Extracting Key Segments of Videos for Event Detection by Learning From Web Sources

Hao Song, Xinxiao Wu, *Member, IEEE*, Wennan Yu, and Yunde Jia, *Member, IEEE*

***Abstract*—In this paper, we present a novel approach of extracting the key segments for event detection in unconstrained videos. The key segments are automatically extracted by transferring the knowledge learned from Web images and Web videos to consumer videos. We propose an adaptive latent structural support vector machine model, where the locations of key segments in videos are regarded as latent variables due to the unavailability of the ground truth of key-segment locations in training data. In order to alleviate the time-consuming and labor-expensive manual annotation of huge amounts of training videos, a large number of loosely labeled Web images as well as videos are collected from the Web sources. Additionally, a limited number of labeled consumer videos are utilized to guarantee the precision of the model. Considering the semantic diversity of key segments, we learn a set of concepts as the semantic description of key segments and explore the temporal information of concepts to capture the sequential relations between the segments. The concepts are automatically discovered by using Web images and videos with their associated tags and description sentences. Comprehensive experiments on the Columbia's consumer video and the TRECVID 2014 Multimedia Event Detection datasets demonstrate that our method outperforms the state-of-the-art methods.**

***Index Terms*—Event detection, key segments, transfer learning, automatic concept discovery.**

I. Introduction

DETECTING complex events in unconstrained videos is an extremely challenging task due to the arbitrariness of consumer videos in computer vision [1], [2]. The complex events are usually composed of various basic actions, objects and scenes [3]–[5]. A single semantic concept such as the event class label is not sufficient to abstract the contents of complex events. Thus, it is essential to employ multiple semantic concepts as the description of events [6]–[8].

Taking the event of "wedding ceremony" for example, it usually consists of the action concepts of "hugging" and "kissing", the object concept of "wedding dress", and the scene concept of "church". The single concept of "kissing" or "hugging" is not able to interpret this event completely. Moreover, in real scenarios, an event video usually lasts for several minutes or even an hour, so there may be irrelevant or redundant information in video which will negatively impact the understanding of events. In this paper, we try to automatically extract the informative segments from consumer videos for event detection. Considering the temporal relations between key segments in a specific event, it is effective to model the temporal constraints between segments for key segment extraction and event detection.

Since the great intra-class variation exists in event videos, a number of videos are required to achieve the promising results by covering all the possible instances of all the event classes [3], [9], [10]. However, manually annotating the training videos is time-consuming and labor-expensive. To alleviate this problem, we propose to explore rich and loosely labeled Web resources as training data for detecting complex events in consumer videos.

This paper presents an adaptive latent structural SVM to extract key segments for event detection by transferring the discriminative model learned by Web resources to consumer videos, in which the locations of key segments in videos are designed as latent variables. A large number of loosely labeled images (from Flickr and Google) and videos (from Youtube) are utilized in the training phase. Additionally, a limited number of labeled consumer videos are also collected to guarantee the precision of the event detection model. A set of semantic concepts is employed to describe the overall content of a specified event video, and each single semantic concept is chosen as the description of each local video segment. Considering the temporal relationship between different segments represented by the concepts, we develop a Temporal Relation Model (TRM) to exploit the temporal relations between the key segments. We also integrate a Segment-Event Interaction Model (SEIM) into the adaptive latent structural SVM model to evaluate the correlations between the key segments and the specified events.

Since the manually defined concept might fail to represent real-world events without the adaptability to different domains [11], [12], we try to discover the concepts automatically by leveraging the tags and description sentences from Web images and videos. We introduce the N adjacent point sample consensus (NAPSAC) [13] to eliminate the noisy images and videos, and then use the hierarchical clustering [14] to generate the last concepts with their associated Web sources by jointly taking account of the similarity of the textual descriptions and the content of their related resource sets. The detailed framework of our method is shown in Fig. 1.

Manuscript received August 3, 2016; revised January 9, 2017 and September 16, 2017; accepted September 21, 2017. Date of publication October 16, 2017; date of current version April 17, 2018. This work was supported in part by the Natural Science Foundation of China under Grants 61673062 and 61472038. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Sen-Ching Samson Cheung. *(Corresponding author: Xinxiao Wu and Yunde Jia.)*

The authors are with Beijing Laboratory of Intelligent Information Technology and the School of Computer Science, Beijing Institute of Technology, Beijing 100081, China (e-mail: songhao@bit.edu.cn; wuxinxiao@bit.edu.cn; yuwennan@bit.edu.cn; jiayunde@bit.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMM.2017.2763322

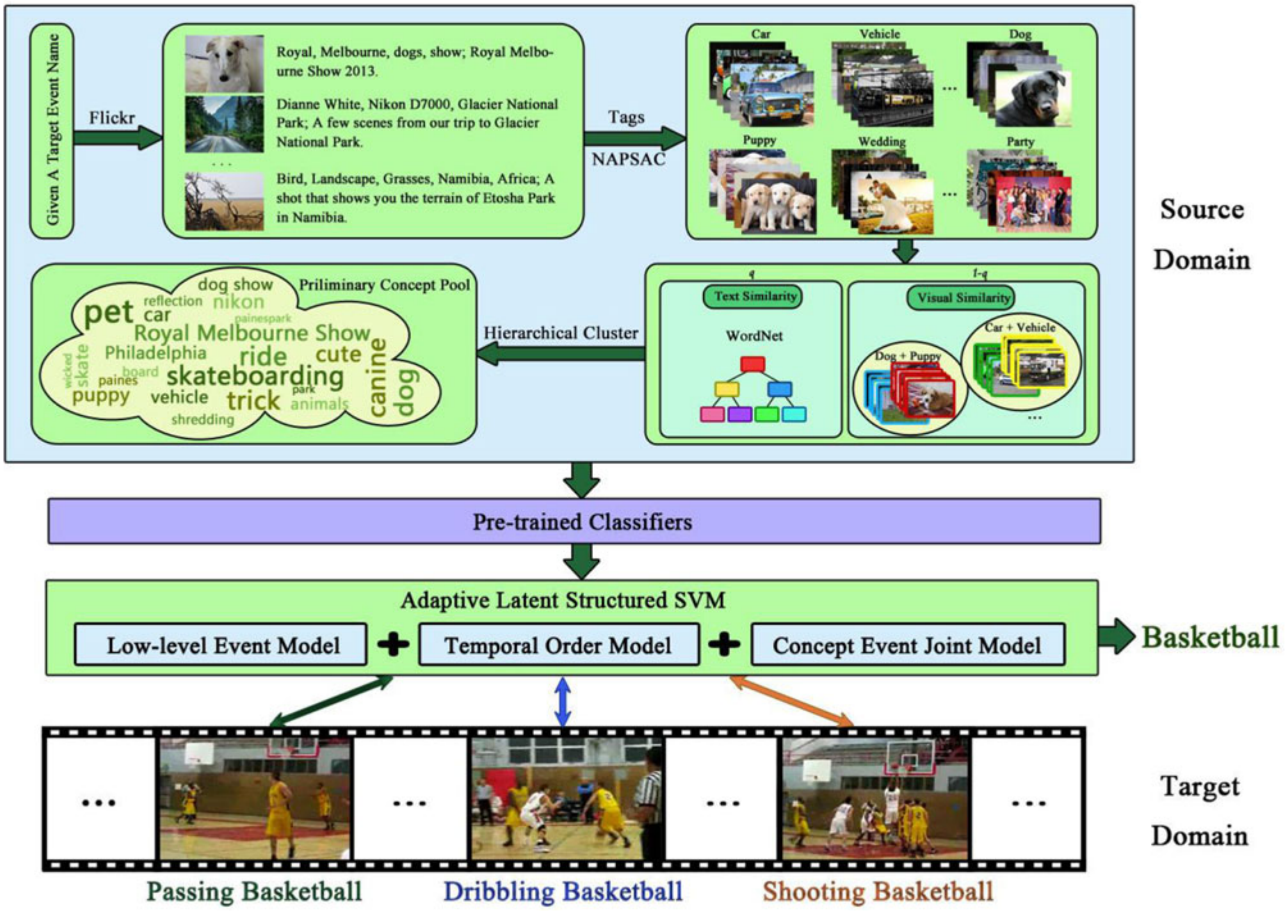


Fig. 1. The framework of the proposed method. We automatically discover concepts by learning from Web images and videos with their associated tags and description sentences. Then the Web sources are used to train the basic concept SVM classifiers. The knowledge learned from the Web sources is transferred to adapt to an optimal target classifier. Also, we explore the temporal relationship to extract the key segments of a video. A discriminative model is learned by using an adaptive latent structural SVM model for high level event classification.

The main contributions of our method are three-fold: (**a**) We propose a novel framework to simultaneously extract the key segments of videos and classify the high-level events. Each segment is described by a concept which is chosen from the automatically discovered concept pool. (**b**) We leverage the knowledge learned from Web videos and images to extract the key segments of videos by exploiting their temporal relationship. (**c**) We combine the NAPSAC and the hierarchical clustering to automatically discover the concepts to extract the segments in videos.

II. Related Work

A. Domain Adaptation for Event Detection

Many researchers draw attention to the topic of detecting complex events in videos using domain adaptation since the limitation number of training examples. Zhang *et al.* [15] leveraged abundant Web images to learn the noise-resistant classifiers for modeling the event-centric semantic concepts. The concepts are encoded in the knowledge base to narrow the semantic gap between complex events. Wang *et al.* [16] presented a set of concept groups to incrementally learn the target classifier, where each concept group consists of the images querying from Web and some simple action videos. Long *et al.* [17] proposed a transfer kernel learning method to learn a domain-invariant kernel by matching source and target distributions in the reproducing kernel Hilbert space. Duan *et al.* [18] proposed a multiple source domain adaption method by selecting the most relevant image source domains for annotating videos. Different from these methods which encode a global video-level feature over the entire video, our method extracts the local segment-based features for recognizing complex events.

B. Extracting Segments of Videos for Event Detection

Due to the complexity of unconstrained videos, many methods focus exploiting the segments of videos for complex event detection. Phan *et al.* [3] divided a video into several segments for feature extraction and classification, using the segment-based approach to produce a video representation for event detection. Li *et al.* [19] detected bursty tweet segments as event segments and clustered the segments into events by considering both the frequency distribution and content similarity of segments. Song *et al.* [20] detected the key segments of complex videos by leveraging image sets. Sun *et al.* [5] proposed an evidence location model to discover the video segments for event classification and recounting. They also leveraged the detected oriented discriminative segments in videos and the descriptions of segments for event detection [21]. Tang *et al.* [22] automatically discovered discriminative and interesting segments of videos on the variable-duration hidden markov model. Li *et al.* [23] leveraged a global dynamic pooling structure to model the temporal relations between segments and the event specific videos. The locations of the segments are also treated as hidden information. Chang *et al.* [24] presented a joint framework which simultaneously detects high-level events and discovers the segments in event videos. An event video is represented as a mid-level concept semantic vector. And all the relationships are built on this concept representations. Different from these segment discovery methods, the concepts in our framework are automatically discovered. Our method introduces a latent structural SVM framework of event detection which utilizes large amounts of loosely labeled Web sources and a few labeled training target videos. Each segment corresponds to one concept, the segment discovery and event detection are processed simultaneously.

C. Collecting Concepts Manually for Event Detection

Many works also try to detect complex events using the concept learning based methods. Mazloom *et al.* [25] proposed an approximate solution to find a set of informative concepts using cross-entropy optimization and they learned the concept prototypes from a set of relevant frames of Web video examples without any annotations [26]. Merler *et al.* [27] presented a mid-level semantic representations with 280 relevant concepts which are trained by labeled Web images. Habibian *et al.* [4] constructed a pool of 1346 concept detectors trained on the ImageNet [28] and TRECVID [29] to create an effective vocabulary for event recognition. Different from the concepts which are manually defined, we present a new algorithm to automatically discover the concepts from loosely labeled Web images with their associated tags.

III. Automatic Concept Discovery

A. Generation of Preliminary Concept Pool

Owing to the rapid growing of Web sources, we try to utilize the textural descriptions of large Web images and videos for generating the preliminary pool of concepts. Particularly, we query images and videos by using the textual descriptions of specified events as keywords. Both the images and videos, which are respectively collected from the Flickr and Youtube Websites, have their own semantic tags or description sentences. Thus we can adopt the Term Frequency-Inverse Document Frequency (TF-IDF) algorithm [30] to extract the keywords from those textural descriptions of Web images and videos. Finally, the preliminary concept pool is generated by collecting all the extracted keywords and selecting only the original tags of Web sources with high frequency. For each tag/keyword, it corresponds to a few images or videos, and for each Web image/video, it relates to several tags/keywords.

Algorithm 1: Pseudo-code of NAPSAC for Eliminating Noisy Web sources.

```
Input: $FS$: Web sources collected from Flickr and Youtube using event textual descriptions at the first stage; $SS$: Web sources queried from Google and Youtube using the concept descriptions in preliminary concept pool at the second stage.
Output: $I$: refined concept training example sets.
For each concept $c$, the related web sources are refined as follows:
    Step 1: Select an initial web source (image/video) $x_1$ from source set $FS_c$.
    Step 2: Find the set of source, $SI_{x_1}$, from the source set $SS_c$ which is lying within a hypersphere of radius $R$ centered on $x_1$.
    Step 3: If the number of sources in $SI_{x_1}$ is less than the prescriptive size then fail. In our experiment, we set the prescriptive size of images as 50 and videos as 10, respectively. Return to Step 1.
    Step 4: Select the sources from $SI_{x_1}$ uniformly until the prescriptive size of the effective source set $I_c$ has been selected, inclusive of $x_1$.
```

B. Refinement of Noisy Concepts and Web Sources

Since the tags and description sentences of Web images/videos are given by various uploaders with subjectivity and arbitrariness, there are a portion of meaningless words that are irrelevant to the event videos. To make the concepts in the preliminary concept pool have reasonable meaning, we simply filter the tags/keywords by choosing the extensive Nouns and Verbs as well as discarding other words because the Nouns and Verbs can effectively represent the semantics of actions for events [31]. Fig. 2 shows some examples of the concepts that we collected after refining.

After refining the noisy concepts, the Web sources belonging to several concepts are not enough and the quality of the Web sources may be poor. To collect enough valid training examples, for each concept, we first search a large number of images from $Google$ and videos from Youtube by querying the corresponding concept. Then we use the NAPSAC [13] to eliminate the noisy images and videos in each concept training set. Specifically, the images/videos collected at the first stage are treated as the center points, and the radius R is chosen empirically to determine the concept circle. The Web images/videos in the circle are selected as the training examples of the concept. The NAPSAC pseudocode algorithm is summarized in Algorithm 1.

C. Hierarchical Clustering of Concepts and Their Associated Images/Videos

In the refined pool of preliminary concepts, some different tags/keywords may have similar semantic meaning, so it is essential to merge them into a single meaningful tag/keyword with the semantic consistence of concepts. Accordingly, the associated images/videos of these tags/keywords should also be

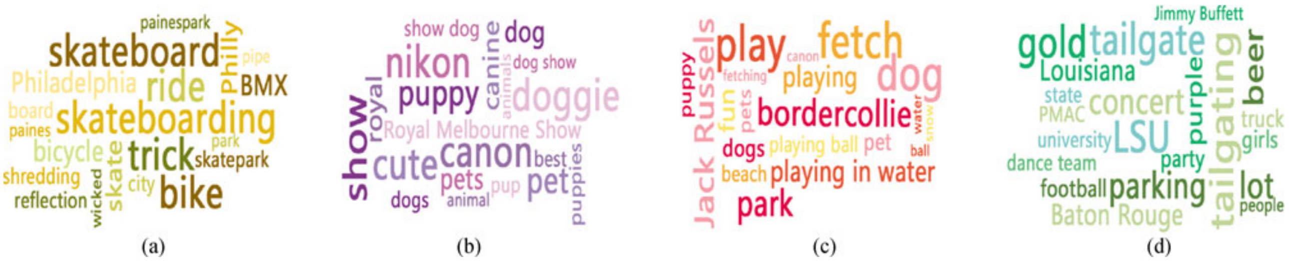


Fig. 2. The concepts we collected corresponding to the target event of "Attempting a bike trick," "Dog Show," "Playing fetch," and "Tailgating" after mining. (a) Attempting a Bike Trick. (b) Dog Show. (c) Playing Fetch. (d) Tailgating.

combined together to enrich the data set for training an informative concept classifier. Therefore, we introduce the hierarchical clustering [14] to cluster tags/keywords based on both the text similarity of textual descriptions and the visual similarity of their associated images/videos.

Text Similarity: We compute the similarity of two words using the interface of the WordNet [32] in Python API. The API provides a packaged method of calculating the distance of two words. For two short phrases of multiple words (e.g. keywords), we select every word from each phase to compute the relevance of the words, and the maximum similarity score is chosen as the relevance of these two phases. The text similarity of concept c_i and c_j is defined as $S(Text_{c_i}, Text_{c_j})$.

Visual Similarity: We use the static and motion features to measure the visual similarity between different concepts. For different concepts c_i and c_j, their associated training example sets are T_{c_i} and T_{c_j}, respectively. We extract the deep CNN features [33] for the images in the image collection and the motion feature IDT [34] of the videos. A mean representation of all the training examples is used to stand for the whole training set. The visual similarity of image/video sets is computed by

$$S(T_{c_i}, T_{c_j}) = \max(\text{cosine}(MI_{c_i}, MI_{c_j}), \text{cosine}(MV_{c_i}, MV_{c_j})), \quad (1)$$

where MI_c and MV_c are the image and video mean vectors of the training set of the concept c, respectively. We have also investigated other similarity metrics such as Euclidean distance, χ^2 distance and histogram intersection. The *cosine* similarity strikes the best trade-off between effectiveness and efficiency. So the final similarity between two concepts c_i and c_j is given by:

$$S(c_i, c_j) = \alpha \cdot S(Text_{c_i}, Text_{c_j}) + (1 - \alpha) \cdot S(T_{c_i}, T_{c_j}), \quad (2)$$

where α is a tradeoff parameter.

We then adopt the hierarchical clustering to cluster the candidate concepts into M concept groups. The collection of the concepts used in event detection is composed of these M concept groups. The distances between the concepts in each group in the clustering are under a distance threshold T_{dis}. We refine the description of each concept group based on the textual of the clustered concepts. Fig. 3 lists several automatically discovered concepts and their image/video training samples we collected in the experiment.

IV. Model Formulation

A. Notation

By using the automatic concept discovery described in Section III, a set of semantic concepts is found, represented by $\mathcal{S} = \{s_1, s_2, \cdots, s_{N_s}\}$, where N_s is the total number of the concepts. For each key segment which is actually a short clip of an video, one related concept from $\mathcal{S}$ is automatically selected to describe its semantic information. Thus the knowledge learned from Web images and videos belonging to one concept is transferred to the related key segment with the same concept.

Formally, the queried images and videos are collected as the training set $\mathcal{X}^s$ of the source domain $\mathcal{D}^s$. Let $\mathcal{X}_p^s \subset \mathcal{X}^s$ represent the p-th training subset related to the p-th concept in $\mathcal{S}$ with N_p, the number of samples of $\mathcal{X}_p^s$. $x_{p,j}^s \in \mathcal{X}_p^s$ denotes the j-th sample in the p-th training subset, and $y_{p,j}^s \in \mathcal{Y}^s$ denotes the event class label of $x_{p,j}^s$. We model the temporal positions of key segments in a video as $H = [h_1, h_2, \cdots, h_{N_g}]$ and the concepts of these segments as $C = [c_1, c_2, \cdots, c_{N_g}]$, where c_i represents the concept of the i-th segment and N_g is the number of the segments. Since the ground truth of H and C are not available in training data, they are treated as latent variables in our method. Then the pre-learned classifier of the p-th concept using the training subset $\mathcal{X}_p^s$ is formulated by $f^s(x_p) = w_p^s \cdot \Phi(x_p)$, where w_p^s is the template and $\Phi(x_p)$ is the feature mapping function.

The consumer videos are collected as the training set $\mathcal{X}^t$ of the target domain $\mathcal{D}^t$ with N_t, the number of samples of $\mathcal{X}^t$. Let x_i^t indicate the i-th sample of $\mathcal{X}^t$ and y_i^t indicate the event class label of x_i^t.

B. Event Detection Model

The event detection can be formulated as predicting the event class label y of an input video x,

$$\begin{aligned} y &= \underset{y,H,C}{\operatorname{argmax}}\, \mathcal{F}(x, y, H, C) \\ &= \underset{y,H,C}{\operatorname{argmax}}\, W^t \cdot \Phi(x, y, H, C). \end{aligned} \quad (3)$$

The discriminative function $\mathcal{F}(x, y, H, C)$ is composed of a low-level event model, a temporal relation model of the key segments, and an interaction model between the key segments

Fig. 3. Examples of image and video concepts collected automatically from Web. These concepts cover the actions, objects, and scenes.

and the entire video:

$$
\begin{aligned}
\mathcal{F}_{(x,y,H,C)} &= W^t \cdot \Phi(x,y,H,C) \\
&= F_d(x,H,C) + F_a(H,C) + F_b(H,C,y) \\
&= \sum_{k=1}^{N_s}\sum_{g=1}^{N_g} w_k^t \varphi(x,h_g) \cdot I_k(c_g) \\
&+ \sum_{k=1}^{N_s}\sum_{l=1}^{N_s}\sum_{p=1}^{N_g}\sum_{q=1}^{N_g} w_a \phi(h_p,h_q) \cdot I_k(c_p) \cdot I_l(c_q) \\
&+ \sum_{k=1}^{N_s}\sum_{g=1}^{N_g} w_b \psi(h_g,y) \cdot I_k(c_g), \qquad (4)
\end{aligned}
$$

where $W^t = [w_1^t, \cdots, w_{N_s}^t, w_a, w_b]$ is the joint weight vector of $\mathcal{F}$. Each model is defined as follows.

1) ***L**ow-Level **E**vent **M**odel (**LEM**):*

$$F_d(x,H,C) = \sum_{k=1}^{N_s}\sum_{g=1}^{N_g} w_k^t \varphi(x,h_g) \cdot I_k(c_g). \qquad (5)$$

The LEM is the most important part in the event detection model, which models the contributions of the low-level features for event detection. w_k^t is a concatenation of $w_{k_{im}}^t$ and $w_{k_{vi}}^t$, the image and video target template vectors, respectively. Each video is divided into N_g shot clips, and each clip can be represented as $\varphi(x,h_g)$, where h_g indicates the index of the g-th clip in video x, and c_g represents the concept of this clip. $\varphi(x,h_g) = [\varphi_{im}(x,h_g), \varphi_{vi}(x,h_g)]$ is also a concatenation of the image and video feature representation of the g-th clip. $I_k(c_g)$ is an indication function, and assigned the value of 1 if $c_g = k$ and 0 otherwise. The variables H and C are not provided during training and testing.

2) ***T**emporal **R**elation **M**odel (**TRM**):*

$$F_a(H,C) = \sum_{k=1}^{N_s}\sum_{l=1}^{N_s}\sum_{p=1}^{N_g}\sum_{q=1}^{N_g} w_a \phi(h_p,h_q) \cdot I_k(c_p) \cdot I_l(c_q). \qquad (6)$$

Exploiting the temporal relationships between segments is significantly essential for accurately extracting meaningful key segments in event detection. For example, in the event of "basketball", the segment corresponding to the concept of "passing the ball" often happens before the segment of "shooting the ball". In the event of "birthday", the segment of "blow out the candles" often occurs after the segment of "burning candles".

Therefore, we leverage the temporal relation model $F_a(H,C)$ in Eq. (6) to capture the temporal relationships between the key segments. $\phi(h_p,h_q)$ is a temporal vector. When the index h_p of the key segment happens before h_q, $\phi(h_p,h_q) = [1,0]$ and $\phi(h_p,h_q) = [0,1]$ otherwise. $w_a = [w_{a1}, w_{a2}]$ is a template vector, where w_{a1} indicates the relation score of the h_p-th segment happening before the h_q-th segment and w_{a2} denotes the score of the h_p-th segment occurring after the h_q-th segment. $I_k(c_p)$ is also an indication function with the value of 1 if $c_p = k$ and 0 otherwise. We initialize w_a according to the temporal relations of the concepts which correspond to the segments in a video.

3) ***S**egment **E**vent **I**nteraction **M**odel (**SEIM**):*

$$F_b(H,C,y) = \sum_{k=1}^{N_s}\sum_{g=1}^{N_g} w_b \psi(h_g,y) \cdot I_k(c_g). \qquad (7)$$

The interaction between segments and events is also an informative factor in event detection. For example, the probability of the segment of "dog" appears in the high-level event of "dog show" is very high while "dog show" hardly contains the segment of "guitar".

So we propose the Segment Event Interaction Model to capture the contextual interactions between the key segments and the entire event. When the segment h_g appears in the event y, $\psi(h_g, y)$ is assigned to 1. $I_k(c_p)$ is an indication function with the value of 1 if $c_g = k$ and 0 otherwise. w_b is the template vector and we initialize w_b according to the c_g-th concept.

V. Model Learning

A. Objective Function

The source event classifier learned from Web images and videos is formulated as $W^s = [w_1^s, w_2^s, \cdots, w_k^s, \cdots, w_{N_s}^s, w_a^s, w_b^s]$, where $w_k^s = [w_{k_{im}}^s, w_{k_{vi}}^s]$ is the k-th SVM classifier trained by images and videos in $\mathcal{X}_p^s$. $w_{k_{im}}^s$ and $w_{k_{vi}}^s$ are the image and video classifiers. w_a^s and w_b^s are the initial temporal template and interaction template, respectively.

The target classifier W^t is trained by using a limited number of labeled training videos. Given a training video x_i from the target domain, y_i is the ground-truth label of x_i and y_i^* is the optimal label of x_i. The parameter vector W^t is learned by the following optimization problem:

$$\min_{W^t} \frac{1}{2}\left\|\Delta W\right\|^2 + \lambda_1 \sum_{i=1}^{N_t} \xi_i + \lambda_2 \sum_{i=1}^{N_t} \zeta_i \tag{8}$$

s.t.

$$\max_{H_i, C_i} \mathcal{F}(x_i, y_i, H_i, C_i) - \max_{H_i^*, C_i^*, y_i^*} \mathcal{F}(x_i, y_i^*, H_i^*, C_i^*)$$

$$\leq L(y_i, y_i^*, H_i^*, C_i^*) - \xi_i, \tag{9}$$

$$0 \leq \xi_i, \forall (x_i, y_i) \in \mathcal{D}^T, \tag{10}$$

$$\sum_{k=1}^{N_s} \sum_{g=1}^{N_g} I_k(C_g) = K, \tag{11}$$

$$\zeta_i = \sum_{m=1}^{N_s} \sum_{n=1}^{N_s} \sum_{p=1}^{N_g} \sum_{q=1}^{N_g} ||f_m(x_i, h_p, c_p) - f_n(x_i, h_q, c_q)||, \tag{12}$$

where λ_1, λ_2 are trade-off parameters. $L(y_i, y_i^*, H_i^*, C_i^*)$ is 0–1 loss function defined by $L(y_i, y_i^*, H_i^*, C_i^*) = 0$ if $y_i = y_i^*$ and 1 otherwise. This loss function is used to enforce the decision value of the newly learned target classifier not far away from the source classifier.

The regularization term $\Delta W = W^t - W^s$ is introduced to indicate that the target classifier W^t should be close to the source hyperplane W^s. The constraint in Eq. (9) can be explained as follows: for the i-th training sample, the score $\max_{H_i, C_i} \mathcal{F}(x_i, y_i, H_i, C_i)$ which is associated with the ground-truth event label y_i, latent locations H_i and concept C_i should be no less than the score $\max_{H_i^*, C_i^*, y_i^*} \mathcal{F}(x_i, y_i^*, H_i^*, C_i^*)$ which is associated with any hypothesized event label y^*, segment locations H_i^* and concept C_i^*.

The constraint in Eq. (11) shows that K key segments are automatically extracted in a video, where $I_k(c_g)$ is an indication function and is assigned to 1 when $c_g = k$, showing that the concept k has been localized in the g-th clip.

We also append another constraint in Eq. (12) for extracting more informative key segments. This indicates that all the selected concepts are important to detect events and the decision values of the detected key segments should be close to each other. $f_m(x_i, h_p, c_p)$ denotes the decision value of the h_p-th key segment in video x_i, defined by

$$f_m(x_i, h_p, c_p) = w_m^t \varphi(x_i, h_p) \cdot I_m(c_p), \tag{13}$$

B. Optimization

In order to solve the non-convex optimization problem in Eq. (8), we first select K fixed concepts, then an iteration optimization algorithm is proposed to alternate between inferring the unobservable (H_i, C_i) given the pair (x_i, y_i) and solving the standard structural SVM when (H_i, C_i) is observable. After learning the fixed target classifier, we re-select K optimal key segments corresponding to the concepts with a dynamic programming algorithm.

1) Finding the Optimal Target Classifier With Fixed K Concepts: We fix the selected K concepts corresponding to the key segments in the video. With the fixed K concepts, Eq. (8) turns to be the latent structural SVM model:

$$\min_{W^t} \mathcal{L}(W^t) = \frac{1}{2}\left\|\Delta W\right\|^2 + \lambda \sum_{i=1}^{N_t} \mathcal{R}_i(W^t), \tag{14}$$

and $\mathcal{R}_i(W^t)$ is a hinge loss function defined as

$$\mathcal{R}_i(W^t) = \max_{H_i, C_i} \mathcal{F}(x_i, y_i, H_i, C_i) - \max_{H_i^*, C_i^*, y_i^*} \mathcal{F}(x_i, y_i^*, H_i^*, C_i^*)$$

$$+ L(y_i, y_i^*, H_i^*, C_i^*)$$

$$+ \sum_{m=1}^{N_s} \sum_{n=1}^{N_s} \sum_{p=1}^{N_g} \sum_{q=1}^{N_g} ||f_m(x_i, h_p, c_p) - f_n(x_i, h_q, c_q)||, \tag{15}$$

where

$$(H_i^*, C_i^*) = \underset{H_i, C_i}{\operatorname{argmax}} \mathcal{F}(x_i, y_i, H_i, C_i), \tag{16}$$

$$y_i^* = \underset{y_i}{\operatorname{argmax}} \mathcal{F}(x_i, y_i, H_i^*, C_i^*). \tag{17}$$

We adopt a non-convex cutting plane method proposed in [35] to solve the non-convex optimization problem. The non-convex cutting plane method aims to iteratively build an increasingly accurate piece-wise quadratic approximation of $\mathcal{R}_i(W^t)$ based on its sub-gradient $\partial_{W^t} \mathcal{R}_i(W^t)$. Now the primary task is to compute the sub-gradient $\partial_{W^t} \mathcal{R}_i(W^t)$. The sub-gradient $\partial_{W^t} \mathcal{R}_i(W^t)$ can be computed by

$$\partial_{W^t} \mathcal{R}_i(W^t) = \sum_{k=1}^{N_s} \sum_{g=1}^{N_g} \varphi(x, h_g) \cdot I_k(c_g)$$

$$- \sum_{k=1}^{N_s} \sum_{g=1}^{N_g} \varphi(x, h_g^*) \cdot I_k(c_g^*)$$

$$+\sum_{k=1}^{N_s}\sum_{l=1}^{N_s}\sum_{p=1}^{N_g}\sum_{q=1}^{N_g}\phi(h_p,h_q)\cdot I_k(c_p)\cdot I_l(c_q)$$

$$-\sum_{k=1}^{N_s}\sum_{l=1}^{N_s}\sum_{p=1}^{N_g}\sum_{q=1}^{N_g}\phi(h_p^*,h_q^*)\cdot I_k(c_p^*)\cdot I_l(c_q^*)$$

$$+\sum_{k=1}^{N_s}\sum_{g=1}^{N_g}\psi(h_g,y)\cdot I_k(c_g)$$

$$-\sum_{k=1}^{N_s}\sum_{g=1}^{N_g}\psi(h_g^*,y)\cdot I_k(c_g^*)$$

$$+\sum_{m=1}^{N_s}\sum_{n=1}^{N_s}\sum_{p=1}^{N_g}\sum_{q=1}^{N_g}|\varphi(x_i,h_p)\cdot I_m(c_p)$$

$$-\varphi(x_i,h_q)\cdot I_n(c_q)|\cdot A, \tag{18}$$

where $A = ||f_m(x_i,h_p,c_p) - f_n(x_i,h_q,c_q)||$.

2) Finding the Optimal K Concepts With Fixed W^t: When fixing W^t, Eq. (8) turns into a 0–1 integer programming problem:

$$\max\sum_{i=1}^{N_t}\sum_{k=1}^{N_s}\sum_{g=1}^{N_g}f_k(x_i,h_g,c_g) \tag{19}$$

s.t.

$$\sum_{k=1}^{N_s}\sum_{g=1}^{N_g}I_k(c_g) = K, \tag{20}$$

$$I_k(c_g) = 0/1, \forall k \in [1,N_s], \forall g \in [1,N_g]. \tag{21}$$

We use the dynamic programming algorithm to resolve this problem. The main issue is how to select the top K concepts relevant to the key segments in videos from the concept collection $\mathcal{S}$ with N_s concepts. There are two conditions to select K concepts from $\mathcal{S}$: (i) Select K concepts from N_s-1 concepts. (ii) Select $K-1$ concepts from N_s-1 concepts and select the N_s-th concept. We choose the max value of these two conditions. Let $EV(N_s)$ represent the decision value of the N_s-th concept in event detection formulated as

$$EV(N_s) = \sum_{i=1}^{N_t}\sum_{g=1}^{N_g}w_{N_s}^t\varphi(x_i,h_g)I_{N_s}(c_g). \tag{22}$$

The procedure of dynamic programming can be defined as

$$E(N_s,K) = \max(E(N_s-1,K), \\ E(N_s-1,K-1)+EV(N_s)), \tag{23}$$

The detailed optimization algorithm for complex event detection is summarized in Algorithm 2.

Algorithm 2: Algorithm for Complex Event Detection.

```
Input: $(x_{p,i}^s, y_{p,i}^s)|_{i=1}^{N_p} \in \mathcal{X}^s$: training examples from the
       source domain, $p \in 1,2,\cdots,N_s$;   $(x_i,y_i)|_{i=1}^{N^t}$:
       the labeled target domain training videos;
Output: $W^t$: the transferred target classifiers;
        $H, C$: the optimal key segments of videos.
Train source domain classifier $w_p^s$ using Web images and videos
repeat
    Step 1: compute $W^t$ with fixed $K$ concepts.
    for $i = 1$ to $N^t$ do
        $(H_i^*, C_i^*) = \operatorname{argmax}_{H_i,C_i} \mathcal{F}(x_i,y_i,H_i,C_i)$
        $y_i^* = \operatorname{argmax}_{y_i} \mathcal{F}(x_i,y_i,H_i^*,C_i^*)$
    end
    Compute $\partial_{W^t}\mathcal{R}_i(W^t)$ according to Eq. (18)
    Update $W^t$ using the cutting plane method proposed
     in [35]
    Step 2: compute $K$ optimal concepts with fixed $W^t$
    Compute $C$ according to Quadratic Programming
     method using the simplified Eq. (23).
until Convergence of objective function Eq. (8) cannot be
 decreased below tolerance $\delta$
```

VI. Experiment

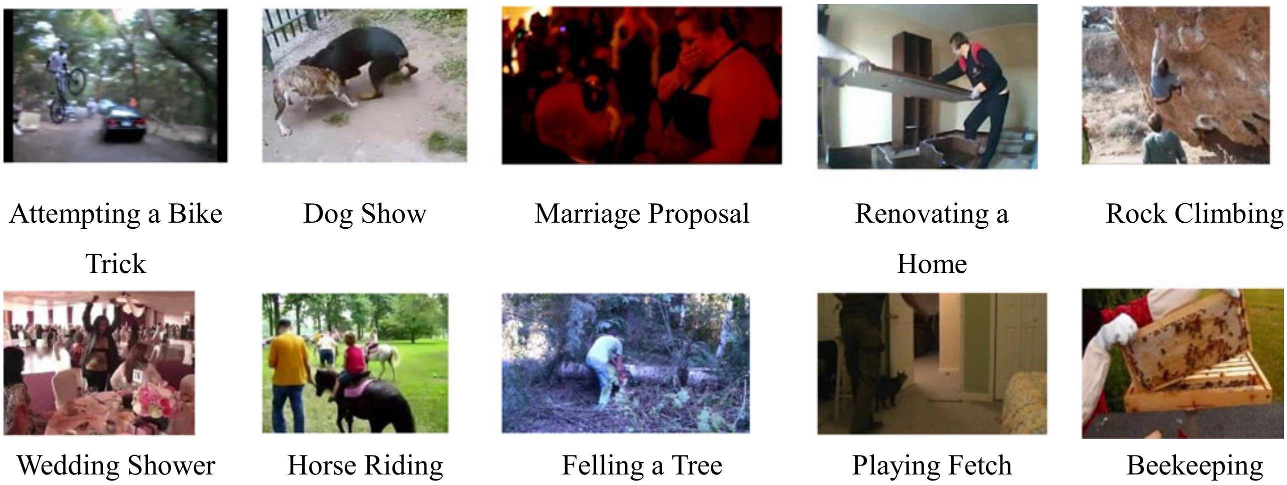
We evaluate the proposed method on two datasets: the TRECVID2014 Multimedia Event Detection dataset [36] and the Columbia's Consumer Video (CCV) dataset [37]. The mean of Average Precision (mAP) for binary classification [8] is applied for performance evaluation.

A. Datasets

The **CCV dataset** contains 9,317 *Youtube* videos over 20 event categories with a training set of 4,659 videos and a test set of 4,658 videos. Since our work focuses on event detection, the object/scene categories such as "bird", "beach", "cat", "dog" and "playground" are discarded. Finally, 15 event categories are utilized in our experiments.

The **TRECVID MED2014 dataset** contains 40 categories of events. The partition of "*Background*" contains 4,983 background videos which do not belong to any events and can be used as negative examples in the training procedure. The partitions of "10EX" and "100EX" respectively consist of 10 and 100 positive videos for each of the pre-defined 20 event classes. In our experiment, the setting of "*10EX*" and "*100EX*" are adopted. The partition of "*MEDTest*" contains 29,200 videos, in which there are about 25 positive samples for each event class and 26717 negative videos. Fig. 4 shows several examples of the frames from the test videos on the CCV and the MED2014 datasets, respectively.

The **Self-Collected Image and Video Dataset** is constructed by the videos collected from the Youtube Website and the images collected from the Google and Flickr Websites. First, we query about 4,000 images (200 images per class) from the Flickr Website and 600 videos (30 videos per class) from the Youtube



Attempting a Bike Trick

Dog Show

Marriage Proposal

Renovating a Home

Rock Climbing

Wedding Shower

Horse Riding

Felling a Tree

Playing Fetch

Beekeeping

Fig. 4. Examples of the frames from the test videos on the CCV dataset and the MED2014 dataset.

website by using the name of event class as search keywords. Then, the associated tags and description sentences of the queried images and videos are collected to construct the preliminary concept pool. To guarantee the adequency of the training data for each concept, we re-search about 40,000 images and 1000 videos based on the concept pool from the Google website and the Youtube website, respectively. Finally, for each concept, there are 200 images and 50 videos collected from the Web source.

B. Experiment Setup

1) Visual Features: The visual description of an image/frame is represented by a 4096-dimensional CNN feature vector which is the output of the 16-layer VGG model [38] implemented by Caffe toolkit [33]. Each image/frame is scaled to 256×256, and cropped to a random 227×227 region. The length of each clip is 10 seconds and there is 5 seconds overlap between two adjacent video clips. For a video segment, we first extract the improved dense trajectory (IDT) [34], and compute three descriptors (i.e. Histogram of Gradients (HOG), Histogram of Optical Flow (HOF), and Motion Boundary Histogram (MBH)) of IDT. Then a local descriptor encoding method called Vector of Locally Aggregated Descriptors [6], [39] is used to encode the three descriptors as the visual feature of the video segment.

2) Concept Classifiers: In this experiment, 120 concepts are automatically discovered by using the methods in the Section III. After collecting the images and videos from Web, there are about 500 images and 20 videos in the training dataset for each concept. These images and videos are treated as positive samples, while the negative samples are constructed by randomly selecting images and videos from the training datasets of other concepts. The proportion of the positive and negative samples is set to 1:5. We use the LIBSVM toolkit [40] with the linear kernel to learn the concept image and video classifiers. The 5-fold cross-validation is utilized to choose the parameter of regularization coefficient C in the SVM.

3) Related Methods: We compare our method with several related methods on both the CCV dataset and the MED2014 dataset, such as the target domain SVM (*SVM_T*), Domain Adaptive SVM (DASVM) [41], Domain Adaptation Machine (DAM) [42], and Domain Selection Machine (DSM) [18]. *SVM_T* is trained on the limited number of the training videos we utilize in our method. DASVM is a single domain method in which the images and videos are gathered together as a single domain. Both DAM and DSM are multi-source domain adaptation methods where the training samples belonging to each concept are treated as a single domain.

In the respect of event detection with key segments, our method is compared with Dynamic Pooling with Segment Pairs (DPSP) [23], Evidence Localization Model (ELM) [5], and Joint Event Detection and Recounting (JEDaR) [24] on the MED 2014 dataset. Additionally, we compare the proposed method with TagBook [43], VideoStory [44], and Fusion for Visual Recognition (RFVR) [45] on the CCV dataset. DPSP and ELM both dynamically locate the key segments of videos for event detection. JEDaR simultaneously detects high-level events and localizes the indicative concepts of the events. We compare these three methods on the MED2014 dataset. TagBook and VideoStory propose to learn the semantic video representation for event detection. RFVR models the dependency based on probabilistic properties of posteriors without any assumption on the data distribution for visual recognition. These three methods are tested on the CCV dataset.

C. Results

1) Comparison With Related Methods: Tables I–III list the per-class average precision of different methods on the TRECVID MED 2014 EK100, EK10 and CCV datasets, respectively. It is obvious that our approach performs better than other related methods. With analysis into more details, we have the following observations:

1) When compared with the SVM_T, our method achieves superior results which verifies the advantage of exploiting

TABLE I
THE RESULTS (%) OF THE METHODS ON TRECVID MED2014 EK100 DATASET

Event	SVM_T	DASVM [41]	DAM [42]	DSM [18]	Ours
Attempting a bike trick	8.2	9.56	5.04	8.6	**21.24**
Cleaning an appliance	1.56	**18.6**	11.45	12.72	8.52
Dog Show	25.4	35.6	30.9	**56.6**	52.56
Giving directions to a location	2.68	**6.8**	4.82	4.24	4.21
Marriage proposal	0.32	5.6	5.3	1.15	**11.82**
Renovating a home	3.85	4.3	6.8	2.48	**9.92**
Rock climbing	8.6	17.3	23.8	14.5	**24.51**
Town hall meeting	18.54	**33.8**	30.2	21.76	25.64
Winning a race without a vehicle	16.82	19.3	35.32	8.48	**38.62**
Working on a metal crafts project	8.64	8.1	12.6	8.82	**13.2**
Beekeeping	55.15	60.8	57.13	72.1	**74.36**
Wedding shower	22.45	20.36	18.49	25.1	**26.83**
Non-motorized vehicle repair	36.46	**44.89**	33.64	43	40.26
Fixing musical instrument	22.92	**28.45**	18.54	28.3	24.61
Horse riding competition	30.22	27.27	28.53	50.2	**52.38**
Felling a tree	16.47	13.62	16.01	16.3	**22.96**
Parking a vehicle	28.63	20.74	26.31	17.8	**67.28**
Playing fetch	6.76	5.17	14.27	13.2	**28.53**
Tailgating	18.28	19.36	10.58	29.6	**62.35**
Tuning musical instrument	6.83	4.8	6.8	4.08	**15.88**
mAP	16.94	20.22	19.83	22.95	**31.29**

TABLE II
THE RESULTS (%) OF THE METHODS ON TRECVID MED2014 EK10 DATASET

Event	DPSP [23]	ELM [5]	JEDaR [24]	Ours
Attempting a bike trick	11.24	14.75	**19.53**	18.62
Cleaning an appliance	2.72	7.54	**8.77**	6.37
Dog Show	30.75	41.29	46.26	**48.93**
Giving directions to a location	3.12	3.16	**3.98**	3.28
Marriage proposal	0.87	1.12	1.27	**5.93**
Renovating a home	4.85	5.88	6.23	**7.86**
Rock climbing	12.56	13.96	15.62	**20.67**
Town hall meeting	20.82	25.25	**27.41**	23.46
Winning a race without a vehicle	14.82	17.84	19.63	**30.84**
Working on a metal crafts project	13.65	**15.92**	15.26	12.6
Beekeeping	61.82	67.85	**69.41**	68.36
Wedding shower	24.64	27.43	**28.28**	22.51
Non-motorized vehicle repair	41.47	**46.54**	46.27	35.72
Fixing musical instrument	27.83	29.78	**31.63**	22.84
Horse riding competition	38.86	42.86	45.32	**48.86**
Felling a tree	15.28	16.52	19.27	**20.49**
Parking a vehicle	37.8	47.81	49.25	**56.48**
Playing fetch	0.78	1.14	1.43	**17.73**
Tailgating	23.22	28.72	30.58	**47.37**
Tuning musical instrument	16.54	15.83	**18.62**	12.46
mAP	20.47	23.56	25.21	**26.57**

the key segments for event detection as well as automatically discovering concepts for event description in our framework.

2) When compared with DASVM, DAM, DSM, Videostory, Tagbook and RFVR, our method performs better. It is possible that our method mainly focuses on detecting the key segments to classify the complex videos with the benefit of efficiently investigating the underlying structural information of videos for the recognition task.
3) Our method outperforms the segment-based methods of DPSP, ELM and JEDaR on the MED 2014 dataset, which validates that it is beneficial to transfer the knowledge learned from Web resources to the target videos for improving the accuracy of event detection model.
4) Our method also achieves better results on the TRECVID EK10 dataset, which validates the effectiveness of our method on few-shot learning.
5) For most of the events such as “marriage proposal”, “beekeeping”, “baseball” and “birthday”, our method obtains significant results owing to the discovered discriminative and descriptive key segments in detecting videos. For some events of “giving direction to a location”, “fixing musical instrument” or “tuning musical instrument”, our method performs a little worse, the possible reason is that

TABLE III
AVERAGE PRECISION (%)OF PER-CLASS PERFORMANCE ON CCV OF DIFFERENT METHODS

Event	SVM_T	DASVM [41]	DAM [42]	DSM [18]	Videostory [44]	Tagbook [43]	RFVR [45]	Ours
Basketball	46.82	42.33	36.59	46.28	55.3	63.3	**77.21**	70.21
Baseball	51.91	55.8	52.70	50.98	29.9	59.4	56.3	**66.5**
Soccer	45.13	46.59	48.82	51.60	50.5	57.4	64.4	**67.84**
Ice-Skating	38.91	41.88	41.69	45.84	67.5	72.2	**87.46**	82.65
Skiing	75.63	83.17	77.90	77.63	67.1	79.6	77.83	**84.34**
Swimming	65.10	**87.04**	71.93	85.93	76.4	76.2	76.29	85.4
Biking	45.28	48.65	45.11	47.42	56.1	**62.1**	48.79	57.9
Graduation	14.55	14.61	16.46	12.47	12.1	29.0	**49.81**	42.68
Birthday	16.96	15.91	26.67	13.98	25.7	49.2	46.91	**51.2**
Reception	11.97	13.84	15.91	21.38	11.7	19.6	36.1	**38.5**
Ceremony	28.16	35.63	44.57	46.29	32.4	45.4	55.59	**58.46**
Dance	49.97	56.11	52.44	58.23	52.1	50.3	**60.05**	57.8
Music	49.05	56.396	51.79	**58.90**	20.1	38.5	34.33	56.5
Non-Music	24.55	32.3	25.7	43.3	28.2	28.9	**72.09**	59.8
Parade	15.47	18.63	23.24	27.78	63.4	52.1	67.75	**72.1**
mAP	38.63	43.16	42.00	45.82	43.23	52.21	60.73	**63.46**

TABLE IV
THE RESULTS OF DIFFERENT COMPONENTS ON COMPLEX EVENT DETECTION MODEL

Dataset	LEM	TRM	SEIM	w/o LEM	w/o TRM	w/o SEIM	$\lambda_1 = 0$	$\lambda_2 = 0$	**Ours**
CCV	50.74	15.43	11.84	18.6	56.81	62.38	15.64	60.53	**63.46**
MED2014 (EK10)	18.76	3.98	2.63	6.84	22.63	25.59	3.63	22.48	**26.57**
MED2014 (EK100)	23.68	4.85	2.92	7.92	25.58	28.85	5.21	28.68	**31.29**

TABLE V
THE RESULTS OF DIFFERENT COMPONENTS IN THE PROCEDURE OF AUTOMATIC CONCEPT DISCOVERY

Dataset	w/o NAPSAC	w/o Cluster	w/o Text	w/o Visual	Ours
CCV	50.73	47.76	59.28	32.83	**63.46**
MED2014(EK10)	16.54	12.38	22.83	5.67	**26.57**
MED2014(EK100)	25.36	20.38	28.47	12.41	**31.29**

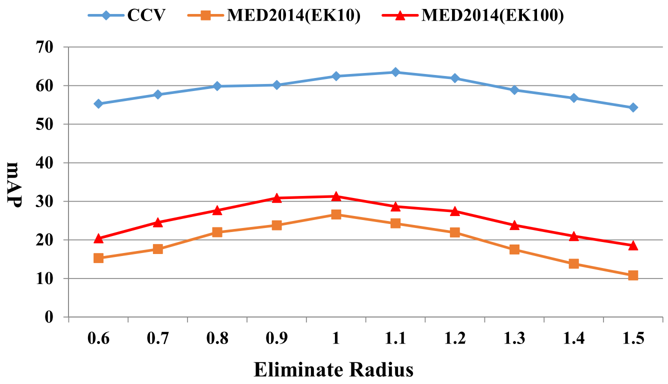


Fig. 5. The changes of mean average precision (%) with the radiuses in the procedure of eliminating noisy images.

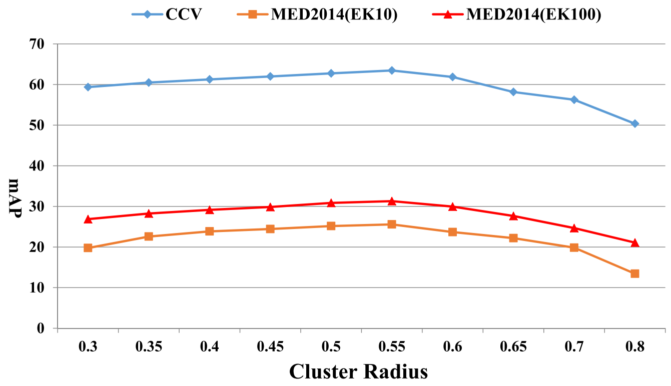


Fig. 6. The changes of mean average precision (%) with the radiuses in the procedure of hierarchical clustering.

it is difficult to describe these events with informative concepts.

2) Analysis of Different Components on Event Detection Model: We investigate the effects of each component and each constraint in Eq. (8). As given in Table IV, "w/o LEM", "w/o TRM" and "w/o SEIM", indicate the methods of removing the low-level event model, removing the temporal model, and removing the concept event joint model, respectively. "$\lambda_1 = 0$" and "$\lambda_2 = 0$" indicate the methods of removing the constraints, respectively. It is interesting to notice that: (i) When discard-

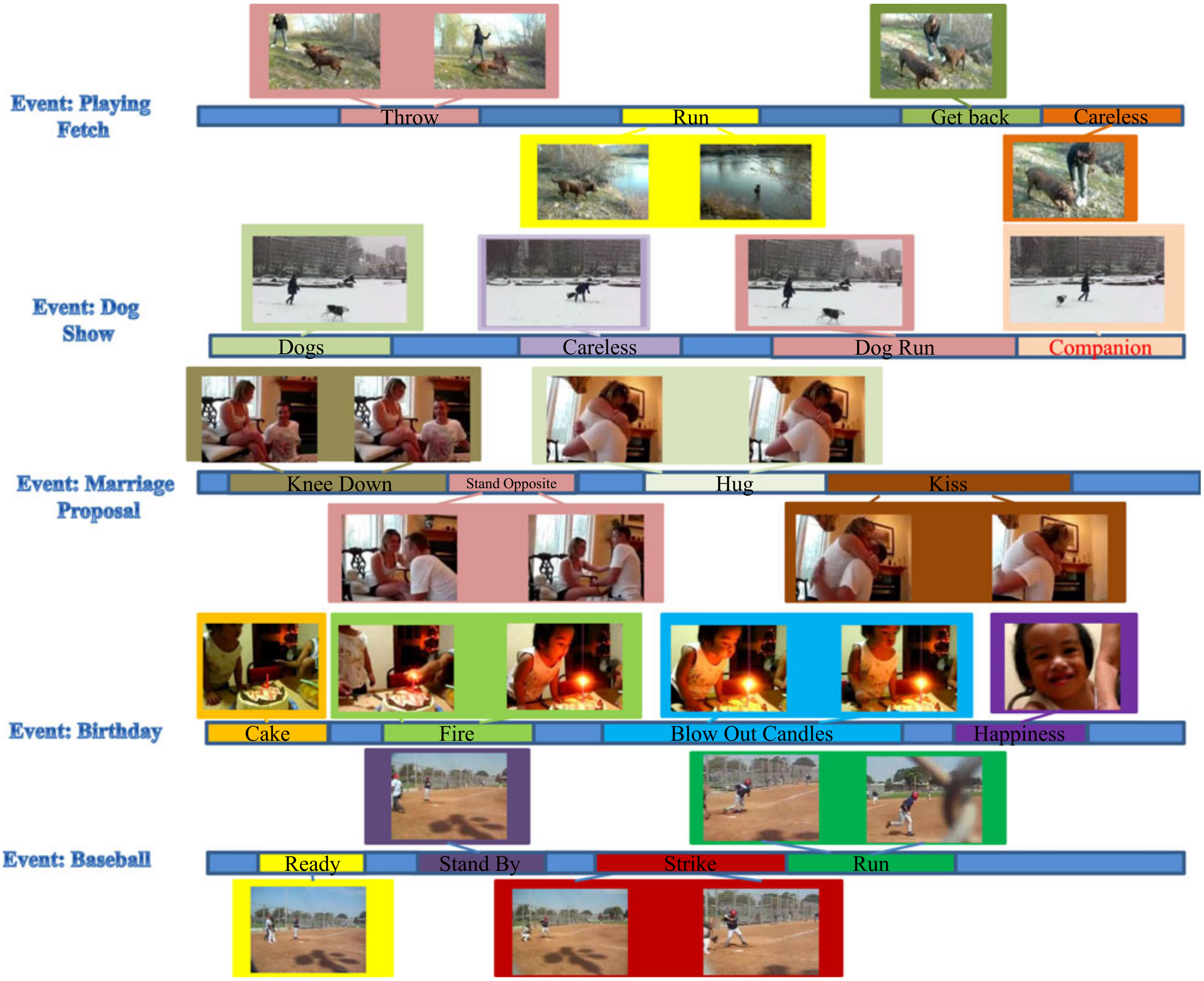


Fig. 7. The key segments (bright color) in videos (blue color) of several events we extracted in our experiment.

ing the LEM model, the detection performances significantly degrade on both the MED 2014 and the CCV datasets, which clearly demonstrates the importance of utilizing the low-level feature representation to train the discriminative concept classifiers for key segments detection. (ii) The performance is obviously improved when integrating either the TRM model or the SEIM model, because either the temporal relationship between different concepts or the context relationship between concepts and specified events is helpful for selecting more discriminative and descriptive concepts for event videos with filtering out the unreasonable and inaccurate concepts. (iii) The mAP decreases when either of the constraints is removed from Eq. (8), which confirms that the contributions of the selected concepts in the specified event are close and the correlations between the target training samples provide effective information to train the target classifier.

3) Analysis of Different Components on Automatic Concept Discovery: Table V lists the results of different components on automatic concept discovery, where "w/o NAPSAC" and "w/o Cluster" respectively represent the methods of excluding the NAPSAC procedure and the hierarchical clustering method in the concept discovery model. "w/o Text" and "w/o Visual" indicate the methods of excluding the text similarity and the visual similarity in hierarchical clustering separately. From the results, we can observe that (i) NAPSAC is very important to find the discriminative concepts by filtering out those noisy videos and images. (ii) The hierarchical clustering of the text description of the concepts as well as the images and videos belonging to each concept is capable of improving the detection accuracy as the hierarchical clustering method could cluster the similar concepts which are similar in text description or visual representation into one new concept. Additionally, during the clustering, it is essential to employ both the textual and visual information of images and videos to measure the similarity between two samples.

4) Evaluations on Different Radiuses in Automatic Concept Discovery: Fig. 5 shows the mean average precision (mAP) of different radiuses for eliminating noisy Web images/videos in automatic concept discovery. The parameter of radius determines the number of relative images/videos in each concept. As shown in Fig. 5, the radius is set to 1 and 1.1 on the CCV and the MED2014 datasets, respectively, to achieve the best performance. When the radius becomes lower or higher from the best value, the mAP will obviously decrease. The possible explanation is that when the radius is smaller, there will be not enough relative training examples to be clustered of each new concept, this will cause over-fitting of the concept classifiers. When the radius is larger, there will be more noisy images and videos in the concept, and the training data from each concept may lose the discriminability.

In Fig. 6, we illustrate the performances of different values of radius for hierarchical clustering of concepts in automatic concept discovery. This radius measures the text similarity of the textual description and the visual similarity of the videos and images belonging to each concept. Once either similarity has reached the radius, the two corresponding concepts will be clustered to generate one new concept. In the CCV and the MED2014 datasets, the best value of radius is 0.55 with the highest detection accuracy. When the radius becomes smaller, the similar concepts may be ignored to be together. On the contrary, when the radius becomes larger, more irrelevant concepts will aggregate into one cluster. In both of these two cases, the accuracy of concept classifiers will degrade resulting in the worse performance of the event detection model.

5) Key Segments of Videos for Event Detection: The segmentation results of several events have been shown in Fig. 7. In Fig. 7, four segments of "knee down", "stand opposite", "hug", and "kiss" are detected in the event of "marriage proposal". For other events, we could also detect each four segments in the same way. The segmentation of videos makes event detection more meaningful and practical.

VII. Conclusion

In this paper, we have proposed a framework of automatically detecting key segments for event detection by leveraging loosely labeled Web sources and a limited number of consumer videos. A discriminative model is presented for complex event detection by using an adaptive latent structural SVM model, where the locations of key segments are regarded as latent variables. The NAPSAC and hierarchical clustering are combined to automatically construct more meaningful concepts which are treated as the semantic descriptions of the key segments and the temporal information of concepts is exploited to capture the temporal relations between segments. The experimental results on the Columbia's Consumer Video dataset and the TRECVID2014 Multimedia Event Detection dataset demonstrate the effectiveness of our method.

References

[1] G. Lavee, E. Rivlin, and M. Rudzsky, "Understanding video events: A survey of methods for automatic interpretation of semantic occurrences in video," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 39, no. 5, pp. 489–504, Sep. 2009.

[2] J. Revaud, M. Douze, C. Schmid, and H. Jégou, "Event retrieval in large video collections with circulant temporal encoding," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2013, pp. 2459–2466.

[3] S. Phan *et al.*, "Multimedia event detection using segment-based approach for motion feature," *J. Signal Process. Syst.*, vol. 74, no. 1, pp. 19–31, 2014.

[4] A. Habibian and C. G. M. Snoek, "Recommendations for recognizing video events by concept vocabularies," *Comput. Vision Image Understanding*, vol. 124, pp. 110–122, 2014.

[5] C. Sun and R. Nevatia, "Discover: Discovering important segments for classification of video events and recounting," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2014, pp. 2569–2576.

[6] Z. Xu, Y. Yang, and A. G. Hauptmann, "A discriminative CNN video representation for event detection," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2015, pp. 1798–1807.

[7] Y. Yang, Z.-J. Zha, Y. Gao, X. Zhu, and T.-S. Chua, "Exploiting web images for semantic video indexing via robust sample-specific loss," *IEEE Trans. Multimedia*, vol. 16, no. 6, pp. 1677–1689, Oct. 2014.

[8] H. Wang, X. Wu, and Y. Jia, "Video annotation via image groups from the web," *IEEE Trans. Multimedia*, vol. 16, no. 5, pp. 1282–1291, Aug. 2014.

[9] Y. S. Sefidgar, A. Vahdat, S. Se, and G. Mori, "Discriminative key-component models for interaction detection and recognition," *Comput. Vision Image Understanding*, vol. 135, pp. 16–30, 2015.

[10] Y. Yan *et al.*, "Event oriented dictionary learning for complex event detection," *IEEE Trans. Image Process.*, vol. 24, no. 6, pp. 1867–1878, Jun. 2015.

[11] Y.-G. Jiang, J. Yang, C.-W. Ngo, and A. Hauptmann, "Representations of keypoint-based semantic concept detection: A comprehensive study," *IEEE Trans. Multimedia*, vol. 12, no. 1, pp. 42–53, Jan. 2010.

[12] C. GM Snoek and M. Worring, "Concept-based video retrieval," *Foundations Trends Inf. Retrieval*, vol. 2, no. 4, pp. 215–322, 2008.

[13] D. Nasuto and JM Bishop R Craddock, "NAPSAC: High noise, high dimensional robust estimation—It's in the bag," in *Proc. Brit. Mach. Vision Conf.*, 2002, pp. 458–467.

[14] S. C. Johnson, "Hierarchical clustering schemes," *Psychometrika*, vol. 32, no. 3, pp. 241–254, 1967.

[15] X. Zhang *et al.*, "Enhancing video event recognition using automatically constructed semantic-visual knowledge base," *IEEE Trans. Multimedia*, vol. 17, no. 9, pp. 1562–1575, Sep. 2015.

[16] H. Wang, H. Song, X. Wu, and Y. Jia, "Video annotation by incremental learning from grouped heterogeneous sources," in *Proc. Asian Conf. Comput. Vision*, 2015, pp. 493–507.

[17] M. Long *et al.*, "Adaptation regularization: A general framework for transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 5, pp. 1076–1089, May 2014.

[18] L. Duan, D. Xu, and S. fu Chang, "Exploiting web images for event recognition in consumer videos: A multiple source domain adaptation approach," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2012, pp. 1338–1345.

[19] C. Li, A. Sun, and A. Datta, "Twevent: segment-based event detection from tweets," in *Proc. 21st ACM Int. Conf. Inf. Knowl. Manage.*, 2012, pp. 155–164.

[20] H. Song, X. Wu, W. Liang, and Y. Jia, "Recognizing key segments of videos for video annotation by learning from web image sets," *Multimedia Tools Appl.*, vol. 76, pp. 6111–6126, 2017.

[21] C. Sun *et al.*, "ISOMER: Informative segment observations for multimedia event recounting," in *Proc. Int. Conf. Multimedia Retrieval*, 2014, p. 241.

[22] K. Tang, L. Fei-Fei, and D. Koller, "Learning latent temporal structure for complex event detection," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2012, pp. 1250–1257.

[23] W. Li, Q. Yu, A. Divakaran, and N. Vasconcelos, "Dynamic pooling for complex event recognition," in *Proc. IEEE Int. Conf. Comput. Vision*, 2013, pp. 2728–2735.

[24] X. Chang, Y.-L. Yu, Y. Yang, and A. G. Hauptmann, "Searching persuasively: Joint event detection and evidence recounting with limited supervision," in *Proc. 23rd Annu. ACM Conf. Multimedia Conf.*, 2015, pp. 581–590.

[25] M. Mazloom, E. Gavves, K. van de Sande, and C. Snoek, "Searching informative concept banks for video event detection," in *Proc. 3rd ACM Conf. Int. Conf. Multimedia Retrieval*, 2013, pp. 255–262.

[26] M. Mazloom, A. Habibian, D. Liu, C. GM Snoek, and S.-F. Chang, "Encoding concept prototypes for video event detection and summarization," in *Proc. 5th ACM Conf. Int. Conf. Multimedia Retrieval*, 2015, pp. 123–130.

[27] M. Merler, B. Huang, L. Xie, G. Hua, and A. Natsev, "Semantic model vectors for complex video event recognition," *IEEE Trans. Multimedia*, vol. 14, no. 1, pp. 88–101, Feb. 2012.

[28] J. Deng *et al.*, "Imagenet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2009, pp. 248–255.

[29] P. Over *et al.*, "Trecvid 2009- goals, tasks, data, evaluation mechanisms and metrics," in Trecvid 2009 papers, pp. 1–42, 2010.

[30] H. C. Wu, R. W. P. Luk, K. F. Wong, and K. L. Kwok, “Interpreting TF-IDF term weights as making relevance decisions,” *ACM Trans. Inf. Syst.*, vol. 26, no. 3, 2008, Art. no. 13.

[31] S. Sadanand and J. J. Corso, “Action bank: A high-level representation of activity in video,” in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2012, pp. 1234–1241.

[32] G. A. Miller, “WordNet: A lexical database for English,” *Commun. ACM*, vol. 38, no. 11, pp. 39–41, 1995.

[33] Yangqing Jia *et al.*, “Caffe: Convolutional architecture for fast feature embedding,” *arXiv preprint arXiv:1408.5093*, 2014.

[34] H. Wang and C. Schmid, “Action recognition with improved trajectories,” in *Proc. IEEE Int. Conf. Comput. Vision*, Sydney, Australia, 2013, pp. 3551–3558.

[35] T.-M.-T. Do and T. Artières, “Large margin training for hidden Markov models with partially observed states,” in *Proc. 26th Annu. Int. Conf. Mach. Learn.*, 2009, pp. 265–272.

[36] TRECVID Multimedia Event Detection 2014, 2014. [Online]. Available: http://www.nist.gov/itl/iad/mig/med14.cfm

[37] Y.-G. Jiang, G. Ye, S.-F. Chang, D. Ellis, and A. C. Loui, “Consumer video understanding: A benchmark database and an evaluation of human and machine performance,” in *Proc. ACM Int. Conf. Multimedia Retrieval, Oral Session*, 2011.

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

[39] H. Jégou, M. Douze, C. Schmid, and P. Pérez, “Aggregating local descriptors into a compact image representation,” in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2010, pp. 3304–3311.

[40] C.-C. Chang and C.-J. Lin, “LIBSVM: A library for support vector machines,” *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, 2011, Art. no. 27.

[41] L. Bruzzone and M. Marconcini, “Domain adaptation problems: A DASVM classification technique and a circular validation strategy,” *Pattern Recognit. Mach. Intell.*, vol. 32, no. 5, pp. 770–787, May 2010.

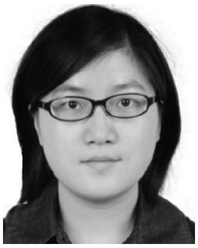
[42] L. Duan, D. Xu, and I. W.-H. Tsang, “Domain adaptation from multiple sources: A domain-dependent regularization approach,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 3, pp. 504–518, Mar. 2012.

[43] M. Mazloom, X. Li, and C. GM Snoek, “TagBook: A semantic video representation without supervision for event detection,” *arXiv preprint arXiv:1510.02899*, 2015.

[44] A. Habibian, T. Mensink, and C. GM Snoek, “VideoStory: A new multimedia embedding for few-example recognition and translation of events,” in *Proc. ACM Int. Conf. Multimedia*, 2014, pp. 17–26.

[45] A. J. Ma and P. C. Yuen, “Reduced analytic dependency modeling: Robust fusion for visual recognition,” *Int. J. Comput. Vision*, vol. 109, no. 3, pp. 233–251, 2014.

Hao Song received the B.S. degree from North China Electric Power University, Baoding, China, in 2012. He is currently working toward the Ph.D. degree under the supervision of Prof. Y. Jia with Beijing Laboratory of Intelligent Information Technology, School of Computer Science, Beijing Institute of Technology, Beijing, China. His research interests include computer vision, machine learning, and video retrieval.



Xinxiao Wu received the B.A. degree in computer science from Nanjing University of Information Science and Technology, Nanjing, China, in 2005, and the Ph.D. degree in computer science from Beijing Institute of Technology, Beijing, China, in 2010. She is currently an Associate Professor with the School of Computer Science, Beijing Institute of Technology. Her research interests include machine learning, computer vision, and human action perception.

Wennan Yu received the B.S. degree in 2015 from Beijing Institute of Technology, Beijing, China, iwhere he is currently working toward the M.S. degree under the supervision of Prof. X. Wu with Beijing Laboratory of Intelligent Information Technology, School of Computer Science. His research interests include computer vision and machine learning.

Yunde Jia received the B.S., M.S., and Ph.D. degrees in mechatronics from Beijing Institute of Technology (BIT), Beijing, China, in 1983, 1986, and 2000, respectively. He is a Professor of computer science with BIT, where he is the Director of Beijing Laboratory of Intelligent Information Technology. He has previously served as the Executive Dean of the School of Computer Science, BIT, from 2005 to 2008. He was a Visiting Scientist with Carnegie Mellon University from 1995 to 1997 and a Visiting Fellow with the Australian National University in 2011. His current research interests include computer vision, media computing, and intelligent systems.