Human Action Recognition by Semi-Latent Topic Models

Yang Wang, *Student Member, IEEE,* and Greg Mori, *Member, IEEE*

Abstract—We propose two new models for human action recognition from video sequences using topic models. Video sequences are represented by a novel "bag-of-words" representation, where each frame corresponds to a "word". Our models differ from previous latent topic models for visual recognition in two major aspects: first of all, the latent topics in our models directly correspond to class labels; secondly, some of the latent variables in previous topic models become observed in our case. Our models have several advantages over other latent topic models used in visual recognition. First of all, the training is much easier due to the decoupling of the model parameters. Secondly, it alleviates the issue of how to choose the appropriate number of latent topics. Thirdly, it achieves much better performance by utilizing the information provided by the class labels in the training set. We present action classification results on five different datasets. Our results are either comparable to, or significantly better than previous published results on these datasets.

Index Terms—Human action recognition, video analysis, bag-of-words, probabilistic graphical models, event and activity understanding

1 Introduction

Recognizing human actions from image sequences is a challenging problem in computer vision. It has applications in many areas, e.g., motion capture, medical bio-mechanical analysis, ergonomic analysis, human-computer interaction, surveillance and security, environmental control and monitoring, sport and entertainment analysis, etc. Various visual cues (e.g., motion [1]–[4] and shape [5]) can be used for recognizing actions. In this paper, we focus on recognizing the action of a person in an image sequence based on motion cues. We develop two novel models for human action recognition based on the "bag-of-words" paradigm.

Our models are motivated by the recent success of "bag-of-words" representations for object recognition problems in computer vision. The common paradigm of these approaches consists of extracting local features from a collection of images, constructing a codebook of visual words by vector quantization, and building a probabilistic model to represent the collection of visual words. While these models of an object as a collection of local patches are certainly not "correct" ones, for example, they only model a few parts of objects and often ignore many structures, they have been demonstrated to be quite effective in object recognition tasks [6]–[8].

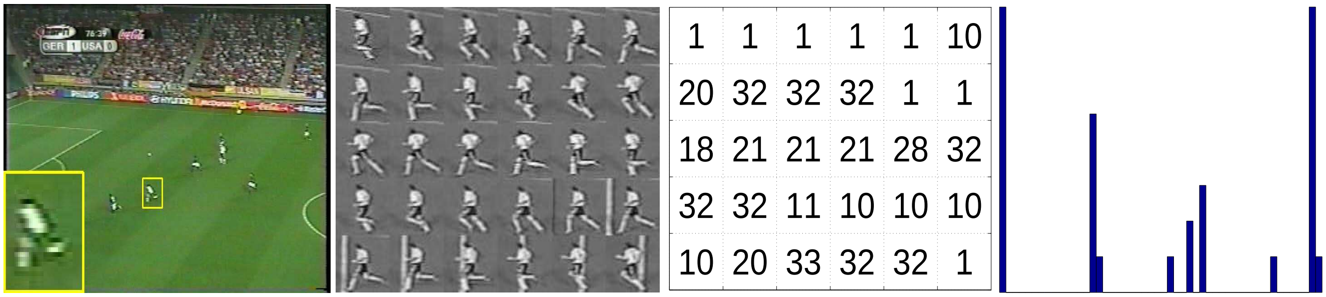
In this paper we explore the use of two similar models for recognizing human actions. Fig. 1 shows an overview of our "bag-of-words" representation. In our models, each frame of an image sequence is assigned to a visual word by analyzing the motion of the person it contains. The unordered set of these words over the image sequence becomes our bag of words. As with the object recognition approaches, some structures have been lost by moving to this representation. However, this representation is much simpler than one that explicitly models temporal structures. Instead we capture "temporal smoothing" via co-occurrence statistics amongst these visual words, i.e., which actions tend to appear together in a single track. For example, in a single track of a person, the combination of "walk left" and "walk right" actions is much more common than the combination of "run left""run right""run up""run down". We note that there has been previous work (e.g., Yamato et al. [9], Bobick & Wilson [10], Xiang & Gong [11]) that tries to model the full dynamics of videos using sophisticated probabilistic models (e.g., hidden Markov models, dynamic Bayesian networks). The problem with this approach is that those sophisticated models impose too many assumptions and constraints (e.g., the independence assumption of hidden Markov models) in order to be tractable. It is also hard to learn those models since there are usually a large number of parameters that need to be set. Instead, our methods can be considered as a way of imposing a "rough" constraint on the overall temporal structures of videos, without worrying about the detailed temporal information between adjacent frames. In this paper, we provide evidence that this simple representation can be quite effective in recognizing actions.

Our models are based on the Latent Dirichlet allocation (LDA) model [12] and the Correlated Topic Model (CTM) [13]. Topic models, such as, LDA, CTM, probabilistic Latent Semantic Analysis (pLSA) [14], and their variants, have been applied to various computer vision applications, such as scene recognition [15], [16], object recognition [17]–[19], action recognition [20], human detection [21], etc.

Despite the great success achieved, there are some

• *Y. Wang and G. Mori are with the School of Computing Science, Simon Fraser University, Burnaby, BC, V5A 1S6, Canada.*
E-mail: {ywang12, mori}@cs.sfu.ca

Manuscript received ; revised .



1	1	1	1	1	10
20	32	32	32	1	1
18	21	21	21	28	32
32	32	11	10	10	10
10	20	33	32	32	1

(a) (b) (c) (d)

Fig. 1. The processing pipeline of the "bag-of-words" representation: (a) given a video sequence, (b) track and stabilize each human figure, (c) represent each frame by a "motion word", (d) ignore the ordering of words and represent the image sequences of a tracked person as a histogram over "motion words".

unsolved, important issues remaining in this line of research. First of all, most of the previous approaches use their models for some specific recognition problem, say object class recognition. However, there is no guarantee that the latent topics found by their algorithms will necessarily correspond to object classes. Secondly, the features used in these approaches are usually SIFT-like local features computed at locations found by interest-point detectors. The only exceptions are histogram of oriented gradients in Bissacco et al. [21] and multiple segmentations in Russell et al. [18]. Features based on local patches may be appropriate for certain recognition problems, such as scene recognition or object recognition. But for human action recognition, it is not clear that they can be sufficiently informative about the action being performed. Instead, we use descriptors that can capture the large-scale properties of human figures, and compare these results to approaches using local patches.

In this paper, we attempt to address the above mentioned issues in two aspects. First of all, we introduce a new "bag-of-words" representation for image sequences. Our representation is dramatically different from previous ones (e.g., Niebles et al. [20]) in that we represent a frame in an image sequence as a "single word", rather than a "collection of words" computed at some spatial-temporal interest points. Our main motivation for this new representation is that human actions may be better characterized by large-scale features, rather than local patches. For example, consider the image in Fig. 2, it is very easy to see that the large scale motion descriptors (described in Section 3.1) capture some important characteristics of this motion, e.g., the movements of the legs. It is not obvious that one can recognize this action by just looking at several small patches in the image. Secondly, we propose two new topic models called *Semi-Latent Dirichlet Allocation (S-LDA)* and *Semi-latent Correlated Topic Model (S-CTM)*, respectively. The major difference between our models and the traditional latent topic model (e.g, *LDA* and *CTM*) is that some of the latent variables in LDA and CTM are observed during the training stage in S-LDA and S-CTM. We show that by naturally pushing the information provided by class labels of training data directly into our model, we can guide the previously latent topics to be our class labels, and consequently achieve much better performance. We notice that the idea of adding supervision to the LDA model has been applied in various ways in other work, e.g., supervised topic models [22], labeled LDA [23].

There are other alternative methods to train an action recognition system. E.g., one can train a discriminative classifer (e.g., SVM) to recognize the action of each frame individually. But this approach ignores the contextual information provided by different frames in a video. We will demonstrate experimentally that our approach performs better than this alternative.

The contributions of this paper are three-fold. First, we propose a novel bag-of-words representation for video sequences. Second, we introduce two semi-latent topic models in which class labels of the frames in a video are naturally exploited in the learning process. Third, we present extensive experimental results to show that the proposed models achieve state-of-the-art recognition accuracies on a large variety of datasets.

The rest of this paper is organized as follows. In Section 2 we review previous work. Section 3 gives the details of our approach. We present experimental results in Section 4 and conclude in Section 5.

A preliminary version [24] of this work appeared at the ICCV'07 workshop on Human Motion Understanding, Modeling, Capture and Animation.

2 Previous Work

A lot of work has been done in recognizing actions from both still images and video sequences. In this paper, we focus on recognition based on motion cues, although we are aware that other cues (e.g., shape cues [5], [25], [26]) have also been used for action recognition.

2.1 Motion-based Action Recognition

In our work, we use the motion descriptor developed by Efros et al. [2]. There are many other approaches that perform action recognition by analyzing patterns of motion. For example, Cutler & Davis [1], and Polana & Nelson [4] detect and classify periodic motions. Little & Boyd [3] analyze the periodic structure of optical flow patterns for gait recognition. There is also work using both motion and shape cues. For example, Bobick & Davis [27] use a representation known as "temporal templates" to capture both motion and shape, represented as evolving silhouettes. Shechtman & Irani [28] propose a space-time correlation method that can detect similarity between video segments. Jhuang et al. [29] apply neurobiological model of motion processing for action recognition using space-time gradient and optical flow features. Schindler & Van Gool [30] perform action recognition by training SVM classifiers based on local shapes and dense optical flows. Rodriguez et al. [31] propose a template-based method based on a maximum average correlation height filter that is capable of capturing intra-class variabilities.

2.2 Temporal Dynamic Models

There is a line of work on building sophisticated temporal dynamic models for modelling and understanding activities. The early work of Yamato et al. [9] uses the hidden Markov model (HMM) for recognizing human actions from image sequences. Feng and Perona [32] build HMM models from vector-quantized image shapes called "movelets". Olivera et al. [33] use a layered HMM to represent office activities. Recently, Xiang & Gong [11] model complex activities of multiple objects in cluttered scenes using dynamic Bayesian networks. Ikizler and Forsyth [34] use finite state models to search for complex activities from videos. Laxton et al. [35] build a dynamic Bayesian network to leverage temporal, contextual and ordering constraints for recognizing complex activities in video.

Although generative models (e.g., HMM) were popular in sequence modelling, recently people have also been applying discriminative models for modeling temporal information, e.g., conditional random fields (CRF) in Sminchisescu et al. [36] and hidden conditional random fields (HCRF) in Wang et al. [37].

A drawback of these models is that they have to make some assumptions (e.g., the independence assumption made by 1-st order HMM) in order to be computationally tractable. It is also hard to learn these models since there are usually many model parameters to be set.

2.3 Interest Point Methods

A popular approach in action recognition is based on spatial-temporal interest points and local feature descriptors. For convenience, those local descriptors are usually vector-quantized to obtain a finite set of "visual words" before they are fed into any classification algorithms. Laptev and Lindeberg [38] propose a space-time interest point operator that detect local structures in space-time that image observations have large local variations in both space and time. Schuldt et al. [39] train an SVM classifier based on this space-time features for recognizing human actions. Dollár et al. [40] propose space-time interest point detector based on a set of linear filters, and use these local features with k-nearest neighbor classifier for action recognition. Ke et al. [41] build a cascade of classifiers based on space-time volumetric features for event detection. Nowozin et al. [42] first detect local interest points, then learn a set of discriminative subsequences for action classification by exploiting the sequence mining techniques from data mining. Liu & Shah [43] exploit mutual information maximization techniques to learn a compact set of visual words. Niebles & Fei-Fei [44] combine shape information with local appearance features by building a hierarchical model that can be characterized as a constellation of bag-of-features. Local descriptors extracted from space-time interest points have also been shown to work well on videos with complex scenes (e.g., movies), e.g., Laptev & Pérez [45] and Laptev et al. [46] learn a boosted classifier based on those local descriptors to do action recognition on movie data.

In contrast, we use large-scale motion descriptors obtained from the whole frame to create the "visual words". We will demonstrate that these large-scale descriptors are better suited for the task of recognizing human actions.

2.4 Topic Models for Visual Recognition

Our approach is directly inspired by a body of work on using generative topic models for visual recognition based on the "bag-of-words" paradigm. The "bag-of-words" model was originally proposed for analyzing text documents, where a document is represented as a histogram over word counts. Generative topic models are then applied on this "bag-of-words" representation, and the topics of the document are denoted as latent variables in these models. Popular topic models include probabilistic Latent Semantic Analysis (pLSA) [14], Latent Dirichlet Allocation (LDA) [12] and Correlated Topic Models (CTM) [13]. Recently, successes have been made in adopting generative topic models with "bag-of-words" framework in solving various recognition problems in computer vision. Fei-Fei & Perona [16] use a variant of LDA for natural scene categorization. The "words" in their model correspond to small local patches in the images, and the "topics" correspond to the intermediate themes (e.g., "rocks") that make up a particular scene (e.g., mountain scene). Their model can learn the intermediate themes that are discriminative for different scene categories. Sivic et al. [19] perform unsupervised learning of object categories using variants of the pLSA model. In their models, the "words" correspond to local patches extracted by interested point operators, and the "topics" correspond to the different object categories. Fergus et al. [17] extend pLSA to incorporate spatial information in a translation and scale-invariant manner and apply them to learn object categories from Google's image search. One major issue with these approaches is that the "visual words" are usually obtained from local patches (e.g., using affine-invariant point descriptors) and are not as discriminative as their counterparts in text analysis. To address this issue, Russell et al. [18] use image segmentation to produce groups of related visual words. For each image, they obtain multiple segmentations by varying the parameters of the segmentation algorithms, and represent the segments from the segmentation algorithms as the "visual words" in their pLSA model. The hope is that some segments in some of the segmentations will correspond to objects of interest. And "topics" corresponding to these segments will be discovered by their pLSA model.

Topic models have also been applied in understanding images and videos containing human figures. Bissacco et al. [21] use LDA for human detection and pose classification. The "visual words" in their model are vector-quantizations of histogram of oriented gradients in the training images. LDA is used to model the intermediate themes that are distinctive for certain human poses. In human action recognition, Niebles et al. [47] recently demonstrate some impressive results on unsupervised learning of human action categories using pLSA and LDA models. The "visual words" in their models are based on features extracted from spatial-temporal interest points. Different human actions are captured by the different "topics" discovered by either pLSA or LDA. Wong et al. [48] adopt and extend pLSA models to capture both semantic (content of parts) and structural (connection between parts) information for recognizing actions and inferring the locations of certain actions.

3 Our Approach

Our approach follows the bag-of-words framework. But our models are different from previous bag-of-words models (e.g., Niebles et al. [20]) in two major aspects. First of all, our method represents a frame as a single word, rather than a collection of words from vector quantization of space-time interest points. In other words, a "word" corresponds to a "frame", and a "document" corresponds to a "video sequence" in our representation. Secondly, our model is trained in a supervised fashion. We will show that by utilizing the class labels, we can greatly simplify the training algorithm, and achieve much better recognition accuracy.

3.1 Motion Features and Codebook

We use the motion descriptor in Efros et al. [2] to represent the video frames. This motion descriptor has been shown to perform reliably with noisy image sequences, and has been applied in various tasks, such as action classification, motion synthesis, etc.

To calculate the motion descriptor, we first need to track and stabilize the persons in a video sequence. We use an automatic human detection method in some of our experiments. But any tracking or human detection methods can be used, since the motion descriptor we use is very robust to jitters introduced by the tracking.

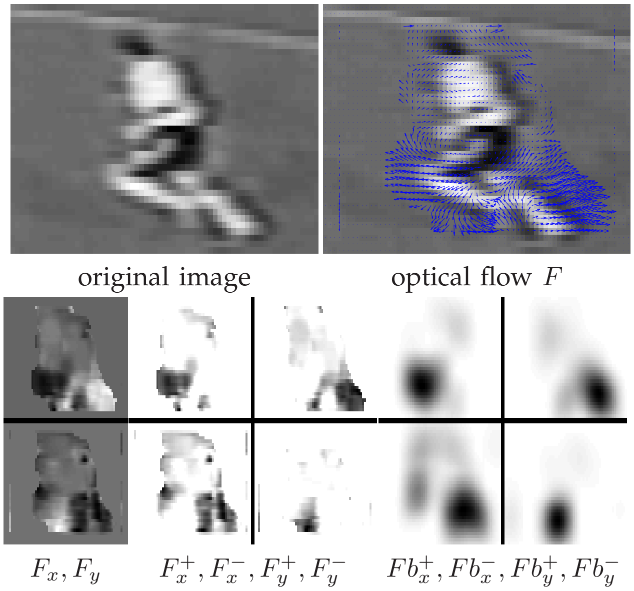

original image
optical flow F
F_x, F_y
$F_x^+, F_x^-, F_y^+, F_y^-$
$Fb_x^+, Fb_x^-, Fb_y^+, Fb_y^-$

Fig. 2. Construction of the motion descriptor. See Section 3.1 for details.

Given a stabilized video sequence in which the person of interest appears in the center of the field of view, we compute the optical flow at each frame using the Lucas-Kanade [49] algorithm. The optical flow vector field F is then split into two scalar fields F_x and F_y, corresponding to the x and y components of F. F_x and F_y are further half-wave rectified into four non-negative channels F_x^+, F_x^-, F_y^+, F_y^-, so that $F_x = F_x^+ - F_x^-$ and $F_y = F_y^+ - F_y^-$. These four non-negative channels are then blurred with a Gaussian kernel and normalized to obtain the final four channels Fb_x^+,Fb_x^-,Fb_y^+,Fb_y^- (see Fig. 2).

The motion descriptors of two different frames are compared using a version of the normalized correlation. Suppose the four channels for frame i of sequence A are a_1, a_2, a_3 and a_4, similarly, the four channels for frame j of sequence B are b_1, b_2, b_3 and b_4, then the similarity between frame A_i and frame B_j is:

$$S(A_i, B_j) = \sum_{t \in T} \sum_{c=1}^{4} \sum_{x,y \in I} a_c^{i+t}(x,y) b_c^{j+t}(x,y) \qquad (1)$$

where T and I is the temporal and spatial extent of the motion descriptors. We choose $T = 10$ in all of our experiments. The final dimensionality of the feature vector is $4 \times T \times I$.

To construct the codebook, we randomly select a subset from all the frames, compute the affinity matrix on this subset of frames, where each entry in the affinity matrix is the similarity between frame i and frame j calculated using the normalized correlation described above. Then we run k-medoid clustering on this affinity matrix to obtain V clusters. Codewords are then defined as the centers of the obtained clusters. In the end, each video sequence is converted to the "bag-of-words" rep-

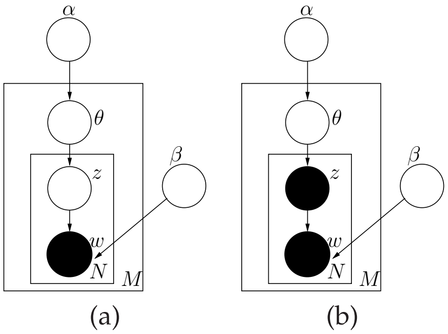


Fig. 3. Graphical representation of: (a) LDA model, adopted from Blei et al. [12]; (b) S-LDA for training. Note the difference from LDA is that z is observed in this case.

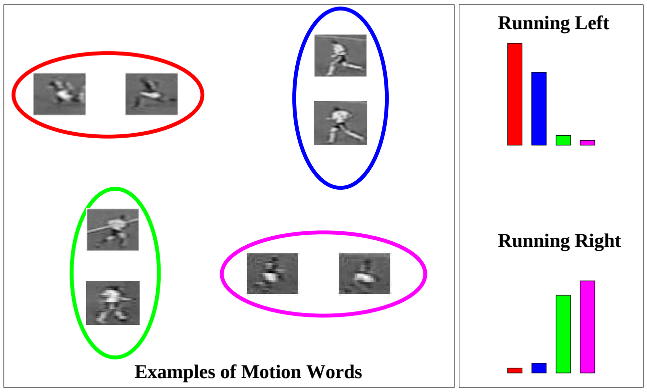


Fig. 4. Illustrations of topic models. Left: examples of clustering frames into “motion words”, where each cluster (denoted by a color ellipse) corresponds to a “word”. The red and blue clusters roughly correspond to two types of “running left”, while the green and pink clusters roughly correspond to “running right”. Right: examples of multinomial distributions (i.e., the parameter β) on words for two difference action labels. For the “running left” action, the multinomial distribution has large components on the red and blue components, while the distribution for “running right” has large components on the green and pink components.

resentation by replacing each frame by its corresponding codeword and removing the temporal information.

3.2 Latent Topic Models

Our models for video sequences are based on latent topics models, in particular, the Latent Dirichlet Allocation (LDA) [12] model and the Correlated Topic Model (CTM) [13]. In the following, we briefly introduce both of them using the terminology in our context.

3.2.1 Latent Dirichlet Allocation

Suppose we are given a collection D of video sequences $\{\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_M\}$. Each video sequence $\mathbf{w}$ is a collection of frames $\mathbf{w} = (w_1, w_2, ..., w_N)$, where w_i is the motion word representing the i-th frame. A motion word is the basic item from a codebook (see Section 3.1) indexed by $\{1, 2, ..., V\}$.

The LDA model assumes there are K underlying action labels (i.e., topics) according to which video sequences are generated. For example, a typical video sequence can be composed of frames with labels “walking left”, “running left”, ’standing still’, etc. Each action is represented by a multinomial distribution over the V motion words. The “motion words” can be thought of as a set of “prototypes” obtained by clustering all the frames in the training set. See Fig. 4 for an illustration. A video sequence is generated by sampling a mixture of these actions, then sampling motion words conditioning on a particular action. The generative process of LDA for a video sequence $\mathbf{w}$ in the collection can be formalized as follows (see Fig. 3(a)):

1) Choose $\theta \sim \mathrm{Dir}(\alpha)$
2) For each of the N motion words w_n:
    a) Choose an action label (i.e., topic) $z_n \sim \mathrm{Mult}(\theta)$;
    b) Choose a motion word w_n from $w_n \sim p(w_n|z_n, \beta)$, a multinomial probability conditioned on z_n.

The parameter θ indicates the mixing proportion of different action labels in a particular video sequence. α is the parameter of a Dirichlet distribution that controls how the mixing proportion θ varies among different video sequences. β is the parameter of a set of multinomial distributions, each of them indicates the distribution of motion words within a particular action label. The probability of a video $\mathbf{w} = \{w_1, w_2, ..., w_n\}$ is:

$$p(\mathbf{w}|\alpha, \beta) = \int p(\theta|\alpha) \left(\prod_{n=1}^{N} \sum_{z_n} p(z_n|\theta) p(w_n|z_n, \beta) \right) d\theta$$

Given a collection of video sequences $D = \{\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_M\}$, learning a LDA model involves finding the model parameters α and β that maximize the log likelihood of the data $\ell(\alpha, \beta) = \sum_{d=1}^{M} \log P(\mathbf{w}_d|\alpha, \beta)$. This parameter estimation problem can be solved by the variational EM algorithm developed in Blei et al. [12].

3.2.2 Correlated Topic Model

Blei & Lafferty [13] point out that the Dirichlet prior on topic proportions $\theta \sim \mathrm{Dir}(\alpha)$ in LDA does not properly model the correlation of different topics in the documents. To address this limitation, they propose a new topic model called the Correlated Topic Model (CTM). CTM uses the logistic normal distribution, rather than the Dirichlet distribution, as the prior distribution of the topic proportions. The generative process of CTM is as follows:

1) Choose $\eta \sim \mathcal{N}(\mu, \Sigma)$
2) For each of the N motion words w_n:
    a) Choose an action label (i.e., topic) $z_n \sim \mathrm{Mult}(\theta)$, where $\theta_i = \exp \eta_i / \sum_j \exp \eta_j$;
    b) Choose a motion word w_n from $w_n \sim \mathrm{Mult}(\beta_{z_n})$

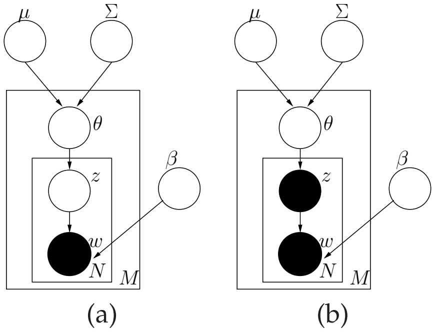


Fig. 5. Graphical representation of: (a) CTM model, adopted from Blei et al. [13]; (b) S-CTM for training. Note the difference from CTM is that z is observed in this case.

This generative process is identical to that of LDA except that the topic proportions are now drawn from a logistic normal rather than a Dirichlet distribution. The parameter estimation problem in CTM can be solved by the variational EM algorithm [13].

3.3 Semi-Latent Topic Models for Training

In the original LDA or CTM, we are only given the words $(w_1, w_2, ..., w_N)$ in each video sequence, but we do not know the topic z_i for the word w_i, nor the mixing proportion θ of topics in the sequence. In order to use LDA for classification problems, people have applied various tricks. For example, Blei et al. [12] use LDA to project a document onto the topic simplex, then train an SVM model based on this new representation, rather than the original vector representation of a document based on words. Although this simplex is a more compact representation for the document, the final SVM classifier based on this new representation actually performs worse than the SVM classifier trained on the original vector representation based on words. Sivic et al. [19] use a simpler method by classifying an image to a topic in which the latent topics of this image is most likely to be drawn from. There are two problems with this approach. First of all, there is no guarantee that a "topic" found by LDA or CTM corresponds to a particular "object class". Secondly, it is not clear how many "topics" to choose.

3.3.1 Semi-Latent Dirichlet Allocation (S-LDA)

In this paper, we are interested in the action classification problem, where all the frames in the training video sequences have action class labels associated with them. In this case, there is no reason to ignore this important information. In this section, we introduce a modified version the LDA model called *Semi-Latent Dirichlet Allocation (S-LDA)*. S-LDA utilizes class labels by enforcing a one-to-one correspondence between topics and class labels. Since we use a word w_i to represent a frame in a video sequence $\mathbf{w} = (w_1, w_2, ..., w_N)$, the topic z_i for the word w_i is simply the class label of w_i. The graphical representation of S-LDA model is shown in Fig. 3(b). We should emphasize that the model in Fig. 3(b) is only for training (i.e., estimating α and β). In testing, we will use the same model shown in Fig. 3(a), together with estimated model parameters α and β.

Our model has several major advantages over previous approaches of using a topic model for classification problems. First of all, the training process of the S-LDA model is much easier than the original LDA. Secondly, we can achieve much better recognition accuracy by taking advantage of the class labels (see Section 4). In addition, we do not have to choose the number of latent topics in S-LDA, since it is simply the number of class labels.

In LDA (see Fig. 3(a)), the parameters α and β are coupled, conditioning on the observed words $\mathbf{w}$. In that case, the model parameters (α and β) have to be estimated jointly, which is difficult. Various approximation approaches (e.g., sampling, variational EM, etc) have to be used. However, in S-LDA (Fig. 3(b)), the parameters α and β become independent, conditioning on observed words $\mathbf{w}$ and their corresponding topics (i.e., class labels) $\mathbf{z}$. So we can estimate α and β separately, which makes the training procedure much easier. In the following, we describe the details of how to estimate these parameters.

The parameter β can be represented by a matrix of size $K \times V$, where K is the number of possible topics (i.e., class labels) and V is the number of possible words. The i-th row of this matrix (β_i) is a V-dimensional vector that sums to 1. β_i is the parameter of a multinomial distribution, which defines the probability of drawing each word in the i-th topic. The maximum-likelihood estimate of β_i can be calculated by simply counting the frequency of each word appearing together with topic z_i, i.e., $\beta_{ij} = n_{ij}/n_{i\cdot}$, where $n_{i\cdot}$ is the count of the i-th topic in the corpus, and n_{ij} is the count of i-th topic with j-th word in the corpus.

The Dirichlet parameter α can be estimated from a "*Dirichlet-multinomial*" distribution (or *Polya* distribution), which is a compound distribution where θ is drawn from a Dirichlet and then a sample of discrete outcomes $\mathbf{z}$ is drawn from a multinomial with probability vector θ. The resulting distribution over $\mathbf{z}$ is $p(\mathbf{z}|\alpha) = \int_\theta p(\mathbf{z}|\theta)p(\theta|\alpha)d\theta$. Given a set of $\{\mathbf{z_1}, \mathbf{z_2}, ..., \mathbf{z_M}\}$, the maximum-likelihood estimate of α can be calculated using Newton-Raphson iterations [50].

3.3.2 Semi-latent Correlated Topic Model (S-CTM)

We can modify the CTM model in a similar way to obtain the *Semi-latent Correlated Topic Model (S-CTM)* for training. The graphical representation of S-CTM is shown in Fig. 5(b).

Similarly, the training algorithm of S-CTM is much simpler than CTM, due to the fact that the parameters μ and Σ are decoupled from the parameters β. The maximum likelihood estimation of β is identical to that in S-LDA. Given a set $\{\mathbf{z_1}, \mathbf{z_2}, ..., \mathbf{z_M}\}$, the parameters μ and Σ can be estimated using the EM algorithm [51].

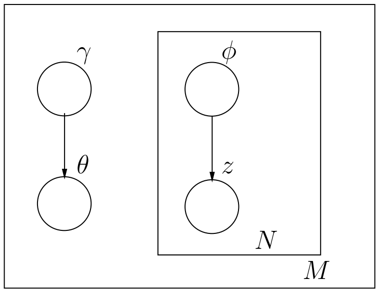


Fig. 6. Graphical representation of the variational distribution for LDA

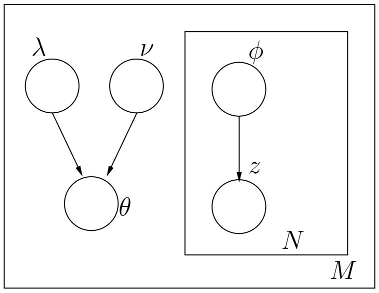


Fig. 7. Graphical representation of the variational distribution for CTM

3.4 Classification of New Video Sequences

Given a new video sequence for testing, we would like to classify each frame in the sequence. Suppose the testing video sequence is represented as $\mathbf{w} = (w_1, w_2, ..., w_N)$, i.e., there are N frames in the sequence, and the n-th frame is represented by the motion word w_n. Then, we need to calculate $p(z_n|\mathbf{w})$ $(n = 1, 2, ..., N)$. The frame w_n is classified to be action class k if $k = \arg\max_j p(z_n = j|\mathbf{w}, \alpha, \beta)$. Notice that we use $p(z_n|\mathbf{w})$ instead of $p(z_n|w_n)$ for classification. This reflects our assumption that the class label z_n not only depends on its corresponding word w_n, but also depends on the video sequence $\mathbf{w} = (w_1, w_2, ..., w_N)$ as a whole.

3.4.1 Inference of S-LDA

To calculate $p(z_n|\mathbf{w}, \alpha, \beta)$ in S-LDA, we use the variational inference algorithm proposed in Blei et al. [12]. The basic idea of the variational inference is to approximate the distribution $p(\theta, \mathbf{z}|\mathbf{w}, \alpha, \beta)$ by a simplified family of variational probability distributions $q(\theta, \mathbf{z})$ with the form $q(\theta, \mathbf{z}) = q(\theta|\gamma) \prod_{n=1}^{N} q(z_n|\phi_n)$. The graphical representation of $q(\theta, \mathbf{z}|\gamma, \phi)$ is shown in Fig. 6.

In order to make the approximation as close to the original distribution as possible, we need to find (γ^*, ϕ^*) that minimize the Kullback-Leibler (KL) divergence between the variational distribution $q(\theta, \mathbf{z}|\gamma, \phi)$ and the true distribution $p(\theta, \mathbf{z}|\mathbf{w}, \alpha, \beta)$, i.e., $(\gamma^*, \phi^*) = \arg\min_{(\gamma,\phi)} D(q(\theta, \mathbf{z}|\gamma, \phi) \| p(\theta, \mathbf{z}|\mathbf{w}, \alpha, \beta))$, where $D(\cdot|\cdot)$ is the KL divergence. Finding (γ^*, ϕ^*) can be achieved by iteratively updating (γ, ϕ) using the following update rules (see Blei et al. [12] for detailed derivations):

$$\hat{\phi}_{ni} \quad \propto \quad \beta_{iv} \exp(\Psi(\gamma_i) - \Psi(\sum_{j=1}^{K} \gamma_j)) \tag{2}$$

$$\hat{\gamma}_i \quad = \quad \alpha_i + \sum_{n=1}^{N} \phi_{ni} \tag{3}$$

Several insights can be drawn from examining the variational parameters $(\gamma^*(\mathbf{w}), \phi^*(\mathbf{w}))$. First of all, $(\gamma^*(\mathbf{w}), \phi^*(\mathbf{w}))$ are video-specific. Also notice that $\mathrm{Dir}(\gamma^*(\mathbf{w}))$ is the distribution from which the mixing proportion θ for the video sequence $\mathbf{w}$ is drawn. We can imagine that if we draw a sample $\theta \sim \mathrm{Dir}(\gamma^*(\mathbf{w}))$, θ will tend to peak towards the true mixing proportion θ^* of actions for the video $\mathbf{w}$. So the true mixing proportion θ^* can be approximated by the empirical mean of a set of samples $\theta^{(i)}$ drawn from $\mathrm{Dir}(\gamma^*(\mathbf{w}))$. The second insight comes from examining the ϕ_n parameters. These distributions approximate $p(z_n|w_n)$. The third insight is that, since the topic z_n is drawn from $\mathrm{Mult}(\theta^*)$, θ^* is an approximation of $p(z_n)$. Then we can get $p(z_n|\mathbf{w}) \propto p(z_n|\theta^*)p(z_n|w_n) \approx \theta^*_{z_n}\phi_{z_n w_n}$.This equation has a very appealing intuition. It basically says the class label z_n is determined by two factors, the first factor $\theta^*_{z_n}$ tells us the probability of generating topic z_n in a document with mixing proportion θ^*, the second factor $\phi_{z_n w_n}$ tells us the probability of generating topic z_n conditioning on a particular word w_n. We can classify frame n according to $p(z_n|\mathbf{w})$. If we know the video $\mathbf{w}$ only contains one action, the classification label for the entire video can be obtained by majority voting.

3.4.2 Inference of S-CTM

For S-CTM, the probability $p(z_n|\mathbf{w}, \mu, \Sigma)$ can be similarly calculated using a variational inference algorithm [13]. For fixed model parameters $\{\beta, \mu, \Sigma\}$, the log probability of a new document $\mathbf{w} = \{w_1, ..., w_n\}$ can be bound using Jensen's inequality:

$$\begin{aligned}
\ell = \quad & \log p(w_{1:N}|\mu, \Sigma, \beta) \\
\geq \quad & \mathbf{E}_q[\log p(\theta|\mu, \Sigma)] \\
+ \quad & \sum_{n=1}^{N} \mathbf{E}_q[\theta^\top z_n] + \sum_{n=1}^{N} \mathbf{E}_q[\log p(w_n|z_n, \beta)] + \mathbf{H}(q) \\
+ \quad & \sum_{n=1}^{N} \Big(-\zeta^{-1}\big(\sum_{i=1}^{K} \mathbf{E}_q[\exp\{\theta_i\}]\big) + 1 - \log(\zeta)\Big) \qquad (4)
\end{aligned}$$

where $\mathbf{H}(\cdot)$ is the entropy of a distribution, $\mathbf{E}_q(\cdot)$ is the expectation with respect to a variational distribution $q(\theta, \mathbf{z})$. The variational probability distribution $q(\theta, \mathbf{z})$ is chosen to have the form $q(\theta, \mathbf{z}) = \prod_{i=1}^{K} q(\theta_i|\lambda_i, \nu_i^2) \prod_{n=1}^{N} q(z_n|\phi_n)$, where $q(z_n|\phi_n)$ is a multinomial distribution, and $q(\theta_i|\lambda_i, \nu_i^2)$ is a univariate Gaussian distribution with mean λ_i and variance ν_i^2. The graphical representation of the variational distribution is shown in Fig. 7.

For fixed model parameters $\{\beta, \mu, \Sigma\}$, the optimal values for the variational parameters $\{\lambda^*, \nu^*, \zeta^*\}$ can be found using coordinate ascent, repeatedly optimizing (4) with respect to each parameter while holding others fixed.

Maximizing (4) with respect to ζ and ϕ_n gives the following update rules:

$$\hat{\zeta} = \sum_{i=1}^{K} \exp\{\lambda_i + \nu_i^2/2\} \quad (5)$$

$$\hat{\phi}_{n,i} \propto \exp\{\lambda_i\}\beta_{i,w_n}, \qquad i \in \{1, ..., K\} \quad (6)$$

Maximizing (4) with respect to λ_i and ν_i^2 does not yield analytic solutions, but conjugate gradient algorithms can be used with derivatives (see [13] for detailed derivations):

$$\frac{d\ell}{\lambda} = -\Sigma^{-1}(\lambda - \mu) + \sum_{n=1}^{N} \phi_{n,1:K} - (N/\zeta)\exp\{\lambda + \nu^2/2\}$$

$$\frac{d\ell}{d\nu_i^2} = -\Sigma_{ii}^{-1}/2 - N/2\zeta \exp\{\lambda + \nu_i^2/2\} + 1/(2\nu_i^2)$$

After obtaining $(\gamma^*(\mathbf{w}), \phi^*(\mathbf{w}))$ for the video $\mathbf{w}$, we can compute $p(z_n|\mathbf{w})$ using the same technique used in S-LDA, i.e. $p(z_n|\mathbf{w}) \propto p(z_n|\theta^*)p(z_n|w_n) \approx \theta^*_{z_n}\phi_{z_n w_n}$, where θ^* is the empirical mean of a set of samples $\theta^{(i)}$ drawn from $\mathrm{Dir}(\gamma^*(\mathbf{w}))$. The frame n is classified according to $p(z_n|\mathbf{w})$. For single-action videos, the classification label of the entire video can obtained by majority voting. Of course, this is not the optimal way to obtain the per-video classification from the per-frame classification – one can build a much more sophisticated model for this purpose. But this work focuses on per-frame classification, so we prefer a simple and straightforward method to obtain the class label of the whole video. We will show experimentally that this simple method already gives very good results.

4 Experiments

We test our algorithm on five datasets: KTH human motion dataset [39], Weizmann human action dataset [52], hockey dataset [25], soccer dataset [2], and a new ballet dataset [53]. See Fig. 8 for sample frames from each dataset. Since the first three datasets (KTH, Weizmann, hockey) only contain single-action video sequences, and most of the previously published results are on per-video classification, we will focus on per-video classification (using majority voting described in the previous section) on these three datasets. The video sequences in the last two datasets (soccer, ballet) contain multiple actions in a video sequence, so we cannot do per-video classification. We will only report per-frame classification results on them. For each dataset, we perform "leave-one-out" cross-validation. For the first two datasets (KTH, Weizmann), we leave the videos of one person as test data each time. And for the last three datasets (hockey, soccer, ballet) we leave one video as test data each time.

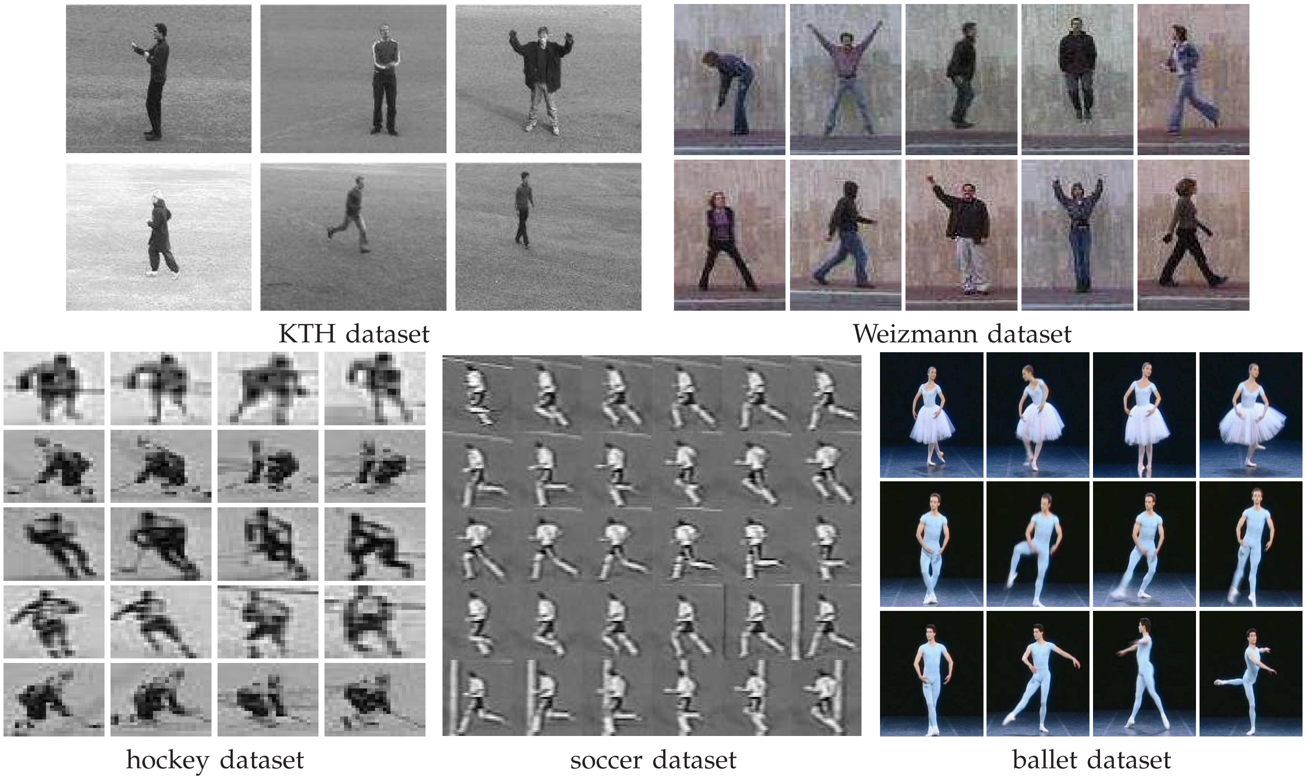
On the KTH and Weizmann datasets, we also show experimental results on using SVM with large-scale features and compare with previous work that uses SVM with local patch features. Since the classification algorithm is identical in this case, this will allow us to compare large-scale features and local patch features directly.

Our approaches are efficient. Most of the computation is spent on the features and codewords. After the bag-of-words representation is obtained, learning the model usually takes less than one minute, and inference on a new video only takes a few seconds in our unoptimized MATLAB implementation combined with existing codes in MATLAB/C [12], [13], [50], [51].

KTH dataset: The KTH human motion dataset contains six types of human actions (walking, jogging, running, boxing, hand waving and hand clapping) performed several times by 25 subjects in four different scenarios: outdoors, outdoors with scale variation, outdoors with different clothes and indoors. Representative frames of this dataset are shown in Fig. 8. We first run an automatic preprocessing step to track and stabilize the video sequences, so that all the figures appear in the center of the field of view. The confusion matrix for per-video classification of S-LDA on the KTH dataset using 1050 codewords is shown in Fig. 9(a). We can see that the algorithm correctly classifies most actions. Most of the mistakes the algorithm makes are confusions between "running" and "jogging" actions. This is intuitively reasonable, since "running" and "jogging" are similar actions. On all of our datasets, the confusion matrices (per-frame or per-video) of S-LDA and S-CTM have similar patterns, so we will only show one confusion matrix on each dataset.

We also test the effect of the codebook size on the overall accuracy for both per-frame and per-video classification. The result is shown in Fig. 9(b). The best accuracy is achieved with 1050 codewords for both S-LDA and S-CTM, but the results are relatively stable. We compare our results with previous approaches on the same dataset, as shown in Table 1. Performances on KTH and Weizmann (see below) datasets are saturating – state-of-the-art approaches achieve near-perfect results. Also we should note that different methods listed in Table 1 have all sorts of variations in their experiment setups, e.g., different splits of training/testing data, whether some preprocessing (e.g., tracking, background subtraction) is needed, with or without supervision, whether per-frame classification can be done, whether a method handles multiple action classes in a video, etc. The results of our methods are comparable to other state-of-the-art approaches,although we emphasize that this is not a precise comparison due to the variations in the experiment setups mentioned before.

The KTH dataset only contains single-action videos. If one only needs per-video classification, the simplest approach is to train a discriminative classifier based on the large-scale features. In order to see how the



KTH dataset

Weizmann dataset

hockey dataset

soccer dataset

ballet dataset

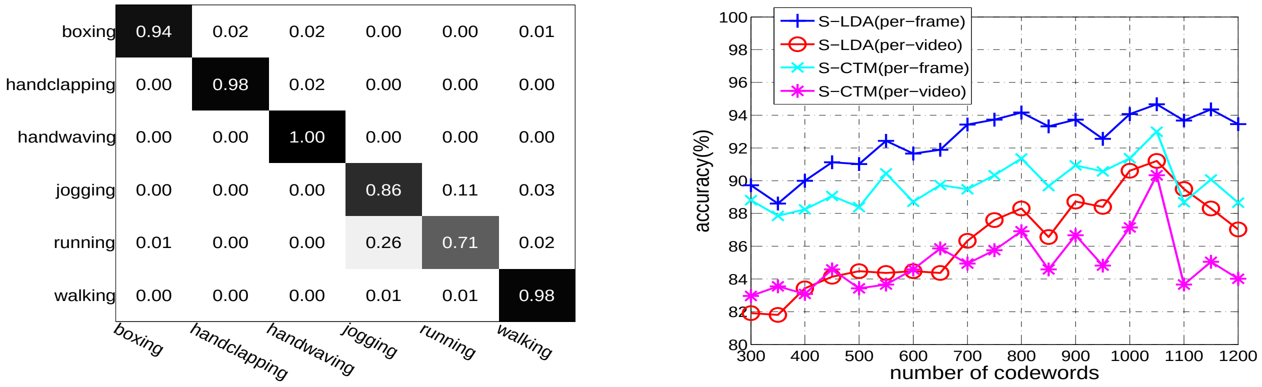
Fig. 8. Sample frames from our datasets. The action labels in each dataset are as follows. (1) KTH dataset: walking, jogging, running, boxing, hand waving, hand clapping; (2) Weizmann dataset: running, walking, jumping-jack, jumping-forward-on-two-legs, jumping-in-place-on-two-legs, galloping-sideways, waving-two-hands, waving-one-hand, bending; (3) hockey dataset: skating down, skating left, skating leftdown, skating leftup, skating right, skating rightdown, skating rightup, skating up; (4) soccer dataset: run left 45°, run left, walk left, walk in/out, run in/out, walk right, run right, run right 45°; (5) ballet dataset: left-to-right hand opening, right-to-left hand opening, standing hand opening, leg swinging, jumping, turning, hopping, standing still.

TABLE 1
Comparison of different reported results (per-video) on KTH dataset. There are many variations in terms of experiment setups among different methods, so a precise comparison between these methods is not possible.

methods	accuracy(%)
S-LDA	91.20
S-CTM	90.33
Liu & Shah [43]	94.16
Schindler & Van Gool [30]	92.70
Wong et al. [48]	91.60
Jhuang et al. [29]	91.70
Nowozin et al. [42]	87.04
Niebles et al. [20]	81.50
Dollár et al. [40]	81.17
Schuldt et al. [39]	71.72
Ke et al. [41]	62.96

large-scale features would perform in this case, we build a histogram representation based on the visual words computed from the large-scale features for each video, then train an SVM classifer based on the histogram. This is similar to what has been done in object recognition [54]. Using a linear SVM with the trade-off parameter $C = 1000$, the best per-video classification performance is 83.31%, which is much better than a similar approach in Schuldt et al. [39] that uses SVM based on histograms of visual words obtained from local patches. We have tried other different values for the parameter C. The results are relatively stable (within $\sim 3\%$). This demonstrates that the large-scale features outperforms local patch features. However, using the same large scale features, our approach (S-LDA/S-CTM) still outperforms SVMs.

Weizmann dataset: The Weizmann human action dataset contains 83 video sequences showing nine different people, each performing nine different actions. We track and stabilize the figures using the background subtraction masks that come with this dataset. Some sample tracked frames are shown in Fig. 8. Fig. 10(b) shows how our per-frame and per-video classification accuracies change as we vary the codebook size. S-LDA achieves a result of 100% per-video classification and 98.91% per-frame classification with 650 codewords. S-CTM achieves 100% accuracies for both per-frame and per-video classifications. In Fig. 10(a), we show the con-



(a) (b)

Fig. 9. Results on KTH dataset: (a) confusion matrix for per-video classification of S-LDA using 1050 codewords (overall accuracy=91.2%). Horizontal rows are ground truths, and vertical columns are predictions; (b) classification accuracy (per-frame and per-video) vs. codebook size.

TABLE 2
Comparison of classification accuracies (per-frame, per-video, and per-cube) with previous methods on Weizmann dataset.

	per-frame(%)	per-video(%)	per-cube(%)
S-LDA	98.91	100	N/A
S-CTM	100	100	N/A
Fathi & Mori [53]	99.9	100	N/A
Jhuang et al. [29]	N/A	98.8	N/A
Niebles & Fei-Fei [44]	55	72.8	N/A
Blank et al. [52]	N/A	N/A	99.64

TABLE 3
Classification accuracies (per-frame and per-video) of S-LDA and S-CTM on the hockey dataset.

	per-frame(%)	per-video (%)
S-LDA	85.43	87.50
S-CTM	77.43	76.04

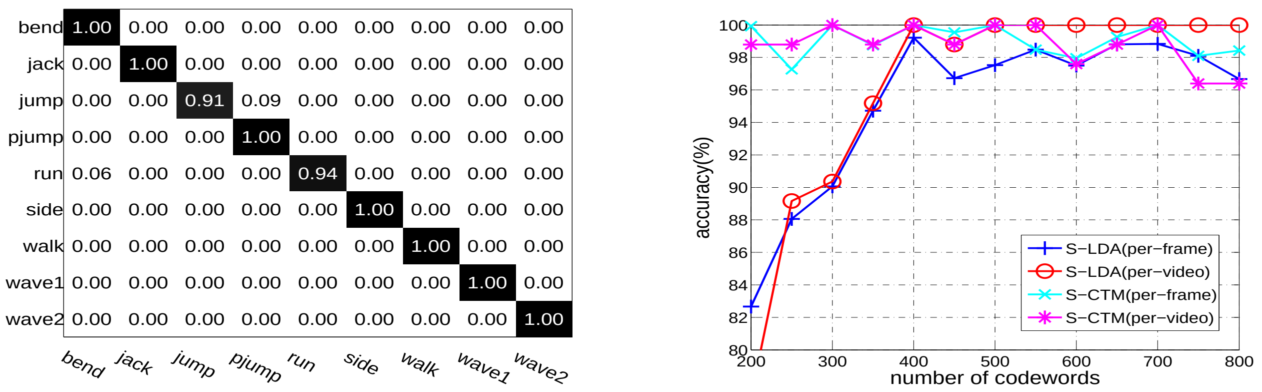
fusion matrix for per-frame classification of S-LDA with 650 codewords. We do not show the confusion matrix for per-video classification, since it is simply a perfect diagonal matrix. We compare our results with previous methods in Table 2. Again, we accept the fact that the comparison to Niebles & Fei-Fei [44] is not completely fair, since their method does not require any tracking or background subtraction. Also notice that [52] classifies space-time cubes. It is not clear how it can be compared with other methods that classify frames or videos.

Again, we conduct another experiment using an SVM classifier with the large-scale features for video classification. The per-video accuracy is 98.8% (linear SVM with $C = 1000$), which is inferior to our approaches (S-LDA/S-CTM).

Hockey dataset: The hockey dataset contains 70 tracks of hockey players performing eight actions (skate down, skate left, skate leftdown, skate leftup, skate right, skate rightdown, skate rightup, skate up). The action labels we obtained from the authors of [25] are slightly different from what they used in [25]. Some sample frames are shown in Fig. 8. The confusion matrix of per-video classification of S-LDA is shown in Fig. 11(a). Again, most of the mistakes are intuitively reasonable, e.g., “skate leftdown” is confused with “skate down”, “skate rightdown” is confused with “skate down”, “skate rightup” is confused with “skate up”, etc. In Fig. 11(b), we show how per-frame and per-video classification accuracies vary with different codebook sizes.

We compare the result of S-LDA and S-CTM in Table 3. The KNN method based on HOG features in Lu et al. [25] achieves a per-frame classification accuracy of 76.37%. However, since Lu et al. [25] use different class labels (they only have four class labels instead of eight) and experiment setup (∼4000 frames for training, ∼1000 frames for testing, frames from a single video could be in both training and testing sets), a direct comparison with their approach is impossible.

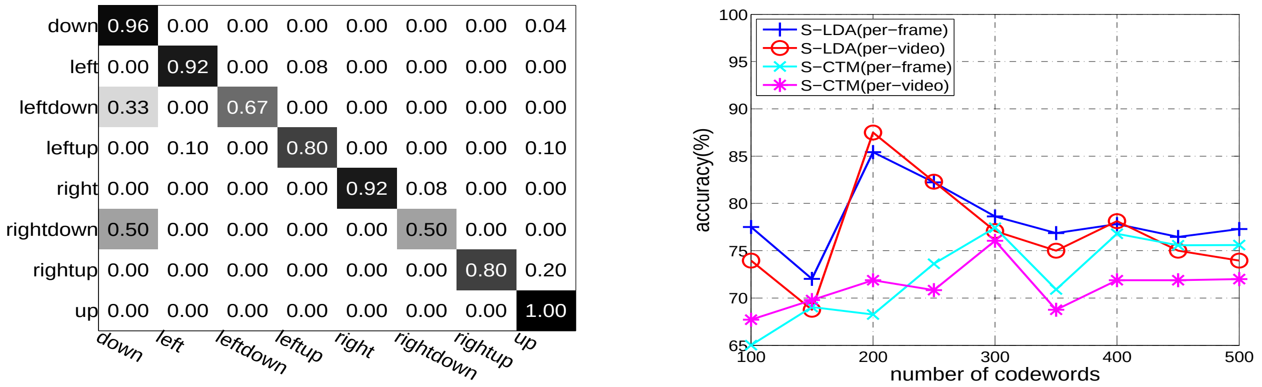
Soccer dataset: The soccer dataset contains several minutes of digitized World Cup football game from an NTSC video tape. A preprocessing step is taken to track and stabilize each human figure. In the end, we obtain 35 video sequences, each corresponding to a person moving in the center of the field of view. We flip the sequences and get 70 video sequences in total. All the frames in these video sequences are hand-labeled with one of 8 action labels: “run left 45 °”, “run left”, “walk left”, “walk in/out”, “run in/out”, “walk right”, “run right”, “run right 45 °”. Representative frames of a single tracked person are shown in Fig. 8. The confusion matrix of S-LDA using 950 codewords is shown in Fig. 12(a). The overall accuracy is 77.81%. The overall accuracy of S-CTM is 78.64%. A previously reported result is in [2], which uses a k-nearest neighbor classifier based on the temporally smoothed motion feature vectors. However,



	bend	jack	jump	pjump	run	side	walk	wave1	wave2
bend	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
jack	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
jump	0.00	0.00	0.91	0.09	0.00	0.00	0.00	0.00	0.00
pjump	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00
run	0.06	0.00	0.00	0.00	0.94	0.00	0.00	0.00	0.00
side	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00
walk	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00
wave1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00
wave2	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00

(a) (b)

Fig. 10. Results on Weizmann dataset: (a) confusion matrix for per-frame classification of S-LDA using 700 codewords (overall accuracy=98.83%); (b) classification accuracy (per-frame and per-video) vs. codebook size.



	down	left	leftdown	leftup	right	rightdown	rightup	up
down	0.96	0.00	0.00	0.00	0.00	0.00	0.00	0.04
left	0.00	0.92	0.00	0.08	0.00	0.00	0.00	0.00
leftdown	0.33	0.00	0.67	0.00	0.00	0.00	0.00	0.00
leftup	0.00	0.10	0.00	0.80	0.00	0.00	0.00	0.10
right	0.00	0.00	0.00	0.00	0.92	0.08	0.00	0.00
rightdown	0.50	0.00	0.00	0.00	0.00	0.50	0.00	0.00
rightup	0.00	0.00	0.00	0.00	0.00	0.00	0.80	0.20
up	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00

(a) (b)

Fig. 11. Results on hockey dataset: (a) confusion matrix for per-video classification of S-LDA using 200 codewords (overall accuracy=87.5%); (b) classification accuracy (per-frame and per-video) vs. codebook size.

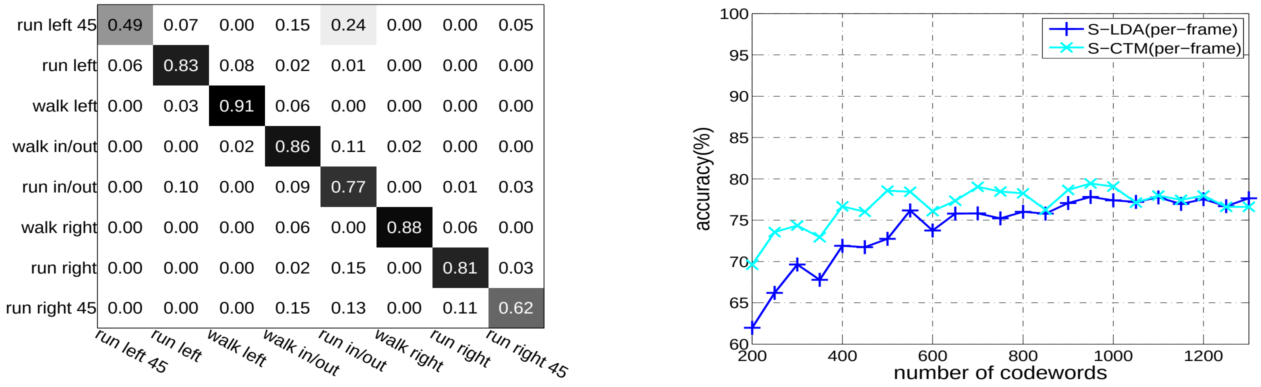
the comparison with [2] is a bit tricky, since [2] only flips some sequences, and there is no way to figure out which sequences are flipped in their experiment. In order to do a fair comparison, we re-run the KNN algorithm in [2] using their own implementation on our dataset. Table 4 shows the overall accuracy and the main diagonals of the confusion matrices of our methods, compared with KNN [2]. We can see that our methods perform better by a large margin. A lot of the mistakes made by our algorithm make intuitive sense. For example, "run left 45 °" is confused with "run left" and "run in/out", "walk right" is confused with "walk in/out" and "run right", etc. Fig. 12(b) shows the effect of codebook size on the accuracy. The best accuracy peaks at around 950 codewords for both S-LDA and S-CTM.

We can visualize the learned parameter θ. Since there is a θ parameter for each video, we cannot show all of them. In Fig. 13, we show the θ parameter of two different videos in the dataset learned by S-LDA. It is obtained as the empirical mean of samples drawn from $\text{Dir}(\gamma)$ (see Sec. 3.4 for more details). It is obvious that the learned θ correctly captures the topic proportions (i.e., the proportion of actions) in each video.

TABLE 4
Comparison of the overall accuracy (per-frame) and the main diagonal of the confusion matrix in our method and the k-nearest neighbor (KNN) method in Efros et al. [2] on the soccer dataset. The result of KNN is obtained by running their own implementation on our dataset, see the text for details.

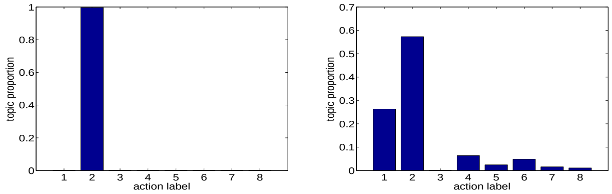
	S-LDA	S-CTM	KNN
run left 45 °	0.4909	0.6208	0.3277
run left	0.8273	0.9557	0.6195
walk left	0.9149	0.8821	0.7585
walk in/out	0.8552	0.8552	0.4202
run in/out	0.7712	0.7390	0.2910
walk right	0.8821	0.8821	0.7585
run right	0.8076	0.8076	0.6195
run right 45 °	0.6208	0.6208	0.3277
Overall	0.7781	0.7864	0.4923

Ballet dataset: Finally, we test our algorithm on a ballet dataset we collected from an instructional ballet DVD. This dataset has been used in Fathi & Mori [53].



(a) (b)

Fig. 12. Results on soccer dataset: (a) confusion matrix for per-frame classification of S-LDA using 950 codewords (overall accuracy=77.81%); (b) per-frame classification accuracy vs. codebook size.



(a) (b)

Fig. 13. Visualization of θ parameters for two different videos in the soccer dataset: (a) a video with all the frames labeled as action 2("run left"); (b) a video with all the frames labeled as action 1 ("run left 45°") or 2 ("run left").

TABLE 5
Comparison of the overall accuracy (per-frame) with Fathi & Mori [53] on the ballet dataset.

	per-frame(%)
S-LDA	88.66
S-CTM	91.36
Fathi & Mori [53]	51

We obtain 44 tracks using a simple tracking algorithm based on color histograms. We manually label each frame with one of the eight action labels:"left-to-right hand opening", "right-to-left hand opening", "standing hand opening", "leg swinging", "jumping", "turning", "hopping", and "standing still". Some sample frames are shown in Fig. 8. Fig. 14(a) shows the confusion matrix of per-frame classification of S-LDA. Our algorithm classifies most actions correctly. The only exception is "standing still". The reason is because it is difficult to reliably obtain optical flow features for this action. Fig. 14(b) shows how the classification accuracy varies with different codebook sizes.

We compare our results with Fathi & Mori [53] in Table 5. Since Fathi & Mori [53] use exactly the same experiment setup, the comparison is fair. We can see that our method performs significantly better.

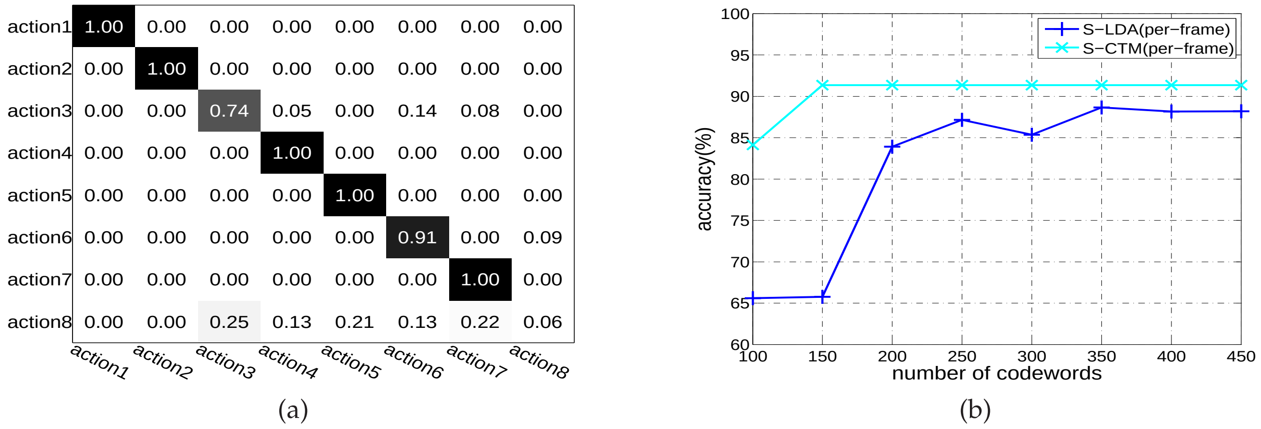
Remarks: From the above experiments, we can see that both S-LDA and S-CTM perform quite well. On the KTH and Weizmann datasets, the results of both models are quite similar. On the soccer and ballet datasets, S-CTM seems to perform better than S-LDA. This is reasonable, since these two datasets have multiple actions in a video sequence, S-LDA does not capture their correlations as well as S-CTM. On the hockey dataset, S-LDA seems to perform much better than S-CTM. We believe it is due to the fact that the size of the dataset is quite small, and there are no strong correlations among different class labels (the video only contains single actions). In this case, S-CTM is prone to overfitting since there are more model parameters to estimate.

We would like to point out that for single-action videos, one can train a discrimative classifier (e.g., SVM) to classify the videos. But our experimental results on KTH and Weizmann datasets show that S-LDA and S-CTM outperform SVM classifiers using the same large-scale features. The real advantage of our models is that we can deal with multiple actions in a video. Of course, one can also train a discriminative classifier to classify each individual frame, e.g., [2], [53], but our experimental results on the soccer and ballet datasets demonstrate empirically that those approaches perform inferior to ours.

Our experimental results also demonstrate that our approach is not very sensitive to the codebook size. The accuracy is quite stable in a large range of codebook sizes.

5 Conclusion

We have presented two supervised hierarchical topic models for action recognition based on motion words. Compared with previous topic models used in visual recognition, our models are different in several aspects. First, a "visual word" in our models is obtained from large-scale motion descriptors from a whole frame,



	action1	action2	action3	action4	action5	action6	action7	action8
action1	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
action2	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00
action3	0.00	0.00	0.74	0.05	0.00	0.14	0.08	0.00
action4	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00
action5	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00
action6	0.00	0.00	0.00	0.00	0.00	0.91	0.00	0.09
action7	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00
action8	0.00	0.00	0.25	0.13	0.21	0.13	0.22	0.06

Fig. 14. Results on ballet dataset: (a) confusion matrix for per-frame classification of S-LDA using 350 codewords (overall accuracy=88.66%), see the text for descriptions of these actions; (b) per-frame classification accuracy vs. codebook size.

rather than small space-time patches. The main motivation for using large-scale descriptors is that they better capture the characteristics of human actions. Second, the "latent topics" in our models directly correspond to different action categories. And class labels in the training data are naturally exploited in the learning process. On five different datasets, our methods consistently achieve superior results – either comparable to, or significantly better than other state-of-the-art results.

Of course, our method has its own limitations. For example, it requires a preprocessing stage of tracking and stabilizing human figures. However, we believe this is a reasonable assumption in many scenarios. In fact, all the video sequences in our experiments are pre-processed by off-the-shelf tracking/detection algorithms. The current datasets we use do not have significant background clutter. It will be interesting to explore the use of our method on more complicated datasets. As future work, we would like to collect and try our approach on more difficult datasets.

Acknowledgments

The authors would like to thank Payam Sabzmeydani for his contribution in the early stage of this research. We also thank the authors of [2], [12], [13], [25], [39], [50]–[52] for making various datasets and source codes used in this paper publicly available. This work is supported by a grant from NSERC (RGPIN-312230).

References

[1] R. Cutler and L. S. Davis, "Robust real-time periodic motion detection, analysis, and applications," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 781–796, August 2000.

[2] A. A. Efros, A. C. Berg, G. Mori, and J. Malik, "Recognizing action at a distance," in *IEEE International Conference on Computer Vision*, 2003, pp. 726–733.

[3] J. L. Little and J. E. Boyd, "Recognizing people by their gait: The shape of motion," *Videre*, vol. 1, no. 2, pp. 1–32, 1998.

[4] R. Polana and R. C. Nelson, "Detection and recognition of periodic, non-rigid motion," *International Journal of Computer Vision*, vol. 23, no. 3, pp. 261–282, June 1997.

[5] J. Sullivan and S. Carlsson, "Recognizing and tracking human action," in *European Conference on Computer Vision LNCS 2352*, vol. 1, 2002, pp. 629–644.

[6] L. Fei-Fei, R. Fergus, and P. Perona, "One-shot learning of object categories," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 4, pp. 594–611, April 2006.

[7] K. Grauman and T. Darrell, "The pyramid match kernel: Discriminative classification with sets of image features," in *IEEE International Conference on Computer Vision*, vol. 2, 2005, pp. 1458–1465.

[8] S. Lazebnik, C. Schmid, and J. Ponce, "A maximum entropy framework for part-based texture and object recognition," in *IEEE International Conference on Computer Vision*, vol. 1, 2005, pp. 832–838.

[9] J. Yamato, J. Ohya, and K. Ishii, "Recognizing human action in time-sequential images using hidden markov model," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1992.

[10] A. F. Bobick and A. D. Wilson, "A state-based approach to the representation and recognition of gesture," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 12, pp. 1325–1337, December 1997.

[11] T. Xiang and S. Gong, "Beyond tracking: Modelling activity and understanding behaviour," *International Journal of Computer Vision*, vol. 67, no. 1, pp. 21–51, 2006.

[12] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation," *Journal of Machine Learning Research*, vol. 3, pp. 993–1022, 2003.

[13] D. M. Blei and J. D. Lafferty, "Correlated topic models," in *Advances in Neural Information Processing Systems 18 (NIPS)*, 2006.

[14] T. Hofmann, "Probabilistic latent semantic indexing," in *Proceedings of Twenty-Second Annual International Conference on Research and Development in Information Retrieval(SIGIR)*, 1999, pp. 50–57.

[15] A. Bosch, A. Zisserman, and X. Munoz, "Scene classification via pLSA," in *European Conference on Computer Vision*, vol. 4, 2006, pp. 517–530.

[16] L. Fei-Fei and P. Perona, "A bayesian hierarchical model for learning natural scene categories," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, 2005, pp. 524–531.

[17] R. Fergus, L. Fei-Fei, P. Perona, and A. Zisserman, "Learning object categories from google's image search," in *IEEE International Conference on Computer Vision*, vol. 2, 2005, pp. 1816–1823.

[18] B. C. Russell, A. A. Efros, J. Sivic, W. T. Freeman, and A. Zisserman, "Using multiple segmentations to discover objects and their extent in image collections," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2006.

[19] J. Sivic, B. C. Russell, A. A. Efros, A. Zisserman, and W. T. Freeman, "Discovering objects and their location in images," in

IEEE International Conference on Computer Vision, vol. 1, 2005, pp. 370–377.

[20] J. C. Niebles, H. Wang, and L. Fei-Fei, "Unsupervised learning of human action categories using spatial-temporal words," in *British Machine Vision Conference*, vol. 3, 2006, pp. 1249–1258.

[21] A. Bissacco, M.-H. Yang, and S. Soatto, "Detecting humans via their pose," in *Advances in Neural Information Processing Systems 19 (NIPS)*. MIT Press, 2007, pp. 169–176.

[22] D. M. Blei and J. D. McAuliffe, "Supervised topic models," in *Advances in Neural Information Processing Systems*. MIT Press, 2008, vol. 20.

[23] P. Flaherty, G. Giaever, J. Kumm, M. I. Jordan, and A. P. Arkin, "A latent variable model for chemogenomic profiling," *Bioinformatics*, vol. 21, no. 15, pp. 3286–3293, 2005.

[24] Y. Wang, P. Sabzmeydani, and G. Mori, "Semi-latent Dirichlet allocation: A hierarchical model for human action recognition," in *The 2nd Workshop on Human Motion Understanding, Modeling, Capture and Animation*, 2007.

[25] W.-L. Lu, K. Okuma, and J. J. Little, "Tracking and recognizing actions of multiple hockey players using the boosted particle filter," *Image and Vision Computing*, vol. 27, no. 1-2, pp. 189–205, January 2009.

[26] C. Thuran and V. Hlaváč, "Pose primitive based human action recognition in videos or still images," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2008.

[27] A. F. Bobick and J. W. Davis, "The recognition of human movement using temporal templates," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 3, pp. 257–267, 2001.

[28] E. Shechtman and M. Irani, "Space-time behavior based correlation," in *International Conference on Computer Vision and Pattern Recognition*, 2005.

[29] H. Jhuang, T. Serre, L. Wolf, and T. Poggio, "A biologically inspired system for action recognition," in *IEEE International Conference on Computer Vision*, 2007.

[30] K. Schindler and L. Van Gool, "Action snippets: How many frames does action recognition require?" in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2008.

[31] M. D. Rodriguez, J. Ahmed, and M. Shah, "Action MACH: A spatial-temporal maximum average correlation height filter for action recognition," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2008.

[32] X. Feng and P. Perona, "Human action recognition by sequence of movelet codewords," in *International Symposium on 3D Data Processing Visualization and Transmission*, 2002.

[33] N. Olivera, A. Garg, and E. Horvitz, "Layered representations for learning and inferring office activity from multiple sensory channels," *Computer Vision and Image Understanding*, vol. 96, no. 2, pp. 163–180, November 2004.

[34] N. Ikizler and D. Forsyth, "Searching video for complex activities with finite state models," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2007.

[35] B. Laxton, J. Lim, and D. Kriegman, "Leveraging temporal, contextual and ordering constraints for recognizing complex activities in video," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2007.

[36] C. Sminchisescu, A. Kanaujia, Z. Li, and D. Metaxas, "Conditional models for contextual human motion recognition," in *IEEE International Conference on Computer Vision*, 2005.

[37] S. B. Wang, A. Quattoni, L.-P. Morency, D. Demirdjian, and T. Darrell, "Hidden conditional random fields for gesture recognition," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2006.

[38] I. Laptev and T. Lindeberg, "Space-time interest points," in *IEEE International Conference on Computer Vision*, 2003.

[39] C. Schuldt, L. Laptev, and B. Caputo, "Recognizing human actions: a local SVM approach," in *IEEE International Conference on Pattern Recognition*, vol. 3, 2004, pp. 32–36.

[40] P. Dollár, V. Rabaud, G. Cottrell, and S. Belongie, "Behavior recognition via sparse spatio-temporal features," in *ICCV'05 Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, 2005.

[41] Y. Ke, R. Sukthankar, and M. Hebert, "Efficient visual event detection using volumetric features," in *IEEE International Conference on Computer Vision*, vol. 1, 2005, pp. 166–173.

[42] S. Nowozin, G. Bakir, and K. Tsuda, "Discriminative subsequence mining for action classification," in *IEEE International Conference on Computer Vision*, 2007.

[43] J. Liu and M. Shah, "Learning human actions via information maximization," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2008.

[44] J. C. Niebles and L. Fei-Fei, "A hierarchical model of shape and appearance for human action classification," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2007.

[45] I. Laptev and P. Pérez, "Retrieving actions in movies," in *IEEE International Conference on Computer Vision*, 2007.

[46] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld, "Learning realistic human actions from movies," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2008.

[47] J. C. Niebles, H. Wang, and L. Fei-Fei, "Unsupervised learning of human action categories using spatial-temporal words," *International Journal of Computer Vision*, vol. 79, no. 3, pp. 299–318, September 2008.

[48] S.-F. Wong, T.-K. Kim, and R. Cipolla, "Learning motion categories using both semantic and structure information," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2007.

[49] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proceedings of the DARPA Image Understanding Workshop*, April 1981, pp. 121–130.

[50] T. P. Minka, "Estimating a Dirichlet distribution," Massachusetts Institute of Technology, Tech. Rep., 2000.

[51] J. Huang and T. Malisiewicz, "Fitting a hierarchical logistic normal distribution," http://www.cs.cmu.edu/∼jch1/research/hln/hlnfit.html.

[52] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri, "Actions as space-time shapes," in *IEEE International Conference on Computer Vision*, 2005.

[53] A. Fathi and G. Mori, "Action recognition by learning mid-level motion features," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2008.

[54] J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid, "Local features and kernels for classification of texture and object categories: A comprehensive study," *International Journal of Computer Vision*, vol. 73, no. 2, pp. 213–238, 2007.

PLACE
PHOTO
HERE

Yang Wang Biography text here.

PLACE
PHOTO
HERE

Greg Mori Biography text here.