Semi-Latent Dirichlet Allocation: A Hierarchical Model for Human Action Recognition

Yang Wang, Payam Sabzmeydani, and Greg Mori

School of Computing Science
Simon Fraser University
Burnaby, BC, Canada
{ywang12,psabzmey,mori}@cs.sfu.ca

Abstract. *We propose a new method for human action recognition from video sequences using latent topic models. Video sequences are represented by a novel "bag-of-words" representation, where each frame corresponds to a "word". The major difference between our model and previous latent topic models for recognition problems in computer vision is that, our model is trained in a "semi-supervised" way. Our model has several advantages over other similar models. First of all, the training is much easier due to the decoupling of the model parameters. Secondly, it naturally solves the problem 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 and irregularity detection results, and show improvement over previous methods.*

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 [6, 8, 16, 20] and shape [26]) 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 a novel model of human actions based on the "bag-of-words" paradigm.

Our model is motivated by the recent success of "bag-of-words" representation 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 parts are certainly not "correct" ones, for example they only model a few parts of objects and often ignore much structure, they have been demonstrated to be quite effective in object recognition tasks [9, 12, 15].

In this paper we explore the use of a similar model, for recognizing human actions. Figure 1 shows an overview of our "bag-of-words" representation. In our model, each frame in 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 structure has been lost by moving to this representation. However, this model is much simpler than one which explicitly models temporal structure. 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". In this paper we provide evidence that this simple model can be quite effective in recognizing actions.

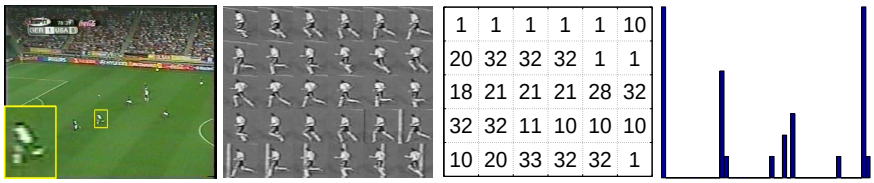


Fig. 1. The processing pipeline of getting the "bag-of-words" representation: (a) given a video sequence, (b) track and stabilize each human figure, (c) represent each frame by a "motion word", (d) ignore the ordering of words and represent the image sequences of a tracked person as a histogram over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

In particular, our model is based on the latent Dirichlet allocation (LDA) [2] model. LDA, the probabilistic Latent Semantic Indexing (pLSI) [13] model, and their variants have been applied to various computer vision applications, such as scene recognition [5, 10], object recognition [11, 22, 25], action recognition [19], human detection [1], etc.

Despite the great success achieved, there are some unsolved, important issues remaining in this line of research. First of all, it is not clear how to choose the right number of latent topics in one of these models. Previous methods usually take a rather ad-hoc approach, e.g., by trying several different numbers. But this is often not possible in realistic settings. Secondly, 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. Thirdly, 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. [1] and multiple segmentations in Russell et al. [22]. 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 by 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. [19]) 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 characterized by large-scale features, rather than local patches. Secondly, we propose a new topic model called *Semi-Latent Dirichlet Allocation (S-LDA)*. The major difference between our model and the *Latent Dirichlet Allocation (LDA)* model is that some of the latent variables in LDA are observed during the training stage in S-LDA. We show that our model naturally solves the problem of choosing the right number of latent topics. Also by pushing the information provided by class labels of training data directly into our model, we can guide the latent topics to be our class labels, and consequently achieve much better performance.

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

2 Previous Work

A lot of work has been done in recognizing actions from both still images and video sequences. Much of this work is focused on analyzing patterns of motion. For example, Cutler & Davis [6], and Polana & Nelson [20] detect and classify periodic motions. Little & Boyd [16] analyze the periodic structure of optical flow patterns for gait recognition. Rao et al. [21] describe a view-invariant representation for 2D trajectories of tracked skin blobs. Others consider the shape of human figure. For example, Sullivan & Carlsson [26] use "order structure" to compare the shape of extracted edges for the purpose of action recognition. There is also work using both motion and shape cues. For example, Bobick & Davis [3] use a representation known as "temporal templates" to capture both motion and shape, represented as evolving silhouettes. Zhong et al. [27] cluster segments of long video sequences by looking at co-occurrences of patterns of motion and appearance.

Our approach is closely related to a body of work on recognition using "bag-of-words". The "bag-of-words" model was originally proposed for analyzing text documents [2, 13]. Recently, researchers in the computer vision community have used "bag-of-words" models for various recognition problems. Fei-Fei & Perona [10] use a variant of LDA for natural scene categorization. Sivic et al. [25], Fergus et al. [11] and Russell et al. [22] use pLSI for unsupervised object class recognition and segmentation. Niebles et al. [19] use pLSI for action recognition

using spatial-temporal visual words. Bissacco et al. [1] use LDA for human pose classification from vector-quantized words from histograms of oriented gradients.

3 Our Approach

Similar to Niebles et al. [19], we represent a video sequence as a "bag of words". But our representation is different from Niebles et al. [19] in two 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 semi-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. [8] to represent the video sequences. 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 the human detection method in Sabzmeydani & Mori [23] 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.

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 [17] 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 fours channels for frame A are a_1, a_2, a_3 and a_4, similarly, the four channels for frame B are b_1, b_2, b_3 and b_4, then the similarity between frame A and frame B is:

$$S(A,B) = \sum_{c=1}^{4} \sum_{x,y \in I} a_c(x,y) b_c(x,y) \tag{1}$$

where I is the spatial extent of the motion descriptors. In Efros et al. [8], a temporal smoothing is also used, but we found the simplified version without temporal smoothing works good enough for our application.

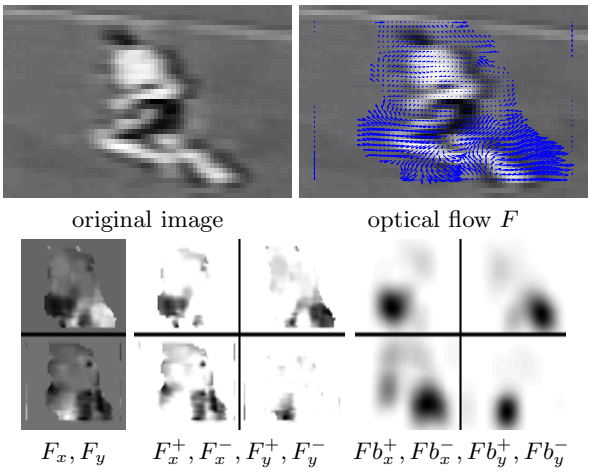


Fig. 2. Construction of the motion descriptor

To construct the codebook, we randomly select a subset from all the frames, compute the affinity matrix A 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, all the video sequences are converted to the "bag-of-words" representation by replacing each frame by its corresponding codeword.

3.2 Latent Dirichlet Allocation

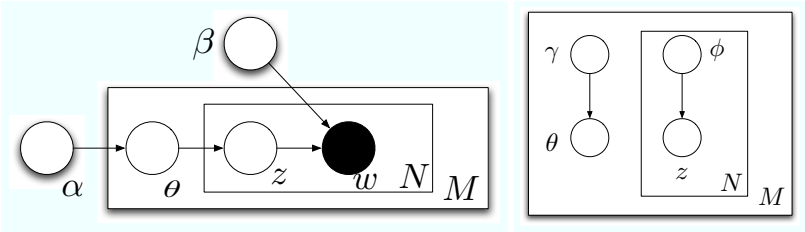
Our model is based the Latent Dirichlet Allocation (LDA) [2]. In the following, we briefly introduce LDA model using the terminology in our context.

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 Sect.3.1) indexed by $\{1, 2, ..., V\}$.

The LDA model assumes there are K underlying latent topics (i.e., action class labels) according to which video sequences are generated. Each topic is represented by a multinomial distribution over the $|V|$ motion words. A video sequence is generated by sampling a mixture of these topics, then sampling motion words conditioning on a particular topic. 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.



(a) (b)

Fig. 3. (a) Graphical representation of LDA model, adopted from Blei et al. [2]; (b) Graphical representation of the variational distribution.

The parameter θ indicates the mixing proportion of different actions labels in a particular video sequence. α is the parameter of a Dirichlet distribution that controls how the mixing proportions θ vary 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. Learning a LDA model from a collection of video sequences $D = \{\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_M\}$ involves finding α and β that maximize the log likelihood of the data $l(\alpha, \beta) = \sum_{d=1}^{M} \log P(\mathbf{w}_d|\alpha, \beta)$. This parameter estimation problem can be solved by the variational EM algorithm developed in Blei et al. [2].

3.3 Semi-Latent Dirichlet Allocation

In the original LDA, we are only given the word $(w_1, w_2, ..., w_N)$ 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. [2] 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 much compact representation for the documents, 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. [25] use a simpler method by classifying an image to a topic in which the latent topics of this document 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 corresponds to a particular "object class". Secondly, it is not clear how many "topics" to choose.

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 semi-supervised version of 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. 4. We should emphasize that the model in Fig. 4 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.

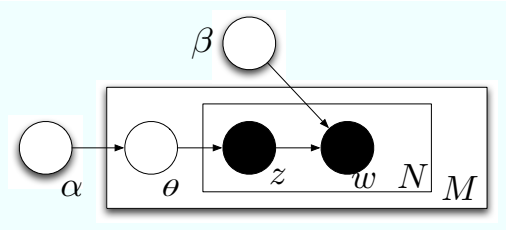


Fig. 4. Graphical representation of the Semi-Latent Dirichlet Allocation (S-LDA) for training. Note the difference from Fig. 3(a) is that z is observed in this case.

Our model has three major advantages over previous approaches of using a topic model for classification problems. First of all, choosing the right number of topics is trivial, since it is simply the number of class labels in the training sequences. Secondly, the training process of the S-LDA model is much easier than the original LDA. Thirdly, we can achieve much better recognition accuracy by taking advantage of the class labels (see Sect. 4).

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. 4), 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 estimating 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 for 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 estimation of the Dirichlet parameter α is a bit more involved. The Dirichlet distribution is a model of how topic mixing proportions θ vary among documents. This distribution has the form $p(\theta|\alpha) = \frac{\Gamma(\sum_k \alpha_k)}{\prod_k \Gamma(\alpha_k)} \prod_k \theta_k^{\alpha_k - 1}$. In order to estimate α, we first need to compute θ for each document $\mathbf{w} = (w_1, w_2, ..., w_N)$. Suppose the topics (i.e., the class labels of the words) of the document are $\mathbf{z} = (z_1, z_2, ..., z_N)$, then the i-th coordinate θ_i of θ can be calculated as $\theta_i = |\{i : z_j = i, j = 1, 2, ..., N\}|/N$. After we collect all the θ^t $(t = 1, 2, ..., M)$ values (as a notation convention, we use subscripts to denote coordinates of θ and superscripts to denote document numbers), the parameter α can be estimated from $\Theta = \{\theta^1, \theta^2, ..., \theta^M\}$ using generalized Newton Raphson iterations [18].

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 test video sequence is represented as $\mathbf{w} = (w_1, w_2, ..., w_N)$, i.e., there are N frames in the sequence, and the i-th frame is represented by the motion word w_i. Then, we need to calculate $p(z_i|\mathbf{w}, \alpha, \beta)$ $(i = 1, 2, ..., N)$. The frame w_i is classified to be action class k if $k = \text{argmax}_j p(z_i = j|\mathbf{w}, \alpha, \beta)$. Notice that we use $p(z_i|\mathbf{w})$ instead of $p(z_i|w_i)$ for classification. This reflects our assumption that the class label z_i not only depends on its corresponding word w_i, but also depends on the video sequence $\mathbf{w} = (w_1, w_2, ..., w_N)$ as a whole.

To calculate $p(z_i|\mathbf{w}, \alpha, \beta)$, we use the variational inference algorithm proposed in Blei et al. [2]. 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. 3(b). 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. [2] for detailed derivation):

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

$$\gamma_i = \alpha_i + \sum_{n=1}^{N} \phi_{ni} \tag{3}$$

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 document-specific. For a particular document $\mathbf{z}$, $\gamma^*(\mathbf{w})$ provides a representation of a document in the topic simplex. Also notice that $\text{Dir}(\gamma^*(\mathbf{w}))$ is the distribution from which the mixing proportion θ for the document $\mathbf{w}$ is drawn. We can imagine that if we

draw a sample $\theta \sim \text{Dir}(\gamma^*(\mathbf{w}))$, θ will tend to peak towards the true mixing proportion θ^* of topics for the document $\mathbf{w}$. So the true mixing proportion θ^* can be approximated by the empirical mean of a set of samples θ_i drawn from $\text{Dir}(\gamma^*(\mathbf{w}))$. 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 $\text{Mult}(\theta^*)$, 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.

3.5 Irregularity Detection in Video Sequences

An interesting application of our method is detecting irregularities (i.e., novelty) in video sequences. This has a lot of potential applications in surveillance and monitoring. Previous approaches to irregularity detection can be broadly classified into two classes: rule-based method and statistical methods [4]. Our method falls into the statistical methods, which try to learn a model of regularity from data, and infer about irregularity using the model.

There are various notions of "irregularity". For example, one possibility is to define all the actions that never appears in the training set to be "irregular". But in this paper, we focus on another case, where "irregularity" is defined by the composition of different actions, rather than the actions themselves. For example, loitering in a parking lot is composed of actions which by themselves are regular. But taken together, those regular actions form an unusual and suspicious behavior. This irregularity is characterized by the unique combination of regular actions. Other irregularity detection algorithms using only low-level cues (e.g. Boiman & Irani [4]) would not be able to identify it.

The application of our method to irregularity detection is quite straightforward. We first build our S-LDA model from a collection of training video sequences that are considered to be "regular", i.e., estimating the model parameters α and β using the method in Sect. 3.3. Given a new testing video sequence $\mathbf{w}$, we calculate the likelihood $l(\mathbf{w}; \alpha, \beta) = p(\mathbf{w}|\alpha, \beta)$ using the method in Sect. 3.4. If $\mathbf{w}$ is very different from those in the training set, i.e., it is not generated by the LDA model defined by α and β, it will probably have a very low likelihood under the model. So the likelihood of this new testing video sequence is an indicator of "irregularity". Lower likelihood means being more "irregular".

4 Experiments

We test our algorithm on two datasets: KTH human motion dataset [24] and soccer dataset [8].

Fig. 5. Representative frames in KTH dataset

4.1 Action Classification on KTH Dataset

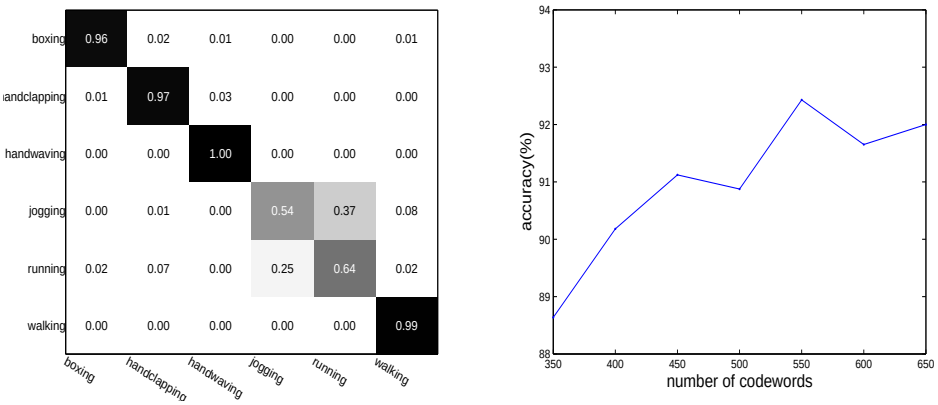
The KTH human motion dataset is one of the largest video datasets of human actions. It contains six types of human actions (walking, jogging, running, boxing, hand waving and hand clapping) performed several times by 25 subjects in four different scenarios: outdoors, outdoors with scale variation, outdoors with different clothes and indoors. Representative frames of this dataset are shown in Fig. 5.

We first run an automatic preprocessing step to track and stabilize the video sequences using the algorithm in Sabzmeydani &Mori [23], so that all the figures appear in the center of the field of view. We perform leave-one-out cross-validation on this dataset. For each run, we choose the video sequences of one subject as the test set, and build our model on the rest of the video sequences. We run the same process on each of the video sequence. For each run, we take the features obtained from Sect. 3.1 as the feature vectors. Then these feature vectors are quantized by k-medoid clustering to form the motion words. Since the number of feature vectors is huge, we randomly select a small number (about 30 frames) from each training video sequence for the k-mediod clustering.

The confusion matrix for the KTH dataset using 550 codewords is shown in Fig. 6(a). We can see that the algorithm correctly classifies most of actions. Most of the mistakes the algorithm makes are confusion between "running" and "jogging" actions. This is intuitively reasonable, since "running" and "jogging" are similar actions.

We also test the effect of the codebook size on the overall accuracy. The result is shown in Fig. 6(b). The best accuracy is achieved with 550 codewords, but is relatively stable.

We compare our results with previous approaches on the same dataset, as shown in Table 1. We would like to point out that the numbers in Table 1 are not directly comparable, since different approaches use different split of training and test data. In particular, the first three approaches use "leave-one-out" cross validation, while the remaining two approaches equally split the dataset into training, validation, and test sets. Nevertheless, Our method achieves better



(a) (b)

Fig. 6. (a) Confusion matrix for KTH dataset using 550 codewords (overall accuracy=92.43%). Horizontal rows are ground truth, and vertical columns are predictions. The action labels are "boxing","handclapping","handwaving","jogging","running","walking"; (b) classification accuracy vs. codebook size for KTH dataset

performance by a large margin. As a sanity check, we also run our experiment by equally splitting the dataset into training and test sets (our algorithm does not need a validation set), the recognition accuracy is similar.

Table 1. Comparison of different methods in terms of recognition accuracy on the KTH dataset

methods	**recognition accuracy**(%)
Our method	**92.43**
Niebles et al. [19]	81.50
Dollár et al. [7]	81.17
Schuldt et al. [24]	71.72
Ke et al. [14]	62.96

4.2 Action Classification on Soccer Dataset

The soccer dataset we use is from Efros et al. [8][1]. This 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

[1] Unfortunately, other datasets (tennis, ballet) used in this paper were not available.

center of the field of view. 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. 1(b).

Again, we perform leave-one-out cross-validation on the dataset. The confusion matrix using 350 codewords is shown in Fig. 7(a). The overall accuracy is 79.19%. Table 2 shows the main diagonal, compared with the main diagonal from Efros et al. [8], which used a k-nearest neighbor classifier based on the temporally smoothed motion feature vectors. We can see that our method performs better by a large margin for most of the class labels. Also, we can see that a lot of the mistakes made by our algorithm makes 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. In addition to achieving a higher accuracy, our algorithm has the added advantage that it is faster to classify a new video sequence, since we do not have to search over all the video sequences in the training set, as required by k-nearest neighbor classifiers.

We test the effect the codebook size on the overall accuracy. The result is shown in Fig. 7(b). The best accuracy peaks at around 350.

Table 2. Comparison of the main diagonal of the confusion matrix of our method and the method in Efros et al. [8] on the soccer dataset

	Our method	Efros et al. [8]
run left 45 °	0.64	0.67
run left	**0.77**	0.58
walk left	**1.00**	0.68
walk in/out	**0.86**	0.79
run in/out	**0.81**	0.59
walk right	**0.86**	0.68
run right	**0.71**	0.58
run right 45 °	0.66	0.66

4.3 Irregularity Detection

Since all the video sequences in the KTH dataset only contain a single action, we only test the irregularity detection of our algorithm on the soccer dataset. Our experimental setting is similar to that in Sect. 4.2. For each run, we choose one video sequence as the test set, and build our model from the remaining video sequences. Then we calculate the likelihood $p(\mathbf{w}|\alpha, \beta)$ of the testing video sequence under the built model. The likelihood value gives us some indication on how "irregular" this testing video sequence is, compared with the remaining video sequences.

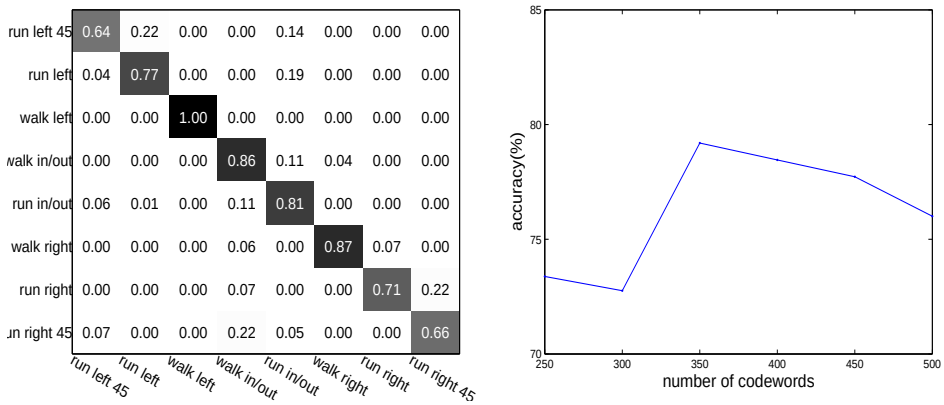


Fig. 7. (a)Confusion matrix for soccer dataset using 350 codewords (overall accuracy=79.19%). Horizontal rows are ground truth, and vertical columns are predictions. The action labels are "run left 45°","run left","walk left", "walk in/out", "run in/out", "walk right", "run right", "run right 45°"; (b) classification accuracy vs. codebook size for the soccer dataset

We repeat the above process for all the video sequences, then rank them according to the increasing order of their likelihood values $p(\mathbf{w}|\alpha, \beta)$. Under our assumption, the top few videos in the list should be considered to be "irregular".

Since there is no ground truth in this experiment, we can only report our results empirically. Table 3(a) shows the frame labels in the top five video sequences (i.e., the most "irregular" ones). We can see that the videos in Table 3(a) are in general "irregular". For example, they all involve a human figure runs/walks out of the scene. In fact, the combinations of the frame labels only appear once or twice in our training set. Table 3(b) shows the frame labels in the bottom five video sequences (i.e., the most "regular" or boring ones). They are obviously "boring" video sequences, since they only contain people running. It is interesting to see that the "boring" video sequences picked out by our algorithms are not necessarily the ones with a single action (see sequence 33 in Table 3(a) and sequence 2 in Table 3(b)).

5 Conclusion

We have presented a hierarchical probabilistic model (semi-latent Dirichlet allocation) for action recognition based on motion words, where each word corresponds to a frame in the video sequence. By naturally exploiting class labels of training data in our model, we are able to achieve much better results, compared with previous "bag-of-words" methods.

Of course, our method has its own limitations. For example, it requires a preprocessing stage of tracking and stabilizing human figures. However, we believe

Table 3. Results of irregularity detection: (a) top five "irregular" video sequences; (b) top five "regular" video sequences.

Sequence No.	frame labels
sequence 6	"run right" "run right 45°" "walk in/out" "walk right"
sequence 5	"walk right" "walk in/out"
sequence 9	"run left" "run left 45°" "run in/out" "run right 45°"
sequence 32	"run in/out" "walk in/out"
sequence 33	"walk in/out"

(a)

Sequence No.	frame labels
sequence 1	"run left"
sequence 2	"run left 45°" "run left"
sequence 29	"run left"
sequence 4	"run left"
sequence 3	"run left"

(b)

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 without much efforts.

References

1. Bissacco, A., Yang, M.H., Soatto, S.: Detecting humans via their pose. In: Advances in Neural Information Processing Systems 19 (NIPS). MIT Press (2007) 169–176
2. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. Journal of Machine Learning Research **3** (2003) 993–1022
3. Bobick, A.F., Davis, J.W.: The recognition of human movement using temporal templates. IEEE Transactions on Pattern Analysis and Machine Intelligence **23**(3) (2001) 257–267
4. Boiman, O., Irani, M.: Detecting irregularities in images and in video. In: IEEE International Conference on Computer Vision. Volume 1. (2005) 462–469
5. Bosch, A., Zisserman, A., Munoz, X.: Scene classification via pLSA. In: European Conference on Computer Vision. Volume 4. (2006) 517–530
6. Cutler, R., Davis, L.S.: Robust real-time periodic motion detection, analysis, and applications. IEEE Transactions on Pattern Analysis and Machine Intelligence **22**(8) (August 2000) 781–796
7. Dollár, P., Rabaud, V., Cottrell, G., Belongie, S.: Behavior recognition via sparse spatio-temporal features. In: ICCV'05 Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance. (2005)
8. Efros, A.A., Berg, A.C., Mori, G., Malik, J.: Recognizing action at a distance. In: IEEE International Conference on Computer Vision. Volume 2. (2003) 726–733
9. Fei-Fei, L., Fergus, R., Perona, P.: One-shot learning of object categories. IEEE Transactions on Pattern Analysis and Machine Intelligence **28**(4) (April 2006) 594–611

10. Fei-Fei, L., Perona, P.: A bayesian hierarchical model for learning natural scene categories. In: IEEE Conference on Computer Vision and Pattern Recognition. Volume 2. (2005) 524–531
11. Fergus, R., Fei-Fei, L., Perona, P., Zisserman, A.: Learning object categories from google's image search. In: IEEE International Conference on Computer Vision. Volume 2. (2005) 1816–1823
12. Grauman, K., Darrell, T.: The pyramid match kernel: Discriminative classification with sets of image features. In: IEEE International Conference on Computer Vision. Volume 2. (2005) 1458–1465
13. Hofmann, T.: Probabilistic latent semantic indexing. In: Proceedings of Twenty-Second Annual International Conference on Research and Development in Information Retrieval(SIGIR). (1999) 50–57
14. Ke, Y., Sukthankar, R., Hebert, M.: Efficient visual event detection using volumetric features. In: IEEE International Conference on Computer Vision. Volume 1. (2005) 166–173
15. Lazebnik, S., Schmid, C., Ponce, J.: A maximum entropy framework for part-based texture and object recognition. In: IEEE International Conference on Computer Vision. Volume 1. (2005) 832–838
16. Little, J.L., Boyd, J.E.: Recognizing people by their gait: The shape of motion. Videre **1**(2) (1998) 1–32
17. Lucas, B.D., Kanade, T.: An iterative image registration technique with an application to stereo vision. In: Proceedings of the DARPA Image Understanding Workshop. (April 1981) 121–130
18. Minka, T.P.: Estimating a Dirichlet distribution. Technical report, Massachusetts Institute of Technology (2000)
19. Niebles, J.C., Wang, H., Fei-Fei, L.: Unsupervised learning of human action categories using spatial-temporal words. In: British Machine Vision Conference. Volume 3. (2006) 1249–1258
20. Polana, R., Nelson, R.C.: Detection and recognition of periodic, non-rigid motion. International Journal of Computer Vision **23**(3) (June 1997) 261–282
21. Rao, C., Yilmaz, A., Shah, M.: View-invariant representation and recognition of actions. International Journal of Computer Vision **50**(2) (2002) 203–226
22. Russell, B.C., Efros, A.A., Sivic, J., Freeman, W.T., Zisserman, A.: Using multiple segmentations to discover objects and their extent in image collections. In: IEEE Conference on Computer Vision and Pattern Recognition. Volume 2. (2006) 1605–1614
23. Sabzmeydani, P., Mori, G.: Detecting pedestrians by learning shapelet features. In: IEEE Conference on Computer Vision and Pattern Recognition. (2007)
24. Schuldt, C., Laptev, L., Caputo, B.: Recognizing human actions: a local SVM approach. In: IEEE International Conference on Pattern Recognition. Volume 3. (2004) 32–36
25. Sivic, J., Russell, B.C., Efros, A.A., Zisserman, A., Freeman, W.T.: Discovering objects and their location in images. In: IEEE International Conference on Computer Vision. Volume 1. (2005) 370–377
26. Sullivan, J., Carlsson, S.: Recognizing and tracking human action. In: European Conference on Computer Vision LNCS 2352. Volume 1. (2002) 629–644
27. Zhong, H., Shi, J., Visontai, M.: Detecting unusual activity in video. In: IEEE Conference on Computer Vision and Pattern Recognition. Volume 2. (2004) 819–826