referred to hereafter as THUMOS)In particular, all videos in the "Validation Data" and "Test Data" sets were labeled. THUMOS training data consists of 3 sets of videos: temporally clipped "Training Data", temporally untrimmed "Validation Data" with temporal annotations, and temporally untrimmed "Background Data" with no temporal annotations. Test data consists of temporally untrimmed "Test Data" with temporal annotations. We annotated all video sets originally including temporal annotations, i.e. "Validation Data" and "Test Data".

Annotations were collected in collaboration with Datatang,[2] a commercial data annotation service. Workers were provided with the name of an action, a brief (up to 1 sentence) description, and 2 annotation examples, and asked to annotate the start and end frame of the action in the videos. An action was annotated if it occurred anywhere in the frame. A single worker was used to annotate each video since the workers are employees of the company, and a second worker verified each annotation as part of Datatang's quality control process after annotation.

In total, we collected 32, 325 annotations of 45 action classes, bringing the total number of annotations from 6, 365 over 20 classes in THUMOS to 38, 690 over 65 classes in MultiTHUMOS. The classes were selected to have a diversity of length, to include hierarchical, hierarchical within a sport, and fine-grained categories, and to include both sport specific and non-sport specific categories. The action classes are described in more detail below. Importantly, it is not just the scale of the dataset that has increased. The *density* of annotations increased from 0.3 to 1.5 labels per frame on average and from 1.1 to 10.5 action classes per video. The availability of such densely labeled videos allows research on interaction between actions that was previously impossible with more sparsely labeled datasets. The maximum number of actions per frame increased from 2 in THUMOS to 9 MultiTHUMOS, and the maximum number of actions per video increased from 3 in THUMOS to 25 in MultiTHUMOS. Figure 3 shows the full distribution of annotation density.

Using these dense multilabel video annotations, we are able to learn and visualize the relationships between actions. The co-occurrence hierarchy of object classes in images based on mutual information of object annotations was learned by Choi et al. (2010); we adapt their method to per-frame action annotations in video. Figure 4 shows the resulting action hierarchy. Classes such as squat and body contract frequently co-occur; in contrast, classes such as run and billiards rarely occur together in the same frame.

MultiTHUMOS is a very challenging dataset for four key reasons.

1. *Long tail data distribution* First, MultiTHUMOS has a long tail distribution in the amount of annotated data per action class. This requires action detection algorithms to effectively utilize both small and large amounts of annotated data. Concretely, MultiTHUMOS has between 27 s

[1] The dataset is available for download at http://ai.stanford.edu/~syyeung/everymoment.html.

[2] http://factory.datatang.com/en/.

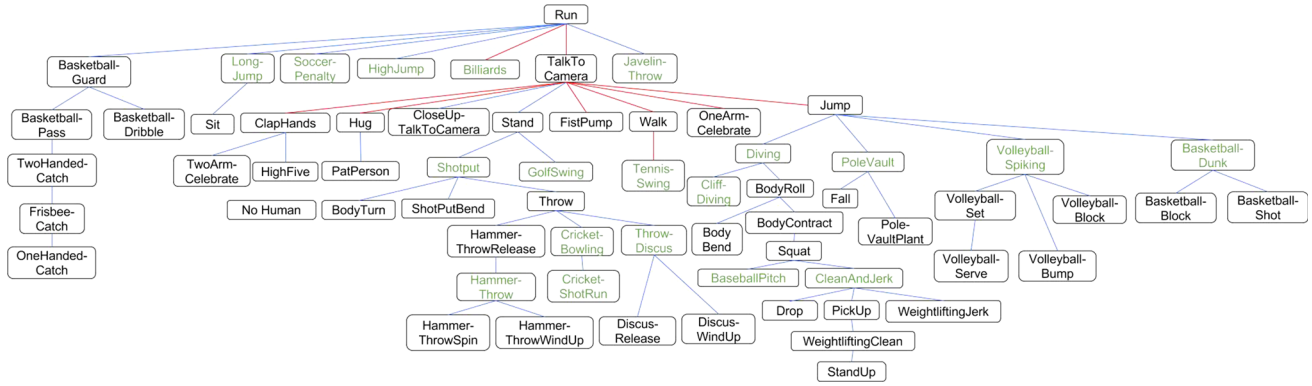

Fig. 4 We use the method of Choi et al. (2010) to learn the relationships between the 65 MultiTHUMOS classes based on per-frame annotations. *Blue* (*red*) means positive (negative) correlation. The 20 original THUMOS classes are in *green* (Color figure online)

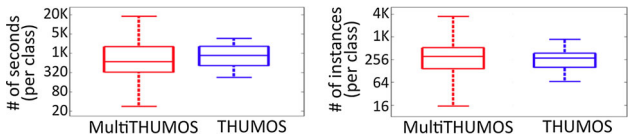

Fig. 5 MultiTHUMOS has a wider range of number of per-class frames and instances (contiguous sequences of a label) annotated than THUMOS. Some action classes like Stand or Run have up to 3.5K instances (up to 18K s, or 5.0 h); others like VolleyballSet or Hug have only 15 and 46 instances (27 and 50 s) respectively

to 5 h of annotated video per action class (with the rarest actions being volleyball bump, a pat, volleyball serve, high five and basketball block, and the most common actions being stand, walk, run, sit and talk to the camera). In contrast, THUMOS is more uniformly annotated: the dataset ranges from the rarest action baseball pitch with 3.7 min annotated to the most common action pole vault with 1 h of annotated video. Figure 5 shows the full distribution.

2. *Length of actions* The second challenge is that MultiTHUMOS has much shorter actions compared to THUMOS. For each action class, we compute the average length of an action instance of that class. Instance of action classes in THUMOS are on average 4.8 s long compared to only 3.3 s long in MultiTHUMOS. Instances of action classes in THUMOS last between 1.5 s on average for clicket bowling to 14.7 s on average for billiards. In contrast, MultiTHUMOS has seven action classes whose instances last less than a second on average: two-handed catch, planting the pole in pole vaulting, basketball shot, one-handed catch, basketball block, high five and throw. Shorter actions are more difficult to detect since there is very little visual signal in the positive frames. There are instances of actions throw, body contract and squat that last only 2 *frames* (or 66 ms) in MultiTHUMOS! Accurately localizing such actions encourages strong contextual modeling and multi-action reasoning.
3. *Fine-grained actions* The third challenge of MultiTHUMOS is the many fine-grained action categories with low visual inter-class variation, including hierarchical (e.g. throw vs. baseball pitch), hierarchical within a sport (e.g. pole vault vs. the act of planting the pole when pole vaulting), and fine-grained (e.g. basketball dunk, shot, dribble, guard, block, and pass). It also contains both sport-specific actions (such as different basketball or volleyball moves), as well as general actions that can occur in multiple sports (e.g. pump fist, or one-handed catch). This requires the development of general action detection approaches that are able to accurate model a diverse set of visual appearances.
4. *High intra-class variation* The final MultiTHUMOS challenge is the high intra-class variation as shown in Fig. 6. The same action looks visually very different across multiple frames. For example, a hug can be shown from many different viewpoints, ranging from extreme close-up shots to zoomed-out scene shots, and may be between two people or a larger group. This encourages the development of models that are insensitive to particular camera viewpoint and instead accurately focus on the semantic information within a video.

With the MultiTHUMOS dataset providing new challenges for action detection, we now continue on to describing our proposed approach for addressing these challenges and making effective use of the dense multilabel annotation.

4 Technical Approach

Actions in videos exhibit rich patterns, both within a single frame due to action label relations and also across frames as they evolve in time. The desire to elegantly incorporate these cues with state-of-the-art appearance-based models has led to recent works (Donahue et al. 2014; Ng et al. 2015;

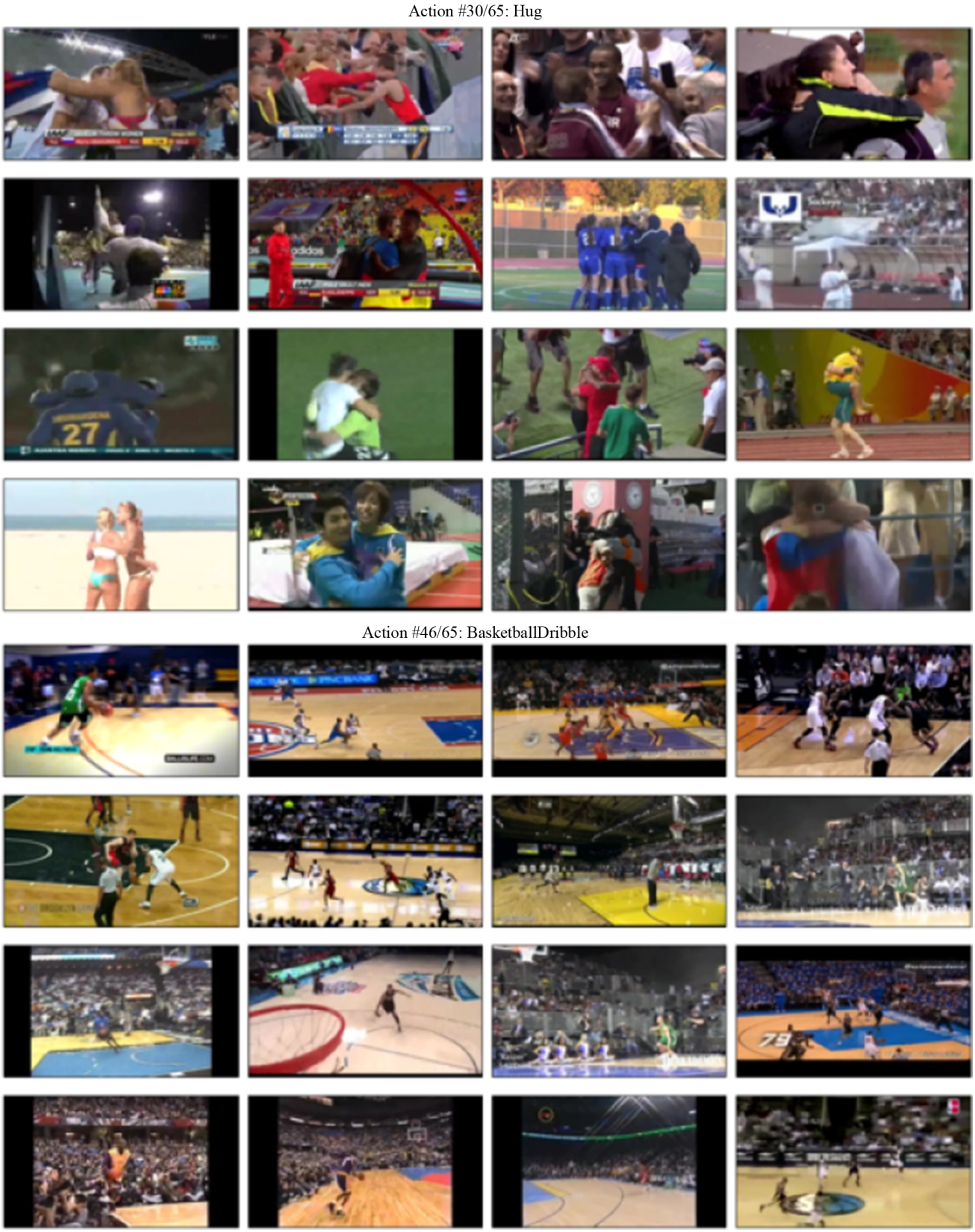

Fig. 6 Our MultiTHUMOS dataset is very challenging due to high intra-class variation

Srivastava et al. 2015) that study combinations of convolutional neural networks (CNN) modeling frame-level spatial appearance and recurrent neural networks (RNN) modeling the temporal dynamics. However, the density of the action labels in our dataset expands the opportunities for more complex modeling at the temporal level. While in principle even a simple instantiation of an ordinary RNN has the capacity to capture arbitrary temporal patterns, it is not necessarily the best model to use in practice. Indeed, our proposed MultiLSTM model extends the recurrent models described in previous work, and our experiments demonstrate its effectiveness.

4.1 LSTM

The specific type of Recurrent architecture that is commonly chosen in previous work is the LSTM, which owing to its appealing functional properties has brought success in a wide range of sequence-based tasks such as speech recognition, machine translation and very recently, video activity classification. Let $\mathbf{x}$ be an input sequence $(x_1, ..., x_T)$ and $\mathbf{y}$ be an output sequence $(y_1, ..., y_T)$. An LSTM then maps $\mathbf{x}$ to $\mathbf{y}$ through a series of intermediate representations:

$$i_t = \sigma\left(W_{xi}x_t + W_{h_i}h_{t-1} + b_i\right) \quad (1)$$
$$f_t = \sigma\left(W_{xf}x_t + W_{hf}h_{t-1} + b_f\right) \quad (2)$$
$$o_t = \sigma\left(W_{xo}x_t + W_{ho}h_{t-1} + b_o\right) \quad (3)$$
$$g_t = \tanh\left(W_{xc}x_t + W_{hc}h_{t-1} + b_c\right) \quad (4)$$
$$c_t = f_t c_{t-1} + i_t g_t \quad (5)$$
$$h_t = o_t \tanh\left(c_t\right) \quad (6)$$
$$y_t = W_{hy}h_t + b_y \quad (7)$$

Here c is the "internal memory" of the LSTM, and the gates i, f, o control the degree to which the memory accumulates new input g, attenuates its memory, or influences the hidden layer output h, respectively. Intuitively, the LSTM has the capacity to read and write to its internal memory, and hence maintain and process information over time. Compared to standard RNNs, the LSTM networks mitigate the "vanishing gradients" problem because except for the forget gate, the cell memory is influenced only by additive interactions that can communicate the gradient signal over longer time durations. The architecture is parametrized by the learnable weight matrices W and biases b, and we refer the reader to Hochreiter and Schmidhuber (1997), Donahue et al. (2014) for further details.

However, an inherent flaw of the plain LSTM architecture is that it is forced to make a definite and final prediction at some time step based on what frame it happens to see at that time step, and its previous context vector.

4.2 MultiLSTM

Our core insight is that providing the model with more freedom in both reading its input and writing its output reduces the burden placed on the hidden layer representation. Concretely, the MultiLSTM expands the temporal receptive field of both input and output connections of an LSTM. These allow the model to directly refine its predictions in retrospect after seeing more frames, and additionally provide direct pathways for referencing previously-seen frames without forcing the model to maintain and communicate this information through its recurrent connections.

4.2.1 Multilabel Loss

In our specific application setting, the input vectors x_t correspond to the 4096-dimensional fc-7 features of the VGG 16-layer Convolutional Network which was first pretrained on ImageNet and then fine-tuned on our dataset on an individual frame level. We interpret the vectors y_t as the unnormalized log probability of each action class. Since each frame of a video can be labeled with multiple classes, instead of using the conventional softmax loss we sum independent logistic regression losses per class:

$$L\left(\mathbf{y}|\mathbf{x}\right) = \sum_{t,c} z_{tc} \log\left(\sigma\left(y_{tc}\right)\right) + \left(1 - z_{tc}\right) \log\left(1 - \sigma\left(y_{tc}\right)\right)$$

where y_{tc} is the score for class c at time t, and z_{tc} is the binary ground truth label for class c at time t.

4.2.2 Multiple Inputs with Temporal Attention

In a standard LSTM network, all contextual information is summarized in the hidden state vector. Therefore, the network relies on the memory vector to contain all relevant information about past inputs, without any ability to explicitly revisit past inputs. This is particularly challenging in the context of more complex tasks such as dense, multilabel action detection.

To provide the LSTM with a more direct way of accessing recent inputs, we expand the temporal dimension of the input to be a fixed-length window of frames previous to the current time step (Fig. 7a). This allows the LSTM to spend its modeling capacity on more complex and longer-term interactions instead of maintaining summary of the recent frames in case it may be useful for the next few frames. Furthermore, we incorporate a soft-attention weighting mechanism that has recently been proposed in the context of machine translation (Bahdanau et al. 2014).

Concretely, given a video $\mathbf{V} = \{v_1, \ldots v_T\}$, the input x_i to the LSTM at time step i is now no longer the

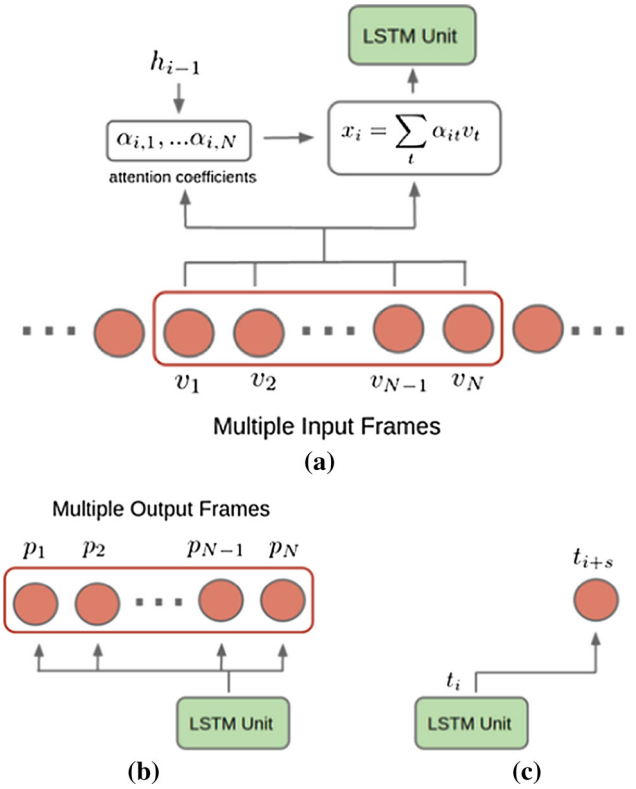

Fig. 7 Components of our MultiLSTM model. **a** Connections to multiple inputs. **b** Multiple outputs. **c** Variant: output offset.

representation of a single frame v_t, but a weighted combination $x_i = \sum_t \alpha_{it} v_t$ where t ranges over a fixed-size window of frames previous to i, and α_{it} is the contribution of frame v_t to input x_i as computed by the soft attention model. To compute the attention coefficients α_{it}, we use a model similar to Bahdanau et al. (2014). The precise formulation that worked best in our experiments is:

$$\alpha_{it} \propto \exp\left(w_{ae}^T \left[\tanh(W_{ha} h_{i-1}) \odot \tanh(W_{va} v_t)\right]\right) \quad (8)$$

Here $\odot$ is element-wise multiplication, $\{w_{ae}, W_{ha}, W_{va}\}$ are learned weights, and α_t is normalized using the softmax function with the interpretation that α_t expresses the relative amount of attention assigned to each frame in the input window. Intuitively, the first term $\tanh(W_{ha} h_{i-1})$ allows the network to look for certain features in the input, while the second term $\tanh(W_{va} v_t)$ allows each input to broadcast the presence/absence of these features. Therefore, the multiplicative interaction followed by the weighted sum with w_{ae} has the effect of quantifying the agreement between what is present in the input and what the network is looking for. Note that the standard LSTM formulation is a special case of this model where all attention is focused on the last input window frame.

4.2.3 Multiple Outputs

Analogous to providing explicit access to a window of frames at the input, we allow the LSTM to contribute to predictions in a window of frames at the output (Fig. 7b). Intuitively, this mechanism lets the network refine its predictions in retrospect, after having seen more frames of the input. This feature is related to improvements that can be achieved by use of bi-directional recurrent networks. However, unlike bi-directional models our formulation can be used in an online setting where it delivers immediate predictions that become refined with a short time lag.[3] Given the multiple outputs, we consolidate the predicted labels for all classes c at time t with a weighted average $y_t = \sum_i \beta_{it} p_{it}$ where p_{it} are the predictions at the ith time step for the tth frame, and β_{it} weights the contribution. β_{it} can be learned although in our experiments we use $\frac{1}{N}$ for simplicity to average the predictions. The standard LSTM is a special case, where β is an indicator function at the current time step. In our experiments we use the same temporal windows at the input and output. Similar to the inputs, we experimented with soft attention over the output predictions but did not observe noticeable improvements. This may be due to increased fragility when the attention is close to the output without intermediate network layers to add robustness, and we leave further study of this to future work.

4.2.4 Single Offset Output

We experimented with offset predictions to quantify how informative frames at time t are towards predicting labels at some given offset in time. In these experiments, the network is trained with shifted labels y_{t+s}, where s is a given offset (Fig. 7c). In our dense label setting, this type of model additionally enables applications such as action prediction in unconstrained internet video [c.f. (Kitani et al. 2012)]. For example, if the input is a frame depicting a person cocking his arm to throw, the model could predict future actions such as Catch or Hit.

5 Experiments

We begin by describing our experimental setup in Sect. 5.1. We then empirically demonstrate the effectiveness of our model on the challenging tasks of action detection (Sect. 5.2) and action prediction (Sect. 5.3).

[3] A similar behavior can be obtained with a bi-directional model by truncating the hidden state information from future time frames to zero, but this artificially distorts the test-time behavior of the model's outputs, while our model always operates in the regime it was trained with.

5.1 Setup

5.1.1 Dataset

We evaluate our MultiLSTM model for dense, multilabel action detection on the MultiTHUMOS dataset. We use the same train and test splits as THUMOS (see Sect. 3 for details) but ignore the background training videos. Clipped training videos (the "Training Data" set in THUMOS) act as weak supervision since they are only labeled with the THUMOS-subset of MultiTHUMOS classes.

5.1.2 Implementation Details

Our single-frame baseline uses the 16-layer VGG CNN model (Simonyan and Zisserman 2014b), which achieves near state of the art performance on ILSVRC (Russakovsky et al. 2015). The model was pre-trained on ImageNet and all layers fine-tuned on MultiTHUMOS using a binary cross-entropy loss per-class. The input to our LSTM models is the final 4096-dimensional, frame-level fc7 representation.

We use 512 hidden units in the LSTM, and 50 units in the attention component of MultiLSTM that is used to compute attention coefficients over a window of 15 frames. We train the model with an exact forward pass, passing LSTM hidden and cell activations from one mini-batch to the next. However we use approximate backpropagation through time where we only backpropagate errors for the duration of a single mini-batch. Our mini-batches consist of 32 input frames (approx. 3.2 s), and we use RMSProp (Tieleman and Hinton 2012) to modulate the per-parameter learning rate during optimization.

5.1.3 Performance Measure

We evaluate our models using average precision (AP) measured on our frame-level labels. The focus of our work is dense labeling, hence this is the measure we analyze to evaluate the performance of our model. We report AP values for individual action classes as well as mean Average Precision (mAP), the average of these values across the action categories.

To verify that our baseline models are strong, we can obtain discrete detection instances using standard heuristic post-processing. Concretely, for each class we threshold the frame-level confidences at λ ($\lambda = 0.1$ obtained by cross-validation) to get binary predictions and then accumulate consecutive positive frames into detections. For each class C, let $\mu(C)$ and $\sigma(C)$ be the mean and standard deviation respectively of frame lengths on the training set. The score of a detection for class C of length L with frame probabilities $p_1 \ldots p_L$ is then computed as

$$score\,(C, p_1, \ldots, p_L) = \left(\sum_i^L p_i\right) \times \exp\left(\frac{-\alpha\,(L - \mu(C))^2}{\sigma(C)^2}\right) \tag{9}$$

where the hyperparameter $\alpha = 0.01$ is obtained by cross-validation. Using this post-processing, our single-frame CNN model achieves 32.4 detection mAP with overlap threshold 0.1 on the THUMOS subset of MultiTHUMOS. Since state of the art performance on THUMOS reports 36.6 detection mAP including audio features, this confirms that our single-frame CNN is a reasonable baseline. Hereafter, we compare our models without this post-processing to achieve a comparison of the models' dense labeling representational ability.

5.2 Action Detection

We first evaluate our models on the challenging task of dense per-frame action labeling on MultiTHUMOS. The MultiLSTM model achieves consistent improvements in mean average precision (mAP) compared to baselines. A model trained on Improved Dense Trajectories features Wang and Schmid (2013) (using a linear SVM trained on top of a temporally pooled and quantized dictionary of pre-computed IDT features, provided by THUMOS'14) performs relatively poorly with 13.3 mAP. This highlights the difficulty of the dataset and the challenge of working with generic hand-crafted features that are not learned for these specific fine-grained actions. Additional variants of IDT could be used to improve performance. For example, Fisher Vector encoding of raw IDT features is commonly used to boost performance. However, these methods can be computationally expensive and are limited due to their reliance on underlying hand-crafted features and lack of opportunity for joint training. Hence, we use neural network-based models for the rest of our experiments.

A single-frame CNN fine-tuned on MultiTHUMOS attains 25.4% mAP. We trained a base LSTM network in the spirit of Donahue et al. (2014) but modified for multilabel action labeling. Specifically, the LSTM is trained using a multilabel loss function and tied hidden context across 32 frame segments, as described in Sect. 4.2. This base LSTM boosts mAP to 28.1%. Our full MultiLSTM model handily outperforms both baselines with 29.7% mAP. Table 2 additionally demonstrates that each component of our model (input connections, input attention and output connections) is important for accurate action labeling.

Figure 8 compares per-class results of the CNN versus MultiLSTM, and the base LSTM versus MultiLSTM. MultiTHUMOS outperforms the CNN on 56 our of 65 action classes, and the LSTM on 50 out of 65 action classes. A sampling of action classes is labeled. It is interesting to note

Table 2 Per-frame mean average precision (mAP) of the MultiLSTM model compared to baselines. Two-stream CNN is computed with single-frame flow. LSTM is implemented in the spirit of Donahue et al. (2014) (details in Sect. 4.2). We show the relative contributions of adding first the input connections with averaging (LSTM + i), then the attention (LSTM + i + a) as in Fig. 7a, and finally the output connections to create our proposed MultiLSTM model (LSTM + i + a + o) as in Fig. 7b

Model	THUMOS mAP	MultiTHUMOS mAP
IDT (Wang and Schmid 2013)	13.6	13.3
Single-frame CNN (Simonyan and Zisserman 2014b)	34.7	25.4
Two-stream CNN (Simonyan and Zisserman 2014a)	36.2	27.6
LSTM	39.3	28.1
LSTM + i	39.5	28.7
LSTM + i + a	39.7	29.1
MultiLSTM	**41.3**	**29.7**

MultiLSTM achieves the highest mAP (bolded) on both THUMOS and MultiTHUMOS

from the two plots that compared with the CNN, the LSTM closes the gap with MultiLSTM on classes such as Frisbee Catch, Pole Vault, and Basetkball Guard, which are strongly associated with temporal context (e.g. a throw proceeds a frisbee catch, and a person usually stands at the track for some time before beginning a pole vault). This shows the benefit of stronger temporal modeling, which MultiLSTM continues to improve on the majority of classes.

Figure 9 analyzes per-frame mAP as the number of attention units (at both input and output) in the MultiLSTM model is varied. We observe that increasing the number of attention units improves performance up to a point (75 units), as would be expected, and then decreases past that as the number of parameters becomes too large. In practice, we use 50 units in our experiments.

Figure 10 visualizes some results of MultiLSTM compared to a baseline CNN. For ease of visualization, we binarize outputs by thresholding rather than showing the per-frame probabilistic action labels our model produces. The CNN often produces short disjoint detections whereas MultiLSTM effectively makes use of temporal and co-occurrence context to produce more consistent detections.

The multilabel nature of our model and dataset allows us to go beyond simple action labeling and tackle higher-level tasks such as retrieval of video segments containing sequences of actions (Fig. 11) and co-occurring actions (Fig. 12). By learning accurate co-occurrence and temporal relationships, the model is able to retrieve video fragments with detailed action descriptions such as Pass and then Shot or frames with simultaneous actions such as Sit and Talk.

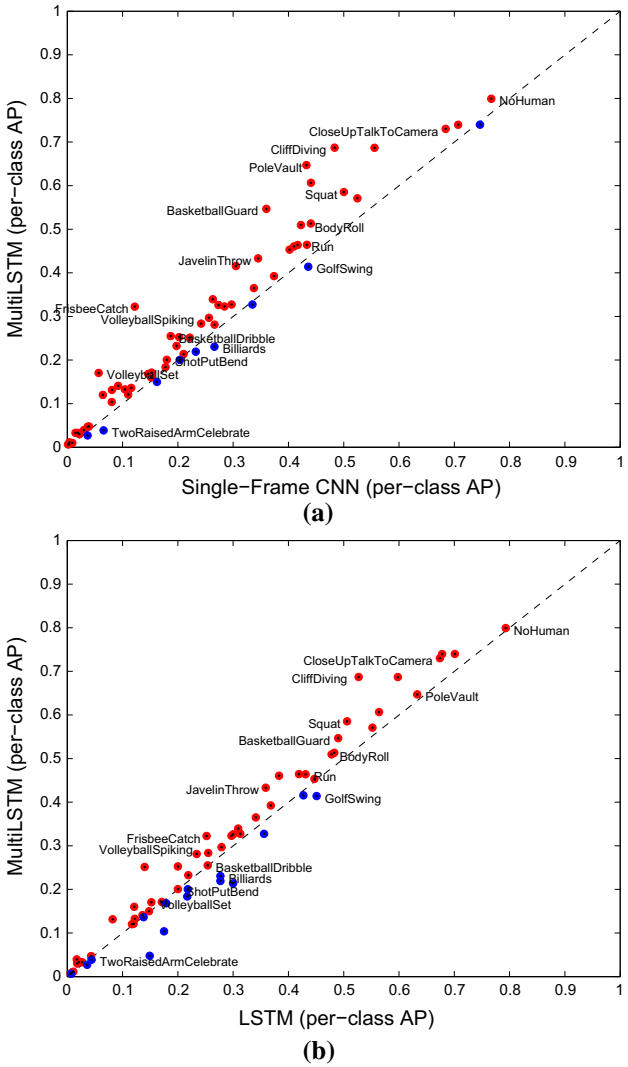

Fig. 8 Per-class average precision of the MultiLSTM model compared to **a** a single-frame CNN model Simonyan and Zisserman (2014b); and **b** an LSTM on MultiTHUMOS. MultiLSTM outperforms the single-frame CNN on 56 out of 65 action classes, and the LSTM on 50 out of 65 action classes

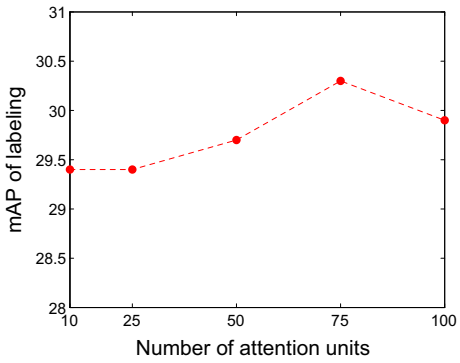

Fig. 9 Number of attention units versus per-frame mAP of the MultiTHUMOS model. Performance increases as the number of units is increased, but decreases past 75 units. We use 50 units in our experiments

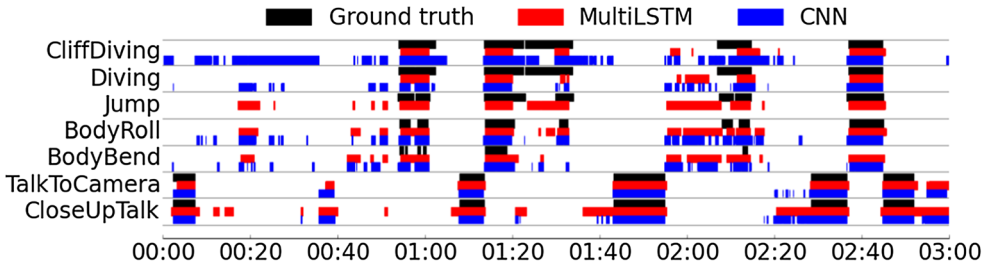

Fig. 10 Example timeline of multilabel action detections from our MultiLSTM model compared to a CNN (*best in color*) (Color figure online)

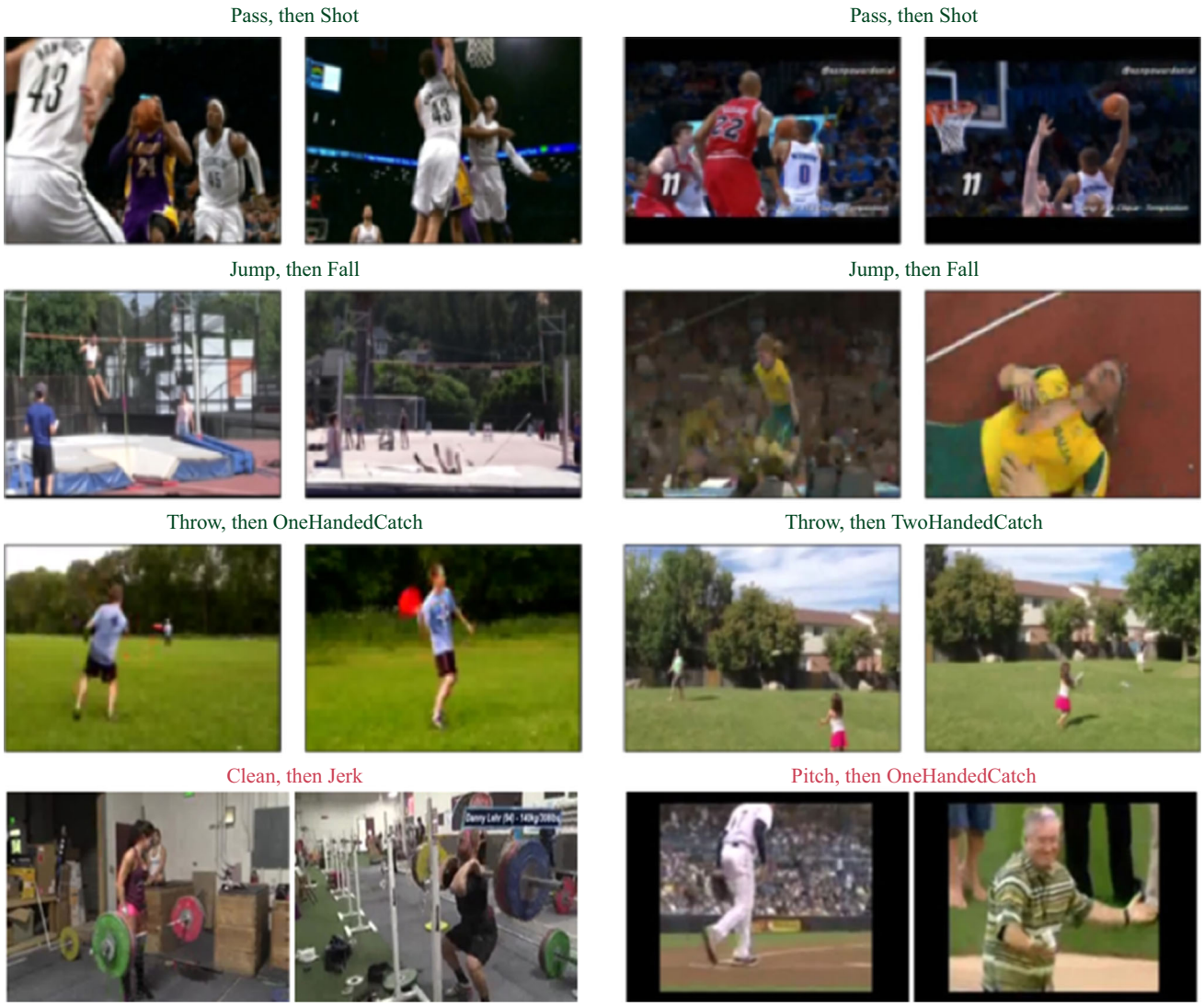

Fig. 11 Examples of retrieved sequential actions (correct in *green*, mistakes in *red*). Results are shown in pairs: first action frame on the *left*, second action frame on the *right* (Color figure online)

5.3 Action Prediction

Dense multilabel action labeling in unconstrained internet videos is a challenging problem to tackle in and of itself. In this section we go one step further and aim to make predictions about what is likely to happen next or what happened previously in the video. By utilizing the MultiLSTM model with offset (Fig. 7c) we are able to use the learned temporal relationships between actions to make inferences about actions likely occurring in past or future frames.

Fig. 12 Examples of retrieved frames with co-occurring actions (correct in *green*, mistakes in *red*). The model is able to distinguish between subtly different scenarios (Color figure online)

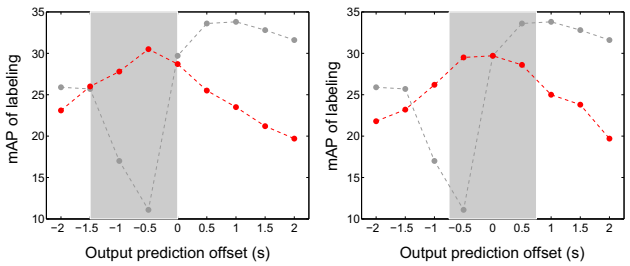

Fig. 13 Action detection mAP when the MultiLSTM model predicts the action for a past (offset < 0) or future (offset > 0) frame rather than for the current frame (offset $= 0$). The input window of the MultiLSTM model is shown in *gray*. Thus, the *left plot* is of a model trained with input from the past, and the *right plot* is of a model trained with the input window *centered* around the current frame. mAP of the MultiLSTM model is shown in *red*, and mAP of a model using ground-truth label distribution is shown in *gray* (Color figure online)

We evaluate the performance of this model as a function of temporal offset magnitude and report results in Fig. 13. MultiLSTM prediction mAP is shown in red. The plot on the left quantifies the prediction ability of the model within a 4 s ($+/-$ 2 s) window, provided an input window of context spanning the previous 1.5 s. The model is able to "see the future" – while predicting actions 0.5 s in the past is easiest (mAP $\approx$ 30%), reasonable prediction performance (mAP $\approx$ 20–25%) is possible 1–2 s into the future. The plot on the right shows the prediction ability of the model using an input context centered around the current frame, instead of spanning only the past. The model is able to provide stronger predictions at past times compared to future times, giving quantitative insight into the contribution of the hidden state vector to providing past context.

It is also interesting to compare MultiLSTM prediction to a model using the ground-truth label distribution (shown in gray). Specifically, this model makes action predictions using the most frequent label for a given temporal offset from the training set, per-class, and weighted by the MultiLSTM prediction probabilities of actions in the current frame. The label distribution-based model has relatively high performance in the future direction as opposed to the past, and at farther offsets from the current frame. This indicates that stronger priors can be learned in these temporal regions (e.g. frisbee throw should be followed by frisbee catch, and 2 s after a dive is typically background (no action)), and MultiLSTM does learn them to some extent. On the other hand, the label distribution-based model has poor performance immediately before the current frame, indicating that there is greater variability in this temporal region, e.g. clapping may be preceded by many different types of sport scoring actions, though a longer offset in the past may be more likely background. In this temporal region, MultiLSTM shows significantly stronger performance than using priors, indicating the benefit of its temporal modeling in this context.

Figure 14 shows qualitative examples of predictions at frames 1 s in the future from the current time. The model is able to correctly infer that a Fall is likely to happen after a Jump, and a BasketballShot soon after a Dribble.

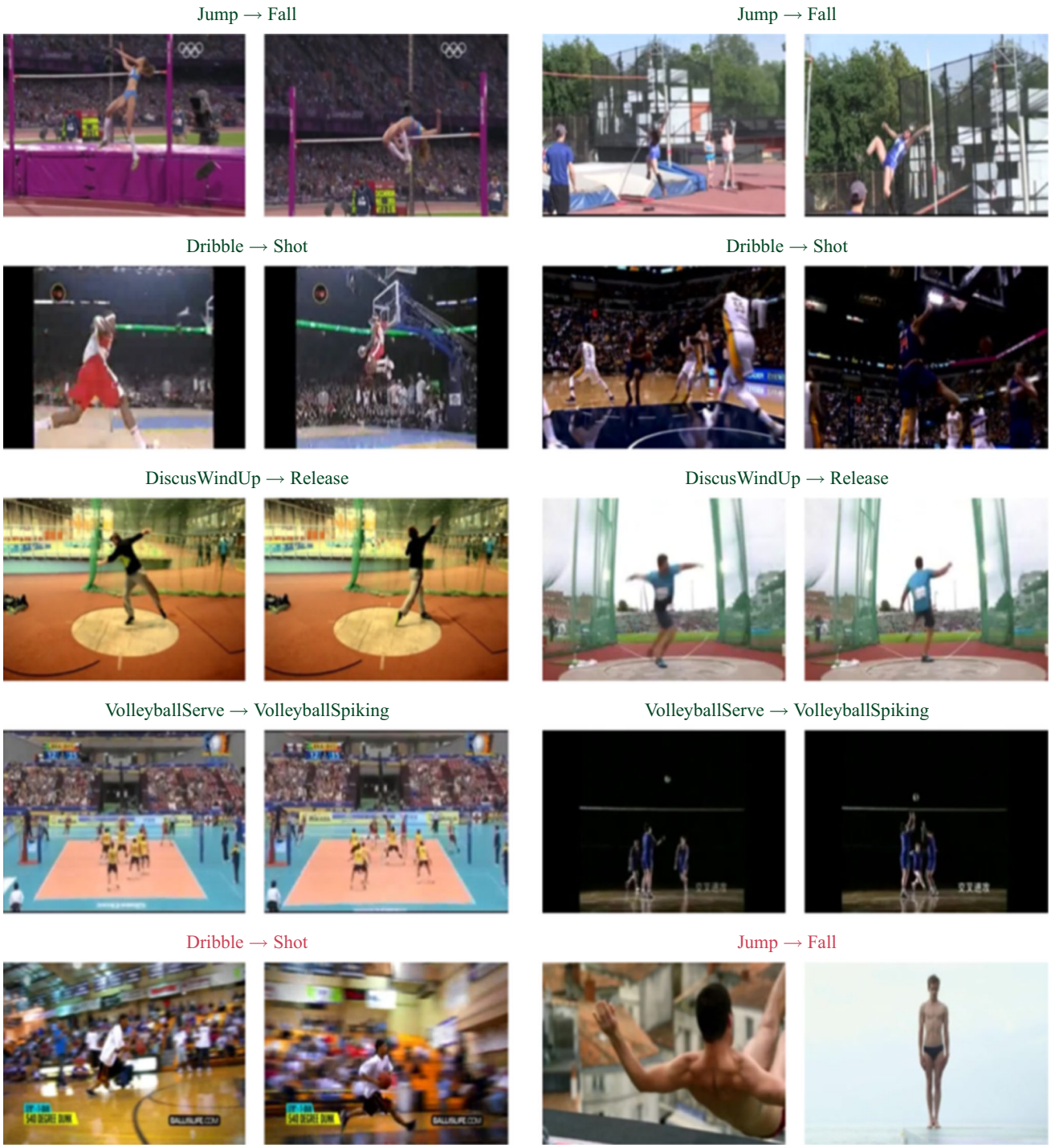

Fig. 14 Examples of predicted actions. For each pair of actions, the first one (*left*) is the label of the current frame and the second one (*right*) is the predicted label 1 s into the future. Correct predictions are shown in *green*, and failure cases are shown in *red* (Color figure online)

6 Conclusion

In conclusion, this paper presents progress in two aspects of human action understanding. First, we emphasize a broader definition of the task, reasoning about dense, multiple labels per frame of video. We have introduced a new dataset MultiTHUMOS, containing a substantial set of labeled data that we will release to spur research in this direction of action recognition. Second, we develop a novel LSTM-based model incorporating soft attention input-output temporal context for

dense action labeling. We show that utilizing this model on our dataset leads to improved accuracy of action labeling and permits detailed understanding of human action.

Acknowledgements We would like to thank Andrej Karpathy and Amir Zamir for helpful comments and discussion.

References

Bahdanau, D., Cho, K., Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. arXiv:1409.0473.

Blank, M., Gorelick, L., Shechtman, E., Irani, M., Basri, R. (2005). Actions as space-time shapes. In *The Tenth IEEE International Conference on Computer Vision (ICCV'05)*.

Choi, M. J., Lim, J. J., Torralba, A., Willsky, A. S. (2010). Exploiting hierarchical context on a large database of object categories. In *CVPR*.

Donahue, J., Hendricks, L. A., Guadarrama, S., Rohrbach, M., Venugopalan, S., Saenko, K., Darrell, T. (2014). Long-term recurrent convolutional networks for visual recognition and description. arXiv:1411.4389.

Fabian Caba Heilbron, B. G., Victor Escorcia and Niebles, J. C. (2015). Activitynet: A large-scale video benchmark for human activity understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, (pp. 961–970).

Gkioxari, G., Malik, J. (2014). Finding action tubes. arXiv:1411.6031.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780.

Itti, L., Koch, C., & Niebur, E. (1998). A model of saliency-based visual attention for rapid scene analysis. *PAMI*, *20*(11), 1254–1259.

Jiang, Y.-G., Liu, J., Roshan Zamir, A., Toderici, G., Laptev, I., Shah, M., Sukthankar, R. (2014). THUMOS challenge: Action recognition with a large number of classes. http://crcv.ucf.edu/THUMOS14/.

Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., Fei-Fei, L. (2014). Large-scale video classification with convolutional neural networks. In *CVPR*.

Ke, Y., Sukthankar, R., Hebert, M. (2007). Event detection in crowded videos. In *ICCV*.

Kitani, K. M., Ziebart, B., Bagnell, J. D., Hebert, M. (2012). Activity forecasting. In *ECCV*.

Kuehne, H., Arslan, A., Serre, T. (2014). The language of actions: Recovering the syntax and semantics of goal-directed human activities. In *CVPR*.

Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., Serre, T. (2011). Hmdb: a large video database for human motion recognition. In *ICCV*.

Lan, T., Wang, Y., Mori, G. (2011). Discriminative figure-centric models for joint action localization and recognition. In *ICCV*.

Lv, F. J., Nevatia, R. (2007). Single view human action recognition using key pose matching and viterbi path searching. In *CVPR*.

Mansimov, E., Srivastava, N., Salakhutdinov, R. (2015). Initialization strategies of spatio-temporal convolutional neural networks. arXiv:1503.07274.

Marszałek, M., Laptev, I., Schmid, C. (2009). Actions in context. In *IEEE Conference on Computer Vision & Pattern Recognition*.

Myers, G. K., Nallapati, R., van Hout, J., Pancoast, S., Nevatia, R., Sun, C., et al. (2014). Evaluating multimedia features and fusion for example-based event detection. *Machine Vision and Applications*, *25*(1), 17–32.

Ng, J. Y.-H., Hausknecht, M., Vijayanarasimhan, S., Vinyals, O., Monga, R., Toderici, G. (2015). Beyond short snippets: Deep networks for video classification. arXiv:1503.08909.

Ni, B., Paramathayalan, V. R., Moulin, P. (2014). Multiple granularity analysis for fine-grained action detection. In *CVPR*.

Niebles, J. C., Chen, C.-W., Fei-Fei, L. (2010). Modeling temporal structure of decomposable motion segments for activity classification. In *ECCV*.

Oh, S., Hoogs, A., Perera, A., Cuntoor, N., Chen, C.-C., Lee, J. T., Mukherjee, S., Aggarwal, J. K., Lee, H., Davis, L., Swears, E., Wang, X., Ji, Q., Reddy, K., Shah, M., Vondrick, C., Pirsiavash, H., Ramanan, D., Yuen, J., Torralba, A., Song, B., Fong, A., Roy-Chowdhury, A., Desai, M. (2011) A large-scale benchmark dataset for event recognition in surveillance video. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Oh, S., Mccloskey, S., Kim, I., Vahdat, A., Cannons, K. J., Hajimirsadeghi, H., et al. (2014). Multimedia event detection with multimodal feature fusion and temporal concept localization. *Machine Vision and Applications*, *25*(1), 49–69.

Over, P., Awad, G., Michel, M., Fiscus, J., Kraaij, W., Smeaton, A. F., Quenot, G. (2011). Trecvid 2011—An overview of the goals, tasks, data, evaluation mechansims and metrics. In *Proceedings of TRECVID 2011*.

Pirsiavash, H., Ramanan, D. (2014). Parsing videos of actions with segmental grammars. In *CVPR*.

Poppe, R. (2010). A survey on vision-based human action recognition. *IVC*, *28*, 976–990.

Qinfeng Shi, L. W. A. S., Cheng, Li (2011). Human action segmentation and recognition using discriminative semi-markov models. *International Journal of Computer Vision*, *93*, May.

Rohrbach, M., Amin, S., Andriluka, M., Schiele, B. (2012). A database for fine grained activity detection of cooking activities. In *CVPR*.

Rohrbach, M., Rohrbach, A., Regneri, M., Amin, S., Andriluka, M., Pinkal, M., Schiele, B. (2015) Recognizing fine-grained and composite activities using hand-centric features and script data. arXiv:1502.06648.

Russakovsky, O., et al. (2015). ImageNet large scale visual recognition challenge. *International Journal of Computer Vision*, *115*(3), 211–252.

Ryoo, M. S., Aggarwal, J. K. (2009). Spatio-temporal relationship match: Video structure comparison for recognition of complex human activities. In *IEEE International Conference on Computer Vision (ICCV)*.

Schuldt, C., Laptev, I., Caputo, B. (2004). Recognizing human actions: A local svm approach. In *ICPR*.

Shapovalova, N., Raptis, M., Sigal, L., Mori, G. (2013). Action is in the eye of the beholder: Eye-gaze driven model for spatio-temporal action localization. In *NIPS*.

Simonyan, K., Zisserman, A. (2014a). Two-stream convolutional networks for action recognition in videos. In *NIPS*.

Simonyan, K., Zisserman, A. (2014b). Very deep convolutional networks for large-scale image recognition. abs/1409.1556.

Soomro, K., Zamir, A. R., Shah, M. (2012). Ucf101: A dataset of 101 human actions classes from videos in the wild. arXiv:1212.0402.

Srivastava, N., Mansimov, E., Salakhutdinov, R. (2015). Unsupervised learning of video representations using LSTMs. arXiv:1502.04681.

Tang, K., Fei-Fei, L., Koller, D. (2012). Learning latent temporal structure for complex event detection. In *CVPR*.

Tian, Y., Sukthankar, R., Shah, M. (2013). Spatiotemporal deformable part models for action detection. In *CVPR*.

Tieleman, T., & Hinton, G. (2012). Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural Networks for Machine Learning*, *4*(2), 26–31.

Tong, W., Yang, Y., Jiang, L., Yu, S.-I., Lan, Z., Ma, Z., et al. (2014). E-lamp: Integration of innovative ideas for multimedia event detection. *Machine Vision and Applications*, *25*(1), 5–15.

Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M. (2015) C3d: Generic features for video analysis. arXiv:1412.0767.

Vahdat, A., Gao, B., Ranjbar, M., Mori, G. (2011). A discriminative key pose sequence model for recognizing human interactions. In *VS*.

Wang, P., Cao, Y., Shen, C., Liu, L., Shen, H. T. (2015). Temporal pyramid pooling based convolutional neural networks for action recognition. arXiv:1503.01224.

Wang, H., Kläser, A., Schmid, C., Liu, C.-L. (2011). Action recognition by dense trajectories. In *CVPR*.

Wang, H., Schmid, C. (2013). Action recognition with improved trajectories. In *IEEE International Conference on Computer Vision, Sydney*.

Weinland, D., Ronfard, R., Boyer, E. (2010). A survey of vision-based methods for action representation, segmentation and recognition. In *CVIU*, *115*(2), (pp. 224–241).

Xu, K. et al. (2015). Show, attend and tell: Neural image caption generation with visual attention. arXiv:1502.03044.

Yamato, J., Ohya, J., Ishii, K. (1992). Recognizing human action in time-sequential images using hidden markov model. In *CVPR*.

Yao, L., Torabi, A., Cho, K., Ballas, N., Pal, C., Larochelle, H., & Courville, A. (2015). Video description generation incorporating spatio-temporal features and a soft-attention mechanism. arXiv:1502.08029.

Zha, S., Luisier, F., Andrews, W., Srivastava, N., Salakhutdinov, R. (2015). Exploiting image-trained cnn architectures for unconstrained video classification. arXiv:1503.04144.