Learning Latent Temporal Structure for Complex Event Detection*

Kevin Tang Li Fei-Fei Daphne Koller
Computer Science Department, Stanford University
{kdtang,feifeili,koller}@cs.stanford.edu

Abstract

In this paper, we tackle the problem of understanding the temporal structure of complex events in highly varying videos obtained from the Internet. Towards this goal, we utilize a conditional model trained in a max-margin framework that is able to automatically discover discriminative and interesting segments of video, while simultaneously achieving competitive accuracies on difficult detection and recognition tasks. We introduce latent variables over the frames of a video, and allow our algorithm to discover and assign sequences of states that are most discriminative for the event. Our model is based on the variable-duration hidden Markov model, and models durations of states in addition to the transitions between states. The simplicity of our model allows us to perform fast, exact inference using dynamic programming, which is extremely important when we set our sights on being able to process a very large number of videos quickly and efficiently. We show promising results on the Olympic Sports dataset [16] and the 2011 TRECVID Multimedia Event Detection task [18]. We also illustrate and visualize the semantic understanding capabilities of our model.

1. Introduction

With the advent of Internet video hosting sites such as YouTube, personal Internet videos are now becoming extremely popular. There are numerous challenges associated with the understanding of these types of videos; we focus on the task of complex event detection. In our problem definition, we are given Internet videos labeled with an event class, where the label specifies the complex event that occurs within the video. This is a weakly-labeled setting, as we are not given *temporally localized* videos. This means that the event can occur anywhere within the video, and we do not have temporal segmentations that indicate the time points at which the event occurs. The *detection* aspect of our problem manifests itself at the video level, where in the testing phase, we are also given large numbers of irrelevant videos, and must *detect* videos that correspond to events of interest. This is in contrast to the typical detection task of localizing the event within the video.

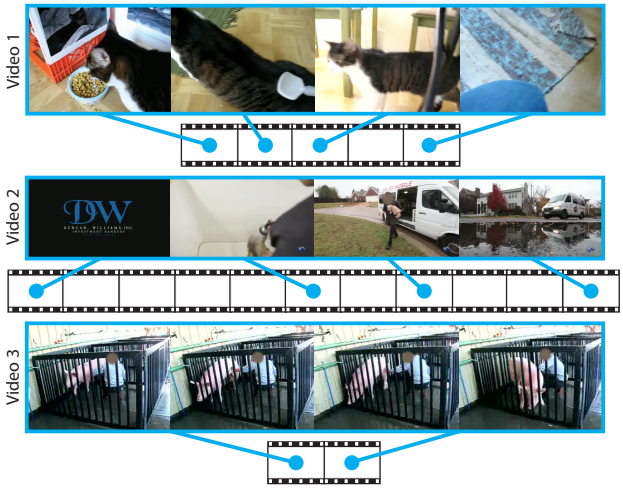

Figure 1. Examples of Internet videos for the event of "Grooming an animal" from the TRECVID MED dataset [18] that illustrate the variance in video length and temporal localization of the event. Video 3 is the only video similar to sequences typically seen in activity recognition tasks, where the event occupies the video in full.

Of the difficulties presented by Internet videos, we focus on two points that have been largely ignored by recent computer vision algorithms. First, there is a large number of videos available on the Internet, creating the need for algorithms that are able to efficiently index and process this wealth of data. Secondly, there is a large amount of variance in these videos, ranging from differences in low-level pro-

*Supported by the Defense Advanced Research Projects Agency under Contract No. HR0011-08-C-0135 and by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior National Business Center contract number D11PC20069. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of DARPA, IARPA, DoI/NBC, or the U.S. Government.

cessing such as length and resolution, to high-level concepts such as activities, events, and contextual information. In addition, there is high intra-class variance when trying to assign class labels to these types of videos, as more often than not the videos are not temporally localized, and will contain varying amounts of contextual or unrelated segments.

These points have not been addressed by much of the recent research on activity recognition and event detection [6, 22]. Although some of the recent works have considered Internet videos, complex activity recognition tasks are typically already temporally localized [13, 16], and event detection tasks focus only on localizing well-defined primitive events [9]. In addition, few of these works deal with large-scale classification.

In order to successfully classify these types of videos, we formulate a model over the temporal domain that is able to discriminatively learn the transitions between events of interest, as well as the durations of these events. We reiterate the challenges associated with complex event detection in Internet videos and highlight key contributions of our model that address these issues:

Extremely large number of difficult videos. Using dynamic programming, our model is able to perform efficient, exact inference, and our max-margin learning framework is based on the linear kernel Support Vector Machine (SVM), which can be optimized very quickly using LIBLINEAR [3]. Together, the inference and learning procedures allow us to process large numbers of videos very quickly. Also, the discriminative nature of our learning enables us to obtain competitive classification results on difficult datasets.

Large amounts of variation in video length. Several previous methods that attempt to model temporal structure assume a video to be of normalized length [12, 16]. However, this is an unrealistic assumption, as the frame rates of the videos are generally on the same scale. Regardless of the duration of a video, a simple motion should still occupy the same number of frames. Our model is able to account for this by representing videos as sequences of fixed length temporal segments.

Weakly-labeled complex events that are not temporally localized. Our model is flexible and allows for sequenced states of interest to transition and occur anywhere within a video, which is crucial for the weakly-labeled setting. The appearance, transitions, and durations of these states are automatically learned with only a class label for the video. In addition, the states can also correspond to semantically meaningful concepts, such as distinguishing between sequences of frames that are relevant and irrelevant for an event of interest.

In summary, the contributions of this paper are two-fold. First, we identify several challenges and difficulties associated with complex event detection in Internet videos, a task of growing importance. And secondly, we formulate a discriminative model that is able to address these issues, and show promising results on difficult datasets.

2. Related Work

We review related work on Hidden semi-Markov Models (HSMMs), Conditional Random Fields (CRFs), and discriminative temporal segments in the context of video, and refer the reader to a recent survey in the area by Turaga *et al*. [24] for a comprehensive review.

HSMMs [2, 8, 14], CRFs [19, 23], semi-CRFs [20], and similar probabilistic frameworks [1] have been previously used to model the temporal structure of videos and text. However, these works differ from ours in that they are applied to different domains such as surveillance video and gesture recognition, and typically require the states to not be latent in order for the models to work. In addition, many of these models were not formulated with large-scale classification in mind, and have complex inference procedures.

Most similar to our method are recent works in video that learn discriminative models over temporal segments [12, 15, 16, 21]. Satkin & Hebert [21] and Nguyen *et al*. [15] attempt to discover the most discriminative portions or segments of videos. Laptev *et al*. [12] divide videos into rigid spatio-temporal bins and compute separate feature histograms from each bin to capture a rough temporal ordering of features. Niebles *et al*. [16] represent videos as temporal compositions of motion segments, and learn appearance models for each of the segments. Their model is tree structured, and assumes fixed anchors for each motion segment, penalizing segments that occur at a distance from their anchors. Our work is different from these previous methods in that in addition to discovering discriminative segments of video, we also model and learn the transitions between and durations of these segments with a chain structured model. Whereas [16] heuristically fixes the anchor points and durations of their temporal segments before training, our approach is completely model-based, and learns all parameters for our transition and duration distributions. There has also been a separate line of work that seeks to model temporal segments of video with the use of additional annotations [5, 7], which we do not require.

Drawing upon recent successes in the field, our model leverages the Bag-of-Words (BoW) feature representation and max-margin learning. Advances in feature representations have utilized the BoW model with discriminative classifiers to achieve state-of-the-art results on popular video datasets [10, 26]. The representation has also been successfully used with semi-latent topic models [28] and unsupervised generative models [17]. We learn parameters for our model using the max-margin framework, which has recently become very popular for latent variable models through the

introduction of general learning frameworks [4, 29].

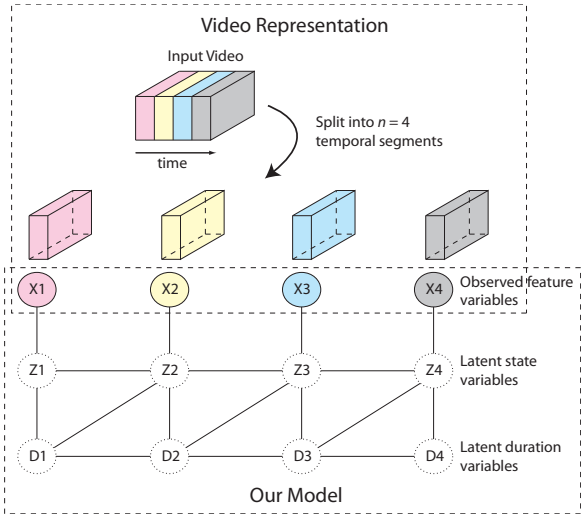

Figure 2. Given an input video, our algorithm divides it into temporal segments and builds a structured temporal model on top of the features. The nodes of the graph represent variables, while the edges denote conditional dependencies between variables. The state variables and duration variables are latent, meaning that they are not observed in training or testing.

3. Our Model

Our model for videos is the conditional variant of the variable-duration hidden Markov model (HMM), also referred to as an explicit-duration HMM or a hidden semi-Markov model [2, 14]. We start by introducing our representation for videos, then give intuition for our model by briefly describing the variable-duration HMM.

3.1. Video representation

Given a video, we first divide it into temporal segments of fixed length l_{seg}, which can be seen in Figure 2. By using fixed length segments, we are able to capture the fact that simple motions should occupy similar numbers of frames, and are invariant to the total length of the video. With this division into segments, a video can be represented by n segments, where the number of segments n is proportional to the video length. For each temporal segment i, we then compute BoW histograms $\boldsymbol{x}_i$ over the features in each segment, and treat these histograms as the observed input variables of our temporal model.

3.2. Variable-duration HMM

A traditional approach is to use an HMM to model transitions between states of a video. However, the HMM suffers because it imposes a geometric distribution on the time within a state, which results when a state continuously transitions to itself. To address this, we use the variable-duration HMM, which allows each state to emit a sequence of observations. This means that we must also model the duration of a state, since a state can generate multiple observations before transitioning into another state. We choose to model the duration of a state using a multinomial distribution. The variable-duration HMM is much more appropriate for our application, since we expect a single state to generate several temporal segments of video that are linked together to form a single, coherent action or event. Our hope is that the latent states and their durations will be able to capture semantically meaningful and discriminative concepts that are shared amongst the videos, as in Figure 3. Note that by restricting the states to have a duration of one, we obtain the standard HMM as a specific instance of the variable-duration HMM.

The conditional variant of the variable-duration HMM is similar to a hidden chain CRF [19]. The difference is in the duration variables, which form an additional chain structure beneath the hidden chain CRF as seen in Figure 2. Since all the v-structures in the conditional variant are moralized, the independencies of the two models are equivalent. Mapping the model onto our video representation, we introduce a latent state for each temporal segment of a video as shown in Figure 2. Since these are latent variables, we are not given labels for them during training or testing.

3.3. Model representation

In our model, there are three types of potentials that define the energy of a particular sequence assignment to the latent state variables $\boldsymbol{z} = \{z_1, z_2, \ldots, z_n\}$ and duration variables $\boldsymbol{d} = \{d_1, d_2, \ldots, d_n\}$ as shown in Figure 2. Intuitively, the duration variable acts as a counter, and decreases after each consecutive state assignment until it reaches zero, after which a new state transition can be made. While it is counting down, the state assignment is not allowed to change. We assume that we are given the maximum duration $d_{\max}$ for all states and the number of states S for our model. The potentials are defined in terms of parameters $\boldsymbol{w}$ of our model that will be learned.

The first potential is a singleton appearance potential on the latent state variables that measures the similarity of the feature histogram $\boldsymbol{x}_i$ for temporal segment i to its assigned state z_i.

$$\psi^a(Z_i = z_i) = \boldsymbol{w}^a_{z_i} \cdot \boldsymbol{x}_i \tag{1}$$

The second potential encompasses both the state and duration variables, and measures the score of transitioning between states, provided we are allowed to transition:

$$\begin{aligned}\psi^t(Z_i =& z_i, Z_{i-1} = z_{i-1}, D_{i-1} = d_{i-1}) = \\ & -\infty \cdot \mathbf{1}[d_{i-1} > 0, z_i \neq z_{i-1}] \\ & + w^t_{z_{i-1}, z_i} \cdot \mathbf{1}[d_{i-1} = 0]\end{aligned} \tag{2}$$

The third potential measures the score of a given duration, provided we are entering a new state:

$$\begin{aligned}\psi^d(Z_i =& z_i, D_i = d_i, D_{i-1} = d_{i-1}) = \\ & -\infty \cdot \mathbf{1}[d_{i-1} > 0, d_i \neq d_{i-1} - 1] \\ & + w^d_{z_i,d_i} \cdot \mathbf{1}[d_{i-1} = 0] \end{aligned} \quad (3)$$

Together, these potentials define the energy of a particular sequence assignment of variables $\boldsymbol{z}$ and $\boldsymbol{d}$ to our model:

$$\begin{aligned}E(\boldsymbol{z}, \boldsymbol{d}|\boldsymbol{w}) = & \sum_i (\psi^a(Z_i = z_i) \\ & + \psi^t(Z_i = z_i, Z_{i-1} = z_{i-1}, D_{i-1} = d_{i-1}) \\ & + \psi^d(Z_i = z_i, D_i = d_i, D_{i-1} = d_{i-1})) \end{aligned} \quad (4)$$

where we initialize $\psi^t(Z_1, Z_0, D_0) = 0$ and $D_0 = 0$.

4. Inference

Exact maximum a posteriori (MAP) inference for our model can be done efficiently using dynamic programming. In MAP inference, we must find the sequence of states $\boldsymbol{z}$ and durations $\boldsymbol{d}$ that maximize the energy function given above in equation 4. This can be done using a recurrence relation that computes the best possible score given that temporal segment j is assigned to state i. The score is computed by searching over all possible durations d and previous states s, assuming that segment j is the last segment in the duration of state i. We can use the following recurrence relation for inference:

$$\begin{aligned}V_{i,j} = \max_{\substack{d \in \{1 \ldots d_{\max}\} \\ s \in \{1 \ldots S\}}} & [\boldsymbol{w}^a_i \cdot (\sum_{k=j-d+1}^{j} \boldsymbol{x}_k) \\ & + w^t_{s,i} + w^d_{i,d} + V_{s,j-d}] \end{aligned} \quad (5)$$

After building up the table of scores V, we can then recover the optimal assignments by backtracking through the table. The runtime complexity for this inference algorithm is $O(n_{\max} d_{\max} S^2)$, where $n_{\max}$ is the maximum number of temporal segments in all videos. By utilizing structure in the duration variables, our inference algorithm achieves a complexity that is linear in $d_{\max}$, whereas a naive implementation would have quadratic dependence.

5. Learning

There are three sets of parameters that we must learn in our model, the appearance parameters $\boldsymbol{w}^a$, the transition parameters $\boldsymbol{w}^t$, and the duration parameters $\boldsymbol{w}^d$, which we can concatenate into a single weight vector:

$$\boldsymbol{w} = [\boldsymbol{w}^a \quad \boldsymbol{w}^t \quad \boldsymbol{w}^d] \quad (6)$$

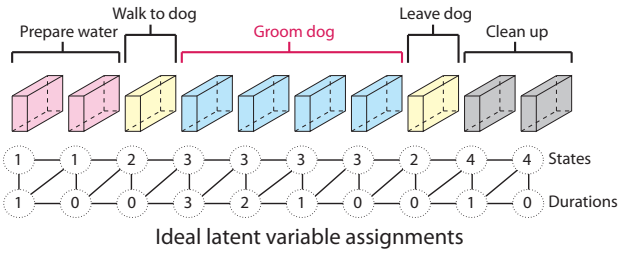

Figure 3. This figure shows the ideal assignments to latent states and durations for a sequence with a known temporal segmentation that we hope our model is able to achieve. By understanding the temporal structure of the video, we are able to classify it as containing the event "Grooming an animal".

Given a training set of N videos and their corresponding binary class labels $y_i \in \{-1, 1\}$, we can compute their feature representations to obtain our dataset $(\langle v_1, y_1 \rangle, ..., \langle v_N, y_N \rangle)$. To learn our parameters, we adopt the binary Latent SVM framework of Felzenszwalb *et al.* [4], which is a specific instance of the Latent Structural SVM with a hinge loss function [29]. The objective we would like to minimize is given by:

$$\min_{\boldsymbol{w}} \frac{1}{2}\|\boldsymbol{w}\|^2 + C \sum_{i=1}^{N} \max(0, 1 - y_i f_{\boldsymbol{w}}(v_i)) \quad (7)$$

where we consider linear classifiers of the form:

$$f_{\boldsymbol{w}}(v) = \max_{\boldsymbol{h}} \boldsymbol{w} \cdot \Phi(v, \boldsymbol{h}) \quad (8)$$

The latent variables $\boldsymbol{h}$ in the classifier are solved for by performing MAP inference on the example v to find the state and duration assignments. Using these assignments, we can construct the feature vector $\Phi(v, \boldsymbol{h})$ for an example v as follows. For the $\boldsymbol{w}^a$ parameters we sum the feature histograms that are assigned to each state, and for the $\boldsymbol{w}^t$ and $\boldsymbol{w}^d$ parameters we count the number of times each state transition and duration occurs. We then normalize each of these features and concatenate them together to form the feature vector $\Phi(v, \boldsymbol{h})$.

The objective function is minimized using the Concave-Convex Procedure (CCCP) [30]. This leads to an iterative algorithm in which we alternate between inferring the latent variables $\boldsymbol{h}$, and optimizing the weight vector $\boldsymbol{w}$. Once the latent variables are inferred and the feature vectors $\Phi(v, \boldsymbol{h})$ are constructed for each example, optimizing the weight vector becomes the standard linear kernel SVM problem, which can be solved very efficiently using LIBLINEAR [3]. This process is repeated for several iterations until convergence or a maximum number of iterations is reached.

5.1. Initialization

In our model, we must initialize the latent states of the temporal segments as well as their durations for each of our

Sport Class	Niebles *et al*. [16]	Our Method
high-jump	**27.0%**	18.4%
long-jump	71.7%	**81.8%**
triple-jump	10.1%	**16.1%**
pole-vault	**90.8%**	84.9%
gymnastics-vault	**86.1%**	85.7%
shot-put	37.3%	**43.3%**
snatch	54.2%	**88.6%**
clean-jerk	70.6%	**78.2%**
javelin-throw	**85.0%**	79.5%
hammer-throw	**71.2%**	70.5%
discus-throw	47.3%	**48.9%**
diving-platform	**95.4%**	93.7%
diving-springboard	**84.3%**	79.3%
basketball-layup	82.1%	**85.5%**
bowling	53.0%	**64.3%**
tennis-serve	33.4%	**49.6%**
Mean AP	62.5%	**66.8%**

Table 1. Average Precision (AP) values for classification on the Olympic Sports dataset [16].

training examples, subject to the constraint that we have S states we can assign and a maximum duration $d_{\max}$. For each video, we begin by initializing each segment to its own state. Then, we use Hierarchical Agglomerative Clustering to merge adjacent segments. This is done by computing the Euclidean distance between feature histograms of all adjacent segments, and repeatedly merging segments with the shortest distance. The number of merges for a given video is fixed to be half the number of segments in the video.

Then, using all the videos, we run k-means clustering to cluster all the states into S clusters, and assign latent states according to their cluster assignments. This gives us the assignments $\boldsymbol{z}$ for the states. We initialize the duration variables by assuming that all consecutive assignments of the same state are a single state assignment with duration equal to the number of consecutive assignments.

6. Experiments

We test our model on two difficult tasks: activity recognition and event detection. In both scenarios, we are only given class labels for the videos. We use the Olympic Sports dataset [16] and the 2011 TRECVID Multimedia Event Detection (MED) dataset [18]. For both datasets, we compare our model to state-of-the-art baselines that consider temporal structure, using the same features for all models.

In our experiments, we use 5-fold cross validation for model selection to select the number of latent states and the C parameter for the SVM. We set the maximum duration to be the average video length, and set the length of temporal segments based on the dataset and density of our sampled features. For the Olympic Sports dataset, we used 20 frames per segment, and for the MED dataset, we used 100 frames per segment. We train a model for each class, and report average precision (AP) numbers on the datasets.

6.1. Activity recognition

Dataset. The Olympic Sports dataset [16] consists of 16 different sport classes of Olympic Sports activities that contain complex motions going beyond simple punctual or repetitive actions. The sequences are collected from YouTube, and class label annotations obtained using Amazon Mechanical Turk. An important point to note is that the sequences are already temporally localized.

Comparisons. We compare our model to the method of decomposable motion segments [16], which achieves state-of-the-art results using local features. Because much of their performance derives from including a BoW histogram over the entire video in their feature vector, we follow protocol and concatenate the BoW histogram to the end of our feature vector $\Phi(v, \boldsymbol{h})$ before classification. For the feature representation, we use the same features used in [16], which consists of an interest point detector [11] and concatenated Histogram of Gradient (HOG) and Histogram of Flow (HOF) descriptors [12]. In addition, because [16] uses a χ^2-SVM, we use the method of additive kernels [25] to approximate a χ^2 kernel for our BoW features to maintain efficient processing while increasing discriminative power. Because the public release of this dataset is not the full dataset used in the paper [16], we obtained results for their model on the public release through personal communication with the authors. The results are given in Table 1.

Results. We obtain better AP numbers for 9 of the 16 classes, as well as better overall mean AP compared to the state-of-the-art baseline model. The promising performance on this dataset shows that, given well-localized videos, our model is able to capture the fine structure between temporal segments that define a complex activity.

Observing the latent states that our model learns, we find that there are three key components that allow us to do better than [16]. First, our model is flexible and allows latent states to appear anywhere within a sequence without penalty. In the "snatch" sequences, the assignment of the first latent state varies approximately equally between two different states. This helps to capture the variability that accompanies the start of a "snatch" sequence, such as differences in preparatory motions of the athletes. The baseline model is unable to easily account for this, as it has a fixed anchor for its segments, and so the beginning of each sequence is almost always modeled by the same segment. The second component is the effect of modeling the duration of the segments. For the same latent state, the durations of the state can vary greatly from sequence to sequence. In some cases, our model is able to realize that the sequence is extremely short and already very discriminative, and assigns

Event Class	Chance	Niebles *et al*. [16]	Laptev *et al*. [12]	Our Method, $d_{\max} = 1$	Our Method
Attempting a board trick	1.18%	5.84%	8.22%	6.24%	**15.44%**
Feeding an animal	1.06%	2.28%	2.45%	**5.28%**	3.55%
Landing a fish	0.89%	9.18%	9.77%	7.30%	**14.02%**
Wedding ceremony	0.86%	7.26%	5.52%	9.48%	**15.09%**
Working on a woodworking project	0.93%	4.05%	4.09%	3.42%	**8.17%**
Mean AP	0.98%	5.72%	6.01%	6.34%	**11.25%**

Table 2. Average Precision (AP) values for detection on the MED DEV-T dataset.

Event Class	Chance	Niebles *et al*. [16]	Laptev *et al*. [12]	Our Method, $d_{\max} = 1$	Our Method
Birthday party	0.54%	2.25%	1.93%	1.97%	**4.38%**
Changing a vehicle tire	0.35%	0.76%	0.98%	**1.01%**	0.92%
Flash mob gathering	0.42%	8.30%	7.60%	7.58%	**15.29%**
Getting a vehicle unstuck	0.26%	1.95%	1.73%	1.82%	**2.04%**
Grooming an animal	0.25%	0.74%	0.72%	0.73%	**0.74%**
Making a sandwich	0.43%	**1.48%**	1.09%	0.80%	0.84%
Parade	0.58%	2.65%	3.77%	**4.17%**	4.03%
Parkour	0.32%	2.05%	1.95%	1.65%	**3.04%**
Repairing an appliance	0.27%	4.39%	1.54%	1.38%	**10.88%**
Working on a sewing project	0.26%	0.61%	1.18%	0.91%	**5.48%**
Mean AP	0.37%	2.52%	2.25%	2.20%	**4.77%**

Table 3. Average Precision (AP) values for detection on the MED DEV-O dataset.

the same state to the entire sequence. This is not allowed in the baseline model, as the lengths of the motion segments are pre-specified parameters. Finally, our model is able to discard unnecessary states and represent most of the sport classes with fewer than 3 states. The baseline model is optimally trained with 6 motion segments, and forces sequences into the temporal structure of its segments, causing the optimization to easily overfit.

We note that our model performs poorly in the "high-jump" and "triple-jump" classes. The reason for this can be attributed to the weak discriminative power of the features extracted from these videos. Visualizing the latent states learned for the "high-jump" class, we find that there are a large number of videos that are all assigned to a single state. This occurs because the underlying BoW histograms at the segment level are too similar, and so our model tends to group them together into a single duration. In addition, the number of videos is skewed for several of the classes, and "triple-jump" is one of the classes with fewer examples in both training and testing, which makes it hard for both discriminative models to learn meaningful parameters.

6.2. Event detection

Dataset. The 2011 TRECVID MED dataset [18] consists of a collection of Internet videos collected by the Linguistic Data Consortium from various Internet video hosting sites. There are 15 events, and they are split into two sets, the DEV-T set and the DEV-O set. The DEV-T set consists of the 5 events "Attempting a board trick", "Feeding an animal", "Landing a fish", "Wedding Ceremony", and "Working on a woodworking project". The DEV-O set consists of the 10 events "Birthday party", "Changing a vehicle tire", "Flash mob gathering", "Getting a vehicle unstuck", "Grooming an animal", "Making a sandwich", "Parade", "Parkour", "Repairing an appliance", and "Working on a sewing project".

The task, although termed event detection, is more similar to that of a retrieval task. We are given approximately 150 training videos for each event, and in the two testing sets for DEV-T and DEV-O, we are given large databases of videos that consist of both the events in the set as well as null videos that correspond to no event. The null videos significantly decrease the chance AP, causing our resulting numbers to be very low. There are a total of 10,723 videos in the DEV-T test set, and 32,061 videos in the DEV-O test set. In the TRECVID task, the DEV-T set is used for development, while the DEV-O set is used for evaluation. We consider the two sets separately, as it is stated that there may be unidentified positive videos of events from the DEV-T set in the DEV-O test set, and vice versa.

Comparisons. We compare our models to strong baseline methods that can capture temporal structure of local features through decomposable motion segments [16], and rigid spatio-temporal bins [12]. For the feature representation, we extract dense HOG3D features [10, 27], and use a linear kernel SVM for all models. To illustrate the effect of the duration variables, we also train a version of our model with the duration variable set to one, corresponding to a standard hidden chain CRF [19]. Results for the MED datasets are given in Table 2 and Table 3 for the DEV-T and

DEV-O sets, respectively.

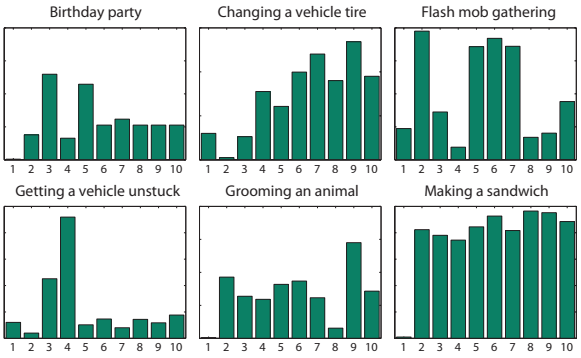

Figure 4. Examples of duration parameters learned for events in the MED dataset. The x-axes are values of the duration parameters, and the height of the bars represent the strength of the parameter, which is averaged over all states of the model.

Effect of duration variables. In a few rare cases, the hidden chain CRF is able to outperform our model by a small margin. This can occur because for some events, the videos that contain them vary between different types of motions very quickly, and so the duration variables will sometimes mistakenly merge these variations into a single state. In relation to the bias-variance tradeoff, the low variance and high bias of the hidden chain CRF allow it to generalize better for certain events. In theory, any model learned using the hidden chain CRF can be learned using our duration model as well, by learning large negative parameters for durations greater than one. However, this does not always occur as the duration variables are initialized to different values, and the inference procedures score assignments differently. On the other hand, the increased performance of the hidden chain CRF also speaks well for our model, as it shows that through better initializations and model selection techniques, it is possible to achieve even better accuracies.

Visualizing the parameters learned for the duration variables, we find that the duration variables are commonly utilized for states that correspond to the contextual and irrelevant portions of videos, as they typically occupy large numbers of consecutive temporal segments. In Figure 4, we show examples of the multinomial duration parameters learned for events in the MED dataset. A hidden chain CRF that imposes a geometric distribution would have a large parameter for the duration of 1, and small parameters for all other durations. Our models learn duration parameters in favor of non-geometric distributions, which suggests that the videos are better modeled with state durations.

Results. Our model achieves the best results for both MED datasets, and achieves significant gains in AP for most of the events. Much of the analysis from the previous section on activity recognition holds for these datasets as well. By

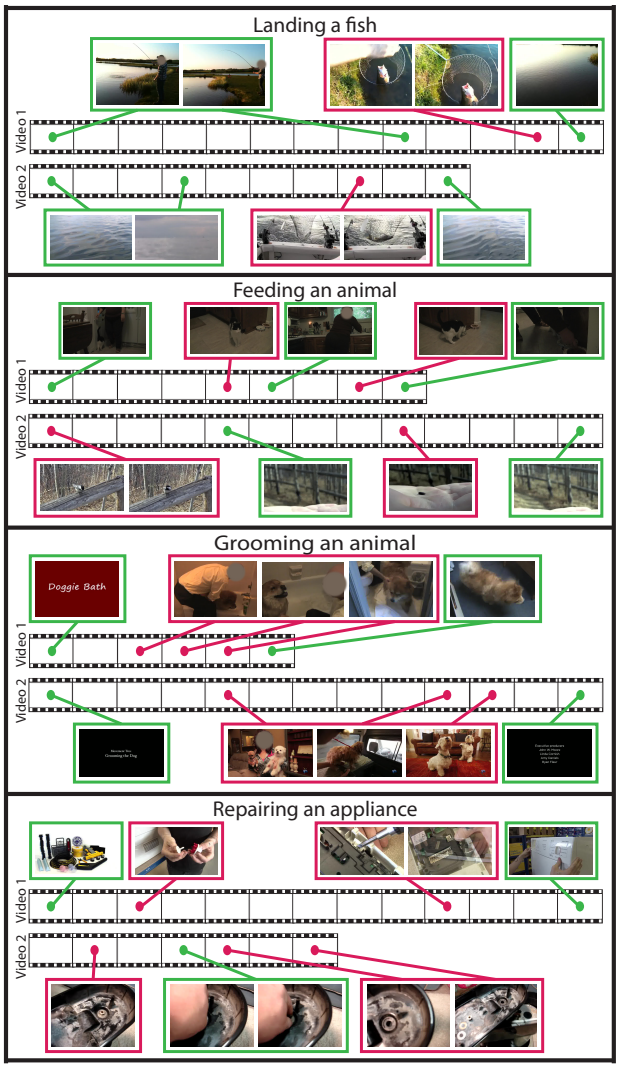

Figure 5. Example inference results on two different videos for four of our models learned on the MED dataset. The red and green boxes represent different latent states that are the same across the two videos, but different across models. Our models are able to learn the transitions and durations of states, and successfully discover discriminative segments at varying points in videos of differing length. This figure is best viewed magnified and in color.

learning state assignments that can occur at any temporal location and by modeling their durations, our model is able to successfully capture the temporal structure of these highly varying Internet videos, as seen in Figure 5. These properties are crucial in MED videos, as events are not temporally localized and there is a large number of contextual segments that we must model. For example, in the "Feeding an animal" visualizations in Figure 5, discriminative segments occur at completely different points in time for the two videos. The fixed structure of the baseline models makes it unable for them to capture the varied temporal structure of these

videos, as they treat segments at the same relative locations of two videos to be the same.

Latent semantic understanding. In addition to achieving competitive accuracies on difficult datasets, our model is also able to capture semantic concepts in the latent states. We find that in many instances, temporal segments assigned to the same latent state are related in semantic content. This occurs at varying locations across different videos, and is shown in Figure 5. The "Landing a fish" class is a particularly nice illustration of this, as we can typically identify a state that corresponds to the actual catching of the fish.

7. Conclusion

In this paper we have introduced a model for learning the latent temporal structure of complex events in Internet videos. Our model is simple, and lends itself to fast, exact inference, which allows us to process large numbers of videos efficiently. In addition, we train our model in a discriminative, max-margin fashion and are able to achieve competitive accuracies on activity recognition and event detection tasks. We've shown competitive results on difficult datasets, as well as examples of semantic structure that our model is able to automatically extract.

Possible directions for future work include incorporating spatial structure into our model. We have tackled temporal understanding of the structure of complex events, but being able to learn spatial structure as well is another step towards our overarching goal of holistic video understanding. Another possible direction is using the semantic understanding capabilities of our model for video summarization.

Acknowledgments. We thank Tianshi Gao and Juan Carlos Niebles for helpful discussions. We also thank Olga Russakovsky, Dave Jackson, and Wei Zeng for helpful comments on the paper.

References

[1] M. Albanese, R. Chellappa, N. P. Cuntoor, V. Moscato, A. Picariello, V. S. Subrahmanian, and O. Udrea. A constrained probabilistic petri net framework for human activity detection in video. *IEEE TMM*, 2008. 2

[2] T. V. Duong, H. H. Bui, D. Q. Phung, and S. Venkatesh. Activity recognition and abnormality detection with the switching hidden semi-markov model. In *CVPR*, 2005. 2, 3

[3] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. LIBLINEAR: A library for large linear classification. *JMLR*, 2008. 2, 4

[4] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *IEEE TPAMI*, 2010. 3, 4

[5] A. Gaidon, Z. Harchaoui, and C. Schmid. Actom Sequence Models for Efficient Action Detection. In *CVPR*, 2011. 2

[6] L. Gorelick, M. Blank, E. Shechtman, M. Irani, and R. Basri. Actions as space-time shapes. *IEEE TPAMI*, 2007. 2

[7] M. Hoai, Z.-Z. Lan, and F. De la Torre. Joint segmentation and classification of human actions in video. In *CVPR*, 2011. 2

[8] S. Hongeng and R. Nevatia. Large-scale event detection using semi-hidden markov models. In *ICCV*, 2003. 2

[9] Y. Ke, R. Sukthankar, and M. Hebert. Event detection in crowded videos. In *ICCV*, 2007. 2

[10] A. Kläser, M. Marszałek, and C. Schmid. A spatio-temporal descriptor based on 3d-gradients. In *BMVC*, 2008. 2, 6

[11] I. Laptev. On space-time interest points. *IJCV*, 2005. 5

[12] I. Laptev, M. Marszaek, C. Schmid, B. Rozenfeld, I. Rennes, I. I. Grenoble, and L. Ljk. B.: Learning realistic human actions from movies. In *CVPR*, 2008. 2, 5, 6

[13] J. Liu, J. Luo, and M. Shah. Recognizing realistic actions from videos "in the wild". *CVPR*, 2009. 2

[14] P. Natarajan and R. Nevatia. Coupled hidden semi markov models for activity recognition. In *WMVC*, 2007. 2, 3

[15] M. H. Nguyen, L. Torresani, F. De la Torre, and C. Rother. Weakly supervised discriminative localization and classification: a joint learning process. In *ICCV*, 2009. 2

[16] J. C. Niebles, C.-W. Chen, and L. Fei-Fei. Modeling temporal structure of decomposable motion segments for activity classification. In *ECCV*, 2010. 1, 2, 5, 6

[17] J. C. Niebles, H. Wang, and L. Fei-Fei. Unsupervised learning of human action categories using spatial-temporal words. *IJCV*, 2008. 2

[18] P. Over, G. Awad, M. Michel, J. Fiscus, W. Kraaij, A. F. Smeaton, and G. Quenot. Trecvid 2011 – an overview of the goals, tasks, data, evaluation mechanisms and metrics. In *Proceedings of TRECVID 2011*. NIST, USA, 2011. 1, 5, 6

[19] A. Quattoni, S. Wang, L.-P. Morency, M. Collins, and T. Darrell. Hidden-state conditional random fields. In *IEEE TPAMI*, 2007. 2, 3, 6

[20] S. Sarawagi and W. W. Cohen. Semi-markov conditional random fields for information extraction. In *NIPS*, 2004. 2

[21] S. Satkin and M. Hebert. Modeling the temporal extent of actions. In *ECCV*, 2010. 2

[22] C. Schüldt, I. Laptev, and B. Caputo. Recognizing human actions: A local svm approach. In *ICPR*, 2004. 2

[23] C. Sminchisescu, A. Kanaujia, and D. Metaxas. Conditional models for contextual human motion recognition. *CVIU*, 2006. 2

[24] P. K. Turaga, R. Chellappa, V. S. Subrahmanian, and O. Udrea. Machine recognition of human activities: A survey. *IEEE TCSVT*, 2008. 2

[25] A. Vedaldi and A. Zisserman. Efficient additive kernels via explicit feature maps. In *CVPR*, 2010. 5

[26] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu. Action recognition by dense trajectories. In *CVPR*, 2011. 2

[27] H. Wang, M. M. Ullah, A. Kläser, I. Laptev, and C. Schmid. Evaluation of local spatio-temporal features for action recognition. In *BMVC*, 2009. 6

[28] Y. Wang and G. Mori. Human action recognition by semi-latent topic models. *IEEE TPAMI*, 2009. 2

[29] C.-N. J. Yu and T. Joachims. Learning structural svms with latent variables. In *ICML*, 2009. 3, 4

[30] A. L. Yuille. The concave-convex procedure. In *NIPS*, 2002. 4