Automatic Visual Concept Learning for Social Event Understanding

Xiaoshan Yang, Tianzhu Zhang, *Member, IEEE*, Changsheng Xu, *Fellow, IEEE*, and M. Shamim Hossain, *Senior Member, IEEE*

***Abstract*—Vision-based event analysis is extremely difficult due to the various concepts (object, action, and scene) contained in videos. Though visual concept-based event analysis has achieved significant progress, it has two disadvantages: visual concept is defined manually, and has only one corresponding classifier in traditional methods. To deal with these issues, we propose a novel automatic visual concept learning algorithm for social event understanding in videos. First, instead of defining visual concept manually, we propose an effective automatic concept mining algorithm with the help of Wikipedia, N-gram Web services, and Flickr. Then, based on the learned visual concept, we propose a novel boosting concept learning algorithm to iteratively learn multiple classifiers for each concept to enhance its representative discriminability. The extensive experimental evaluations on the collected dataset well demonstrate the effectiveness of the proposed algorithm for social event understanding.**

***Index Terms*—Event analysis, video recognition.**

I. Introduction

With the recent boom of smart phones, digital cameras, and Social Media sites (e.g., Flickr, YouTube, and Facebook), it is convenient for people to capture and share social media data online, which successfully facilitates information generation, sharing and propagation. As a result, a popular event that is happening around us and around the world can spread very fast, and there are substantial amounts of events with multi-modality (e.g., images, videos, and texts) in Internet. Most of these social events uploaded by users are related to some specific topics, and it is time-consuming to manually identify or cluster them. Therefore, automatically understanding social events from massive social media data is important and helpful to better browse, search and monitor social events by users or governments. However, it is difficult to achieve this goal because the substantial amounts of events are very complex and diverse, which makes it difficult to mine effective information for social event understanding. For example, for the social event "Kate and Wiliam wedding", videos may contain images of Kate and Wiliam together on the wedding's day, in an official setting (such as in the church or waiving at the crowd from the balcony.

Recently, many researchers have proposed different methods by using different kinds of information. In [1]–[7], media data, such as photos, text descriptions, tags, geographical locations, and time stamp, are adopted for social event detection (such as, "soccer events taking place in Barcelona (Spain) and Rome (Italy)"), which is the topic of MediaEval Benchmark.[1] In [8], the authors propose a framework to discover social events (such as, "Larry Page and Sergey Brin founded Google Inc. in 1998") from unstructured text based on several existing Web sources, such as social networks, blogs, wikis, and search engines. In [9]–[16], videos are adopted to detect and recognize general categories of social events, such as "Birthday party", "Making a sandwich" and "Rock climbing" in the popular multimedia event detection (MED) dataset from NIST. Video based social event understanding becomes more and more popular. However, based on the above work, we can observe that textual information is much more discriminative than visual information and achieves much better performance. As a result, more and more researchers attempt to combine textual and visual information, and adopt attribute as description for social event understanding due to its simplicity and promising performance.

In this paper, we call the various event related attributes, such as objects, actions and scenes contained in videos, as concepts. The basic idea of traditional concept based methods is: (1) define concept based on the textual information; (2) train visual classifier for each concept; and (3) represent each data sample using the learned classifiers. Even though the concept based method can show good performance, it has the following problems. (1) Concept definition: most of the previous methods define the concept manually, which may be not enough to describe the videos. For a video, only several frames are related to a specific concept. Let's take the event "presidential election" in Fig. 1 as an example. We can define a number of specific event concepts, such as "presidential", and "Obama talk". However,

Manuscript received July 01, 2014; revised October 23, 2014; accepted December 29, 2014. Date of publication January 16, 2015; date of current version February 12, 2015. This work was supported in part by the National Program on Key Basic Research Project under the 973 Program, Project 2012CB316304, by the National Natural Science Foundation of China under Grant 61225009, Grant 61303173, Grant 61373122, and Grant 61432019, by the Beijing Natural Science Foundation under Grant 4131004, and by the Deanship of Scientific Research, King Saud University under the research group Project RGP VPP-228. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. K. Selcuk Candan.

X. Yang, T. Zhang, and C. Xu are with the National Lab of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China, and also with the China-Singapore Institute of Digital Media, 119613, Singapore (e-mail: xiaoshang.yang@nlpr.ia.ac.cn; tzzhang@nlpr.ia.ac.cn; csxu@nlpr.ia.ac.cn).

M. S. Hossain is with the SWE Department, College of Computer and Information Sciences, King Saud University, Riyadh 12372, Saudi Arabia (e-mail: mshossain@ksu.edu.sa).

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.2015.2393635

[1][Online] Available: http://www.multimediaeval.org/

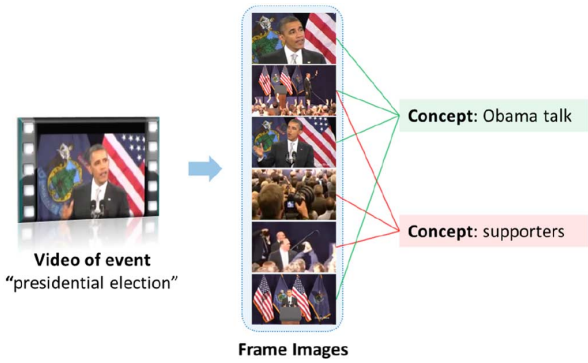

Fig. 1. Video related to event "presidential election." Each concept "Obama talk" or "supporters" corresponds to only several frames of the video. Besides, the frames corresponding to one concept, such as "Obama talk," are different due to illumination changes, different views, and scale variations. (The video via YouTube under Creative Commons License.[2])

each of them only corresponds to several frames of the video. Moreover, some meaningful concepts may be ignored due to the manual definition, such as "supporters" in Fig. 1. Therefore, the concept definition should include all meaningful event related concepts, such that all video frames can be represented well. (2) Concept diversity: the traditional methods train one classifier for each concept, which ignores the diversity of concept. For some concepts, their visual images may be very complex and diverse, and reflect different aspects of the semantic information of a given event. As a result, only one classifier cannot represent them very well. As shown in Fig. 1, visual images corresponding to concept "Obama talk" are totally different due to illumination changes, different views, and scale variations, and so on. Therefore, it is necessary to train multiple classifiers for each concept to consider its diversity and enhance the representation discriminability.[2]

To deal with the above issues, we propose a novel automatic visual concept learning method for social event understanding in videos. Social events considered in this paper are specific public events, such as "Concert of Shakira in Kiev 2011", which are much more complex and diverse than the conventional events defined in the MED dataset. Fig. 2 shows the main steps of the proposed algorithm, which includes automatic concept mining and boosted concept learning.[3]

Automatic concept mining. First, we collect an auxiliary image set with corresponding text descriptions from Flickr.[4] Then, based on the text information, we automatically extract compact semantic phrase segments as concepts. In our implementation, the phrase segments are learned by considering both words stickiness and visual representativeness. The stickiness of phrase segments is measured according to the key phrases value extracted from Wikipedia[5] and the Microsoft N-gram services.[6] The visual representativeness is measured according to the visual similarity of images returned from Flickr when a phrase segment is used as the search query. The finalist of our concepts is the selected subset of phrase segments with large stickiness and visual representativeness.

Boosted concept learning. The basic idea is to integrate the concept classifier learning process in a boosting framework. Each iteration of boosting begins by learning a classifier for each concept according to the instance weights assigned by the previous step. Then, the learned classifier is applied to the instances to obtain their representations. Finally, the resulting feature representation is applied to learn a new weak classifier and the new weights of instances are updated by using the classification scores. Based on the above procedure, it is clear that the instances are iteratively reused with different weights to learn multiple classifiers to enhance the representation discriminability of each concept.

Compared with the existing methods, our contributions are three-fold.

1. To avoid defining concept manually, we propose an automatic visual concept mining method for social event understanding in videos by taking the advantages of the social media sites (Flickr and Wikipedia) and the cloud-based Web N-gram service platform provided by Microsoft.
2. Different from the traditional concept representation only based on single classifiers, we propose a boosted concept learning algorithm to iteratively learn multiple classifiers to enhance the representation discriminability of each concept. As a result, the learned representation can model different aspects of visual concepts.
3. Extensive experimental results demonstrate the effectiveness of our boosted concept learning algorithm for social event understanding.

The rest of the paper is organized as follows. In Section II, we summarize the work most related to this paper. The proposed automatic concept mining is described in detail in Section III, while the boosted concept learning method is presented in Section IV. Experimental results are reported and analyzed in Section V. Finally, we conclude the paper with future work in Section VI.

II. Related Work

In this section, we review the related work about event analysis, attribute concept, and knowledge transfer, which are the three areas most related to our algorithm.

Event Analysis: Recently, many methods have been proposed for vision based event detection and recognition [16]–[18], [12]. There are two methods more related to ours by leveraging on auxiliary dataset. In [19], Domain Selection Machine (DSM) for event recognition in videos is proposed by leveraging on web images. Compared with our algorithm, this method is based on only visual features without considering the context information of videos and images, while our method uses video descriptions to mine visual concepts. The event recognition problem is done in [9], where visual features and text descriptions are used to understand video event, and videos with only visual content are tested. Though the video descriptions are used, the previously labeled "atomic event" dataset are needed while we endeavor to mine visual concepts automatically.

[2][Online] Available: https://www.youtube.com/watch?v=SLQsdW4AySE

[3][Online] Available: https://drive.google.com/file/d/0B0os9DsfRuirVVZsV05wZEJxT2M/view?usp=sharing

[4][Online] Available: http://www.flickr.com/

[5][Online] Available: http://www.wikipedia.org/

[6][Online] Available: http://web-ngram.research.microsoft.com/

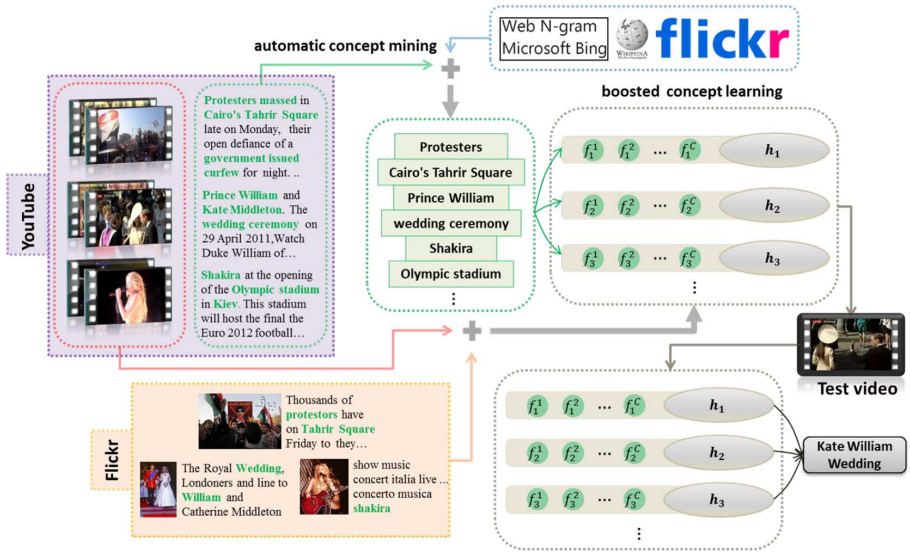

Fig. 2. Our proposed social event understanding framework including automatic concept mining and boosted concept learning. For *automatic concept mining*, we collect an auxiliary image set with corresponding text descriptions from Flickr. Based on the text information, we automatically extract compact semantic phrase segments as concepts with the help of Wikipedia, Microsoft N-gram services, and Flickr. For *boosted concept learning*, the basic idea is to integrate the concept classifier learning process in a boosting framework to iteratively learn multiple classifiers to model concept diversity. Given a test video, multiple concept classifiers are applied to create concept descriptors. Then, the test video is recognized using these concept descriptors. (Videos via YouTube and images via Flickr under Creative Commons License.)

Attribute Concept: The concept in our work can be seemed as visual attribute which is used in object recognition, action recognition. In [20], a detection method for unseen object classes is proposed based on a human specified high-level descriptor of target objects. More recently, a ranking function is learned for each attribute of object to model the relative attributes in [21]. It has been suggested in [22], that proper usage of semantic concepts is likely to improve video analysis. In [23], [10], [24], concept based event representation and natural languages summarization method is studied. In [11], [13], semantic labels of external videos are used as attributes to characterize video complex properties. The method in [25]is more related to our work at attribute concept mining, where the associations between object classes and attributes are determined using semantic relatedness. Compared with our automatic concept mining method, their attributes are mined for general objects.

Knowledge Transfer: Many methods have been proposed for knowledge transfer between different domains. Most of the existing methods are designed to improve classification accuracies for unlabeled instances in target domain by leveraging on the labeled instances in source domain [26], [27]. In multimedia community, there are also several algorithms [14], [28], [29]. In [29], the authors use a graph to model the distribution discrepancy problem of the social stream between Twitter and YouTube. In our work, the image set collected from Flickr is source domain, which is combined with the frames in video set (target domain) for concept learning. Different from above mentioned domain transfer methods, the image set in our method are related to the video set by video descriptions and image meta data, such as tags, tiles and descriptions.

III. Automatic Concept Mining

Different from conventional concept based event analysis methods where concept labels are defined manually, we attempt to automatically mine visual concepts from text descriptions. Our visual concepts are obtained by the following two steps. Firstly, we segment text descriptions into semantic phrases with the help of Wikipedia and N-gram web services. Then we use social media site Flickr to pick out a subset of these phrase segments as our visual concepts for event understanding.

A. Description Segmentation

Given text description $\mathcal{D}e$ of a video, the problem of description segmentation is to split $\mathcal{D}e$ into m non-overlapping and consecutive segments. Specifically, this can be denoted as the expression $\mathcal{D}e =< \mathbf{se}_1, \mathbf{se}_2, \ldots, \mathbf{se}_m >$. Here, each segment $\mathbf{se}_i$ is a compact semantic unit. This kind of semantic units are more expressive than single word, and contain much more compact semantic information than the whole sentence. Practically, the descriptions of a given video could be segmented into many phrase segments. Any two or three words could construct a phrase segment. However, we want to find some valid phrases which probably denote an object, a celebrity or a building name related to event. The description segmentation problem can be formulated as an optimization problem.

$$\arg \max_{\mathbf{se}_1,\ldots,\mathbf{se}_m} Stc(\mathcal{D}e) = \sum_{i=1}^{m} Stc(\mathbf{se}_i) \tag{1}$$

where Stc is a function that measures the stickiness of a segment. The high stickiness score of a segment $\mathbf{se}$ indicates the high probability of being a celebrity, a building name or an object. It is worth to note that the summation in Eq. (1) will not make the stickiness of each segment be high, because all these segments are not independent. If the stickiness of a specific segment is enhanced by adding or removing one word, the stickiness of the adjacent segments will probably be decreased. Specifically, given the text description $\mathcal{D}e$ of a video,

the formulation Eq. (1) in Section III-A aims to split $\mathcal{D}e$ into m non-overlapping and consecutive segments. Practically, the m and the words contained in each segment are all unknown before solving Eq. (1). For a specific word in $\mathcal{D}e$, it will be assigned to a segment which can contribute larger stickiness value to the summarization. Thus, through solving Eq. (1), we obtain m segments which have the highest stickiness value compared with any other kinds of consecutive word groups. Though many segments which do not contain much useful information will be generated by this method, the solutions of Eq. (1) give us the clues for choosing concepts as illustrated in Section III-B. As illustrated in Section III-B, the event concepts can be the segments with higher stickiness scores. One idea of computing Stc is to count the appearance of segment in a very large corpus. The other one is to look up the segments in a knowledge base where valid segments are more easily recognized. Wikipedia can be exploited for this purpose, which is by far the largest online encyclopedia in the World Wide Web. Specifically, to compute the stickiness score, we adopt two large corpus. One is the document collection in the World Wide Web indexed by Microsoft Bing, which provides a good estimate of the statistics of commonly used phrases in English. The other one is a dictionary extracted from Wikipedia. If a segment matches any entry in the dictionary, it has a higher prior probability of being a valid phrase for concept. Here, the stickiness function Stc is defined as

$$Stc(\mathbf{se}) = L(\mathbf{se}) \cdot e^{Q(\mathbf{se})} \cdot Sigmod(SCP(\mathbf{se})) \tag{2}$$

where $Q(\mathbf{se})$ is the probability that $\mathbf{se}$ appears as the anchor text in the Wikipedia articles that contain $\mathbf{se}$. $Q(\mathbf{se})$ can be obtained by using the Wikipedia Keyphraseness dataset created in [30]. The $Q(\mathbf{se})$ will be assigned to zero if $\mathbf{se}$ does not appear in the Keyphraseness dataset. $SCP(\cdot)$ denotes the symmetric conditional probability for N-grams (N = 3 in our experiment). $SCP(\cdot)$ is defined to measure the "cohesiveness" of a segment $\mathbf{se} =< w_1, \ldots, w_n > (n > 1)$ by considering all possible binary segmentations.

$$SCP(\mathbf{se}) = log \frac{Pr(\mathbf{se})^2}{\frac{1}{n-1}\sum_{i=1}^{n-1} Pr(w_1, \ldots, w_i) Pr(w_{i+1} \ldots w_n)} \tag{3}$$

Here, $Pr(\cdot)$ denotes the prior probability of the word sequence $< w_1, \ldots, w_n >$ derived from Microsoft Web N-Gram service. The item $L(\mathbf{se})$ in Eq. (2) is used to prefer longer segments. Here, $|\mathbf{se}|$ denotes the number of words contained in $\mathbf{se}$.

$$L(\mathbf{se}) = \begin{cases} \frac{(|\mathbf{se}|-1)}{|\mathbf{se}|}, for |\mathbf{se}| > 1 \\ 1, for |\mathbf{se}| = 1 \end{cases} \tag{4}$$

B. Concept Selection

The phrase segments for text descriptions illustrated in Section III-A are obtained by only considering the text information. These segments may be useful to explain a specific event, but they probably cannot be used for visual information for event analysis. Now, we introduce how to select the visual concepts from these segments considering both textual and visual information. The probability of a segment $\mathbf{se}$ to be selected as a concept is computed by a product of segment stickiness and visual representativeness as in Eq. (5).

$$Score(\mathbf{se}) = Stc(\mathbf{se}) \cdot V_{flickr}(\mathbf{se}) \tag{5}$$

Here, $Stc(\mathbf{se})$ is the segment stickiness computed in Section III-A, $V_{flickr}(\mathbf{se})$ is the visual representativeness which is defined as the effectiveness of segment by describing the visual content of the videos according to [31]. Specifically, $V_{flickr}(\mathbf{se})$ is computed as the visual similarities of returned images $I_{\mathbf{se}}$ from Flickr when segment $\mathbf{se}$ is used as search query. As shown in Eq. (6), we compute $V_{flickr}(\mathbf{se})$ according to the centroid-based cohesion due to its effectiveness and efficiency [31]. Visual representativeness in [31]is computed for general semantic tags related to large number of images while concepts for specific events are considered in our method.

$$V_{flickr}(\mathbf{se}) = \sum_{i \in I_{\mathbf{se}}} sim(i, Cent(I_{\mathbf{se}})) \tag{6}$$

Here, similarity function $sim()$ can be computed by distance function using low level visual features, and $Cent(I_{\mathbf{se}})$ is the centroid of $I_{\mathbf{se}}$. Consider that a segment $\mathbf{se}$ may be only relevant to a specific region of an image, it is reasonable to measure the visual similarity for different image regions separately. However, it is time-consuming using traditional methods. Inspired by [32], we use a similarity measurement method through Fourier transformation. We denote the feature vector of image i as $\mathbf{x}_i$, which is obtained using Bag-of-Word [33], [34]and SPM [35]scheme. SPM divides image into regions to consider spatial information, and features in different regions are concatenated to obtain the feature vector. Thus, each part of $\mathbf{x}_i$ denotes visual feature of one region in image i. To measure correlations of segments which are only related to local image regions, it is reasonable to align their image features according to different image regions. As shown in Eq. (7), we use the 1-dimensional circulate encoding method [32]to achieve the visual similarity of an image pair (i, j).

$$sim(i,j) = \mathcal{F}^{-1}\left(\frac{\mathcal{F}(\mathbf{x}_i)^* \odot \mathcal{F}(\mathbf{x}_j)}{\mathcal{F}(\mathbf{x}_i)^* \odot \mathcal{F}(\mathbf{x}_i) + \lambda}\right) \tag{7}$$

where, $*$ denotes the conjugation, $\odot$ denotes the element-wise multiplication, $\mathcal{F}$ is the 1D discrete fourier transformation, and $\mathcal{F}^{-1}$ is its inverse, $\mathbf{x}_i$ and $\mathbf{x}_j$ are the feature vectors of image pair (i, j), and λ is the regularization coefficient which ensures the stability of the filter. The similarity is measured for all possible alignments of two image features rapidly due to Fourier domain processing.

Note that we do not use the number of images in $I_{\mathbf{se}}$ to normalize the sum in Eq. (6). This is due to the following reasons. Flickr is a photo sharing website where users can upload photos to record the events happened in the world around them. The tags and descriptions are mainly used to clarify the information contained in these photos. One or several words of segment $\mathbf{se}$ must be contained in the descriptions, titles or tags related to images returned from Flickr when $\mathbf{se}$ is used as query. The more images contained in Flickr site for a single phrase segment $\mathbf{se}$, the more likely $\mathbf{se}$ can be represented by visual information.

Thus the number of images in $I_{\mathbf{se}}$ can reflect the visual representativeness of the phrase segment $\mathbf{se}$. Another possible choice to measure the visual representativeness is using the query logs of image search engine for each segment. However, it is difficult to collect search logs in occurrence time of all events.

IV. Boosted Concept Learning

As illustrated in Section III, the concepts in *Cpts* are extracted automatically from text descriptions of the videos. In practice, some image frames are not truly related to the concepts assigned to a video as shown in Fig. 1. These images will add too much noise if we learn concept classifiers using the extracted visual features from the whole video. In this section, we introduce a boosted concept learning algorithm to iteratively obtain multiple classifiers for each concept. With the help of auxiliary web images, our concept classifiers are trained using the most related image frames in videos. We first overview the whole process of our algorithm. Then, we introduce each step in details.

A. Overview

Our goal is to learn multiple classifiers for each concept for social event understanding on video dataset by leveraging on an auxiliary image dataset. To achieve this goal, we make use of boosting to learn one classifier at each iteration by considering instance weight distributions of the two datasets. After iteratively learning, we can obtain multiple classifiers to describe each concept.

Let $\mathcal{V} = \{\mathbf{v}_i\}_{i=1}^n$ be the event dataset including n videos. Here, $\mathbf{v}_i$ denotes a video instance and consists of l_i frames. The event class labels of videos in $\mathcal{V}$ are denoted as $\mathbf{Y}_\mathcal{V} = \{y_i\}_{i=1}^n, y_i \in \{1, 2, \ldots, K\}$, K is the number of event classes and n is the number of video instances. We use $\mathbf{X}_\mathcal{V} = \{\mathbf{x}_i\}_{i=1}^l$ to denote visual feature vectors of all frame images extracted from all n videos. Here, $\mathbf{x}_i$ denotes the visual feature of the ith image, and $l = \sum_{l=1}^n l_i$ is the number of all frames in $\mathcal{V}$. Let $\mathbf{A}_\mathcal{V} = \{\mathbf{a}_i\}_{i=1}^l$ be visual concept annotations for images of all videos. $\mathbf{a}_i$ denotes the index of concepts in *Cpts* related to the ith image in $\mathbf{X}_\mathcal{V}$. Here, *Cpts* is the concept set extracted in Section III. Since different videos may be related to several different concepts, frame images in different videos probably have different concepts. Let I be an auxiliary image set including m images. We use $\mathbf{X}_I = \{\mathbf{x}_i\}_{i=l+1}^{l+m}$ to denote m feature vectors for all images in I. Concept annotations for all images in the auxiliary image set are denoted as $\mathbf{A}_I = \{\mathbf{a}_i\}_{i=1}^m$ according to *Cpts*. Let $\mathbf{d}^V$ and $\mathbf{d}^I$ denote the distribution vectors for all images in $\mathcal{V}$ and I, respectively.

Given $\mathcal{V}$, I, $\mathbf{d}^V$, and $\mathbf{d}^I$, we introduce how to learn effective concept representation under boosting framework [36]. At each iteration t of the boosting process, (1) we first learn domain adaptation features, which can alleviate the domain difference between $\mathcal{V}$ and I. According to $\mathbf{d}^V$ and $\mathbf{d}^I$ updated at the $(t-1)$th iteration, we sample a subset $\mathbf{S}_\mathcal{V}$ of $\mathbf{X}_\mathcal{V}$ and a subset $\mathbf{S}_I$ of $\mathbf{X}_I$ to lean a shared feature representation $\mathbf{F}_t(\mathbf{x})$ via a recent method mSDA [37]. (2) Based on the learned feature representation $\mathbf{F}_t(\mathbf{x})$ and image distributions $\mathbf{d}^V$ and $\mathbf{d}^I$, we train concept classifiers $\{\mathbf{f}_t^i(\mathbf{x})\}_{i=1}^C$. Here, C is the number of all concepts. Then, the image weight distributions $\mathbf{d}^V$ and $\mathbf{d}^I$ are updated by using the concept classifier scores. Moreover, the concept classifier scores can also be used to describe each frame of video. By using sum-pooling among all frames in a video, we can obtain the representation for each video. (3) Then, based on the video representations, a weak classifier $\mathbf{h}_t(\mathbf{x})$ and the corresponding combination coefficient α_t can be learned for social events. (4) Finally, We update the distribution $\mathbf{d}^V$ based on the error ε_t of this event classifier. In our update, the values of the distribution $\mathbf{d}^V$ related to images of videos which are not classified correctly by $\mathbf{h}_t(\mathbf{x})$ are increased. Thus, images of these videos are more likely to be selected for training a new feature representation function $\mathbf{F}_{t+1}(\mathbf{x})$ at the next iteration.

Once this procedure converges, we obtain a set of concept classifiers $\{\mathbf{f}_t^c(\mathbf{x})\}_{t=1}^T$ for each concept c, feature representation functions $\{\mathbf{F}_t(\mathbf{x})\}_{t=1}^T$, weak event learners $\{\mathbf{h}_t(\mathbf{x})\}_{t=1}^T$ and their corresponding combination coefficients $\{\alpha_t\}_{t=1}^T$. The weak event classifier can be combined as a strong classifier $\mathbf{H}(\mathbf{x})$ as shown in Eq. (19) to classify new videos. To classify a new video, each frame of the video is mapped with feature functions $\{\mathbf{F}_t(\mathbf{x})\}_{t=1}^T$ to obtain T feature spaces (T is the number of boosting iterations). Then, concept classifiers $\{\mathbf{f}_t^c(\mathbf{x})\}_{t=1}^T$ are used to create a concept descriptor for the new video, which can be classified by its corresponding weak classifier $\mathbf{h}_t(\mathbf{x})$. Via the corresponding combination coefficients $\{\alpha_t\}_{t=1}^T$, the final class label of this video can be assigned by $\mathbf{H}(\mathbf{x})$. The details of the above process are summarized in Algorithm 1. In short, at each iteration t, our algorithm has 4 major components, learn domain adaptation feature, learn concept classifier, learn event weak classifier, update weight distribution. Finally, we can construct a strong social event classifier. The details of each component are introduced in the following subsections.

Algorithm 1 Boosted Concept Learning.

input: concept set $\mathbf{Cpts} = \{c\}_{c=1}^C$, videos $\mathcal{V} = \{\mathbf{v}_i\}_{i=1}^n$, labels $\mathbf{Y}_\mathcal{V} = \{y_i\}_{i=1}^n$, frame features $\mathbf{X}_\mathcal{V} = \{\mathbf{x}_i\}_{i=1}^l$, concept annotations $\mathbf{A}_\mathcal{V} = \{\mathbf{a}_i\}_{i=1}^n$, image set I and its features $\mathbf{X}_I = \{\mathbf{x}_i\}_{i=l+1}^{l+m}$, concept annotation $\mathbf{A}_I = \{\mathbf{a}_i\}_{i=1}^m$, weight distribution $\mathbf{d}^V \leftarrow 1/l$, $\mathbf{d}^I \leftarrow 1/m$.

output: $\{\mathbf{F}_t(x)\}_{t=1}^T$, $\{\mathbf{f}_t^i(x)\}_{i=1}^C$, $\{\alpha_t\}_{t=1}^T$, and $\{\mathbf{h}_t(x)\}_{t=1}^T$.

1 **for** $t = 1$ to T **do**
2 Sample $\mathbf{S}_V$ and $\mathbf{S}_I$ from $\mathbf{X}_\mathcal{V}$ and $\mathbf{X}_I$ according to $\mathbf{d}^V$ and $\mathbf{d}^I$, respectively.
3 Learn domain adaptation feature representation $\mathbf{F}_t(\mathbf{x})$ on $\mathbf{S}_V \cup \mathbf{S}_I$ as in Section IV-B.
4 Learn concept classifiers $\{\mathbf{f}_t^i(x)\}_{i=1}^C$ according to $\mathbf{F}_t(\mathbf{x})$ by considering $\mathbf{d}^V$ and $\mathbf{d}^I$ as in Section IV-C.
- Compute the error $\{\varepsilon_t^i\}_{i=1}^C$ and $\{\alpha_t^i\}_{i=1}^C$ for all concepts according to Eq. (11) and Eq. (12).
- Update distribution $\mathbf{d}^V$ and $\mathbf{d}^I$ as in Eq. (13).

5 Learn weak classifier $\mathbf{h}_t(\mathbf{x})$ according to $\mathbf{d}_t^V$ as in Section IV-D.
- Compute the error ε_t and α_t according to Eq. (16) and Eq. (17), respectively.
- Update distribution $\mathbf{d}^V$ as in Eq. (18).

6 **end**
7 Obtain social event classifier $\mathbf{H}(\mathbf{x})$ as in Eq. (19).

B. Domain Adaptation Feature Learning

The simple yet effective marginalized stacked denoising auto-encoder (mSDA) method has been successfully applied for transfer learning in document sentiment analysis. The basic idea of mSDA is to combine the instances in the source and target domains together to learn a common feature representation [38], [37]. The SDAs are able to disentangle hidden factors which explain the variations in the input data, and automatically group features in accordance with their relatedness to these factor. This helps transfer across domains as these generic concepts are invariant to domain-specific features [38]. More specifically, as mentioned in [37], the learned common representations would tend to reconstruct, and be reconstructed by, co-occurring features, typically of similar sentiment. Hence, the source-trained classifier can assign weights even to features that never occur in its original domain representation, which is "re-constructed" by mSDA.

In this work, we go further beyond mSDA and SDA, and propose a distribution sensitive feature learning method within our boosting framework. Specifically, we sample a subset $\mathbf{S}_{\mathcal{V}}$ of $\mathbf{X}_{\mathcal{V}}$ and a subset $\mathbf{S}_I$ of $\mathbf{X}_I$ according to image weight distributions $\mathbf{d}^V$ and $\mathbf{d}^I$ to learn domain adaptation feature representation. Let $\{\mathbf{x}_i\}_{i=1}^S = \mathbf{S}_{\mathcal{V}} \cup \mathbf{S}_I$ be feature vectors of all sampled image instances. The mSDA method reconstructs the original image features with a single mapping function by minimizing the following squared reconstruction loss.

$$\frac{1}{s}\sum_{i=1}^{s}\|\mathbf{x}_i - \mathbf{W}\tilde{\mathbf{x}}_i\|^2 \tag{8}$$

Here, $\tilde{\mathbf{x}}_i$ is a corrupt version of $\mathbf{x}_i$ and is obtained by stochastically setting some elements of the input $\mathbf{x}_i$ to zero. Hence, denoising auto-encoder is trying to predict the missing values from the non-missing values. The corruptions are useful for capturing the statistical dependencies between the inputs [39]. W denotes the mapping matrix which projects the corrupted feature $\tilde{\mathbf{x}}_i$ to $\mathbf{x}_i$. Though it is just a single linear mapping, more representative domain invariant features can be learned when combined with the non-linear activation function and the layer-wise stacking scheme [37]. With r different corruptions, Eq. (8) can be written as

$$\mathcal{L}_{sq}(\mathbf{W}) = \frac{1}{2sr}\sum_{j=1}^{r}\sum_{i=1}^{s}\|\mathbf{x}_i - \mathbf{W}\tilde{\mathbf{x}}_{ij}\|^2 \tag{9}$$

This equation can be solved using the closed-form solution for ordinary least squares. A more simplified solution is given in [37]by marginalizing all the noises when $r \to \infty$.

$$\mathbf{W} = E[\mathbf{P}]E[\mathbf{Q}]^{-1}, \mathbf{Q} = \tilde{\mathbf{X}}\tilde{\mathbf{X}}^\top, \mathbf{P} = \bar{\mathbf{X}}\tilde{\mathbf{X}}^\top \tag{10}$$

Here, $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_s]$, $\bar{\mathbf{X}} = [\mathbf{X}, \ldots, \mathbf{X}]$ is r copies of $\mathbf{X}$, and the corrupted version of $\bar{\mathbf{X}}$ is denoted as $\tilde{\mathbf{X}}$.

The $\mathbf{W}$ obtained by minimizing Eq. (9) can be considered practically as a linear mapping function. After the linear feature mapping, as in traditional deep learning methods, a non-linear activation function (e.g., $\tanh(\cdot)$) is applied. To construct a deep layer-wise structure, such one layer auto-encoders are stacked together. In practice, the mSDA structure for feature representation is fixed by weight matrices where each layer has one weight matrix and a nonlinear $\tanh(\cdot)$ function. For simplicity, we denote the multi-layer mSDA feature representation as a single function $\mathbf{F}(\mathbf{x})$. Take a 2-layer mSDA as an example, the function $\mathbf{F}(\mathbf{x}) = \tanh(\mathbf{W}_2\tanh(\mathbf{W}_1\mathbf{x}))$ is the learned feature representation.

C. Concept Classifier Learning

Based on the learned domain adaptation feature representation $\mathbf{F}(\mathbf{x})$, we can learn concept classifier $\mathbf{f}^c(\mathbf{x})$ for each concept c on two sampled subsets $\mathbf{S}_{\mathcal{V}}$ and $\mathbf{S}_I$ obtained by considering weight distributions $\mathbf{d}^V$ and $\mathbf{d}^I$. For simplicity, we make use of the linear SVM due to its effectiveness and efficiency to learn concept classifier for each concept in *Cpts*. For concept c, images in both $\mathbf{S}_{\mathcal{V}}$ and $\mathbf{S}_I$ related to concept c are adopt to learn the corresponding concept classifier $\mathbf{f}^c(\mathbf{x})$. For efficiency, the classifier training for different concepts can be conducted in parallel. After training, the classifier $\mathbf{f}^c(\mathbf{x})$ can assign a binary value 1 or -1 to indicate whether an input image $\mathbf{x}$ is related to concept c or not.

At each iteration of the boosting process, we need to obtain the corresponding feature descriptors $\{\mathbf{z}_i\}_{i=1}^n$ for all videos $\{\mathbf{v}_i\}_{i=1}^n$. For the ith video, its images can be described by the scores of the concept classifiers $\{\mathbf{f}^i(\mathbf{x})\}_{i=1}^C$. Then, we can obtain the video representation through sum-pooling scheme among its images. Moreover, based on the learned concept classifier $\mathbf{f}^c(\mathbf{x})$ for each concept c, we can compute the corresponding classification error ε^c and the weight α^c as in Eq. (11) and Eq. (12), respectively.

$$\varepsilon^c = \frac{1}{\sum_{i\in image(c)} d_i}\sum_{i\in image(c)} d_i \cdot \mathbb{I}(1 \neq \mathbf{f}^c(\mathbf{x}_i)) \tag{11}$$

$$\alpha^c = \ln((1-\varepsilon^c)/\varepsilon^c) \tag{12}$$

In Eq. (11), $image(c)$ denotes the index of images related to concept c in $\mathbf{X}_{\mathcal{V}}$ and $\mathbf{X}_I$. Features in $\mathbf{X}_{\mathcal{V}}$ and $\mathbf{X}_I$ are firstly transformed through $\mathbf{F}(\mathbf{x})$. Then all concept classifiers are used to predict their labels. Here, $\mathbb{I}(.)$ is the indicator function.

Update distributions $\mathbf{d}^V$ *and* $\mathbf{d}^I$**:** Rewrite $\mathbf{d}^V = \{d_i\}_{i=1}^l$, the d_i can be updated as in Eq. (13). In the same way, the update scheme for $\mathbf{d}^I$ can be derived directly.

$$d_i = d_i \cdot \exp(\alpha^c \cdot \mathbb{I}(1 \neq \mathbf{f}^c(\mathbf{x}_i))) \qquad \forall i \in image(c) \tag{13}$$

Here, $image(c)$ denotes the index of images in $\mathbf{X}_{\mathcal{V}}$ related to concept c. Note that Eq. (13) is different from the update scheme as in conventional boosting method [36]. Here, the weights of correctly classified images will be increased. By this kind of update, the most representative images for each concept, which are always correctly predicted by concept classifiers, will be selected to learn new representations.

D. Event Weak Classifier Learning

Once the concept descriptors $\{\mathbf{z}_i\}_{i=1}^n$ for all videos are obtained as shown in Section IV-C, we can learn a corresponding weak classifier for social event classification. Now, we introduce how to design an effective and efficient weak classifier $\mathbf{h}(\mathbf{x})$ for social event by considering the weight distribution $\mathbf{d}^V$ of images in the video set. For simplicity, we adopt the linear

weighted support vector machine (WSVM) [40] due to its efficiency. Descriptors of all videos are denoted as $\{\mathbf{z}_i\}_{i=1}^n$. The event class labels of the training instances are $\{y_i\}_{i=1}^n$. Then, a 2-class linear support vector machine can be written as

$$\begin{aligned}&\min_{\mathbf{u},b,\xi} \frac{1}{2}\mathbf{u}^\top \mathbf{u} + C\sum_{i=1}^{n} \hat{d}_i \xi_i \\ &s.t. \quad y_i(\mathbf{u}^\top \mathbf{z}_i + b) \geq 1 - \xi_i, \quad i = 1, \ldots, n \\ &\xi_i \geq 0, \quad i = 1, \ldots, n. \end{aligned} \tag{14}$$

Note that Eq. (14) can be viewed as assigning a penalty parameter $\hat{d}_i C$ to $\mathbf{z}_i$. Thus different instances will be constrained with different penalties in the learning process. For the multi-class problem, one-vs-the-rest strategy can be adopted.

Now we illustrate the $\hat{d}_i$ in Eq. (14). The event classifiers need to be trained according to the distribution of videos. However, the distribution is only defined for frames contained in the videos for the convenience of the distribution update according to errors of concept classifiers. Thus, we need to transform the distribution $\mathbf{d}^V = \{d_i\}_{i=1}^l$ of frames to the distribution $\hat{\mathbf{d}}^V = \{\hat{d}_i\}_{i=1}^n$ of videos. Rewrite $\mathbf{d}^V = \{d_i\}_{i=1}^l$ and $\hat{\mathbf{d}}^V = \{\hat{d}_i\}_{i=1}^n$. Here, $\{d_i\}_{i=1}^l$ are l weight values of the images in video set $\mathcal{V}$, and $\{\hat{d}_i\}_{i=1}^n$ denote weight values of videos. $\hat{\mathbf{d}}^V$ can be obtained by accumulating all the weight values in $\{d_i\}_{i=1}^l$ of images in the corresponding video. This can be illustrated as Eq. (15), where $image(j)$ denotes index of all images in $\mathbf{X}_\mathcal{V}$ belonging to the jth video in $\mathcal{V}$.

$$\hat{d}_j = \sum_{i \in image(j)} d_i, \forall j = 1, \ldots, n \tag{15}$$

Similar to the conventional multi-class AdaBoost scheme [36], after constructing the weak classifier $\mathbf{h}(\mathbf{x})$ according to distribution $\hat{\mathbf{d}}^V$ of videos, we compute the classification error ε and assign a weight α for the weak learner $\mathbf{h}(\mathbf{x})$ as shown in Eq. (16) and Eq. (17), respectively. Here, $\mathbb{I}(.)$ is the indicator function, K is the number of event classes.

$$\varepsilon = \frac{1}{\sum_{i=1}^n \hat{d}_i} \sum_{i=1}^{n} \hat{d}_i \cdot \mathbb{I}(y_i \neq \mathbf{h}(\mathbf{z}_i)) \tag{16}$$

$$\alpha = \ln((1-\varepsilon)/\varepsilon) + \ln(K-1) \tag{17}$$

Then, we update all weight values of $\mathbf{d}^V$ as Eq. (18) according to the learned weak classifier.

$$d_i = d_i \cdot \exp(\alpha \cdot \mathbb{I}(y_j \neq \mathbf{h}(\mathbf{z}_j))), i \in image(j) \qquad \forall i = 1, \ldots, l \tag{18}$$

Here, i denotes the index of images in $\mathbf{X}_\mathcal{V}$, j denotes the index of videos in $\mathcal{V}$, and $image(j)$ denotes index of all images in $\mathbf{X}_\mathcal{V}$ related to the jth video in $\mathcal{V}$.

E. Social Event Classifier

Once the boosting procedure converges, we obtain a set of feature representation functions $\{\mathbf{F}_t(\mathbf{x})\}_{t=1}^T$, concept classifiers $\{\mathbf{f}_t^c(\mathbf{x})\}_{t=1}^T$, and a set of weak learners $\{\mathbf{h}_t(\mathbf{x})\}_{t=1}^T$ and their corresponding combination coefficients $\{\alpha_t\}_{t=1}^T$. Then, the learned social event classifier $\mathbf{H}(\mathbf{x})$ is

$$\mathbf{H}(\mathbf{x}) = \arg\max_k \sum_{t=1}^{T} \alpha_t \cdot \mathbb{I}(\mathbf{h}_t(\mathbf{x}) = k), k \in \{1, \ldots, K\}. \tag{19}$$

F. Discussion

Firstly, we give the more detailed explanations to the distributions $\mathbf{d}^V$ and $\mathbf{d}^I$ which are used in our algorithm to sample image instances and iteratively updated in the boosting procedure. (1) The distribution $\mathbf{d}^V$ is maintained for all frame images of the event videos in $\mathcal{V}$. The $\mathbf{d}^V$ is firstly updated by using the errors of the concept classifiers as shown in Eq. (13). In the boosting process, the image frames with most representative visual appearance for concepts are increased. With the help of $\mathbf{d}^V$, though the concept annotations are only available for videos, we can obtain the most related frames in the videos for each concept to train the corresponding concept classifier. Moreover, by updating $\mathbf{d}^V$ using the errors of the event weak classifiers as shown in Eq. (18), the misclassified videos in $\mathcal{V}$ will be paid more attention at next iteration of the boosting process. (2) We also maintain a distribution $\mathbf{d}^I$ similar as $\mathbf{d}^V$ for all images in image set I. The image set I crawled from Flickr is used to enrich the instances for concept classifier training. The concept annotations for images in I are obtained by assigning the concepts in *Cpts* according to the text descriptions, tags, and titles on Flickr. As we know that the text descriptions or tags of the images crawled from Flickr are provided by common users, this will inevitably introduce inaccurate descriptions which lead to the unreliable concept annotations for the images in the image set I. By updating $\mathbf{d}^I$ based on the concept classifiers as shown in Eq. (13), we can obtain the most related images in the image set to each concept for concept classifier training. It is worth noting that the image dataset and the event video dataset may have different data distributions. To deal with this issue, we adopt the mSDA to bridge the two domains to reduce the domain difference as shown in Section IV-B.

In Fig. 3, we show an example to illustrate how the learned multiple concept classifiers can improve event recognition performance step by step.[7] For simplicity, we only show the concept classifier training process and event recognition for training videos related to event "Victory speech of Obama" from the tth iteration to the $(t+1)$th iteration. At the tth iteration, the learned concept classifier for "Obama talk" can classify *video1* and *video*2 correctly. Due to the representative discriminability of the concept classifier, it cannot describe *video*3 well. By considering the weight distributions of the misclassified instances, our algorithm can learn a new concept classifier for "Obama talk", which can classify the *video*3 correctly. Therefore, our boosted concept learning can iteratively learn multiple classifiers for each concept to enhance its representative discriminability.

[7][Online] Available: https://drive.google.com/file/d/0B0os9DsfRuirQVJmRURXdW52dE0/view?usp=sharing

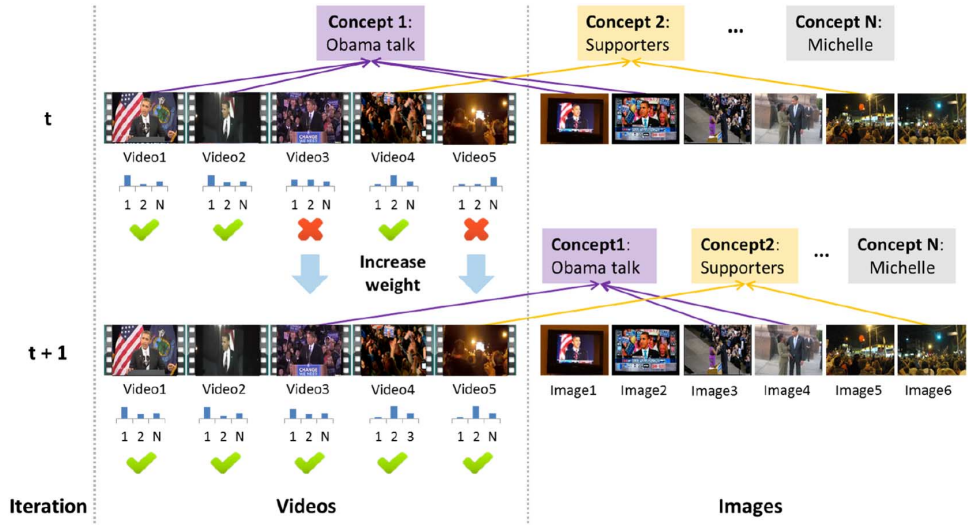

Fig. 3. Process of our boosted concept learning. Here, for simplicity, we only show the process of concept classifier training and event recognition for training videos related to event " Victory speech of Obama" from the tth iteration to the $(t+1)$th iteration. At the tth iteration, the learned concept classifier for "Obama talk" can classify *video1* and *video2* correctly. Due to the representative discriminability of the concept classifier, it cannot describe *video3* well. By considering the weight distributions of the misclassified instances, our algorithm can learn a new concept classifier for "Obama talk," which can classify the *video3* correctly. (Videos via YouTube and images via Flickr under Creative Commons License.)

TABLE I
DETAILS OF EVVE VIDEO DATASET

ID	Event name	Abbr.
1	Austerity riots in Barcelona, 2012	RB
2	Concert of Die toten Hosen, Rock am Ring, 2012	CD
3	Arrest of Domminique Strauss-Kahn	AD
4	Egyptian revolution: Tahrir Square demonstrations	ER
5	Concert of Johnny Hallyday stade de France, 2012	CJ
6	Wedding of Prince William and Kate Middleton	WP
7	Bomb attack in the main square of Marrakech, 2011	BM
8	Concert of Madonna in Rome, 2012	CM
9	Presidential victory speech of Barack Obama 2008	PB
10	Concert of Shakira in Kiev 2011	CS
11	Eruption of Strokkur geyser in Iceland	ES
12	Major autumn flood in Thailand, 2011	MT
13	Jurassic Park ride in Universal Studios theme park	JP

V. EXPERIMENTS

In this section, we firstly introduce the two datasets including a video dataset and an image dataset used in our experiments. Then we show the experimental results and analysis.

A. Dataset and Feature Extraction

For the video dataset, we use the videos of the EVVE dataset collected in [32], which includes 13 specific events. In Table I, we show the details of all these 13 events. Since original videos are not provided by authors, we crawl videos according to the URLs provided in [32] by ourselves. Finally, we obtain 1659 videos with the above event class labels as declared in [32]. Besides, more than 1000 negative videos provided in [32] are used to train concept learners combined with 2000 negative videos crawled from YouTube.

In our implementation, all videos are formatted uniformly to a maximum size of 300 pixels in height and width while the original aspect ratios remain unchanged. Frame images are sampled for all videos with a fixed step size (every 5 seconds). Then, to obtain image representation, we adopt the localized soft-assignment coding(LSC) [41] with a codebook comprised of 1024 keywords to encode the dense SIFT features. Finally, via $1 \times 2 \times 4$ SPM [35], the visual feature dimension is 21504. Since the visual features are extremely sparse, we reduce their dimensions to 200 using PCA which does not decrease the performance much. In all experiments, we choose two-thirds of the videos to be our training instances while the remaining videos are used for testing.

To compute the visual representativeness of each description segment illustrated in Section III-A, we crawl about 100 images per segment from Flickr. Totally, there are more than 1000 K images used in the experiment. Then, the same visual features as above are extracted for each image. For the auxiliary image set which is used to train concept learners, we collect about 40 K images from Flickr according to the keywords of each event. Specifically, we used the nouns (e.g., time, place, and people) contained in the event name as the keywords to search the images on Flickr. The event name is defined by the author of the EVVE dataset [32], such as "die toten hosen rock am ring 2012". For some events, using all words of the event name may lead to invalid search on Flickr and no images are returned. In this case, we randomly ignored several words. As a result, the returned images may not to be strictly relevant to the event. This issue can be slightly dealt with the proposed boosted concept learning method, which can pick out the most related images in both video set and image set for event recognition as illustrated in Section IV. Though this kind of method for collecting images from Flickr really needs some manual work, it is still much more convenient than the traditional methods by annotating the event related concepts manually. The images collected for each event are restricted with the condition that the uploading time cannot be earlier than the practical happening time of the event. Visual features for all crawled images are also extracted as the above method.

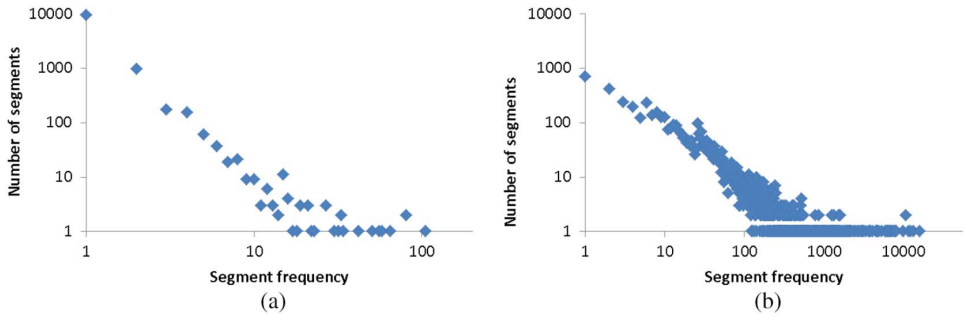

Fig. 4. Segments frequency. (a) Frequencies of the segments appeared in all videos follow the power-law distribution. We can see that above 90% of these segments only appear in a few videos. (b) Frequencies of the segments with enriched instances by leveraging on the Flickr image set. We can see that more related instances are available for specific segments.

TABLE II
EVENT CLASSIFICATION ACCURACIES OF DIFFERENT CONCEPT SELECTION METHODS

Event ID	1	2	3	4	5	6	7	8	9	10	11	12	13	Avg
CommonSegments	0.17	0.23	0.18	0.44	0.06	0.15	0.16	0.09	0.73	0.20	0.04	0.19	0.17	0.22
SpecificSegments	0.12	0.17	0.09	0.47	0.25	0.11	0.20	0.02	0.73	0.73	0.71	0.66	0.13	0.34
StickyConcepts	0.58	0.64	0.44	0.59	0.74	0.84	0.11	0.64	0.91	0.67	0.99	0.88	0.87	0.68
CombineConcepts	0.58	0.73	0.50	0.53	0.69	0.86	0.50	0.61	0.82	0.80	0.97	0.81	0.87	**0.71**

B. Automatic Concept Mining

In this section, we introduce the segment results for video descriptions and the visual concepts selected from them.

Segments of Text Descriptions for Videos: Before segment extraction from video descriptions, the stop-words and words with non-English character in the text description of all videos are removed. After segmentation of text description for each video as illustrated in Section III-A, we obtain a total of 14852 distinct segments. The statistics on the description segments learned by our segmentation method are given as follows. There are 10895 unique segments, 311 of them are unigram (single word), 9230 are 2-grams, and 1354 are 3-grams. We observed that 2-grams account for most part of the segments which is consistent with [30].

As shown in Fig. 4(a), the frequencies of the segments appeared in all videos follow the power-law distribution. We can see that above 90% of these segments only appear in a few videos which are referred as specific segments for simplicity. Though these segments are much more efficient for distinguishing classes of the corresponding videos, there are few instances for learning concept classifiers. Segments with more related video instances are called common segments. Compared with specific segments, these common segments are not efficient to represent a specific event. This can be proved by the event recognition results shown in the first two rows of the Table II. *specificSegments* denotes the method using segments appeared between 5 to 10 times while *commonSegments* denotes the method using segments appeared more than 20 times. We obtain about 40 segments for *commonSegments* and *specificSegments*, respectively. In both methods, the segments used as the concept labels to train concept classifiers are similar to the method in [10]. Then, the learned concept classifiers are combined to decide the event class. We can see that the *specificSegments* method performs better. The simple experiment gives us an insight that if we want to improve the event recognition performance, we should enrich the instances for specific segments.

TABLE III
STICKINESS AND REPRESENTATIVENESS OF SEGMENTS

Event ID	Segments	Stickiness	VR.
1 (#707)	shops attacked	0.87	0.15
	riot police	0.75	0.23
	starbucks fire	0.68	0.11
	scuffles strike	0.67	0.44
2 (#178)	hier kommt alex	2.03	0.21
	rock im park	1.91	0.14
	rock ring	0.88	0.05
	ring festival	0.63	0.20
3 (#668)	diallo dsk pulled	0.89	0.05
	kahn resigned imf	0.88	0.05
	house arrest	0.78	0.14
	sexual assault	0.72	0.18

Concept Selection: As shown in Fig. 4(a), most segments only correspond to a few videos. By leveraging on the Flickr image set, we can enrich instances for specific segments as illustrated in Section V-B1. This is also the reason why we introduce an auxiliary image set to learn concept classifiers. By annotating images from Flickr with the segments introduced in Section V-B1, we get more visual instances for each segment. The segment frequencies on Flickr are shown in Fig. 4(b). Compared with Fig. 4(a), we can see that more related instances are available for specific segment.

We obtain our concepts by choosing a subset of these segments with the help of stickiness and visual representativeness as illustrated in Section III. In Table III, we show four examples of segments for each event with high stickiness score computed as in Section III-A. Due to the space limitation, we only show segment examples of 3 events. The number of unique segments is shown below the ID for each event. We can see that the visual representativeness is always inconsistent with the stickiness value. For the first event as shown in the first row of Table III, the segment "shops attacked" has the highest stickiness value 0.87 and the relative small visual representativeness value 0.15 compared with other three segments. On the contrary, the segment "scuffles strike" has the smallest stickiness value

TABLE IV
COMPARISONS OF SEVERAL TRADITIONAL METHODS FOR EVENT CLASSIFICATION

Event ID	1	2	3	4	5	6	7	8	9	10	11	12	13	Avg
PoolFeature	0.13	0.23	0.05	0.26	0.73	0.76	0.07	0.39	0.82	0.20	0.97	0.73	0.61	0.46
DenseTraj	0.33	0.45	0.50	0.56	0.91	0.68	0.75	0.73	0.82	0.73	0.96	0.71	0.78	0.69
SVM-2K	0.08	0.18	0.38	0.26	0.71	0.76	0.50	0.27	0.72	0.60	0.90	0.53	0.52	0.49
Co-train	0.25	0.45	0.88	0.35	0.63	0.73	0.25	0.52	0.82	0.53	0.90	0.69	0.70	0.59
CTE-KNN	0.28	0.60	0.22	0.41	0.63	0.41	0.19	0.36	1.00	0.33	1.00	0.53	0.57	0.50
CTE-KNN-cpt	0.50	0.68	0.13	0.41	0.71	0.78	0.13	0.64	0.64	0.73	0.95	0.78	0.87	0.61
DSM	0.48	0.52	0.33	0.39	0.64	0.84	0.17	0.47	0.91	0.51	0.92	0.02	0.61	0.52
DSM-cpt	0.42	0.64	0.38	0.50	0.66	0.65	0.25	0.68	0.73	0.73	0.91	0.76	0.96	0.64
BoostConcepts-mFL	0.67	0.73	0.44	0.65	0.81	0.84	0.25	0.59	1.00	0.67	0.98	0.85	0.91	0.72
BoostConcepts	0.67	0.82	0.56	0.76	0.79	0.92	0.25	0.64	1.00	0.73	1.00	0.88	0.91	**0.76**

and the largest representativeness value. From these examples, we can see that some segments with big stickiness value, such as "shops attacked", cannot be expressed easily with visual content due to the various scenes related to the segments. Thus, this kind of segments are not suitable to be selected as our visual concepts.

To further illustrate the importance of the visual representativeness for concept selection, we set two simple baselines, *StickyConcepts* and *CombineConcepts*, for event recognition on the video dataset. *StickyConcepts* denotes the event recognition method on the video dataset using concepts selected according to the stickiness value while *CombineConcepts* denotes the method using concepts selected according to both the stickiness value and the visual representativeness. In these two methods, the concept classifiers are trained with the similar scheme as in conventional methods [23], [10], [24], and the feature descriptions for videos are obtained using the scheme introduced in Section IV-C. Accuracy results for these two methods are shown in Table II. These results demonstrate the necessity of introducing visual representativeness for our visual concept selection. Impact of the number of the selected segments will be illustrated in Section V-D.

C. Boosted Concept Learning

In this section, we evaluate the effectiveness of our boosted concept learning method by comparing it with several recently proposed event recognition methods.

1) *PoolFeature* is a simple baseline, which combines visual features of all frames by sum-pooling to describe each video. Then, the SVM classifier is trained and tested.
2) *DenseTraj* is a baseline method implemented using dense trajectory features [42] based on the code provided by the authors.
3) *SVM-2K* denotes the revised version of the conventional two-view learning method [43] using classifier of the video features for predicting. Specifically, classifiers for two views (video and text in our experiment) are learned for training while only the classifier corresponding to the visual features of videos is used for predicting. This is different from the conventional SVM-2 K [43]. Because in our experiment, text information is only available for training. We use the code provided by the authors to implement the baseline. All parameters are tuned to give best results.
4) *Co-training* denotes the revised method of [44]. Similar as the SVM-2 K method, classifiers for two views (video and text in our experiment) are learned for training while only the classifier corresponding to the visual features of videos is used for predicting. Besides, the auxiliary images are used as unlabeled instances. We use the code provided in [45]. All parameters are tuned to give best results.
5) *CTE-KNN* denotes the method where we combine circulate temporal encoding (CTE) and KNN for event classification. Specifically, the CTE is used as distance function and KNN is used to decide the class for a given video. CTE method is originally proposed in [32] for event based video retrieval on EVVE dataset.
6) *CTE-KNN-cpt* denotes the baseline method based on *CTE-KNN* using the learned concept features.
7) *DSM* is an event recognition method proposed in [19] which is leveraged on the web images from different web sites. Since an auxiliary image set is also adopted in our algorithm, the *DSM* method is used to measure the effectiveness for utilizing auxiliary images of our algorithm.
8) *DSM-cpt* denotes the baseline method based on *DSM* using the learned concept features.
9) *BoostConcepts-mFL* is a simple implementation of our algorithm introduced in Section IV. Concept classifiers are trained without considering multiple feature learning via mSDA. Specifically, original low level visual features of videos and images are used to train concept learners. In the boosting iteration process, distributions of video instances and image instances are updated similarly as the distribution update scheme introduced in Section IV. Then the concept classifiers are used to create feature representations of videos to train event learners.
10) *BoostConcepts* is the implementation of our whole algorithm. Concept classifiers are trained based on the common feature representations (mSDA) of images in video set combined with the images in auxiliary image set. More detailed illustrations could be found in Section IV.

Experimental results for all these methods are shown in Table IV. We can see that our *BoostConcepts-mFL* and *BoostConcepts* methods perform much better than all other methods. Compared with *PoolFeature*, the dense trajectory features *DenseTraj* really improve the event recognition performance, but still cannot outperform our proposed boosted concept features. Besides, it is very time-consuming to extract the dense trajectory features for video understanding. The experimental results are shown in Table IV. The two methods *SVM-2K* and *co-training* perform better than the baseline method *PoolFeature* which only uses the visual features in the video for training classifiers, but are still not comparable with our concept feature based method. From the results of *DSM-cpt* and *CTE-KNN-cpt*,

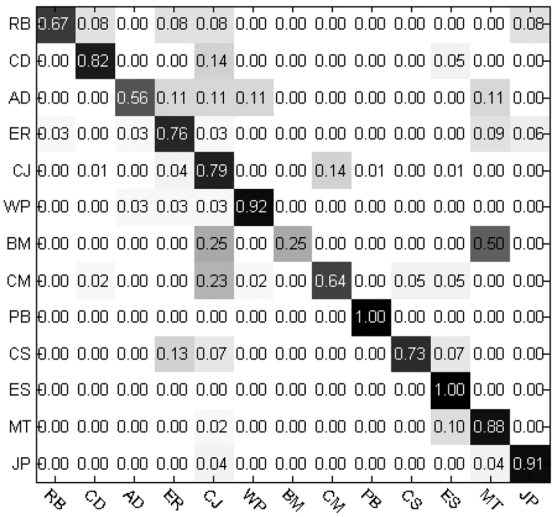

Fig. 5. Confusion matrix of the event recognition results by our *BoostConcepts* method.

we can see that *DSM* and *CTE-KNN* methods can be improved dramatically by using the concept features. Compared with the conventional event recognition methods, our *BoostConcepts-mFL* achieves much better performance, which shows the effectiveness of the concept boosting scheme that effective visual images for each concept are selected iteratively according to the distributions $\mathbf{d}^V$ and $\mathbf{d}^I$ as discussed in Section IV. The only difference between *BoostConcepts-mFL* and *BoostConcepts* is the multiple feature representation learning via mSDA at each iteration. Our *BoostConcepts* shows much better performance, which demonstrates that feature learning via mSDA can reduce the domain difference between the video set and the image set.

In Fig. 5, we show the confusion matrix of the event recognition results for our *BoostConcepts* method. We can see that event "BM" (the full name is "Bomb attack in the main square of Marrakech, 2011") has the maximum confusion value where 50% videos are misclassified as "MT" (the full name is "Major autumn flood in Thailand, 2011"). Intuitively, the events "BM" and "MT" are not similar according to the semantic information contained in their event names. The confusion is because videos of the two events have very similar background, which increases the difficulty to distinguish these two events.

D. Parameter Analysis

In Section V-B, we show event recognition accuracies for different concept selection methods without considering the impact of the number of selected concepts. To make clear how many concepts should be used for event recognition, we show the average accuracies of *CombineConcepts* method with different number of concepts in Fig. 6. We can see that the best accuracy is obtained with 550 concepts and more concepts even up to 1000 do not improve the performance. This can be explained by reviewing the concept mining step of our algorithm in Section III. Generally speaking, more concepts lead to a more complete semantic representation for the videos. However, in our method, the concepts with low stickiness value are inaccurate and will decrease the performance. According to this rule, we select 550 concepts in all experiments.

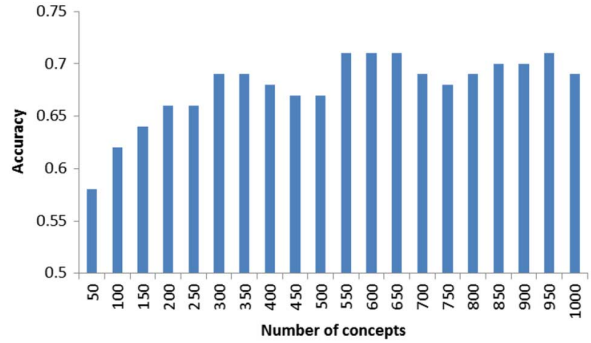

Fig. 6. Average accuracy results for the *CombineConcepts* method with different number of concepts.

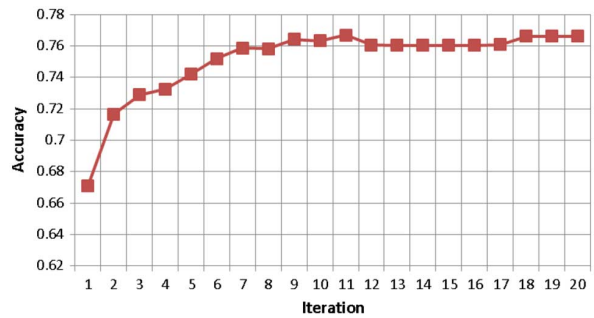

Fig. 7. Accuracy results of the *BoostConcepts* method with different iterations of the boosting framework.

Our boosted concept learning algorithm is based on the boosting framework as illustrated in Section IV. To verify its convergence, in Fig. 7, we show the recognition accuracies of the *BoostConcepts* method explained in Section V-C with different iterations. We can see that our algorithm converges quickly within about 10 iterations, more iterations will not improve the performance much.

E. Event Description by Visual Concept

As illustrated in Section IV, two distributions are adopted in the boosting process of our algorithm. One is the distribution for frames of the video set and the other one is the distribution for images of the image set. As shown in Section V-C, these distributions actually contribute much to the high performance of our algorithm for event recognition in videos. Besides, after convergence of the boosting process, we can obtain a weight vector which is an intermediate output according to the event classifier. Elements in the weight vector denote the importance of different concepts for recognizing event videos. For linear SVM classifier, this vector could be simply obtained by summing the transformation matrix according to the column of each concept. Here, we show another application of the weight vector of concepts. In Fig. 8, we show a visual concept description for social event "kate and william wedding" using the weight vector for all concepts. The edge reflects the weights of concepts. This concept based event description is a more effective visual summarization of the social event "kate and william wedding". We can see that concepts "horse guards parade", "clarence house", "queen elizabeth ii", "buckingham" and "kate williams" have big weights, which are consistent with our observation that these

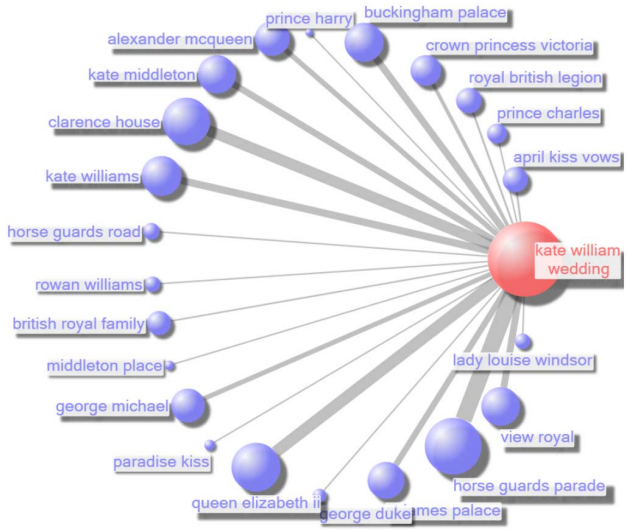

Fig. 8. Relationship among event and its concepts. The thick edges mean that the concepts and the event have strong relationship. We show visual concept description for social event "kate and william wedding" using the weight vector for all concepts.

concepts are the most discriminative semantic key phrases for the wedding event.

VI. Conclusion

In this paper, we have proposed an automatic visual concept learning method for social event understanding. To achieve this goal, we first do automatic concept mining. Then, we propose a boosted concept learning method to learn multiple classifiers for each visual concept to enhance its representation discriminability. The experimental results demonstrate the effectiveness of our proposed method. In the future, we will test our algorithm for other applications, such as image retrieval and domain adaptation.

References

[1] B.-K. Bao, W. Min, K. Lu, and C. Xu, "Social event detection with robust high-order co-clustering," in *Proc. 3rd ACM Int. Conf. Multimedia Retrieval*, 2013, pp. 135–142.

[2] M. Zaharieva, M. Zeppelzauer, and C. Breiteneder, "Automated social event detection in large photo collections," in *Proc. ACM Int. Conf. Multimedia Retrieval*, 2013, pp. 167–174.

[3] G. Petkos, S. Papadopoulos, and Y. Kompatsiaris, "Social event detection using multimodal clustering and integrating supervisory signals," in *Proc. 2nd ACM Int. Conf. Multimedia Retrieval*, New York, NY, USA, 2012, pp. 23:1–23:8.

[4] M. Brenner and E. Izquierdo, "Social event detection and retrieval in collaborative photo collections," in *Proc. 2nd ACM Int. Conf. Multimedia Retrieval*, 2012, pp. 21:1–21:8.

[5] Y. Wang, H. Sundaram, and L. Xie, "Social event detection with interaction graph modeling," in *Proc. ACM Int. Conf. Multimedia*, 2012, pp. 865–868.

[6] T. Reuter and P. Cimiano, "Event-based classification of social media streams," in *Proc. ACM Int. Conf. Multimedia Retrieval*, 2012, pp. 22:1–22:8.

[7] X. Liu and B. Huet, "Heterogeneous features and model selection for event-based media classification," in *Proc. 3rd ACM Conf. Int. Conf. Multimedia Retrieval*, 2013, pp. 151–158.

[8] S. Orlando, F. Pizzolon, and G. Tolomei, "Seed: A framework for extracting social events from press news," in *Proc. Int. World Wide Web Conf.*, 2013, pp. 1285–1294.

[9] V. Ramanathan, P. Liang, and F.-F. Li, "Video event understanding using natural language descriptions," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 905–912.

[10] J. Liu, Q. Yu, O. Javed, S. Ali, A. Tamrakar, A. Divakaran, H. Cheng, and H. S. Sawhney, "Video event recognition using concept attributes," in *Proc. IEEE Workshop Appl. Comput. Vis.*, Jan. 2013, pp. 339–346.

[11] Z. Ma, Y. Yang, Z. Xu, S. Yan, N. Sebe, and A. G. Hauptmann, "Complex event detection via multi-source video attributes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2013, pp. 2627–2633.

[12] Y. Yang, Z. Ma, Z. Xu, S. Yan, and A. G. Hauptmann, "How related exemplars help complex event detection in web videos?," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 2104–2111.

[13] Z. Ma, Y. Yang, Z. Xu, N. Sebe, and A. G. Hauptmann, "We are not equally negative: Fine-grained labeling for multimedia event detection," in *Proc. ACM Int. Conf. Multimedia*, 2013, pp. 293–302.

[14] Z. Ma, Y. Yang, Y. Cai, N. Sebe, and A. G. Hauptmann, "Knowledge adaptation for ad hoc multimedia event detection with few exemplars," in *Proc. ACM Int. Conf. Multimedia*, 2012, pp. 469–478.

[15] Y. Yang, Y. Yang, Z. Huang, J. Liu, and Z. Ma, "Robust cross-media transfer for visual event detection," in *Proc. ACM Int. Conf. Multimedia*, 2012, pp. 1045–1048.

[16] L. Jiang, A. G. Hauptmann, and G. Xiang, "Leveraging high-level and low-level features for multimedia event detection," in *Proc. ACM Int. Conf. Multimedia*, 2012, pp. 449–458.

[17] J. Luo, J. Yu, D. Joshi, and W. Hao, "Event recognition: Viewing the world with a third eye," in *Proc. 16th ACM Int. Conf. Multimedia*, 2008, pp. 1071–1080.

[18] N. Imran, J. Liu, J. Luo, and M. Shah, "Event recognition from photo collections via pagerank," in *Proc. ACM Int. Conf. Multimedia*, 2009, pp. 621–624.

[19] L. Duan, D. Xu, and S.-F. Chang, "Exploiting web images for event recognition in consumer videos: A multiple source domain adaptation approach," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2012, pp. 1338–1345.

[20] C. H. Lampert, H. Nickisch, and S. Harmeling, "Learning to detect unseen object classes by between-class attribute transfer," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2009, pp. 951–958.

[21] D. Parikh and K. Grauman, "Relative attributes," in *Proc. IEEE Int. Conf. Comput. Vis.*, Nov. 2011, pp. 503–510.

[22] A. G. Hauptmann, "Lessons for the future from a decade of informedia video analysis research," in *Proc. ACM Int. Conf. Image Video Retrieval*, 2005, pp. 1–10.

[23] Q. Yu, J. Liu, H. Cheng, A. Divakaran, and H. S. Sawhney, "Multimedia event recounting with concept based representation," in *Proc. ACM Conf. Multimedia*, 2012, pp. 1073–1076.

[24] D. Ding, F. Metze, S. Rawat, P. F. Schulam, S. Burger, E. Younessian, L. Bao, M. G. Christel, and A. Hauptmann, "Beyond audio and video retrieval: Towards multimedia summarization," in *Proc. 2nd ACM Int. Conf. Multimedia Retrieval*, 2012, pp. 2:1–2:8.

[25] M. Rohrbach, M. Stark, G. Szarvas, I. Gurevych, and B. Schiele, "What helps where — and why? Semantic relatedness for knowledge transfer," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2010, pp. 910–917.

[26] B. Gong, Y. Shi, F. Sha, and K. Grauman, "Geodesic flow kernel for unsupervised domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2012, pp. 2066–2073.

[27] B. Gong, K. Grauman, and F. Sha, "Connecting the dots with landmarks: Discriminatively learning domain-invariant features for unsupervised domain adaptation," in *Proc. Int. Conf. Mach. Learn.*, 2013, pp. 222–230.

[28] W. Lu, J. Li, T. Li, W. Guo, H. Zhang, and J. Guo, "Web multimedia object classification using cross-domain correlation knowledge," *IEEE Trans. Multimedia*, vol. 15, no. 8, pp. 1920–1929, Dec. 2013.

[29] S. D. Roy, T. Mei, W. Zeng, and S. Li, "Socialtransfer: Cross-domain transfer learning from social streams for media applications," in *Proc. ACM Int. Conf. Multimedia*, 2012, pp. 649–658.

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

[31] A. Sun and S. S. Bhowmick, "Quantifying tag representativeness of visual content of social images," in *Proc. ACM Int. Conf. Multimedia*, 2010, pp. 471–480.

[32] J. Revaud, M. Douze, C. Schmid, and H. Jegou, "Event retrieval in large video collections with circulant temporal encoding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2013, pp. 2459–2466.

[33] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2003, vol. 2, pp. 1470–1477.

[34] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," in *Proc. Eur. Conf. Comput. Vis., Workshop*, 2004, pp. 1–22.

[35] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2006, vol. 2, pp. 2169–2178.
[36] J. Zhu, H. Zou, S. Rosset, and T. Hastie, "Multi-class adaboost," *Statist. Interface*, vol. 2, no. 3, pp. 349–360, 2009.
[37] M. Chen, Z. E. Xu, K. Q. Weinberger, and F. Sha, "Marginalized denoising autoencoders for domain adaptation," in *Proc. ICML*, 2012, pp. 767–774.
[38] X. Glorot, A. Bordes, and Y. Bengio, "Domain adaptation for large-scale sentiment classification: A deep learning approach," in *Proc. Int. Conf. Mach. Learn.*, 2011, pp. 513–520.
[39] Y. Bengio, "Learning deep architectures for ai," *Found. Trends Mach. Learn.*, vol. 2, no. 1, pp. 1–127, 2009.
[40] X. Yang, Q. Song, and Y. Wang, "A weighted support vector machine for data classification," *Int. J. Pattern Recog. AI*, vol. 21, no. 5, pp. 961–976, 2007.
[41] L. Liu, L. Wang, and X. Liu, "In defense of soft-assignment coding," in *Proc. IEEE Int. Conf. Comput. Vis.*, Nov. 2011, pp. 2486–2493.
[42] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu, "Action recognition by dense trajectories," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2011, pp. 3169–3176.
[43] J. D. R. Farquhar, D. R. Hardoon, H. Meng, J. Shawe-Taylor, and S. Szedmák, "Two view learning: Svm-2k, theory and practice," in *Advances Neural Inf. Process. Syst. 18*, Vancouver, BC, Canada, Dec. 2005, pp. 355–362.
[44] A. Blum and T. M. Mitchell, "Combining labeled and unlabeled sata with co-training," in *Proc. 11th Annu. Conf. Comput. Learn. Theory*, Madison, WI, USA, Jul. 1998, pp. 92–100.
[45] W. Wang and Z. Zhou, "Co-training with insufficient views," in *Asian Conf. Mach. Learn.*, Canberra, Australia, Nov. 2013, pp. 467–482.

Xiaoshan Yang received the M.S. degree in computer science from the Beijing Institute of Technology, Beijing, China, in 2012, and is currently working toward the Ph.D. degree at the Multimedia Computing Group, National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China.

He was an Intern with the China-Singapore Institute of Digital Media, Singapore, from September 2013 to April 2014. His research interests include multimedia and computer vision.

Tianzhu Zhang (S'09–M'11) received the B.S. degree in communications and information technology from the Beijing Institute of Technology, Beijing, China, in 2006, and the Ph.D. degree in pattern recognition and intelligent systems from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2011.

He is currently an Associate Professor with the Institute of Automation, Chinese Academy of Sciences. His current research interests include computer vision and multimedia, especially action recognition, object classification, and object tracking.

Changsheng Xu (M'97–SM'99–F'14) is a Professor with the National Lab of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China, and Executive Director of China-Singapore Institute of Digital Media, Singapore. He holds 30 granted/pending patents and has authored or coauthored over 200 refereed research papers. His research interests include multimedia content analysis/indexing/retrieval, pattern recognition, and computer vision.

Dr. Xu is an Associate Editor for the IEEE TRANSACTIONS ON MULTIMEDIA, *ACM Transactions on Multimedia Computing, Communications and Applications*, and *ACM/Springer Multimedia Systems Journal*. He received the Best Associate Editor Award of *ACM Transactions on Multimedia Computing, Communications and Applications* in 2012 and the Best Editorial Member Award of *ACM/Springer Multimedia Systems Journal* in 2008. He has served as Associate Editor, Guest Editor, General Chair, Program Chair, Area/Track Chair, Special Session Organizer, Session Chair, and TPC Member for over 20 IEEE and ACM multimedia journals, conferences, and workshops. He served as Program Chair of ACM Multimedia 2009. He is an IAPR Fellow and ACM Distinguished Scientist.

M. Shamim Hossain (S'03–M'07–SM'09) received the Ph.D. degree in electrical and computer engineering from the University of Ottawa, Ottawa, ON, Canada.

He is an Associate Professor with King Saud University, Riyadh, Saudi Arabia. He has authored or co-authored more than 70 publications including refereed IEEE/ACM/Springer/Elsevier journals, conference papers, books, and book chapters. His research interests include serious games, cloud and multimedia for health care, resource provisioning for big data processing on media clouds, and biologically inspired approach for multimedia and software system.

Dr. Hossain has served as a member of the organizing and technical committees of several international conferences and workshops. He served as a Co-Chair of the 1st, 2nd, 3rd, 4th, and 5th IEEE ICME Workshop on Multimedia Services and Tools for E-Health. He served as a Co-Chair of the 1st Cloud-Based Multimedia Services and Tools for E-Health Workshop 2012 with ACM Multimedia. He currently serves as a Co-Chair of the 4th IEEE ICME Workshop on Multimedia Services and Tools for E-Health. He is on the editorial board of the *Springer International Journal of Multimedia Tools and Applications*. He was on the editorial board of some journals including the *International Journal of Advanced Media and Communication*. Previously, he served as a Guest Editor of the IEEE TRANSACTIONS ON INFORMATION TECHNOLOGY IN BIOMEDICINE and *Springer Multimedia Tools and Applications*. Currently, he serves as a Lead Guest Editor of the *Elsevier Future Generation Computer Systems*, *International Journal of Distributed Sensor Networks*, and *Springer Cluster Computing*. He is a member of ACM and ACM SIGMM.