Boosted Exemplar Learning for Action Recognition and Annotation

Tianzhu Zhang, Jing Liu, *Member, IEEE,* Si Liu, Changsheng Xu, *Senior Member, IEEE,* and Hanqing Lu, *Senior Member, IEEE*

***Abstract*—Human action recognition and annotation is an active research topic in computer vision. How to model various actions, varying with time resolution, visual appearance, and others, is a challenging task. In this paper, we propose a boosted exemplar learning (BEL) approach to model various actions in a weakly supervised manner, i.e., only action bag-level labels are provided but action instance level ones are not. The proposed BEL method can be summarized as three steps. First, for each action category, amount of class-specific candidate exemplars are learned through an optimization formulation considering their discrimination and co-occurrence. Second, each action bag is described as a set of similarities between its instances and candidate exemplars. Instead of simply using a heuristic distance measure, the similarities are decided by the exemplar-based classifiers through the multiple instance learning, in which a positive (or negative) video or image set is deemed as a positive (or negative) action bag and those frames similar to the given exemplar in Euclidean Space as action instances. Third, we formulate the selection of the most discriminative exemplars into a boosted feature selection framework and simultaneously obtain an action bag-based detector. Experimental results on two publicly available datasets: the KTH dataset and Weizmann dataset, demonstrate the validity and effectiveness of the proposed approach for action recognition. We also apply BEL to learn representations of actions by using images collected from the Web and use this knowledge to automatically annotate action in YouTube videos. Results are very impressive, which proves that the proposed algorithm is also practical in unconstraint environments.**

***Index Terms*—Action annotation, action recognition, AdaBoost, mi-SVM, multiple instance learning (MIL).**

I. Introduction

HUMAN MOTION analysis has attracted increasing interest from computer vision researchers [1], [2]. In particular, human action recognition has a wide range of promising applications, e.g., video surveillance, intelligent interface, and video retrieval. Generally, there are two important components in action recognition. One is how to extract useful information from raw video data, and the other is how to model different actions and measure their similarities for recognition. We focus on the latter one in this paper.

Recent papers [3]–[8] have shown that action recognition based on key poses from single video frame is feasible. This kind of method attempts to represent an action video with a set of representative frames called exemplars and then models various actions into a space defined by distances (or similarities) to these exemplars. However, varying from actors, environments, or cameras, videos of the same action may contain dissimilar frames as well as different lengths or time resolutions [as shown in the sequences (a) and (b) of Fig. 1]. Furthermore, videos from different actions may also include similar frames [as in sequences (a)–(d) of Fig. 1]. All these issues, but not limited to them, will increase the difficulties to recognize various actions from videos.

Inspired by the exemplar-based approach, we try to build a more generic system to recognize action in different kinds of environments. So far, most research in human action recognition has focused on videos taken in controlled environments. Standard datasets, like Weizmann [9] and KTH [10], supplied for this purpose are well-explored in various studies [5], [11]–[14]. However, real-world uncontrolled videos seldom exhibit such consistent and relatively simple settings. Therefore, action recognition in real-world videos is more difficult and challenging. To tackle this problem, the proposed methods require training with large amounts of videos and select many discriminative exemplars. It is quite challenging to find enough labeled video data covering a diverse set of poses. The Web may provide this information including many action images taken under various conditions and their roughly annotations, e.g., their surrounding text is a clue to the semantics of their content. We can make full use of such a collection of images to learn the exemplar-based action model. By doing this, we can benefit our action recognition model learning from the Web.

The images collected from the Web contain a huge range of variability (see Fig. 2), which include images of actions taken from multiple viewpoints in a range of environments, performed by people who have varying body proportions and different clothing. Thus, we believe that these images have covered various key poses in the action although some noises may exist. In many cases, the background clutter impedes good exemplar-based action recognition using existing algorithms. From these views, how to select the key-pose images as the suitable exemplars and how to learn a suitable distance

Manuscript received May 21, 2010; revised August 6, 2010 and November 12, 2010; accepted January 4, 2011. Date of publication April 15, 2011; date of current version July 7, 2011. This work was supported in part by the National Natural Science Foundation of China, under Grants 60835002, 90920303, and 60903146, and in part by the 973 Program Project, under Grant 2010CB327905. This paper was recommended by Associate Editor J. Luo.

The authors are with the National Laboratory 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: jliu@nlpr.ia.ac.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/TCSVT.2011.2133090

Fig. 1. Some examples of different actions from different subjects and different cameras. (a) Walk. (b) Walk. (c) Skip. (d) Run.

metric between the key-pose images are two important and challenging issues in the exemplar-based model.

Many research efforts have been conducted in the literature to select the exemplars. For instance, some methods proposed to sub-sample or cluster the space of exemplars [15], [16]. Such methods required nevertheless very large sets of exemplars. Moreover, the clustering might miss some important exemplars [4]. Daniel and Edmond [7] and Weinland *et al.* [4] selected some discriminative exemplars with forward selection, which was particularly robust against over-fitting. However, the forward selection algorithm was slow calculation because of the repetitive learning and evaluation cycles.

For the second issue, usually, heuristic distance metrics or specified matching approaches, such as squared Euclidean distance [7] or hidden Markov model (HMM)-based matching [4], were proposed to measure the relation among exemplars. However, these approaches ignored the distribution of frames in the feature space and might fail to achieve the best discriminativity for action recognition. As shown in Fig. 3, given a frame m, not all its similar frames evaluated by a heuristic distance, e.g., Euclidean distance, belong to the same action to the frame m. Therefore, it is necessary for every frame to learn its similar frames across all samples from the same action, instead of using a predefined and heuristic distance.

In this paper, we propose a boosted exemplar learning (BEL) approach (as shown in Fig. 4) to recognize various actions. First, amount of class-specific candidate exemplars are learned through an optimization formulation considering their discrimination and co-occurrence to each action category. Second, for each candidate exemplar, we employ the multiple instance learning (MIL) to learn the exemplar-based classifier to measure the similarity. For the MIL problem, each action, such as in a video clip, is considered as an action bag, and components of action, such as frames of the video clip, are viewed as action instances. As a result, if we obtain M candidate exemplars, each action bag is described as a M-dimensional vector of similarities between the M exemplars and the action bag. Third, we apply AdaBoost algorithm to integrate the further selection of representative exemplars and action modeling together. That is, through the boosting learning, the most discriminative exemplars are selected, and simultaneously the similarities based on the selected exemplars as the weak classifier are combined to obtain an action bag-based detector. Experimental results on publicly available challenging datasets demonstrate the validity and effectiveness of the proposed approach.

In our previous paper [17], preliminary results of video action recognition on two public datasets were reported. Compared with [17], a number of improvements have been made in this paper. First, in [17], the candidate exemplars were all randomly selected from the datasets, which was indiscriminative and less efficient. In this paper, we develop a novel method to select semantic exemplars. The proposed method takes the discrimination and co-occurrence of exemplars into consideration, hence, it is efficient to obtain semantic candidate exemplars and reduce the computational cost, especially, selecting exemplars from the Web. Second, we broaden our application to action annotation in videos of uncontrolled environments, like YouTube videos. This application shows that our algorithm is also suitable for real-world videos except for videos taken in controlled environments working with limited action vocabularies.

The paper is organized as follows. In Section II, we review related work. The detailed implementation of the proposed BEL method is introduced in Section III. In Section IV, we evaluate our approach on three publicly available datasets. We conclude the paper with future work in Section V.

II. Related Work

Exemplar-based embedding methods have already been proposed in computer vision field [15], [18]. Athitsos and Sclaroff [15] presented an approach for hand pose estimation based on Lipschitz embeddings. Guo *et al.* [18] used an exemplar-base embedding approach to match images of cars over different viewpoints. In these approaches, complex distances between signals were approximated in a Euclidean embedding space that was spanned by a set of distances to exemplar measures.

Recently, some attempts have been made to apply such exemplar-based approaches to action recognition. Wang *et al.* [19] utilized deformable template matching for computing the distance between human poses, so that similar poses could be grouped together. Thurau and Hlavac [6] approached the problem by using non-negative matrix factorization on pose primitives. In the work of Carlsson and Sullivan [3], class representative silhouettes were matched against video frames to recognize forehand and backhand strokes in tennis recordings. Dedeoglu *et al.* [20] proposed a real-time system for action recognition based on key-poses and histograms. Daniel and Edmond [7] adopted the exemplar-based approach to transform length-variant orderless feature set of action videos into matching distances to exemplars, and then a classifier was trained based on this fixed length representation. Essentially, the pose primitives were learned from non-cluttered videos and applied to images to find the closest pose. In this paper, we try to select representative exemplars from the Web, fit an action model, and use this to annotate actions in the cluttered videos.

To select the discriminative exemplars, Daniel *et al.* [7] and Weinland *et al.* [4] used the forward selection algorithm. Other exemplar-based approaches [4], [16], [21], [22] attempted to learn HMMs with observation probabilities based on matching distances to exemplars. However, the similarities between

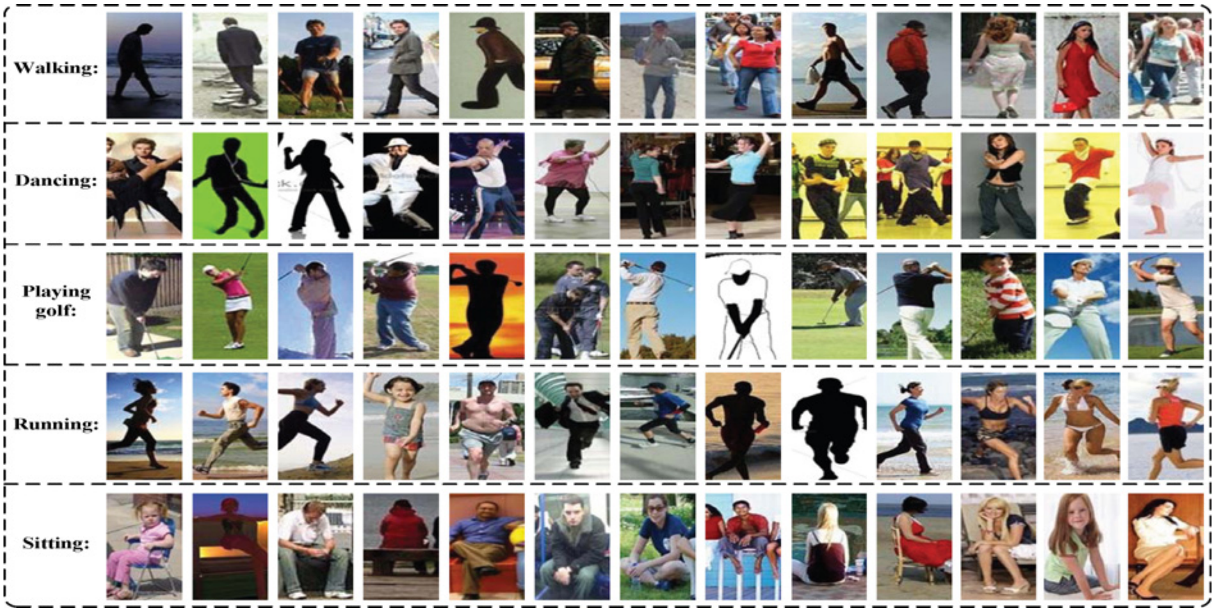


Fig. 2. Some examples of collected images from the Web in [8]. These show the output of the person detector. The rows correspond to actions (a) walking, (b) dancing, (c) playing golf, (d) running, and (e) sitting, respectively.

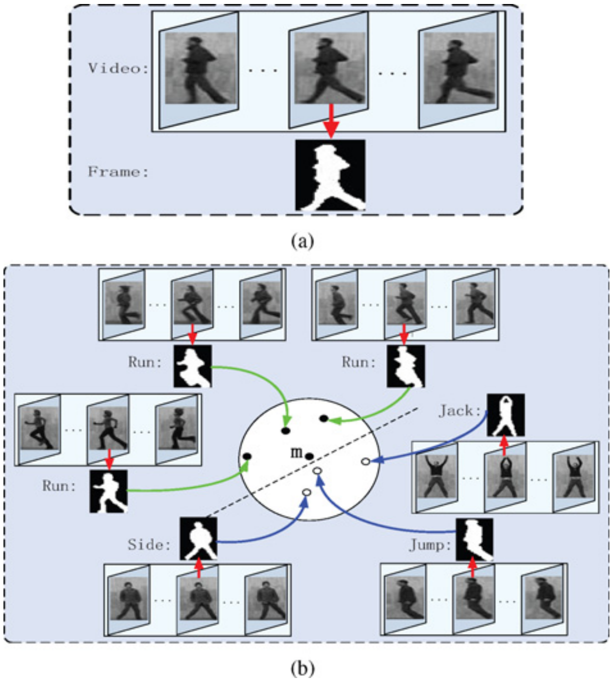


Fig. 3. Learning similar frames to frame m. (a) Frame m, which is one frame of run action video. (b) Learning a classifier to describe the similarity of other frames to m. " $\bullet$" represents similar frames from the same kind of action with m, and " $\circ$" represents unrelated frames.

frames and exemplars were measured using heuristic distance. This might be not correct (as shown in Fig. 3). Therefore, it is necessary to adopt an efficient approach to select the discriminative exemplars. Instead of using heuristic distance, we learn the similarity metric by MIL.

There is little work in literature dealing with generic videos like YouTube videos, where the resolution is low and the recording environment is nonuniform. A lot of existing work [23], [24] recognized human action in such videos. Hu *et al.* [25] proposed a novel multiple-instance learning framework, named simulated annealing multiple instance learning support vector machines (SVMs), to learn human action detector based on imprecise action locations on real-world video database. Tran *et al.* [14] detected actions in YouTube Badminton videos with fairly static backgrounds. Our method is applicable to videos with a broader range of settings. Ikizler *et al.* [8] used images collected from the Web to learn representations of actions and used this knowledge to automatically annotate actions in videos. Different from the previous methods, we adopt MIL to describe different kinds of actions from complexity data sources and present a BEL method to learn the similarity metric and select some representative exemplars from the Web. Experimental results show that our method is much better than [8].

III. BEL FOR ACTION MODELING

In this section, we explain how to select a set of discriminative exemplars for action modeling. There are four challenges for our discriminative exemplar-based action model: 1) how to describe each action instance; 2) how to select candidate exemplars for large-scale dataset to reduce computational cost; 3) how to describe action bag based on the candidate exemplars when only action bag label is given; and 4) how to explore action bags into an overall classifier for the action of interest given descriptions for action bags. In this paper, we present a unified and effective solution to these challenges.

Our approach proceeds as illustrated in Fig. 4. Each action instance is described using simple features, such as histogram of silhouette, histogram of edge, or histogram of oriented gradients. Based on this description, for each kind of action, some candidate exemplars are selected and their corresponding classifiers are trained via MIL. Based on the classifier of each

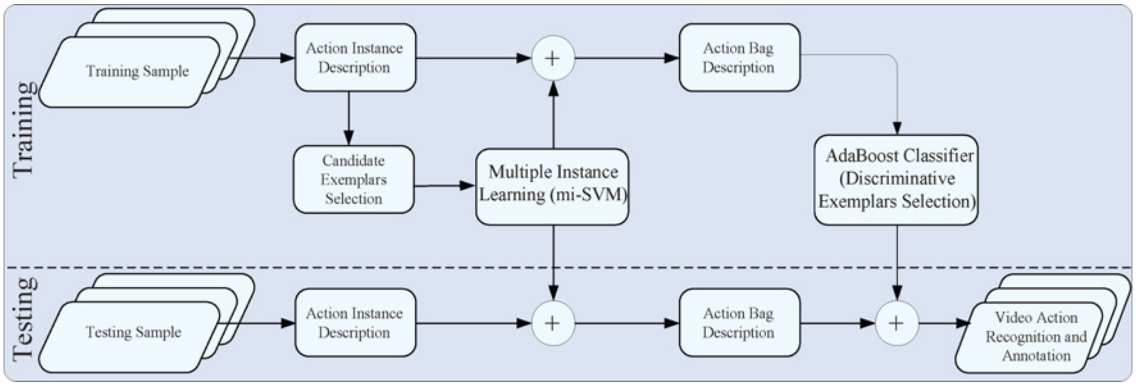

Training
Training Sample
Action Instance Description
Candidate Exemplars Selection
Multiple Instance Learning (mi-SVM)
Action Bag Description
AdaBoost Classifier (Discriminative Exemplars Selection)
Testing
Testing Sample
Action Instance Description
Action Bag Description
Video Action Recognition and Annotation

Fig. 4. Framework of our approach.

exemplar, similarities between the exemplar and instances in an action bag can be obtained. Then, the action bag can be described using the similarities as its features. Considering the large intra-class variation of different actions, AdaBoost is employed to select the most discriminative features to form a strong classifier.

First, we introduce some concepts and notation. An action bag can be a video clip or a set of sampled patches from a single image, and an action instance is a frame of a video clip or a sampled patch of a single image. Denote v_i as the i action bag and $I_{v_i,j}$ is the feature of the jth instance of bag v_i. Based on this description, for an action bag v_i, it can be represented as a set of histogram features. The formal definition of action v_i is as follows: $v_i = \{I_{v_i,j} | j = 1, 2, \ldots, n_i\}$, where n_i is the number of instances from action bag v_i. Let v_i^+ denote a positive action bag and v_i^- denote a negative action bag. v_{ij}^+ is the jth instance of a positive action bag v_i^+ and v_{ij}^- denotes the jth instance of a negative action bag v_i^-. Let $\{v_1^+, v_2^+, \ldots, v_s^+, v_1^-, v_2^-, \ldots, v_t^-\}$ denote the set of s positive and t negative training action bags. $l(v_i) \in \{+1, -1\}$ is the bag label of v_i and $l(v_{ij}) \in \{+1, -1\}$ is the instance label of v_{ij}. For the negative action bags, their all instances are negative. However, for the positive action bags, their all instances must contain at least one true positive instance, and they may also contain many negative instances.

Next, we begin by introducing how to select candidate exemplars in Section III-A, then we present the MIL for action bag description in Section III-B, and the AdaBoost-based action classifier in Section III-C.

A. Candidate Exemplars Selection

If we take each action instance as an exemplar, it leads to a large number of candidate exemplars. Therefore, the computational cost is very high to train each exemplar-based classifier. One possible remedy is to select a representative set of instances (candidate exemplars). The candidate exemplars selection has two phases. In the first phase, we use k-means to create an initial vocabulary by grouping similar action instances based on their features for each action category, and select instances nearest to each cluster as initial exemplar set. For an exemplar m, I_m is denoted as its feature vector. The initial exemplar set has two drawbacks. First, the performance is sensitive to the size of the exemplar set. Generally, larger exemplar set size performs better since the most discriminative instances are contained. Second, the instances in the exemplar set are not necessarily semantically meaningful, because k-means only considers the similarity in the feature space. In the second phase, compact yet discriminative candidate exemplars are required to obtain from the initial exemplar set for the sake of efficiency and effectiveness. Next, we introduce how to select meaningful exemplars from the initial set as the candidate exemplars.

1) *Problem Formulation:* The goal of the selection of candidate exemplars related to an action class is attempting to select the most informative exemplars to represent the corresponding action class. Therefore, two criteria are desired for selecting the exemplars: 1) the exemplars in the class-specific candidate exemplars should have much discrimination to classify action bags labeled with the given class and action bags without the class, and 2) they should appear more simultaneously in action bags labeled with the given class than in action bags without the class. In the following, we propose an optimization scheme to effectively incorporate these two criteria in the process of selection of class-specific candidate exemplars from an initial exemplar set obtained by k-means with the given class.

Assume we have a collection of action bags $V = \{V_1, \ldots, V_c, \ldots, V_C\}$, in which the action bags annotated with a given class c form a subset $V_c = \{v_1, v_2, \ldots, v_{N_c}\}$ and N_c is the number of the action bags in V_c. The number of action bag is $N = N_1 + \ldots + N_c + \ldots + N_C$, and C is the number of action class. We suppose there is an initial exemplar set $E_c = \{e_1, e_2, \ldots, e_{M_c}\}$ for class c, where M_c denotes the size of the exemplar set. We define $f_m \in [0, 1]$ as the probabilistic score to measure the possibility of an exemplar e_m being descriptive to the action corresponding to class c, and the scores for the all exemplars in E_c can be represented as a vector $f = [f_1, f_2, \ldots, f_m, \ldots f_{M_c}]^T$.

Based on the first criterion, the discriminative information of each exemplar is important for identifying its descriptiveness. Besides, co-occurrence information between a pair of exemplars is another important clue, since co-occurrence exemplars in a given action class c are more likely to appear simultaneously on the action bag corresponding to the class c. Based on these two clues, the task of selection of class-specific candidate exemplars from a large-scale instance is formulated as

$$\begin{aligned} E(f) &= \sum_{m=1}^{M_c} (f_m - p_m)^2 + \lambda \sum_{mn} w_{mn}(f_m - f_n)^2 \\ s.t. 0 &\leqslant f_m \leqslant 1, m = 1, \ldots, M_c \end{aligned} \quad (1)$$

where the value of p_m measures the discrimination of exemplar e_m and can be adopted as a prior probability to estimate the descriptive capability of exemplar e_m. How to obtain the value of p_m is introduced in Section III-A2. w_{mn} denotes the co-occurrence frequency between exemplars e_m and e_n which is defined as $w_{mn} = Co(e_m, e_n)/N_c$, where $Co(e_m, e_n)$ denotes the number of action bags that simultaneously contain exemplars e_m and e_n. λ is a parameter that controls the tradeoff among these two terms and is manually set to be 0.8 by experimental validation. The first term in the objective function measures the data fitting capability, namely, the deviation between the estimated probabilistic confidence score and the prior probability. The second term measures the smoothness of confidence score for different exemplars, i.e., two exemplars with high co-occurrence should also have similar confidence scores of being selected to be descriptive.

2) *Discrimination p_m:* If the exemplar e_m represented by I_m is a positive exemplar, it will be discriminative. Therefore, the discrimination of p_m can be calculated by its classification capability.

We assume that given a positive exemplar m, the probability that an instance v_{ij} is positive is calculated as follows:

$$\Pr(l(v_{ij}) = +1|m) = \exp(-\frac{\left\|I_m - I_{v_i,j}\right\|^2}{\delta_m^2}) \quad (2)$$

where $\|\bullet\|$ represents L2-norm, and δ_m is a parameter learned from the training data. The probability that an action bag v_i is a positive action bag is defined as follows:

$$\begin{aligned} \Pr(l(v_i) = +1|m) &= \max_{v_{ij} \in v_i} \Pr(l(v_{ij}) = +1|m) \\ &= \max_{v_{ij} \in v_i} \exp(-\frac{\left\|I_m - I_{v_i,j}\right\|^2}{\delta_m^2}) = \exp(-\frac{d^2(m, v_i)}{\delta_m^2}) \end{aligned} \quad (3)$$

where $d(m, v_i) = \min_{v_{ij} \in v_i} \left\|I_m - I_{v_i,j}\right\|$. In other words, the distance $d(m, v_i)$ between an exemplar m and all instances of an action bag v_i is simply equal to the distance between m and the nearest instance of v_i. Then $\Pr(l(v_i) = +1|m) - \Pr(l(v_i) = -1|m) = 2\exp(-\frac{d^2(m,v_i)}{\delta_m^2}) - 1$. If $\Pr(l(v_i) = +1|m) \geqslant \Pr(l(v_i) = -1|m)$, we get $d(m, v_i) \leqslant \delta_m\sqrt{\ln 2}$. For a negative instance (i.e., false positive instance), however, its distances to the positive and negative action bags do not exhibit the same distribution as those from m. Since some positive action bags may also contain negative instances just like the negative action bags, the distances from the negative instance to the positive action bags may be as random as those to the negative action bags. This distributional difference provides an informative hint for identifying the positive exemplars (true positive instances). Therefore, given a positive exemplar m, there exists a threshold θ_m which makes the decision function defined in (4) labels the action bags according to the Bayes decision rule as follows:

$$\eta_{\theta_m}^m(v_i) = \begin{cases} +1, & \text{if } d(m, v_i) \leqslant \theta_m \\ -1, & \text{otherwise} \end{cases} \quad (4)$$

where $\theta_m = \delta_m\sqrt{\ln 2}$ determined by training data as follows:

$$p_m = \max_{\theta_m} P_m(\theta_m) \quad (5)$$

where $P_m(\theta_m)$ is an empirical precision and defined as follows:

$$P_m(\theta_m) = \frac{1}{s+t}\sum_{i=1}^{s+t} \frac{1 + \eta_{\theta_m}^m(v_i)l(v_i)}{2}. \quad (6)$$

In this way, for each exemplar, we can obtain its discrimination p_m as shown in (5).

3) *Optimization Solution:* Since the objective function in (1) with respect to f is a formulation of quadratic programs [26], which can be solved efficiently with global optimum using existing convex optimization packages, such as Mosek [27]. Based on the above solution, the class-specific candidate exemplars for the given class c is constructed by selecting the top $M_c^{'}$ exemplars with the highest confidence scores. In this way, we can obtain candidate exemplars for each kind of action, and the total number of exemplars is $M = M_1^{'}+, \ldots, +M_c^{'}+, \ldots, +M_C^{'}$.

B. MIL for Action Bag Description

In this section, we introduce how to describe each action bag as a set of similarities between its instances and candidate exemplars, which are selected for each class c in Section III-A.

We assume that $M_c^{'}$ exemplars from positive action bags are obtained $\{I_m|m = 1, \ldots, M_c^{'}\}$, where I_m represents the feature for the mth exemplar. For exemplar m in the action bag of category c, it is possible that some instances from the same action to m are less similar than the ones from other actions when a uniform distance metric is adopted (as shown in Fig. 3). To tackle this problem, we propose a discriminative solution to get semantic similarity by learning exemplar-based classifiers. Here, we formulate the similarity measure learning as a problem of MIL [28] and mi-SVM [29] is employed to solve the problem.

We introduce how to train an exemplar-based classifier for the action category c. This process can be repeated for training all exemplar-based classifiers for different kinds of actions. For an exemplar m ($m = 1, \ldots, M_c^{'}$) from the action of category c, a corresponding mi-SVM classifier is trained and denoted by $mi - SVM_m$. The training samples are the bags in c denoted as positive bags and those in other categories denoted as negative ones. Some action bags may contain a large number of instances. If we use all the instances to train the mi-SVM classifier, its computational and storage requirements may become too large. To reduce the computational burden and learn efficient classifier, we adopt an efficient strategy to obtain action bags by filtering out the instances which are very different from m for each action bag v_i. This strategy enables the classifier to be learned only in the local feature space. Specifically, mi-SVM classifier is trained in the hypersphere centered at I_m with radius of r_m in the feature space (as shown in Figs. 6 and 14). Define the distance from exemplar

m to action bag v_i as $d_{v_i,j,m} = \min_j \left\|I_m - I_{v_i,j}\right\|$, where $\|\bullet\|$ represents L2-norm. In practice, it is found quite robust and in majority of cases the positive instance in the positive action bag v_i is just

$$\left\{ I_{v_i,j^*} | j^* = \arg\min_j \left\|I_{v_i,j} - I_m\right\| \right\}. \tag{7}$$

Based on this observation, r_m is set as follows:

$$r_m = \underset{v_i \in pos}{mean}(d_{v_i,j,m}) + \beta \times \underset{v_i \in pos}{std}(d_{v_i,j,m}) \tag{8}$$

where β is a tradeoff between efficiency and accuracy. The larger β, the less probability that one positive instance will be filtered out before training, and the more instances will be involved during solving the mi-SVM.

It is observed in experiments that there may be too many instances falling into the hyper-sphere. To be more efficient and make classifiers different, within the hyper-sphere, at most k_p nearest instances to I_m are selected for each positive action bag, and k_n for each negative action bag. Figs. 6 and 14 show two examples for each exemplar and its corresponding action bag of v_i. Experiments demonstrate that this strategy can greatly reduce the computational burden and has no significant impact on the final results.

Denote $y_{v_i,j}$ to be the instance label of $I_{v_i,j}$ and Y_{v_i} the label of action bag v_i, where $I_{v_i,j}$ is the feature of the instance j in the action bag v_i. mi-SVM is formulated as follows:

$$\begin{aligned}
&\min_{\{y_{v_i,j}\}} \min_{w,b,\xi} \frac{1}{2}\|w\|^2 + C\sum_{v_i,j} \xi_{v_i,j} \\
&s.t. \sum_j \frac{y_{v_i,j}+1}{2} \geqslant 1, \forall v_i \; s.t. \; Y_{v_i} = 1 \\
&\mathrm{y}_{v_i,j} = -1, \forall v_i \; s.t. \; Y_{v_i} = -1 \\
&\forall j : \mathrm{y}_{v_i,j}(\langle w, I_{v_i,j}\rangle + b) \geqslant 1 - \xi_{v_i,j}, \xi_{v_i,j} \geqslant 0 \\
&\mathrm{y}_{v_i,j} \in \{-1, 1\}
\end{aligned} \tag{9}$$

where $\xi_{v_i,j}$ is slack variable.

For the mth classifier $mi - SVM_m$, an action bag v_i can be projected to real value with a function. For simplicity, the projection function is defined as follows:

$$g_m(v_i) = \begin{cases} \max_j mi\_SVM_m(I_{v_i,j}) \; \exists I_{v_i,j}, s.t. \left\|I_{v_i,j} - I_m\right\| \leqslant r_m \\ -1, \quad \text{otherwise} \end{cases} \tag{10}$$

where $mi\_SVM_m(I_{v_i,j}) \in \mathbb{R}$ is the output of $mi - SVM_m$ with the input $I_{v_i,j}$.

In this way, we can obtain $g_m(v_i)$ as the similarity for action bag v_i with the learned classifier of exemplar m. If we have selected $M = M_1^{'}+, \ldots, +M_c^{'}+, \ldots, +M_C^{'}$ exemplars for all kinds of action in training dataset, we can get M classifiers trained using mi-SVM, respectively. Based on the M classifiers, each action bag v_i can be measured using M-dimensional features $(g_1(v_i), \ldots, g_m(v_i), \ldots, g_M(v_i))^T$.

C. AdaBoost Classifier

To learn a diverse collection of features we turn to boosting [30]. In boosting, multiple weak learners, each of which may have fairly high error, are combined into a single strong classifier with a low overall error. Weak classifiers are trained sequentially with the weights of the training samples adjusted so that incorrectly classified examples receive more weight. Boosting is ideally suited for combining diverse classifiers into an overall classifier.

Algorithm 1 Proposed BEL algorithm

```
Given: $N$-labeled training examples $(v_i, y_i)$ with $y_i \in \{-1, 1\}$ and $v_i = \{I_{v_i,1}, \ldots, I_{v_i,n}\}$, and initial distribution of weights $w_i = \frac{1}{N}, i = 1, \ldots, N$.
Select $M$ candidate exemplars for all kinds of action using the proposed method in Section III-A and train their corresponding classifiers using mi-SVM to obtain $\{g_m(v_i) | i = 1, \ldots, N, m = 1, \ldots, M\}$ for action bag description. The $g_m(v_i)$ can be viewed as the $m$th feature of action bag $v_i$.
for $t = 1, \ldots, T$ : do
    Train: Find $M$ hypotheses $h_m$, by training the base learner on each feature $g_m$ of the given training set, using current weighting $w_i$.
    Calculate: The weighted training error for each hypothesis $h_m$
    $\varepsilon_m = \sum_{i=1}^{N} w_{t,i} 1(y_i \neq h_m(g_m(v_i)))$.
    Select: Hypothesis $h_m$ with the lowest $\varepsilon_m$, set $h_t = h_m$ and $\varepsilon_t = \varepsilon_m$.
    Calculate: Hypothesis coefficient $\alpha_t = \frac{1}{2}\log(\frac{1-\varepsilon_t}{\varepsilon_t})$.
    Update: Sample weights $w_{t+1,i} = \frac{1}{Z_t} w_{t,i} \exp(-\alpha_t y_i h_t(v_i))$, where $Z_t$ is a normalization coefficient such that $\sum_i (w_{t+1,i}) = 1$.
end for
Output: The BEL classifier $H(v) = sign(\sum_{t=1}^{T} \alpha_t h_t(v))$.
```

The discrete version of AdaBoost [30] defines a strong binary classifier H as follows:

$$H(v_i) = \mathrm{sgn}(\sum_{t=1}^{T} \alpha_t h_t(g_m(v_i))) \tag{11}$$

using a weighted combination of T weak learners h_t with weights α_t. Each weak learner

$$h_t(g_m(v_i)) = \begin{cases} 1, & \text{if } g_m(v_i) > threshold \\ -1, & \text{otherwise} \end{cases} \tag{12}$$

may explore any feature $g_m(v_i)$ of the action bag v_i.

Based on the features of action bags, the optimal threshold in (12) is determined and weak learners are trained and combined to get a strong classifier for action recognition, In fact, because each feature corresponds to an exemplar, the discriminative exemplars from the candidate frames are selected during the AdaBoost learning process. For details, please see the proposed BEL algorithm 1.

IV. Experimental Results

Video action recognition and video action annotation are adopted to validate the efficiency and effectiveness of our

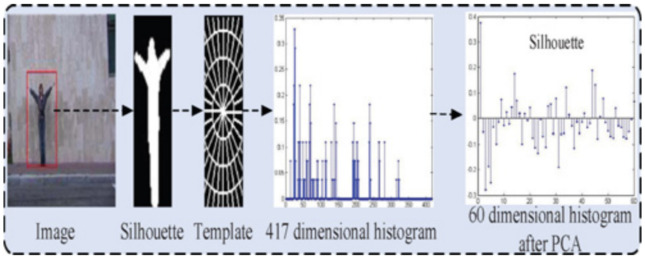


Fig. 5. Feature extraction.

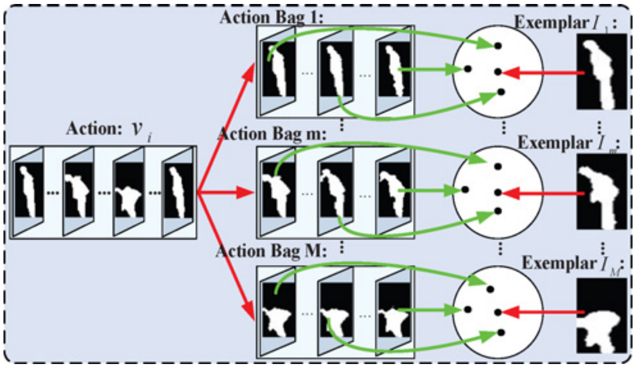


Fig. 6. Action v_i and its corresponding action bag for each exemplar. Note each action has M action bags corresponding to M exemplars.

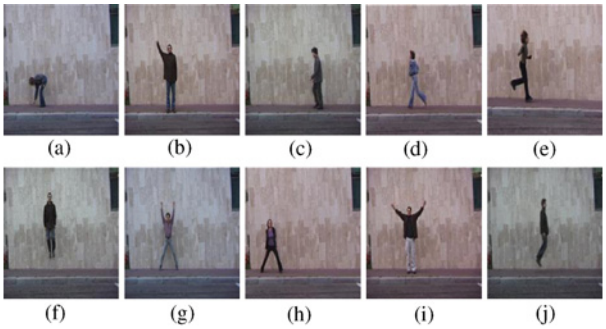


Fig. 7. Example actions from the Weizman dataset. (a) Bend. (b) Wave1. (c) Walk. (d) Run. (e) Skip. (f) Pjump. (g) Jack. (h) Side. (i) Wave2. (j) Jump.

proposed method. All of our experiments are conducted on a server with four Quad-Core Intel Xeon E7320 (2.13 GHz) processors and 16 GB memory. In all experiments, β in (8) is empirically set to be 2.5. For the multi-class classification problem, we deal with it as a series of two-class problems, for which one-against-all strategy is adopted.

A. Video Action Recognition

For video action recognition, we have tested our algorithm on two standard datasets: Weizmann human action dataset [9] and KTH human motion dataset [10]. Weizmann and KTH provide a few action classes recorded in controlled and simplified settings. We use simple features to describe each frame, such as histogram of silhouette on Weizmann dataset and histogram of edge on KTH dataset. Next, we introduce how to describe action instance and construct action bag in Section IV-A1. Then, experimental results on Weizmann and KTH datasets are reported in Sections IV-A2 and IV-A3, respectively.

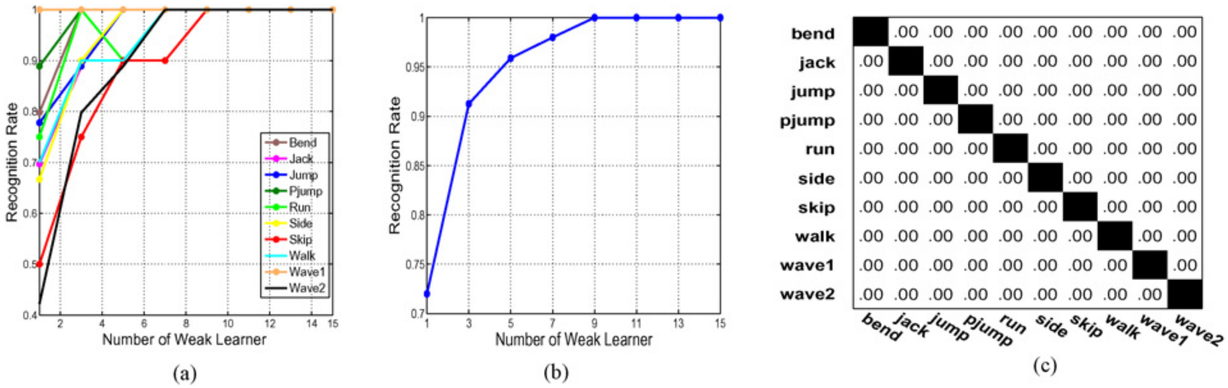
1) *Action Instance Description and Bag Construction:* The input to our action recognition algorithm is a stabilized sequence of cropped frames, which are centered on the human figure. For each cropped frame, a template image as in [31] with similar size is adopted to describe the silhouette or edge. Fig. 5 shows an example of the template image, which is divided into many pie slices covering some degrees each. The maximum distance between the pixels and the center in each cropped frame is quantized into R_{bin} bins, which makes the description insensitive to the scale variance. For the rth bin, each pie slice covers $\frac{\theta}{r}$ degrees without overlaps. In our experiments, θ is 30°, and R_{bin} is 8. Then $\sum_{r=1}^{R_{bin}} \frac{360\times r}{\theta}$ bins are used to generate a histogram-based descriptor. The value of each bin is integrated over the domain of every slice. Then we obtain a 417-D histogram as the descriptor, as shown in Fig. 5. To obtain compact description and efficient computation, the dimension of the feature is further reduced using principal component analysis (PCA).

For a candidate exemplar m and a video, at most k_p nearest frames to I_m are selected for each positive action bag, and k_n for each negative action bag. Fig. 6 shows an example for each exemplar and its corresponding action bag of video v_i.

2) *Results on Weizmann Dataset:* The Weizmann dataset [9] (see Fig. 7) contains ten actions: bend (bend), jumping-jack (jack), jump-in-place (pjump), jump-forward (jump), run (run), gallop-sideways (side), jump-forward-one-leg (skip), walk (walk), wave one hand (wave1), wave two hands (wave2), performed by nine actors. In these experiments, the background-subtracted silhouettes provided by the Weizmann dataset are used. All recognition rates are computed with the leave-one-out cross-validation. Details are as follows. Eight out of the nine actors in the database are used to select the discriminative exemplars and train the AdaBoost classifier, the ninth is used for the evaluation. This is repeated for all nine actors and the rates are averaged. The discriminative exemplars are constantly selected from all eight actors, but never from the ninth that is used for the evaluation.

Considering the temporal correlation, we first uniformly subsample the sequences by a factor 1/2 and use k-means to cluster the remaining frames. The cluster number is set to be 200 for each class, and the proposed method in Section III-A is used to select 20 candidate exemplars for each class. To train each exemplar-based classifier, $k_p = 10$ and $k_n = 20$ are adopted to obtain training samples. It takes about 98 ms to train an mi-SVM for one exemplar using a single core. Finally, the most discriminative exemplars are selected by AdaBoost to form a strong classifier.

Experimental results show that our approach can reach recognition rates up to 100% with approximately nine discriminative exemplars. In Fig. 7, we have shown some sample frames of the Weizmann dataset. In Fig. 8(a), we show recognition rates for the individual classes. The average recognition rate on the test set and with respect to the number of weak learners is shown in Fig. 8(b). The confusion matrix of our result is shown in Fig. 8(c). In comparison, the space-time volume approach proposed by Blank *et al.* [9] had a recognition



	bend	jack	jump	pjump	run	side	skip	walk	wave1	wave2
bend		.00	.00	.00	.00	.00	.00	.00	.00	.00
jack	.00		.00	.00	.00	.00	.00	.00	.00	.00
jump	.00	.00		.00	.00	.00	.00	.00	.00	.00
pjump	.00	.00	.00		.00	.00	.00	.00	.00	.00
run	.00	.00	.00	.00		.00	.00	.00	.00	.00
side	.00	.00	.00	.00	.00		.00	.00	.00	.00
skip	.00	.00	.00	.00	.00	.00		.00	.00	.00
walk	.00	.00	.00	.00	.00	.00	.00		.00	.00
wave1	.00	.00	.00	.00	.00	.00	.00	.00		.00
wave2	.00	.00	.00	.00	.00	.00	.00	.00	.00	

Fig. 8. Experimental results on Weizman dataset. (a) Recognition rates per action versus number of weak learner. (b) Average recognition rates versus number of weak learner. (c) Confusion matrix for action recognition.

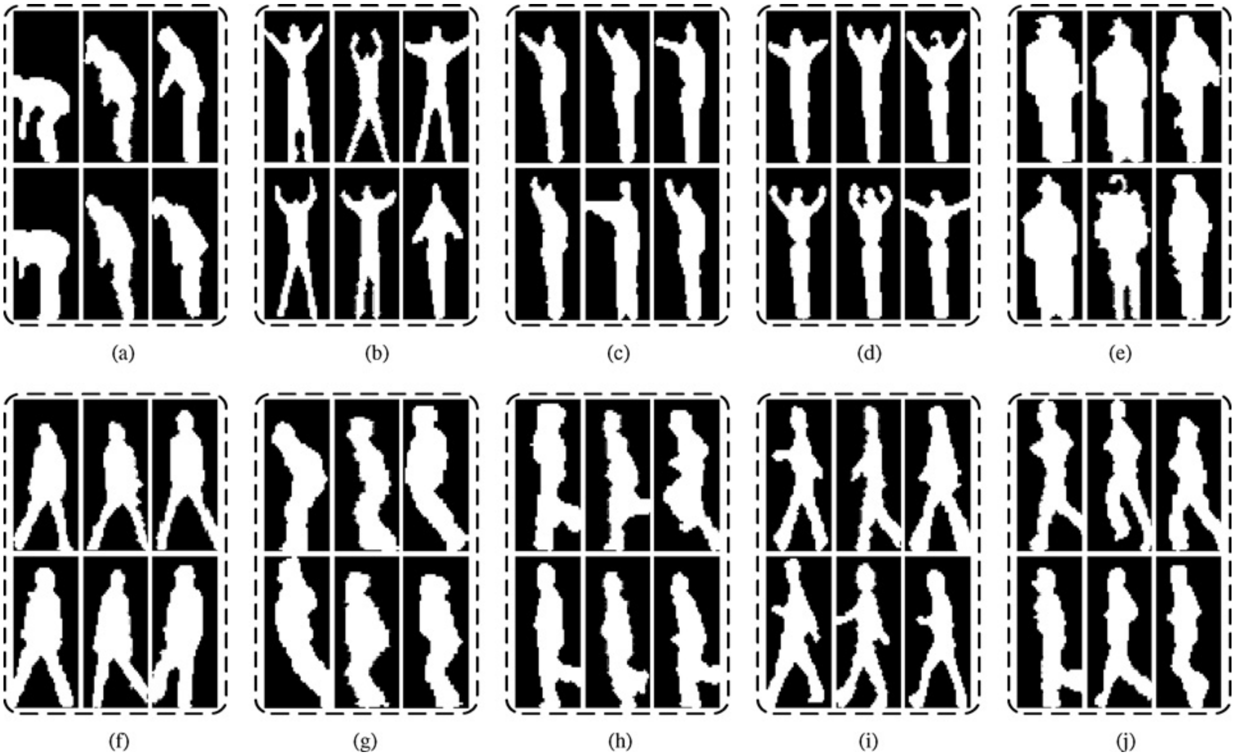


Fig. 9. Discriminative exemplars are selected for different actions by the final AdaBoost classifiers for the Weizmann dataset. (a) Bend. (b) Jack. (c) Wave1. (d) Wave2. (e) Pjump. (f) Side. (g) Jump. (h) Skip. (i) Walk. (j) Run.

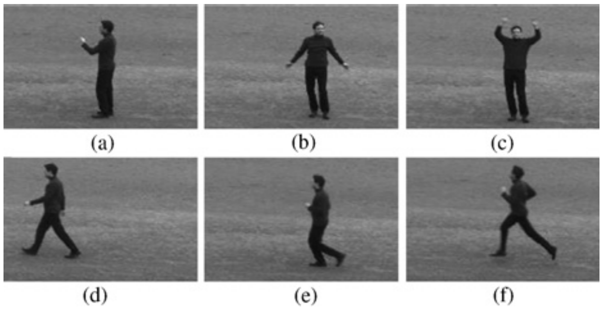
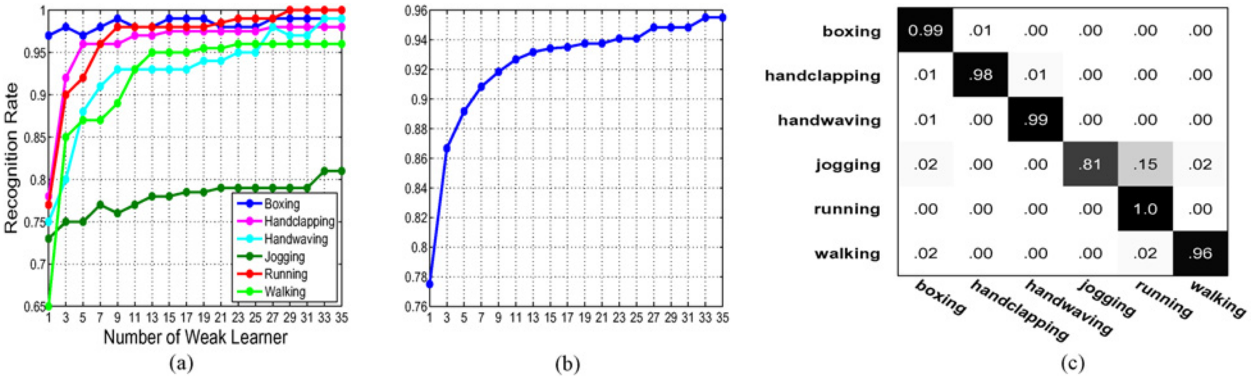


Fig. 10. Example actions from the KTH data set. (a) Boxing. (b) Hand clapping. (c) Hand waving. (d) Walking. (e) Jogging. (f) Running.

rate of 99.61%. Wang and Suter [32] reported a recognition rate of 97.78% with an approach that used kernel-PCA for dimensional reduction and factorial conditional random fields to model motion dynamics. The work of Ali *et al.* [33] used a motion representation based on chaotic invariants and reports 92.6%. Daniel and Edmond [7] reported a recognition rate of 100% with approximately 120 exemplars, which were more than the number of our approach. Note, however, that a precise comparison between different approaches is difficult, since experimental setups, e.g., number of actions and length of segments, slightly differ with each approach.

In Fig. 9, we show the selected discriminative exemplars in the first six iterations of AdaBoost for ten different actions on Weizman dataset only using histogram of silhouette. It is observed that all these exemplars are representative. This phenomenon partly proves that our algorithm is capable of discovering intrinsic characteristics of the videos belonging to the same category. When specifically looking into "walk" and "run," "wave2" and "jack," one might note that they may



	boxing	handclapping	handwaving	jogging	running	walking
boxing	0.99	.01	.00	.00	.00	.00
handclapping	.01	.98	.01	.00	.00	.00
handwaving	.01	.00	.99	.00	.00	.00
jogging	.02	.00	.00	.81	.15	.02
running	.00	.00	.00	.00	1.0	.00
walking	.02	.00	.00	.00	.02	.96

Fig. 11. Experimental results on KTH dataset. (a) Recognition rates per action versus number of weak learner. (b) Average recognition rates versus number of weak learner. (c) Confusion matrix for action recognition.

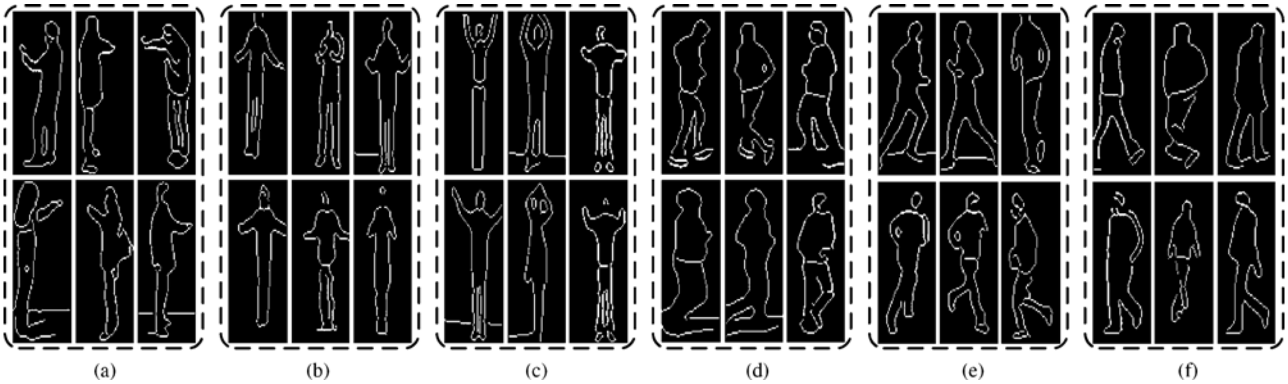


Fig. 12. Discriminative exemplars are selected for different actions by the final AdaBoost classifiers for the KTH dataset. (a) Boxing. (b) Handclapping. (c) Handwaving. (d) Jogging. (e) Running. (f) Walking.

TABLE I

COMPARISON OF DIFFERENT METHODS ABOUT MEAN ACCURACY ON KTH DATASET

Methods	Mean Accuracy (%)	Feature
Schuldt *et al.* [10]	71.71	Spatio-temporal interest points
Niebles and Li [11]	81.50	Spatio-temporal interest points
Saad and Mubarak [35]	87.70	Optical Flow
Liu and Mubarak [36]	94.15	3-D interest points
Our method	**95.33**	Histogram of edge

have some similar exemplars. However, these actions are distinguished by AdaBoost classifier via a weighted combination of these exemplars.

3) *Results on KTH Dataset:* The KTH human motion dataset (see Fig. 10) contains six types of human actions (walking, jogging, running, boxing, hand waving, and hand clapping). Each action is performed several times by 25 subjects in four different conditions: outdoors, outdoors with scale variation, outdoors with different clothes, and indoors. In this experiment, we use edge filtered sequences instead of background subtracted silhouettes. Edges are detected independently using a Canny edge detector. Based on the locations of people detected by using the method in Sabzmeydani and Mori [34], the histogram of edge for each cropped frame can be extracted.

Because this dataset contains tens of thousands of cropped frames, we first adopt k-means to cluster frames for each kind of action. The cluster number is set to 500 for each class, and the proposed method in Section III-A is used to select 80 candidate exemplars for each class. For each exemplar, $k_p = 3$ and $k_n = 3$ are adopted to obtain training samples for training an mi-SVM classifier. After obtaining all mi-SVM classifiers, each action video can be described. We use videos of 24 actors as training dataset and the rest as testing videos, and the results are reported as the average accuracy of 25 runs.

In Fig. 11(a), we show recognition rates for the individual classes. The average recognition rate on the test set and with respect to the number of weak learners is shown in Fig. 11(b). From Fig. 11(b), we can see that our recognition rate is about 95.33% with only 33 exemplars. The confusion matrix for this experiment is shown in Fig. 11(c) and the average accuracy is 95.33%. In Fig. 12, we show the selected discriminative exemplars in the first six iterations of AdaBoost for six different actions on KTH dataset just using histograms of edge. From Fig. 12, we can see that our approach is effective to select discriminative exemplars. Moreover, it is worth noting that "jogging" and "running" share similar exemplars, which results from the essential similarities between these two actions.

We also compare our performance with other state-of-art algorithms on KTH dataset. The performance is reported in Table I. It can be seen that performance using our proposed BEL approach exceeds other methods. We believe that the improvement attributes to the efficient similarity measure

TABLE II
COMPARISON OF DIFFERENT METHODS ABOUT MEAN ACCURACY AND COMPUTATIONAL TIME ON WEIZMANN AND KTH DATASETS

Dataset	Methods	#Exemplars	Accuracy (%)	Computational Time
Weizmann	[17]	2500	100	5 m
	Our method	200	**100**	0.5 m
KTH	[17]	6000	94.33	13.8 h
	Our method	480	**95.33**	1.1 h

"#Exemplars" represents the number of candidate exemplars.

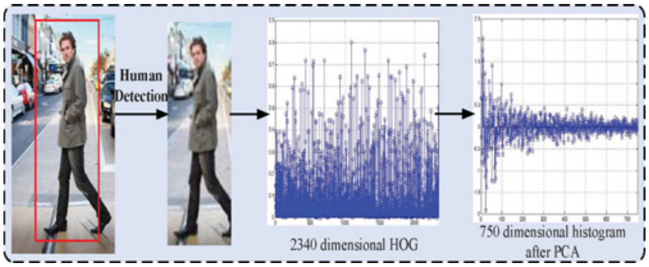


Fig. 13. Feature extraction.

learning and the effectively selected and combined discriminative exemplars via BEL approach. The results demonstrate the effectiveness of our method for recognizing different actions.

For video action recognition, we compare our method with [17] on accuracy and computational time on Weizmann and KTH datasets. The computational cost for each exemplar-based classifier depends on the number of training samples and the dimension of histogram feature. The computation time includes both training and testing time for all exemplar-based classifiers. On Weizmann dataset, our approach and [17] reach recognition rates up to 100%. However, [17] requires about 13 exemplars, and our method just uses nine discriminative exemplars. Because the dataset is small scale, there is not much difference in computational time. The computational time is 5 min and 0.5 min, respectively. On KTH dataset, our method costs 1.1 h to train exemplar-based classifiers compared with 13.8 h used in [17], and the performance has also been improved.

B. Video Action Annotation

In this experiment, we make use of image dataset (see Fig. 2) from the Web provided by Ikizler *et al.* [8] for training our action models, and annotate human actions in uncontrolled videos, such as YouTube videos, which compose Web dataset [37]. How to describe action instance and construct action bag is presented in Section IV-B1, and experimental results on the Web dataset are given in Section IV-B2.

1) *Action Instance Description and Bag Construction:* We use the implementation of Felzenswalb *et al.*'s human detector [38], which has been shown to be effective in detecting people in different poses. Once the humans are centralized, we extract an image descriptor for each detected area. In human detection, the histogram of oriented gradients (HOGs) has been shown to be successful [39]. We follow the construction in [39] to define a dense representation of an image at a particular resolution. The image is first divided into 8 × 8 non-overlapping pixel regions, or cells. For each cell we accumulate a 1-D histogram of gradient orientations over pixels in that cell. These histograms capture local shape properties but are also somewhat invariant to small deformations.

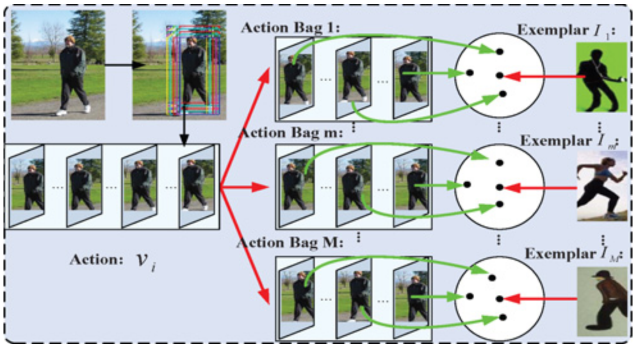


Fig. 14. Action v_i and its corresponding action bag for each exemplar. Note each action has M action bags corresponding to M exemplars.

The gradient at each pixel is discretized into one of nine orientation bins, and each pixel "votes" for the orientation of its gradient, with a strength that depends on the gradient magnitude at that pixel. For color images, we compute the gradient of each color channel and pick the channel with the highest gradient magnitude at each pixel. Finally, the histogram of each cell is normalized with respect to the gradient energy in a neighborhood around it. We look at the four 2 × 2 blocks of cells that contain a particular cell and normalize the histogram of the given cell with respect to the total energy in each of these blocks. This leads to a 9 × 4-D vector representing the local gradient information inside a cell. In our implementation, we resize each image to 128 × 64 and then extract HOGs in 8 × 8 cells. Our final feature vector is the 2340-D normalized HOG cell vector. After PCA, the dimension of the feature is further reduced to 750 to obtain compact description and efficient computation. Fig. 13 shows the process of feature extraction.

Next, we introduce how to construct action bags given a detector location. By shifting the obtained detection bounding box by ten pixels in x and y direction five times, respectively, we get 25 cropped frames in total. These cropped frames are considered as an action v_i. Fig. 14 shows an example for person detection and its corresponding action v_i. For each candidate exemplar, at most k_p nearest frames to it are selected for each positive action bag, and k_n for each negative action bag.

2) *Results on Web Dataset:* We use the image dataset (see Fig. 2) provided by Ikizler *et al.* [8] for training our action models. The image dataset contains five different actions: running, walking, sitting, playing golf, and dancing, which are collected from the Web. The final set contains 384 running, 307 walking, 313 sitting, 162 playing golf, and 561 dancing images. After person detection, we obtain 2454 action bags: 746 running, 479 walking, 349 sitting, 327 playing golf, and 746 dancing bags.

We use the video dataset provided by Niebles *et al.* [37] for testing our action models. This dataset consists of YouTube videos that have considerably low resolution and moving cameras. This dataset has been used for person detection purposes and does not include action annotations. Ikizler

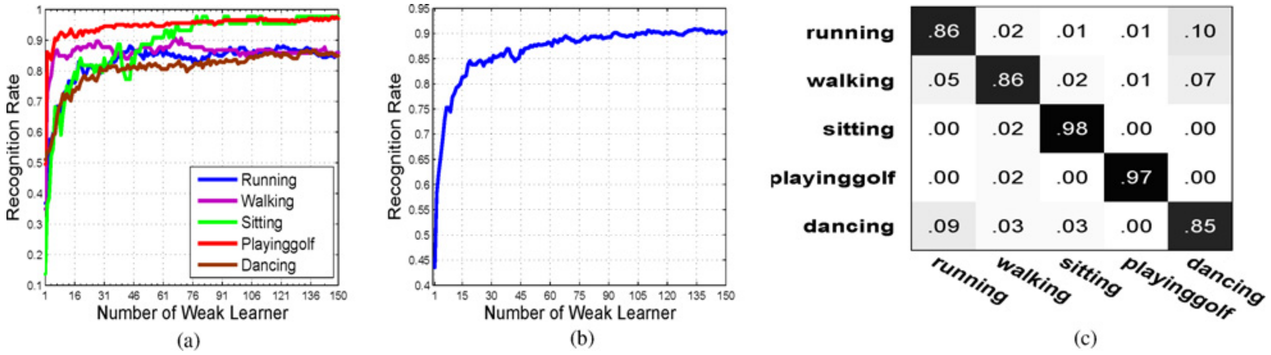


Fig. 15. Experimental results on Web dataset. (a) Recognition rates per action versus number of weak learner. (b) Average recognition rates versus number of weak learner. (c) Per frame confusion matrix for action annotation on YouTube videos.

TABLE III

COMPARISON OF DIFFERENT METHODS ABOUT MEAN ACCURACY AND COMPUTATIONAL TIME ON WEB DATASETS

Methods	# Exemplars	Mean Accuracy (%)	Computational Time
Zhang *et al.* [17]	2000	84.16	28.9 h
Our method	500	**90.40**	6.8 h

"#Exemplars" represents the number of all candidate exemplars.

TABLE IV

COMPARISON OF DIFFERENT METHODS ON YOUTUBE ACTION ANNOTATIONS

Methods	Mean Accuracy (%)	Feature
Ikizler *et al.* [8]	75.87	PbHOGs
Our method	**90.40**	HOG

Percentages shown are the average accuracies per frame.

et al. [8] annotated 11 videos from this dataset, 775 frames in total, which includes the five actions in combination. Based on this dataset, we obtain 777 action bags: 193 running, 106 walking, 44 sitting, 202 playing golf, and 232 dancing. Note that each video may contain more than one action, and since we will do frame by frame annotation, our method does not require action segmentation prior to application.

Because the training dataset contains tens of thousands of cropped frames, we use k-means to cluster the cropped frames and the cluster number is manually set to 400 for each kind of action. Then, the frames closest to the cluster are viewed as exemplars. After candidate exemplar selection in Section III-A, we obtain 100 candidate exemplars for each kind of action. For each exemplar, $k_p = 4$ and $k_n = 2$ are adopted to obtain some training samples for training an mi-SVM classifier. After obtaining all mi-SVM classifiers, each action bag is described and the discriminative exemplars are selected by AdaBoost classifier.

Compared with [17], our method proposes a discriminative process for candidate exemplars selection. The results are given in Table III, and we can see that our method is more effective and efficient than [17]. This is because the proposed candidate exemplar selection considers the discrimination of exemplars. Using k-means to obtain candidate exemplars in [17] loses some discriminative exemplars and leads to the lower performance. In addition, to avoid losing some discriminative exemplars, the number of clusters of k-means is manually set to 400 for each action class. This leads to the cost of computational time about 28.9 h. However, our method adopts an efficient strategy to obtain discriminative exemplars and reduce the computational time (6.8 h) and improve the performance (90.40%).

The experimental results on Web dataset are shown in Fig. 15. The recognition rates for the individual classes are given in Fig. 15(a), and the average recognition rate with respect to the number of weak learners is shown in Fig. 15(b). From the Fig. 15(b), we can notice that our recognition rate is increasing with the number of exemplars and is up to 90.40% with about 150 exemplars. The confusion matrix for the five different kinds of action is shown in Fig. 15(c). From the confusion matrix, we can see that the average accuracy is 90.40%, and most of the confusion occur between dancing and running actions. This is not surprising, because some of the dancing poses look very similar to running. We also show the selected discriminative exemplars in the first eight iterations of AdaBoost for five different kinds of action in Fig. 16. From Fig. 16, some more discriminative exemplars are obtained and this validates the effectiveness of our proposed method. We also observe that some selected discriminative exemplars are not the precision detection obtained by human detection, and most of discriminative exemplars are obtained by shifting the precision detection bounding box. This phenomenon shows that it is effective and true to adopt action bag instead of the precision detection, and formulate the selection of discriminative exemplars as the MIL problem. Moreover, one might note that "running" and "dancing" have some similar exemplars. That is because the two actions are very similar, and it leads to the confusion between dancing and running actions.

The comparison between our method with other state-of-art algorithm on the YouTube dataset is reported in Table IV. It can be seen that performance using our proposed BEL approach exceeds the method in [8]. Especially, we do not create separate tracks for each person and use temporal smoothing over each track. We get the final annotations for each frame using our action models learnt from Web images. Some results are shown in Fig. 17. Compared with [8], our method does not require head detection to achieve an alignment of the poses. In addition, the clutter in Web images makes it difficult to obtain

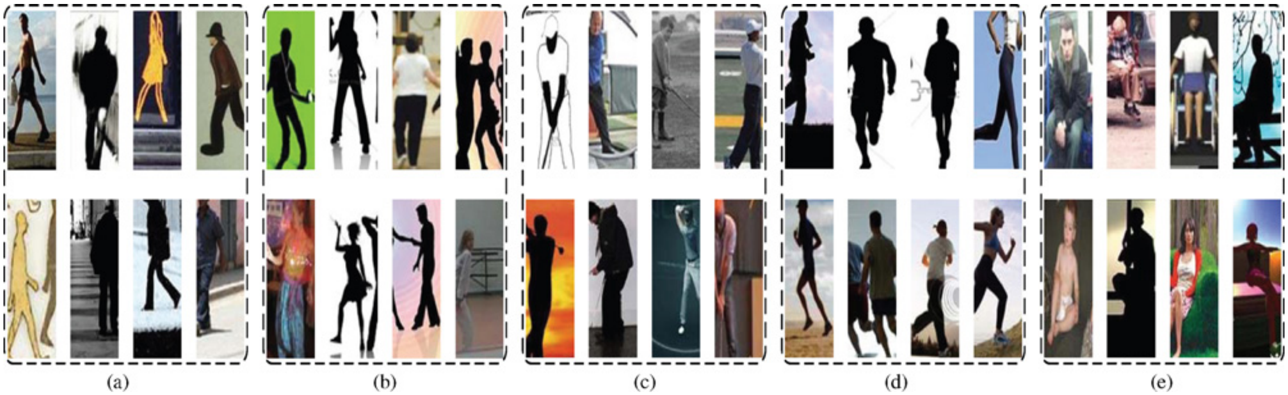


Fig. 16. Discriminative exemplars are selected for different actions by the final AdaBoost classifiers for the Web dataset. (a) Walking. (b) Dancing. (c) Playing golf. (d) Running. (e) Sitting.

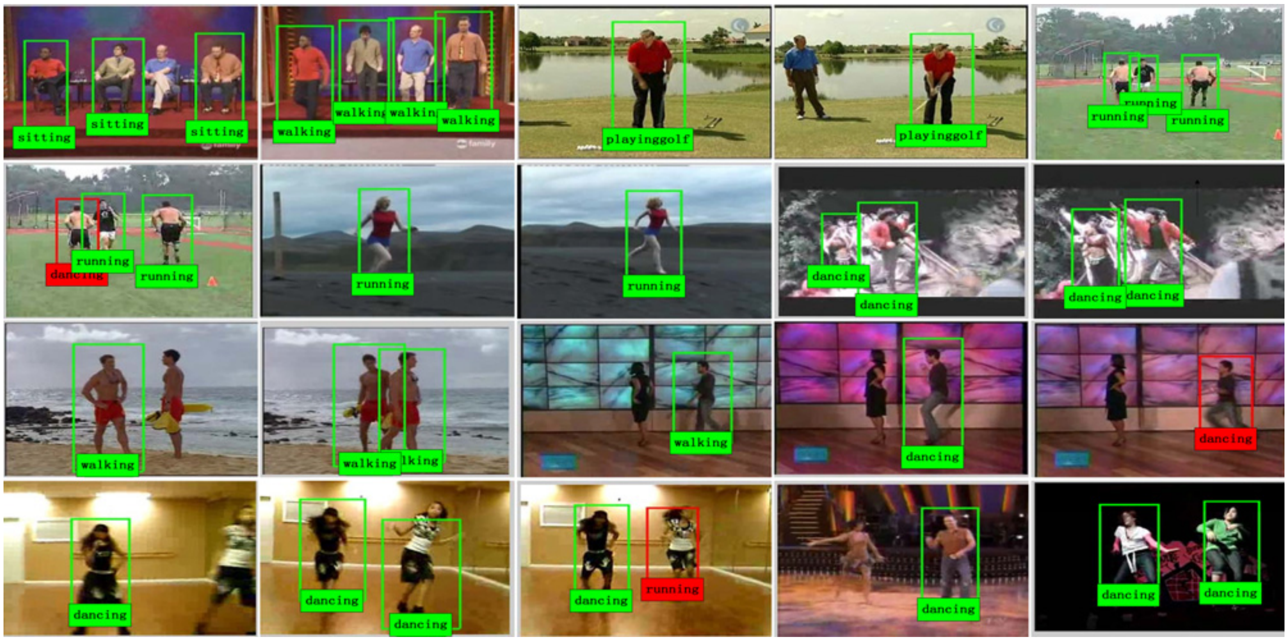


Fig. 17. Example annotated frames from YouTube videos of Niebles *et al.* [37]. We run the person detector [38] on these frames. Then, by applying our action models learnt from Web images, we get the final annotations. Note that, our method inherently handles multiple people and multiple actions. Correct classifications are shown in green and misclassifications are in red.

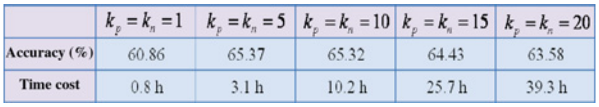
a useful frame description. In most cases, a simple gradient filtering based HOG descriptor is affected significantly by noisy responses. Therefore, Ikizler *et al.* [8] used the probability of boundary (Pb) operator and then extracted HOG features based on the responses. However, we only use the HOG feature and obtain a significant performance. Again, we believe that the improvement attributes to the efficient similarity measure learning and the effectively selected and combined discriminative exemplars via BEL approach. The results demonstrate the effectiveness of our method for recognizing different actions. As for the complex Web dataset, we think our method is more efficient and effective.

Here, we show some experimental results to demonstrate the filtering strategy proposed in Section III-B on Web dataset. First, we select 200 action bags for each class as the training samples and use k-means to cluster the cropped frames on each action category as an initial exemplar set. The cluster number is manually set to 300 for each kind of action, and the frames closest to the cluster are viewed as exemplars. After candidate exemplar selection in Section III-A, we obtain 100 candidate exemplars for each kind of action. For each action bag, there are 25 instances at most. To train each exemplar-based classifier, we manually set $k_p = k_n$ for simplicity and do five different experiments to show how k_p and k_n affect the recognition performance and efficiency. The results are shown in Table V. We can see that the five different experimental results have similar recognition performances, but the computational cost is increasing rapidly with the increased value of k_p and k_n. We do not set $k_p = k_n = 25$, because it causes that each exemplar-based classifier shares the same training samples.

In addition, our primary aim is to select some discriminative exemplars. These discriminative exemplars exist only in positive bag, and not in negative bag. Therefore, for an

TABLE V

ACCURACY AND COMPUTATIONAL TIME FOR DIFFERENT k_p AND k_n



	$k_p = k_n = 1$	$k_p = k_n = 5$	$k_p = k_n = 10$	$k_p = k_n = 15$	$k_p = k_n = 20$
Accuracy (%)	60.86	65.37	65.32	64.43	63.58
Time cost	0.8 h	3.1 h	10.2 h	25.7 h	39.3 h

exemplar, if it is similar to one instance in each positive bag, and no instance in each negative bag, we can confirm that this exemplar is very discriminative. Therefore, MIL is very suitable for training each exemplar-based classifier. To demonstrate this point, we compare our method with SVM classifier on Web dataset. When $k_p = k_n = 5$, the accuracy is about 61.53% using SVM classifier. We can see the mi-SVM has much better result 65.37%. The improvement is because our method is very suitable to select some discriminative exemplars compared with SVM.

V. CONCLUSION

We have presented a discriminative BEL approach for action recognition. Each cropped frame centered on the human figure was described using simple histogram features. Based on this description, the MIL was employed to learn the similarities between frames. Based on the learned exemplar-based classifiers, each action bag can be described, then AdaBoost was employed to select the most discriminative features to form a strong classifier. Experimental results illustrated the effectiveness and efficiency of the proposed method for video action recognition and annotation. For video action annotation, we showed that Web images can be used to annotate the videos taken in uncontrolled environments. However, it should be noted that not all actions can be discriminated with the exemplar-based approach. A typical example is an action and its reversal, e.g., sit-down and get-up. Without taking temporal ordering into account, it will be very difficult to discriminate them. In the future, we will investigate to incorporate motion information to improve our approach.

REFERENCES

[1] J. K. Aggarwal and Q. Cai, "Human motion analysis: A review," *Comput. Vision Image Understand.*, vol. 73, no. 3, pp. 428–440, 1999.

[2] T. B. Moeslund, A. Hilton, and V. Krüger, "A survey of advances in vision-based human motion capture and analysis," *Comput. Vision Image Understand.*, vol. 104, no. 2, pp. 90–126, 2006.

[3] S. Carlsson and J. Sullivan, "Action recognition by shape matching to key frames," in *Proc. Workshop Models Versus Exemplars Comput. Vision*, Dec. 2001, pp. 263–270.

[4] D. Weinland, E. Boyer, and R. Ronfard, "Action recognition from arbitrary views using 3-D exemplars," in *Proc. ICCV*, 2007, pp. 1–7.

[5] K. Schindler and L. van Gool, "Action snippets: How many frames does human action recognition require?" in *Proc. CVPR*, 2008, pp. 1–8.

[6] C. Thurau and V. Hlavac, "Pose primitive based human action recognition in videos or still images," in *Proc. CVPR*, 2008, pp. 1–8.

[7] D. Weinland and E. Boyer, "Action recognition using exemplar-based embedding," in *Proc. CVPR*, 2008, pp. 1–7.

[8] N. I. Cinbis, R. G. Cinbis, and S. Sclaroff, "Learning actions from the Web," in *Proc. ICCV*, 2009, pp. 995–1002.

[9] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri, "Actions as space-time shapes," in *Proc. ICCV*, 2005, pp. 1395–1402.

[10] C. Schuldt, I. Laptev, and B. Caputo, "Recognizing human actions: A local SVM approach," in *Proc. ICPR*, 2004, pp. 32–36.

[11] J. Niebles and L. Fei-Fei, "Unsupervised learning of human action categories using spatial-temporal words," in *Proc. BMVC*, 2006, pp. 299–318.

[12] H. Jhuang, T. Serre, L. Wolf, and T. Poggio, "A biologically inspired system for action," in *Proc. ICCV*, 2007, pp. 1–8.

[13] T. K. Kim, S. F. Wong, and R. Cipolla, "Tensor canonical correlation analysis for action classification," in *Proc. CVPR*, 2007, pp. 1–8.

[14] D. Tran and A. Sorokin, "Human activity recognition with metric learning," in *Proc. ECCV*, 2008, pp. 548–561.

[15] V. Athitsos and S. Sclaroff, "Estimating 3-D hand pose from a cluttered image," in *Proc. CVPR*, 2003, pp. 432–439.

[16] K. Toyama and A. Blake, "Probabilistic tracking in a metric space," in *Proc. ICCV*, 2001, pp. 50–57.

[17] T. Zhang, J. Liu, S. Liu, Y. Ouyang, and H. Lu, "Boosted exemplar learning for human action recognition," in *Proc. Workshop Video-Oriented Object Event Classification Conjunction ICCV*, 2009, pp. 538–545.

[18] Y. Guo, Y. Shan, H. Sawhney, and R. Kumar, "PEET: Prototype embedding and embedding transition for matching vehicles over disparate viewpoints," in *Proc. CVPR*, 2007, pp. 1–8.

[19] Y. Wang, H. Jiang, M. S. Drew, Z.-N. Li, and G. Mori, "Unsupervised discovery of action classes," in *Proc. CVPR*, 2006, pp. 1654–1661.

[20] Y. Dedeoglu, B. Toreyin, U. Gudukbay, and A. Cetin, "Silhouette-based method for object classification and human action recognition in video," in *Proc. HCI*, 2006, pp. 64–77.

[21] A. M. Elgammal, V. D. Shet, Y. Yacoob, and L. S. Davis, "Learning dynamics for exemplar-based gesture recognition," in *Proc. CVPR*, 2003, pp. 571–578.

[22] A. Fathi and G. Mori, "Human pose estimation using motion exemplars," in *Proc. ICCV*, 2007, pp. 1–8.

[23] J. Liu, J. Luo, and M. Shah, "Recognizing realistic actions from videos in the wild," in *Proc. CVPR*, 2009, pp. 1996–2003.

[24] J. Liu, Y. Yang, and M. Shah, "Learning semantic visual vocabularies using diffusion distance," in *Proc. CVPR*, 2009, pp. 461–468.

[25] Y. Hu, L. Cao, F. Lv, S. Yan, Y. Gong, and T. Huang, "Action detection in complex scenes with spatial and temporal ambiguities," in *Proc. IEEE Int. Conf. Comput. Vision*, Sep.–Oct. 2009, pp. 128–135.

[26] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2003.

[27] *MOSEK Optimization Software* [Online]. Available: http://www.mosek.com

[28] T. Dietterich, R. Lathrop, and T. Lozano-Perez, "Solving the multiple-instance problem with axis parallel rectangles," *Artif. Intell.*, vol. 89, nos. 1–2, Jan. 1997, pp. 31–71.

[29] S. Andrews, I. Tsochantaridis, and T. Hofmann, "Support vector machines for multiple-instance learning," in *Proc. NIPS*, 2002, pp. 561–568.

[30] J. Friedman, T. Hastie, and R. Tibshirani, "Additive logistic regression: A statistical view of boosting," *Ann. Statist.*, vol. 28, p. 2000, 1998.

[31] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 24, no. 4, pp. 509–522, Apr. 2002.

[32] L. Wang and D. Suter, "Recognizing human activities from silhouettes: Motion subspace and factorial discriminative graphical model," in *Proc. CVPR*, 2007, pp. 1–8.

[33] S. Ali, A. Basharat, and M. Shah, "Chaotic invariants for human action recognition," in *Proc. ICCV*, 2007, pp. 1–8.

[34] P. Sabzmeydani and G. Mori, "Detecting pedestrians by learning shapelet features," in *Proc. CVPR*, 2007, pp. 1–8.

[35] S. Ali and M. Shah, "Human action recognition in videos using kinematic features and multiple instance learning," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 32, no. 2, pp. 288–303, Feb. 2010.

[36] J. Liu and M. Shah, "Learning human actions via information maximization," in *Proc. CVPR*, 2008, pp. 1–8.

[37] J. C. Niebles, B. Han, A. Ferencz, and L. Fei-Fei, "Extracting moving people from Internet videos," in *Proc. ECCV*, 2008, pp. 527–540.

[38] P. Felzenszwalb, D. McAllester, and D. Ramanan, "A discriminatively trained, multiscale, deformable part model," in *Proc. CVPR*, 2008, pp. 1–8.

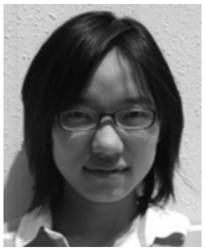
[39] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. CVPR*, 2005, pp. 886–893.

Tianzhu Zhang received the B.S. degree from the Beijing Institute of Technology, Beijing, China, in 2006. He is currently pursuing the Ph.D. degree at the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing.

In 2009, he was an Intern Student in the China-Singapore Institute of Digital Media, Singapore. His current research interests include multimedia, computer vision and machine learning.

Jing Liu (M'08) received the B.E. and M.E. degrees from Shandong University, Shandong, China, in 2001 and 2004, respectively, and the Ph.D. degree from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2008.

He is an Assistant Professor with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences. Her current research interests include machine learning, image content analysis and classification, multimedia information indexing and retrieval, and others.



Si Liu received the B.E. degree from the Beijing Institute of Technology, Beijing, China, in 2008. She is currently pursuing the Ph.D. degree at the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing.

In 2009, she was an Intern Student with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore. Her current research interests include multimedia, computer vision, and machine learning.

Changsheng Xu (M'97–SM'99) is a Professor with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China. He is also the Executive Director of the China-Singapore Institute of Digital Media, Singapore. His current research interests include multimedia content analysis, image processing, pattern recognition, and computer vision. He has published over 200 refereed book chapters, journal, and conference papers in these areas.

He is an Associate Editor of the *ACM/Springer Multimedia System Journal* and is on the Editorial Board of the *International Journal of Multimedia Intelligence and Security*. He was the Program Co-Chair of ACM Multimedia in 2009, the Program Co-Chair of the International Conference on Internet Multimedia Computing and Services in 2009, the General Co-Chair of the Pacific-Rim Conference on Multimedia (PCM) in 2008, the Short Paper Co-Chair of ACM Multimedia in 2008, the General Co-Chair of the Asia-Pacific Workshop on Visual Information Processing in 2007, the Program Co-Chair of Asia-Pacific Workshop on Visual Information Processing, and the Industry Track Chair and Area Chair of the International Conference on Multimedia Modeling in 2007. He is on the organizing committees and program committees in many prestigious multimedia conferences including ACM Multimedia, ICME, PCM, CIVR, and MMM, and others. He is the Director of Programs of the ACM SIG Multimedia Beijing Chapter. He is a member of ACM.

Hanqing Lu (M'05–SM'06) received the Ph.D. degree from Huazhong University of Sciences and Technology, Wuhan, China, in 1992.

Currently, he is a Professor of the Institute of Automation, Chinese Academy of Sciences, Beijing, China. His current research interests include image similarity measure, video analysis, and object recognition and tracking. He has published more than 200 papers in these areas.