# Unsupervised Activity Perception in Crowded and Complicated Scenes Using Hierarchical Bayesian Models

Xiaogang Wang, Xiaoxu Ma, and Eric Grimson

Massachusetts Institute of Technology,

Computer Science and Artificial Intelligence Laboratory

## Abstract

We propose a novel unsupervised learning framework to model activities and interactions in crowded and complicated scenes. Under our framework hierarchical Bayesian models are used to connect three elements in visual surveillance: low-level visual features, simple "atomic" activities, and interactions. Atomic activities are modeled as distributions over low-level visual features, and multi-agent interactions are modeled as distributions over atomic activities. These models are learnt in an unsupervised way. Given a long video sequence, moving pixels are clustered into different atomic activities and short video clips are clustered into different interactions. In this paper, we propose three hierarchical Bayesian models, *Latent Dirichlet Allocation* (*LDA*) mixture model, *Hierarchical Dirichlet Process* (*HDP*) mixture model, and two dimensional *HDP* (*2D-HDP*) model. They advance existing language models, such as *LDA* [1] and *HDP* [2]. Directly using existing *LDA* and *HDP* models under our framework, only moving pixels can be clustered into atomic activities. Our models can cluster both moving pixels and video clips into atomic activities and interactions. *LDA* mixture model assumes that it is already known how many different types of atomic activities and interactions occur in the scene. *HDP* mixture model automatically decides the number of categories of atomic activities. *2D-HDP* automatically decides the numbers of categories of both atomic activities and interactions.

Our data sets are challenging video sequences from crowded traffic scenes and train station scenes with many kinds of activities co-occurring. Without tracking and human labeling effort, our framework completes many challenging visual surveillance tasks of board interest such as: (1) discovering and providing a summary of typical atomic activities and interactions happening in the scene; (2) segmenting long video sequences into different interactions; (3) segmenting motions into different activities; (4) detecting abnormality; and (5) supporting high-level queries on activities and interactions. In our work, these surveillance problems are formulated in a transparent, clean and probabilistic way compared with the ad hoc nature of many existing approaches.

## Index Terms

Hierarchical Bayesian model, Visual surveillance, Activity analysis, Abnormality detection, Video segmentation, Motion segmentation, Clustering, Dirichlet Process, Gibbs sampling, Variational inference.

## I. Introduction

The goal of this work is to understand activities and interactions in a crowded and complicated scene, e.g. a crowded traffic scene, a busy train station or a shopping mall (see Figure 1). In such scenes it is often not easy to track individual objects because of frequent occlusions

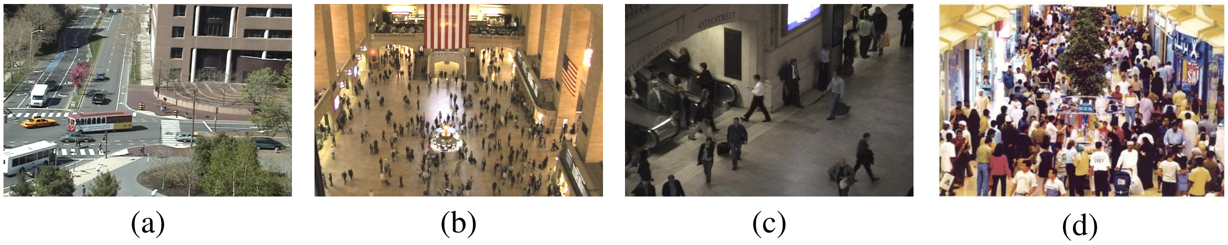

(a) (b) (c) (d)

Fig. 1. Examples of crowded and complicated scenes, such as traffic scenes, train stations, and shopping malls.

among objects, and many different types of activities often happen simultaneously. Nonetheless, we expect a visual surveillance system to: (1) discover typical types of single-agent activities (e.g. car makes a U-turn) and multi-agent interactions (e.g. vehicles stop waiting for pedestrians to cross the street) in these scenes, and provide a summary of them; (2) label short video clips in a long sequence as different interactions, and localize different activities involved in an interaction; (3) detect abnormal activities, e.g. pedestrians cross the road outside the crosswalk; and abnormal interactions, e.g. jay-walking (people cross the road while vehicles pass by); (4) support queries about interactions which have not yet been discovered by the system. Ideally, a system would learn models of the scene to answer such questions in an unsupervised way. These visual surveillance tasks become extremely difficult in crowded and complicated scenes. Most of the existing activity analysis approaches are expected to fail in these scenes (see more details in Section I-A).

To answer these challenges, we must determine how to model activities and interactions in crowded and complicated scenes. In this work, we refer to atomic activities, such as cars stopping, cars turning right, pedestrians crossing the street, etc., as the basic units for describing more complicated activities and interactions. An atomic activity usually causes temporally continuous motion and does not stop in the middle. Interaction is defined as a combination of different types of co-occurring atomic activities, such as a car stops to wait for a pedestrian passing by. However we do not consider interactions with complicated temporal logic, such as two people meet each other, walk together, and then separate. Instead, just model co-occurrence of atomic activities. Atomic activities and interactions are modeled using hierarchical Bayesian models under our framework.

Our system diagram is shown in Figure 2. We compute local motions (moving pixels) as our low-level visual features. This avoids difficult tracking problems in crowded scenes. We

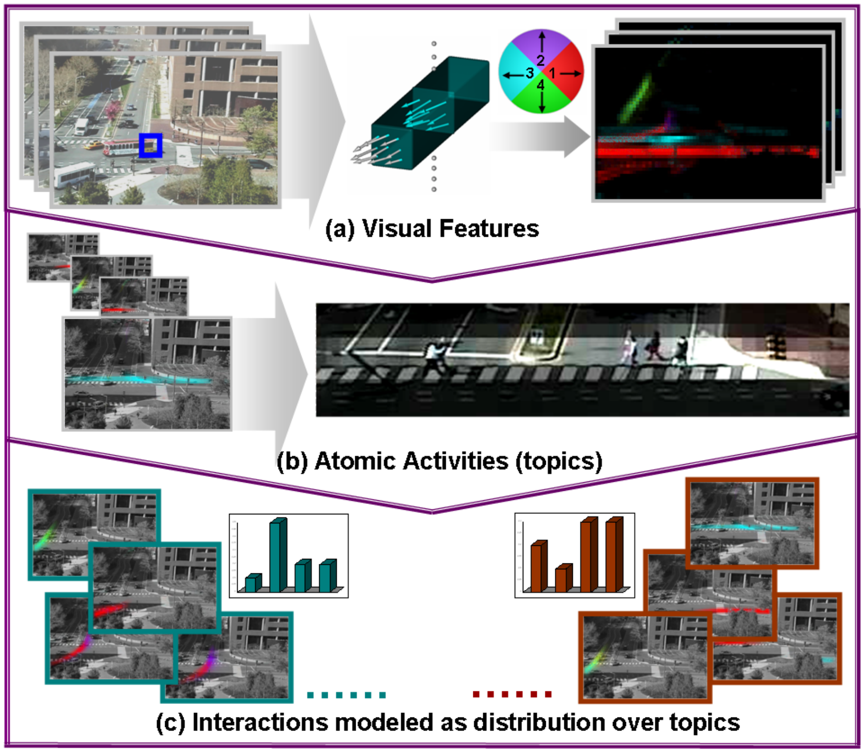

2
3
1
4
(a) Visual Features
(b) Atomic Activities (topics)
(c) Interactions modeled as distribution over topics

Fig. 2. Our framework connects: low-level visual features, atomic activities and interactions. (a) The video sequence is divided into short clips as documents. In each clip, local motions are quantized into visual words based on location and motion direction. The four quantized directions are represented by colors. Each video clip has a distribution over visual words. (b) Atomic activities (e.g. pedestrians cross the road) are discovered and modeled as distributions over visual words. (c) Each video clip is labeled by type of interaction, modeled as a distribution over atomic activities.

do not adopt global motion features ([3], [4]), because in these complicated scenes multiple different types of activities often occur simultaneously and we want to separate them. Eaching moving pixel is labeled by location and direction of motion to form our basic feature set. A long video sequence can be divided into many short video clips. Local motions caused by the same kind of atomic activities often co-occur in the same short video clips, since atomic activities cause temporally continuous motions. Interaction is a combination of atomic activities occurring in the same video clip. Thus there exist two hierarchical structures in both our data set (long video sequence → short video clips → moving pixels) and visual surveillance tasks (interactions → atomic activities). So it is natural to employ a hierarchical Bayesian approach to connect three elements in visual surveillance: low-level visual features, atomic activities and interactions. Atomic activities are modeled as distributions over low-level visual features, and interactions are modeled as distributions over atomic activities. Moving pixels are clustered into atomic activities and video clips are clustered into interactions. As explained in [5] a hierarchical Bayesian model learnt from a data set with hierarchical structure has the advantage of using enough parameters to fit the data well while avoiding overfitting problems, since it is able to use a population distribution to structure some dependence into the parameters. In our case, the same type of atomic activities repeatedly occur in different video clips. By sharing a common set of atomic activity models across different video clips, the models of atomic activities can be well learnt from enough data. On the other hand, atomic activities are used as components to further model more complicated interactions, which are clusters of video clips. This is a much more compact representation than directly clustering high dimensional motion feature vectors computed from video clips. Under hierarchical Bayesian models, surveillance tasks such as video segmentation, activity detection and abnormality detection are formulated in a transparent, clean and probabilistic way compared with the ad hoc nature of many existing approaches.

There are some hierarchical Bayesian models for language processing, such as *LDA* [1] and *HDP* [2], from which we can borrow. Under *LDA* and *HDP* models, words often co-existing in the same documents are clustered into the same topic. *HDP* is a nonparametric model and automatically decides the number of topics while *LDA* requires knowing that in advance. We perform word-document analysis on video sequences. Moving pixels are quantized into visual words and short video clips are treated as documents. Directly applying *LDA* and *HDP* to our problem, atomic activities (corresponding to topics) can be discovered and modeled, however modeling interactions is not straightforward, since these models cannot cluster documents. Although *LDA* and *HDP* allow inclusion of more hierarchical levels corresponding to groups of documents, they require first manually labeling documents into groups. For example, [2] modeled multiple corpora but required knowing to which corpus each document belonged; [6] used *LDA* for scene categorization, but had to label each image in the training set into different categories. These are supervised frameworks. We propose three novel hierarchical Bayesian models, *LDA* mixture model, *HDP* mixture model and two dimensional *HDP* (*2D-HDP*) model. They co-cluster words and documents in an unsupervised way. In the case of visual surveillance, this means we can learn atomic activities as well as interactions without supervision. In fact, the problems of clustering moving pixels into atomic activities and clustering video clips into interactions are closely related. The interaction category of a video clip provides a prior for possible activities happening in that clip. On the other hand, first clustering moving pixels into atomic activities provides an efficient representation for modeling interactions since it dramatically reduces the data dimensionality. We solve these two problems together under a co-clustering framework. *LDA* mixture model assumes that the number of different types of atomic activities and interactions happening in the scene is known. *HDP* mixture model automatically decides the number of categories of atomic activities. *2D-HDP* automatically decides the numbers of categories of both atomic activities and interactions.

### *A. Related Work*

Most existing approaches to activity analysis fall into two categories. In the first, objects of interest are first detected, tracked, and classified into different object categories. Then object tracks are used to model activities [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18], [19]. For example, Stauffer and Grimson [7] classified tracks into different activity categories based on the positions, speeds, moving directions, sizes and silhouettes of objects along the tracks. Wang and Grimson [9] used the modified Hausdorff distance to compare the distance between two tracks and clustered tracks into activities. Oliver and Pentland [8] used a coupled HMM to model the interaction between two tracks. Intille and Bobick [12] used a Bayesian network to analyze the strategies in a football game. Since it was hard to track objects in such a crowded scene, they manually marked tracks. With the help of tracking, the activity of one object can be separated from other co-occurring activities. However, tracking based approaches are sensitive to tracking errors. If tracking errors happen only at a few frames, the future track could be completely wrong. These approaches fail when object detection, tracking, and/or recognition do not work well, especially in crowded scenes. Many of these approaches are supervised. Some systems model primitive events, such as "move, stop, enter-area, turn-left", which are similar to our atomic activities, and use these primitives as components to model complicated activities and interactions [11], [20]. However, these primitive events were learnt from labeled training examples, or their parameters were manually specified. When switching to a new scene, new training samples must be labeled and parameters must be tuned or re-learnt.

The second kind of approaches [3], [4], [21], [22], [23], [24] directly use motion feature vectors instead of tracks to describe video clips. For example, Zelnik-Manor and Irani [4] modeled and clustered video clips using multi-resolution histograms. Zhong et. al. [3] also computed global motion histograms and did word-document analysis on video. However, their words were frames instead of moving pixels. They clustered video clips through the partition of a bipartite graph. Without object detection and tracking, a particular activity can not be separated from other activities simultaneously occurring in the same clip, as is common in crowded scenes. These approaches treat a video clip as an integral entity and flag the whole clip as normal or abnormal. They are often applied to simple data sets where there is only one kind of activity in a video clip. It is difficult for theese approaches to model both single-agent activities and multi-agent interactions. Although there are actions/events modeling approaches [25], [26], [27], [28], [29], which allowed one to detect and separate co-occurring activities, they are usually supervised. At the training stage, they required manually isolating activities or a training video clip only contained one kind of activity.

In computer vision, hierarchical Bayesian models have been applied to scene categorization [6], object recognition [30], [31], [32], and human action recognition [27]. [6], [32], [33], and [27] are supervised learning frameworks in the sense that they need to manually label the documents. The video clip in [27] usually contains a single activity and [27] did not model interactions among multiple objects. [30] and [31], which directly applied an LDA model, was unsupervised frameworks assuming a document contains only one major topic. These methods will not directly transfer to our problem where each document typically contains several topics. These approaches could not model interactions either.

Our approach avoids tracking in crowded scenes, using only local motion as features. It can separate co-occurring activities in the video clip by modeling activities and interactions. The whole learning procedure is unsupervised without manual labeling of video clips or local motions. The rest of this paper is organized as following. Section II describes how to compute the low-level visual features. Three novel hierarchical Bayesian models are proposed in Section III. Section IV explains how to employ these models to solve visual surveillance tasks and shows experimental results from a traffic scene and a train station scene. In Section V, we discuss the limitations and possible extensions of this work.

## II. Low-Level Visual Features

Our data sets are video sequences from far-field traffic scenes (Figure 1 (a)) and train station scenes (Figure 1 (c)) recorded by a fixed camera. There are myriads of activities and interactions in the video data. It also involves many challenging problems, such as lighting changes, occlusions, a variety of object types, object view changes and environmental effects.

We compute local motions as our low-level features. Moving pixels are detected in each frame as follows. We compute the intensity difference between two successive frames, on a pixel basis. If the difference at a pixel is above a threshold, that pixel is detected as a moving pixel. The motion direction at each moving pixel is obtained by computing optical flow [34]. The moving pixels are quantized according to a codebook, as follows. Each moving pixel has two features: position and direction of motion. To quantize position, the scene$(480 \times 720)$ is divided into cells of size 10 by 10. The motion of a moving pixel is quantized into four directions as shown in Figure 2(a). Hence the size of the codebook is $48 \times 72 \times 4$, and thus each detected moving pixel is assigned a word from the codebook based on rough position and motion direction. The whole video sequence is uniformally divided into non-overlapping short clips, each 10 seconds in length. In our framework, video clips are treated as documents and moving pixels are treated as words for word-document analysis as described in Section III.

## III. Hierarchical Bayesian Models

*LDA* [1] and *HDP* [2] were originally proposed as hierarchical Bayesian models for language processing. In these models, words that often co-exist in the same documents are clustered into the same topic. We extend these models by enabling clustering of both documents and words, thus finding co-occurring words (topics) and co-occurring topics (interactions). For far-field surveillance videos, words are quantized local motions of pixels; moving pixels that tend to co-occur in clips (or documents) are modeled as topics. Our goal is to infer the set of activities (or topics) from video by learning the distributions of features that co-occur, and to learn distributions of activities that co-occur, thus finding interactions. Three new hierarchical Bayesian models are proposed in this section: *LDA* mixture model, *HDP* mixture model, and *2D-HDP* model.

### A. LDA Mixture Model

Figure 3(a) shows the *LDA* model of [1]. Suppose the corpus has $M$ documents. Each document is modeled as a mixture of $K$ topics, where $K$ is assumed known. Each topic $k$ is modeled as a multinomial distribution $\beta_k = [\beta_{k1}, \ldots, \beta_{kW}]$ over a word vocabulary of size $W$. $\beta = \{\beta_k\}$. $\alpha = [\alpha_1, \ldots, \alpha_K]$ is a Dirichlet prior on the corpus. For each document $j$,

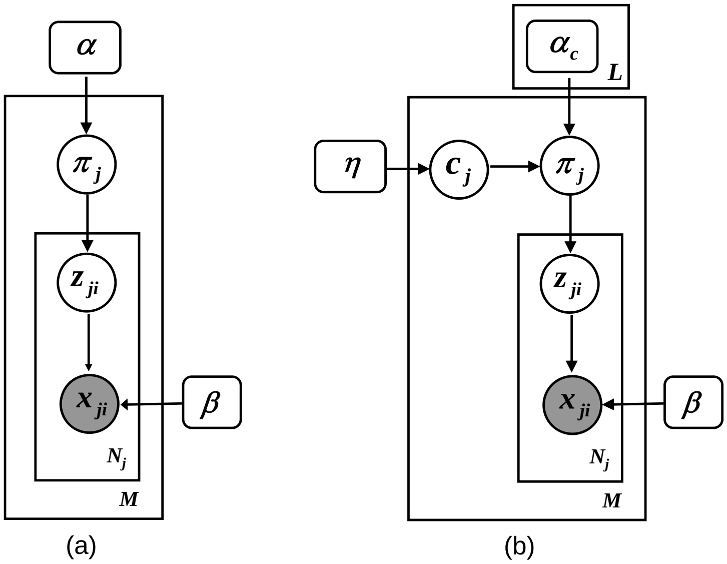


Fig. 3. (a) *LDA* model proposed in [1]; (b) our *LDA* mixture model.

a parameter $\pi_j = [\pi_{j1}, \ldots, \pi_{jK}]$ of the multinomial distribution over $K$ topics is drawn from Dirichlet distribution $Dir(\pi_j|\alpha)$. For each word $i$ in document $j$, a topic label $z_{ji} = k$ is drawn with probability $\pi_{jk}$, and word $x_{ji}$ is drawn from a discrete distribution given by $\beta_{z_{ji}}$. $\pi_j$ and $z_{ji}$ are hidden variables. $\alpha$ and $\beta$ are hyperparameters to be optimized. Given $\alpha$ and $\beta$, the joint distribution of topic mixture $\pi_j$, topics $\mathbf{z}_j = \{z_{ji}\}$, and words $\mathbf{x}_j = \{x_{ji}\}$ is:

$$p(\mathbf{x}_j, \mathbf{z}_j, \pi_j|\alpha, \beta) = p(\pi_j|\alpha) \prod_{i=1}^{N_j} p(z_{ji}|\pi_j) p(x_{ji}|z_{ji}, \beta)$$

$$= \frac{\Gamma(\sum_{k=1}^{K} \alpha_k)}{\prod_{k=1}^{K} \Gamma(\alpha_k)} \pi_{j1}^{\alpha_1 - 1} \cdots \pi_{jK}^{\alpha_K - 1} \prod_{i=1}^{N_j} \pi_{j z_{ji}} \beta_{z_{ji} x_{ji}} \quad (1)$$

where $N_j$ is the number of words in document $j$. Unfortunately, the marginal likelihood $p(\mathbf{x}_j|\alpha, \beta)$ and thus the posterior distribution $p(\pi_j, \mathbf{z}_j|\alpha, \beta)$ are intractable for exact inference. Thus in [1], a Variational Bayes (VB) inference algorithm used a family of variational distributions:

$$q(\pi_j, \mathbf{z}_j|\gamma_j, \phi_j) = q(\pi_j|\gamma_j) \prod_{i=1}^{N_j} q(z_{ji}|\phi_{ji}) \quad (2)$$

to approximate $p(\pi_j, \mathbf{z}_j|\alpha, \beta)$, where the Dirichlet parameter $\gamma_j$ and multinomial parameters $\{\phi_{ji}\}$ are free variational parameters. The optimal $(\gamma_j, \phi_j)$ is computed by finding a tight lower bound on $\log p(\mathbf{x}_j|\alpha, \beta)$.

This LDA model in [1] does not model clusters of documents. All the documents share the same Dirichlet prior $\alpha$. In activity analysis, we assume that video clips (documents) of the same type of interaction would include a similar set of atomic activities (topics), so they could be grouped into the same cluster and share the same prior over topics. Our *LDA* model is shown in Figure 3(b). The $M$ documents in the corpus will be grouped into $L$ clusters. Each cluster $c$ has its own Dirichlet prior $\alpha_c$. For a document $j$, the cluster label $c_j$ is first drawn from discrete distribution $\eta$, and $\pi_j$ is drawn from $Dir(\pi_j|\alpha_{c_j})$. Given $\{\alpha_c\}$, $\beta$, and $\eta$, the joint distribution of hidden variables $c_j$, $\pi_j$, $\mathbf{z}_j$ and observed words $\mathbf{x}_j$ is

$$p(\mathbf{x}_j, \mathbf{z}_j, \pi_j, c_j|\{\alpha_c\}, \beta, \eta) = p(c_j|\eta)p(\pi_j|\alpha_{c_j})\prod_{i=1}^{N} p(z_{ji}|\pi_j)p(x_{ji}|z_{ji}, \beta) \tag{3}$$

The marginal log likelihood of document $j$ is:

$$\log p(\mathbf{x}_j|\{\alpha_c\}, \beta, \eta) = \log \sum_{c_j=1}^{L} p(c_j|\eta)p(\mathbf{x}_j|\alpha_{c_j}, \beta) \tag{4}$$

Using VB [1], $\log p(\mathbf{x}_j|\alpha_{c_j}, \beta)$ can be approximated by a tight lower bound $L_1(\gamma_{jc_j}, \phi_{jc_j}; \alpha_{c_j}, \beta)$,

$$\begin{aligned}
&\log p(\mathbf{x}_j|\alpha_{c_j}, \beta) \\
&= \log \int_{\pi_j} \sum_{\mathbf{z}_j} p(\pi_j, \mathbf{z}_j, \mathbf{x}_j|\alpha_{c_j}, \beta) d\pi_j \\
&= \log \int_{\pi_j} \sum_{\mathbf{z}_j} \frac{p(\pi_j, \mathbf{z}_j, \mathbf{x}_j|\alpha_{c_j}, \beta) q(\mathbf{z}_j, \pi_j|\gamma_{jc_j}, \phi_{jc_j})}{q(\mathbf{z}_j, \pi_j|\gamma_{jc_j}, \phi_{jc_j})} d\pi_j \\
&\geq \int_{\pi_j} \sum_{\mathbf{z}_j} q(\mathbf{z}_j, \pi_j|\gamma_{jc_j}, \phi_{jc_j}) \log p(\mathbf{x}_j, \mathbf{z}_j, \pi_j|\alpha_{c_j}, \beta) d\pi_j \\
&- \int_{\pi_j} \sum_{\mathbf{z}_j} q(\mathbf{z}_j, \pi_j|\gamma_{jc_j}, \phi_{jc_j}) \log q(\mathbf{z}_j, \pi_j|\gamma_{jc_j}, \phi_{jc_j}) d\pi_j \\
&= L_1(\gamma_{jc_j}, \phi_{jc_j}; \alpha_{c_j}, \beta).
\end{aligned} \tag{5}$$

However because of the marginalization over $c_j$, hyperparameters are still coupled even using VB. So we use both EM and VB to estimate hyperparameters. After using VB to compute the lower bound of $\log p(\mathbf{x}_j|\alpha_{c_j}, \beta)$, an averaging distribution $q(c_j|\gamma_{jc_j}, \phi_{jc_j})$ can provide a further lower bound on the log likelihood,

$$\begin{aligned}\log p(\mathbf{x}_j|\{\alpha_c\},\beta,\eta) \geq \log \sum_{c_j=1}^{L} p(c_j|\eta) e^{L_1(\gamma_{jc_j},\phi_{jc_j};\alpha_{c_j},\beta)}\\ =\log \sum_{c_j=1}^{L} q(c_j|\gamma_{jc_j},\phi_{jc_j}) \frac{p(c_j|\eta)e^{L_1(\gamma_{jc_j},\phi_{jc_j};\alpha_{c_j},\beta)}}{q(c_j|\gamma_{jc_j},\alpha_{jc_j})}\\ \geq \sum_{c_j=1}^{L} q(c_j|\gamma_{jc_j},\phi_{jc_j}) \left[\log p(c_j|\eta) + L_1(\gamma_{jc_j},\phi_{jc_j};\alpha_{c_j},\beta)\right]\\ -\sum_{c_j=1}^{L} q(c_j|\gamma_{jc_j},\phi_{jc_j}) \log q(c_j|\gamma_{jc_j},\phi_{jc_j})\\ =L_2(q(c_j|\gamma_{jc_j},\phi_{jc_j}),\{\alpha_c\},\beta,\eta) \qquad (6)\end{aligned}$$

$L_2$ is maximized when choosing

$$q(c_j|\gamma_{jc_j},\phi_{jc_j}) = \frac{p(c_j|\eta)e^{L_1(\gamma_{jc_j},\phi_{jc_j};\alpha_{c_j},\beta)}}{\sum_{c_j} p(c_j|\eta)e^{L_1(\gamma_{jc_j},\phi_{jc_j};\alpha_{c_j},\beta)}}. \qquad (7)$$

Our EM algorithm for hyperparameters estimation is:

1) For each document $j$ and cluster $c_j$, find the optimal values of the variational parameters $\{\gamma^*_{j,c_j},\phi^*_{j,c_j} : j=1,\ldots,M, c_j=1,\ldots,L\}$ to maximize $L_1$ (using VB [1]).
2) Compute $q(c_j|\gamma^*_{jc_j},\phi^*_{jc_j})$ using (7) to maximize $L_2$.
3) Maximize $L_2$ with respect to $\{\alpha_c\}$, $\beta$, and $\eta$. $\beta$ and $\eta$ are optimized by setting the first derivative to zero,

$$\eta_c \propto \sum_{j=1}^{M} q(c_j=c|\gamma^*_{jc},\phi^*_{jc}) \qquad (8)$$

$$\beta_{kw} \propto \sum_{j=1}^{M}\sum_{c_j=1}^{L} q(c_j|\gamma^*_{jc_j},\phi^*_{jc_j}) \left[\sum_{i=1}^{N} \phi^*_{jc_jik} x^w_{ji}\right] \qquad (9)$$

where $x^w_{ji}=1$ if $x_{ji}=w$, otherwise it is $0$. The $\{\alpha_c\}$ are optimized using a Newton-Raphson algorithm. The first and second derivatives are:

$$\frac{\partial L_2}{\partial \alpha_{ck}} = \sum_{j=1}^{M} q(c_j=c|\gamma_{jc},\phi_{jc})[\Psi(\sum_{k=1}^{K}\alpha_{ck}) - \Psi(\alpha_{ck}) + \Psi(\gamma_{jck}) - \Psi(\sum_{j=1}^{k}\gamma_{jck})] \qquad (10)$$

$$\frac{\partial^2 L_2}{\partial \alpha_{ck_1}\alpha_{ck_2}} = \sum_{j=1}^{M} q(c_j=c|\gamma_{jc},\phi_{jc})[\Psi'(\sum_{k=1}^{K}\alpha_{ck}) - \delta(k_1,k_2)\Psi'(\alpha_{ck_1})] \qquad (11)$$

where $\Psi$ is the first derivative of log Gamma function.

$L_2$ monotonously increases after each iteration.

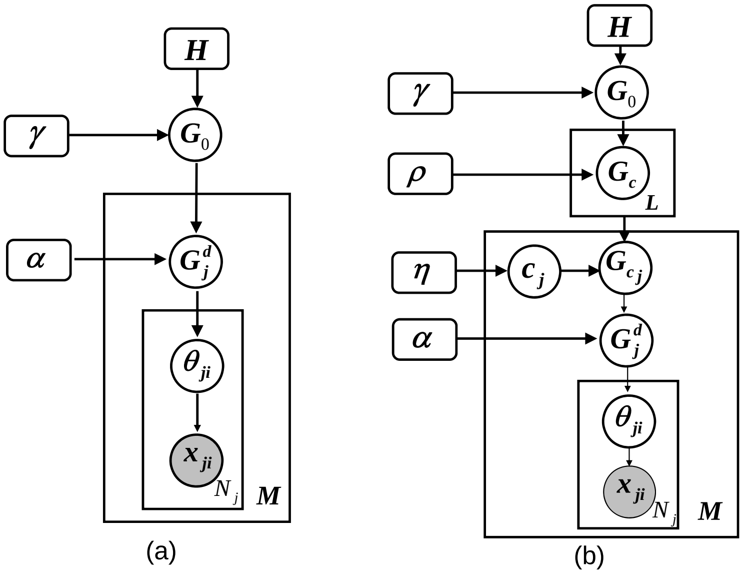


Fig. 4. (a) *HDP* model proposed in [2]; (b) our *HDP* mixture model.

### B. HDP Mixture Model

*HDP* is a nonparametric hierarchical Bayesian model and automatically decides the number of topics. The *HDP* model proposed in [2] is shown in Figure 4 (a). A global random measure $G_0$ is distributed as a Dirichlet Process with concentration parameter $\lambda$ and base probability measure $H$ (*H* is a Dirichlet prior in our case):

$$G_0|\gamma, H \sim DP(\gamma, H).$$

$G_0$ can be expressed using a stick-breaking representation,

$$G_0 = \sum_{k=1}^{\infty} \pi_{0k} \delta_{\phi_k}, \tag{12}$$

where $\{\phi_k\}$ are parameters of multinomial distributions and $\delta_{\phi_k}(\cdot)$ is the Delta function with support point at $\phi_k$. $\{\phi_k\}$ and $\{\pi_{0k}\}$ are called locations and masses. $\{\phi_k\}$ model topics of words. $\{\pi_{0k}\}$ are mixtures over topics. They are sampled from a stick-breaking construction.

$$\phi_k \sim H,$$
$$\pi_{0k} = \pi'_{0k} \prod_{l=1}^{k-1} (1 - \pi'_{0l}),$$
$$\pi'_{0k} \sim Beta(1, \lambda).$$

$G_0$ is a prior distribution over the whole corpus. For each document $j$, a random measure $G_j^d$ is drawn from a Dirichlet process with concentration parameter $\alpha$ and base probability measure $G_0$:

$$G_j^d|\alpha, G_0 \sim DP(\alpha, G_0).$$

Each $G_j^d$ has support at the same locations $\{\phi_k\}_{k=1}^{\infty}$ as $G_0$, i.e. all the documents share the same set of topics, and can be written as

$$G_j^d = \sum_{k=1}^{\infty} \pi_{jk}\delta_{\phi_k}.$$

$G_j^d$ is a prior distribution of all the words in document $j$. For each word $i$ in document $j$, a topic $\theta_{ji}$ is drawn from $G_j^d$ ($\theta_{ji}$ is sampled as one of the $\phi_k$'s). Word $x_{ji}$ is drawn from discrete distribution $Discrete(\theta_{ji})$. In [2], Gibbs sampling schemes were used to do inference under an *HDP* model.

In our *HDP* mixture model, as shown in Figure 4 (b), clusters of documents are modeled and each cluster $c$ has a random probability measure $G_c$. $G_c$ is drawn from Dirichlet process $DP(\rho, G_0)$. For each document $j$, a cluster label $c_j$ is first drawn from discrete distribution $p(c_j|\eta)$. Document $j$ chooses $G_{c_j}$ as the base probability measure and draws its own $G_j^d$ from Dirichlet process $G_j^d \sim DP(\alpha, G_{c_j})$. We also use Gibbs sampling for inference. In our *HDP* mixture model, there are two kinds of hidden variables to be sampled: (1) variables $\mathbf{z} = \{z_{ij}\}$ assigning words to topics, base distributions $G_0$, $\{G_k\}$; and (2) cluster label $c_j$. The key issue to be solved in this paper is how to sample $c_j$. Given $c_j$ fixed, the first kind of variables can be sampled using the same scheme described in [2]. We will not repeat the details in this paper. We focus on the step of sampling $c_j$, which is the new part of our model compared with *HDP* in [2].

At some sampling iteration, suppose that there have been $K$ topics, $\{\phi_k\}_{k=1}^{K}$, generated and assigned to the words in the corpus ($K$ is variable during the sampling procedure). $G_0$, $G_c$, and $G_j^d$ can be expressed as,

$$G_0 = \sum_{k=1}^{K} \pi_{0k}\delta_{\phi_k} + \pi_{0u}G_{0u},$$

$$G_c = \sum_{k=1}^{K} \pi_{ck}\delta_{\phi_k} + \pi_{cu}G_{cu},$$

$$G_j^d = \sum_{k=1}^{K} \omega_{jk}\delta_{\phi_k} + \omega_{ju}G_{ju}^d,$$

where $G_{0u}$, $G_{cu}$, and $G_{ju}^d$ are distributed as Dirichlet process $DP(\gamma, H)$. Note that the prior over the corpus ($G_0$), a cluster of document ($G_c$) and a document $G_j^d$ share the same set of topics $\{\phi_k\}$. But they have different mixtures over topics.

Using the sampling schemes in [2], topic mixtures $\pi_0 = \{\pi_{01}, \ldots, \pi_{0K}, \pi_{0u}\}$, $\pi_c = \{\pi_{c1}, \ldots, \pi_{cK}, \pi_{cu}\}$ are sampled, while $\{\phi_k\}$, $G_{0u}$, $G_{cu}$, $G_{ju}^d$, and $\omega_j^d = \{\omega_{j1}, \ldots, \omega_{jK}, \omega_{ju}\}$ can be integrated out without sampling. In order to sample the cluster label $c_j$ of document $j$, the posterior $p(c_j = c|(m_{j1}, \ldots, m_{jK}), \pi_0, \{\pi_c\})$ has to be computed where $m_{jk}$ is the number of words assigned to topic $k$ in document $j$ and is computable from **z**.

$$\begin{aligned}
&p(c_j = c|(m_{j1}, \ldots, m_{jK}), \pi_0, \{\pi_c\})\\
\propto& p(m_{j1}, \ldots, m_{jK}|\pi_c)p(c_j = c)\\
=&\eta_c \int p(m_{j1}, \ldots, m_{jK}|\omega_j^d)p(\omega_j^d|\pi_c)d\omega_j^d
\end{aligned}$$

$p(m_{j1}, \ldots, m_{jK}|\omega_j^d)$ is a multinomial distribution. Since $G_j^d$ is drawn from $DP(\alpha, G_c)$, $p(\omega_j^d|\pi_c)$ is a Dirichlet distribution $Dir(\omega_j^d|\alpha \cdot \pi_c)$. Thus we have

$$\begin{aligned}
&p(c_j = c|(m_{j1}, \ldots, m_{jK}), \pi_0, \{\pi_c\})\\
\propto&\eta_c \int \frac{\Gamma(\alpha\pi_{cu} + \alpha\sum_{k=1}^{K}\pi_{ck})}{\Gamma(\alpha\pi_{cu})\prod_{k=1}^{K}\Gamma(\alpha\pi_{ck})}\omega_{ju}^{\alpha\pi_{cu}-1}\prod_{k=1}^{K}\omega_{jk}^{\alpha\pi_{ck}+m_{jk}-1}d\omega_j^d\\
\propto&\frac{\Gamma(\alpha\pi_{cu} + \alpha\sum_{k=1}^{K}\pi_{ck})}{\Gamma(\alpha\pi_{cu})\prod_{k=1}^{K}\Gamma(\alpha\pi_{ck})}\frac{\Gamma(\alpha\pi_{cu})\prod_{k=1}^{K}\Gamma(\alpha\pi_{ck}+m_{jk})}{\Gamma(\alpha\pi_{cu}+\sum_{k=1}^{K}(\alpha\pi_{ck}+m_{jk}))}\\
=&\eta_c\frac{\Gamma(\alpha)}{\Gamma(\alpha+N_j)}\frac{\prod_{k=1}^{K}\Gamma(\alpha\cdot\pi_{ck}+m_{jk})}{\prod_{k=1}^{K}\Gamma(\alpha\cdot\pi_{ck})}\\
\propto&\eta_c\frac{\prod_{k=1}^{K}\Gamma(\alpha\cdot\pi_{ck}+m_{jk})}{\prod_{k=1}^{K}\Gamma(\alpha\cdot\pi_{ck})}.
\end{aligned}\qquad(13)$$

where $\Gamma$ is the Gamma function.

So the Gibbs sampling procedure repeats the following two steps alternatively at every iteration:

1) given $\{c_j\}$, sample **z**, $\pi_0$, and $\{\pi_c\}$ using the schemes in [2];
2) given $\mathbf{z}, \pi_0$, and $\{\pi_c\}$, sample cluster labels $\{c_j\}$ using posterior Eq 13.

In this section, we assume that the concentration parameters $\gamma$, $\rho$, and $\alpha$ are fixed. In actual implementation, we give them a vague gamma prior $Gamma(1,1)$ and sample them using the scheme proposed in [2]. Thus these concentration parameters are sampled from a broad distribution instead of being fixed at a particular point.

### *C. 2D-HDP*

In this section, we propose a two dimensional *HDP* model which automatically decides both the number of word topics and the number of document clusters. In addition to the hierarchical Dirichlet process which models the word topics, there is another hierarchical Dirichlet process modeling the clusters of document. Hence we call this a *2D HDP* model. The graphical model of *HDP* is shown in Figure 5. In *HDP* mixture model, each document $j$ has a prior $G_{c_j}$ drawn from a finite mixture $\{G_c\}_{c=1}^{L}$. In the *2D-HDP* model, $G_{c_j}$ is drawn from an infinite mixture,

$$Q = \sum_{c=1}^{\infty} \epsilon_c \delta_{G_c} \tag{14}$$

Notice that $G_c$ itself is a random distribution with infinite parameters. When a Dirichlet process was first developed by Ferguson [35], the location parameters (such as $\phi_k$ in Eq 12) could only be scalars or vectors. MacEachern [36] made an important generalization and proposed the *Dependent Dirichlet Process* (*DDP*). *DDP* replaces the locations in the stick-breaking representation with stochastic processes and introduces dependence in a collection of distributions. The parameters $\{(\pi_{ck}, \phi_{ck})\}_{k=1}^{\infty}$ of $G_c$ can be treated as a stochastic process with index $k$. $Q$ can be treated as a set of dependent distributions, $Q = \{Q_k = \sum_{c=1}^{\infty} \epsilon_c \delta_{(\pi_{ck}, \phi_{ck})}\}_{k=1}^{\infty}$. So we can generate $Q$ through *DDP*.

As shown in Figure 5 (a), $Q$ is sampled from $DDP(\mu, \Psi)$. $\mu$ is the concentration parameter, and

$$\epsilon_c = \epsilon'_c \prod_{l=1}^{c-1} (1 - \epsilon'_l),$$

$$\epsilon'_c \sim Beta(1, \mu).$$

As shown in Figure 5 (b), $\Psi = DP(\rho, G_0)$ is a Dirichlet process, and

$$G_c \sim DP(\rho, G_0).$$

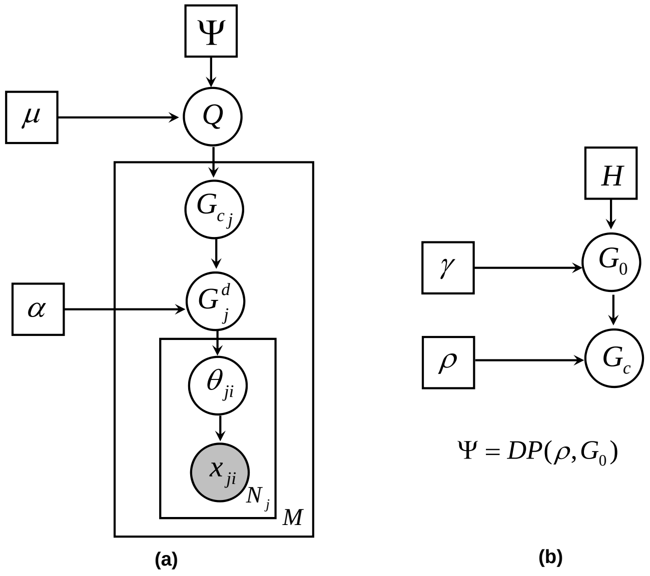


Fig. 5. The graphical model of *2D-HDP*. $Q = \sum_{c=1}^{\infty} \epsilon_c \delta_{G_c}$ and $G_0 = \sum_{k=1}^{\infty} \pi_{0k} \delta_{\phi_k}$ are two infinite mixtures modeling clusters of documents and words respectively. $Q$ is generated from $DDP(\mu, \Psi)$. $\Psi = DP(\rho, G_0)$ is a Dirichlet process.

Similar to the *HDP* mixture model in Figure 4 (b), $G_0 \sim DP(\lambda, H)$ is the prior over the whole corpus and generates topics shared by all of the words. $\{G_c\}_{c=1}^{\infty}$ all have the same topics in $G_0$, i.e. $\phi_{ck} = \phi_k$. However they have different mixtures $\{\pi_{ck}\}_{k=1}^{\infty}$ over these topics.

Each document $j$ samples a probability measure $G_{c_j}$ from $Q$ as its prior. Different documents may choose the same prior $G_c$, thus they form one cluster. So in *2D-HDP*, the two infinite mixtures $Q$ and $G_0$ model the clusters of documents and words respectively. The following generative procedure is the same as *HDP* mixture model. Document $j$ generates its own probability measure $G_j^d$ from $G_j^d \sim DP(\alpha, G_{c_j})$. Word $i$ in document $j$ samples topic $\phi_k$ from $G_j^d$ and sample its word value from $Discrete(\phi_k)$.

Gibbs sampling was also used to inference and learning on *2D-HDP*. The Gibbs sampling procedure can be divided into two steps:

1) given the cluster assignment $\{c_j\}$ of documents fixed, sample the word topic assignment **z**, masses $\pi_0$ and $\pi_c$ on topics using the schemes in [2];
2) given **z**, masses $\pi_0$ and $\pi_c$, sample the cluster assignment $\{c_j\}$ of documents. $c_j$ can be assigned to one of the existing clusters or a new cluster. We use the Chinese restaurant franchise for sampling. See details in the Appendix.

### *D. Discussion on the words-documents co-clustering framework*

We propose three words-documents co-clustering models. Readers may ask why we need a co-clustering framework? Can we first cluster words into topics and then cluster documents based on their distributions over topics, or solve the two problems separately? In visual surveillance applications, the issue is about simultaneously modeling activities and interactions. In language processing literature, there has been considerable work dealing with words clustering [37], [1], [2] and document clustering [38], [39], [40] separately. Dhillon [41] showed the duality of words and documents clustering: "word clustering induces document clustering while document clustering induces words clustering". Information on the category of documents helps to solve the ambiguity of word meaning and vice versus. Thus a co-clustering framework can solve the two closely related problems in a better way. Dhillon [41] co-clustered words and documents by partitioning a bipartite spectral graph with words and documents as vertices. However, one cluster of documents only corresponded to one cluster of words. [37], [1] showed that one document may contain several topics. In a visual surveillance data set, one video clip may contain several atomic activities. Our co-clustering algorithms based on hierarchical Bayesian models can better solve these problems.

### *E. Example of synthetic data*

We use an example of synthetic data to demonstrate the strength of our hierarchical Bayesian models (see Figure 6). The toy data is similar as that used in [42]. The word vocabulary is a set of $5 \times 5$ cells. There are 10 topics with distributions over horizontal bars and vertical bars (Figure 6 (a)), i.e., words tend to co-occur along the same row or column, but not arbitrarily. The document is represented by a image with 25 pixels in a $5 \times 5$ grid. Each pixel is a word and the intensity of a pixel is the frequency of the word. If we generate documents by randomly choosing several topics from the ten, adding noise to the bar distributions, and sample words from these bars, there are only two levels of structures (topics and words) in the data and the *HDP* model in [2] can perfectly discover the 10 topics. However, in our experiments in Figure 6, we add one more level, clusters of documents, to the data. Documents are from two clusters: a vertical-bars cluster and a horizontal-bars cluster. If a document is from the vertical-bars cluster, it randomly combines several vertical bar topics and sample words from them, otherwise, it randomly combines horizontal bar topics. As seen in Figure 6 (c), *HDP* in [2] has much worse performance on this data. There are two kinds of correlation among words: if words are on the same bar, they often co-exist in the same documents; if words are all on horizontal bars or vertical bars, they are also likely to be in the same documents. It is improper to use a two-level *HDP* to model data with a three-level structure. 15 topics are discovered and many of the topics include more than one bar. Using our *HDP* mixture model and *2D-HDP* model to co-cluster words and documents, the 10 topics are discovered nearly perfectly as shown in Figure 6(d). In the meanwhile, the documents are grouped into two clusters as shown in Figure 6 (e) and (f). The topic mixtures $\pi_1$ and $\pi_2$ of these two clusters are shown in Figure 6 (g). $\pi_1$ only has large weights on horizontal bar topics while $\pi_2$ only has large weights on vertical bar topics. Thus our approach recovers common topics (i.e. words that co-occur) and common documents (i.e. topics that co-occur). For *2D-HDP*, we tried different numbers of document clusters as initialization, and found it always converges to two clusters.

## IV. Visual Surveillance Applications and Experimental Results

After computing the low-level visual features as described in Section II, we divide our video sequence into 10 second long clips, each treated as a document, and feed these documents to the hierarchical Bayesian models described in Section III. In this section, we explain how to use the results from hierarchical Bayesian models for activity analysis. We will mainly show results from *2D-HDP*, since it automatically decides the number of word topics and the number of document clusters, while *LDA* mixture model and *HDP* mixture model need to know those in advance. However, if the number of word topics and the number of document clusters are properly set in *LDA* mixture model and *HDP* mixture model, they provide very similar results. Most of the experimental results are from a traffic scene. Some results from a train station scene is shown at the end of this section. We have video results which can be found be the supplementary material.

### A. Discover Atomic Activities

In visual surveillance, people often ask “what are the typical activities and interactions in this scene?” The parameters estimated by our hierarchical Bayesian models provide a good answer to this question.

As we explained in Section I, an atomic activity usually causes temporally continuous motion and does not stop in the middle. So the motions caused by the same kind of atomic activity often

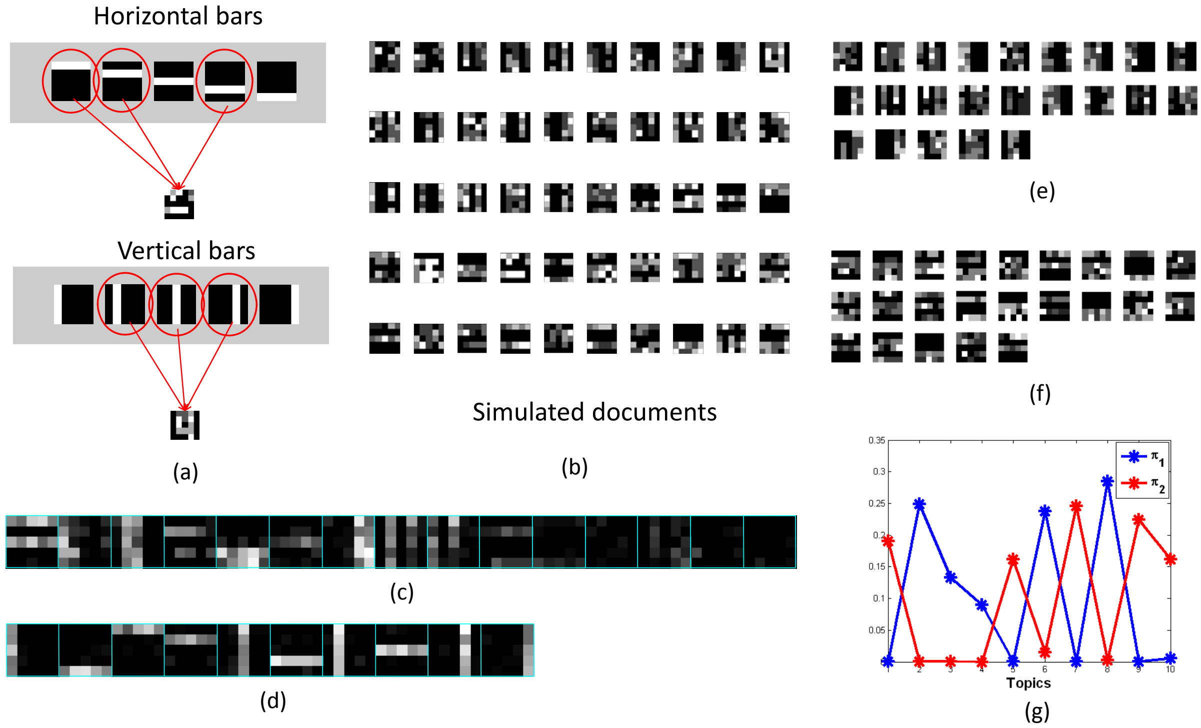


Fig. 6. Experiment on synthetic data. (a) There are ten topics with distributions along horizontal bars and vertical bars. A synthetic document can be generated in one of the two ways. It randomly randomly combines several vertical bar topics and sample words from them or randomly combines several horizontal bar topics. (b) The simulated documents. (c) Topic distributions learnt by the *HDP* model in [2]. (d) Topics distributions learnt by the *2D-HDP* model. Documents are grouped into two clusters shown in (e) and (f). (g) Topic mixtures of two clusters $\pi_1$ and $\pi_2$.

co-occur in the same video clip. Since the moving pixels are treated as words in our hierarchical Bayesian models, the topics of words are actually a summary of typical atomic activities in the scene. Each topic has a multinomial distribution over words (i.e., visual motions), specified by $\beta$ in *LDA* mixture model and $\{\phi_k\}$ in our *HDP* models. ($\phi_k$ can be easily estimated given the words assigned to topic $k$ after sampling).

Our *HDP* models automatically discovered 29 atomic activities in the traffic scene. In Figure 7, we show the motion distributions of these topics. The topics are sorted by size (the number of words assigned to the topic) from large to small. The numbers of moving pixels assigned to topics are shown in Figure 8. Topic 2 explains vehicles making a right turn. Topics 5, 14, and 20 explain vehicles making left turns. Topics 6 and 9 explain vehicles crossing road *d*, but along different lanes. Topics 1 and 4 explain "vehicles pass road *d* from left to right". This activity is

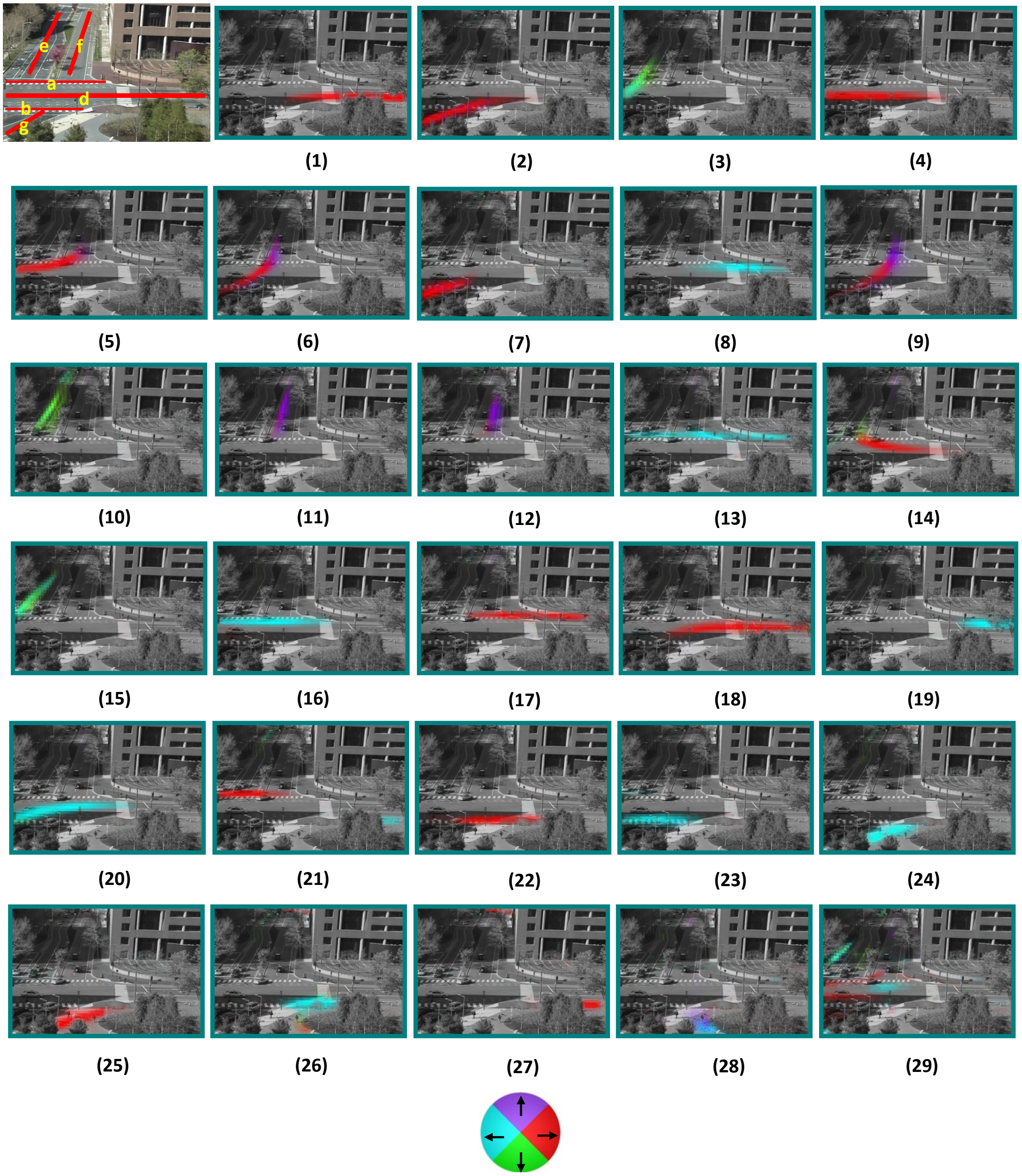


Fig. 7. Motion distributions of some topics discovered by our *HDP* models. The motion is quantized into four directions represented by four colors: red (⟶), magenta (↑), cyan (⟵), and green (↓). The topics are sorted according to how many words in the corpus are assigned to them (from large to small). For convenience, we label roads and crosswalks as $a, b, \ldots$ in the first image.

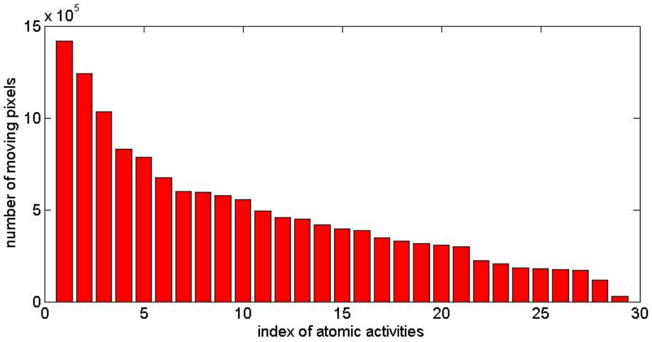


Fig. 8. Histogram of moving pixels assigned to 29 topics in Figure 7.

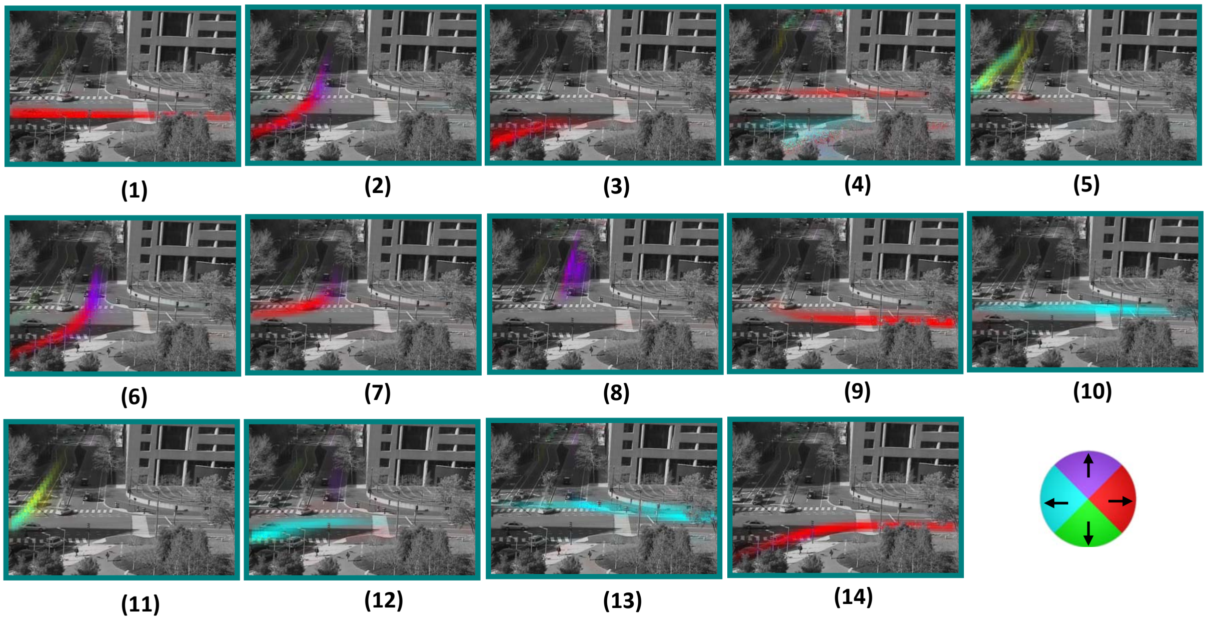


Fig. 9. Motion distributions of topics discovered by our *LDA* model when the topic number is fixed as 14.

broken into two topics because when vehicles from $g$ make a right turn (see topic 2) or vehicles from road $e$ make a left turn (see topic 14), they also share the motion in 4. From topic 10 and 19, we find vehicles stopping behind the stop lines during red lights. Topics 13, 17, 21 explain that pedestrians walk on crosswalks. When people pass the crosswalk $a$, they often stop at the divider between roads $e$ and $f$ waiting for vehicles to pass by. So this activity breaks into two topics 17 and 21. For readers' convenience, the semantics of these discovered topics are listed in a table as supplementary material. When the number of topics is set as 29, *LDA* model provides

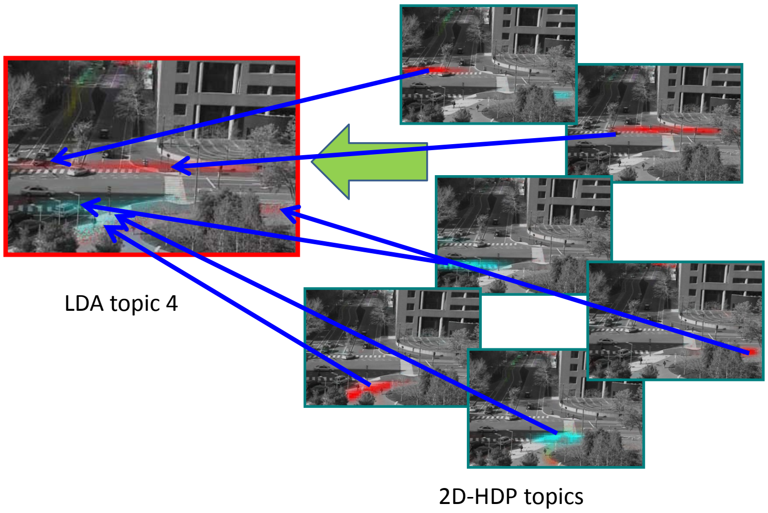


Fig. 10. When the number of word topics is set as 14 in *LDA*, *HDP* topics 17, 21, 23, 24, 25, 26, and 27 related to pedestrian walking are merged into one *LDA* topic 14.

similar result as *HDP*. In Figure 9, we show the results from *LDA* when choosing 14 instead of 29 as the number of topics. Several topics discovered by *HDP* merge into one topic in *LDA*. For example, as shown in Figure 10, *HDP* topics 17, 21, 23, 24, 25, 26, and 27 related to pedestrian walking in Figure 7 merge into *LDA* topic 4 in Figure 9. Topics 8, 16, 19 in Figure 7 merge into topic 10 in Figure 9.

### *B. Discover Interactions*

Multi-agent interactions can be well explained as combinations of atomic activities, or equivalently, topics, under our framework. In our hierarchical Bayesian models, the video clips are automatically clustered into different interactions. The topics mixtures ($\{\alpha_c\}$ in *LDA* mixture model and $\{\pi_c\}$ in *HDP*) as priors of document clusters provide a good summary of interactions. Figure 11 plots the topic mixtures $\pi_c$ of five clusters under our *HDP* models. Cluster 1 explains traffic moving in a vertical direction. Vehicles from $e$ and $g$ move vertically, crossing road $d$ and crosswalk $a$. 3, 6, 7, 9 and 11 are major topics in this interaction, while the prior over other topics related to horizontal traffic(1, 4, 5, 8, 16, 20), and pedestrians walking on crosswalk $a$ and $b$ (13, 17, 21, 23), is very low. Cluster 2 explains "vehicles from road $g$ make a right turn to road $a$ while there is not much other traffic". At this time, vertical traffic is forbidden because of the red light while there are no vehicles traveling horizontally on road *d*, so these vehicles

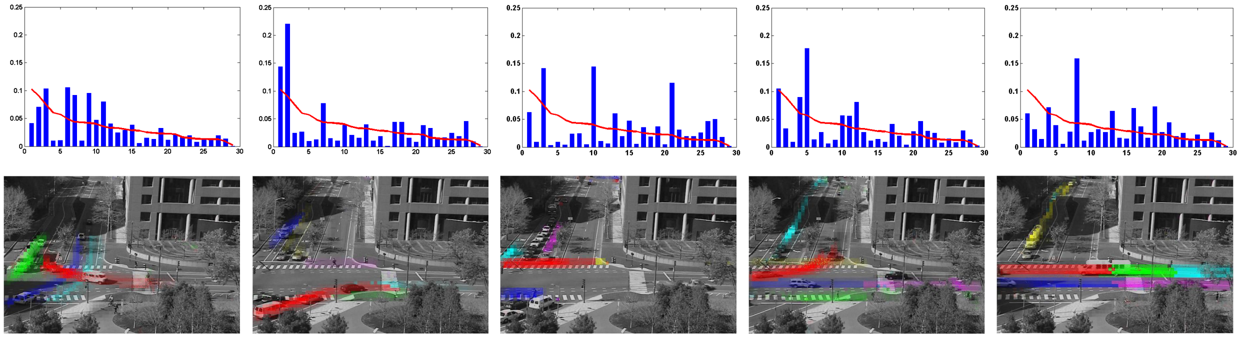
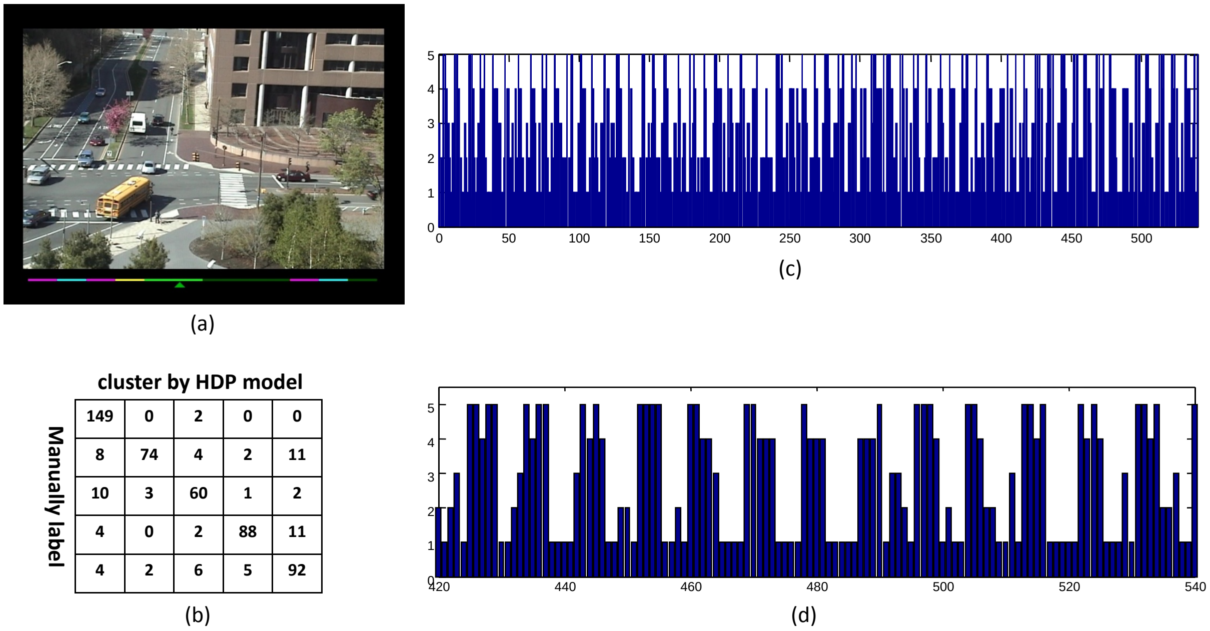


Fig. 11. The short video clips are grouped into five clusters. **In the first row**, we plot the mixtures $\{\pi_c\}$ over 29 topics as prior of each cluster represented by blue bars. For comparison, the red curve in each plot is the average topic mixture over the whole corpus. The x-axis is the index of atomic activities. The y-axis is the mixture over atomic activities. **In the second row**, we show a video clip as an example for each type of interaction and mark the motions of the five largest topics in that video clip. Notice that colors distinguish different topics in the same video (the same color may correspond to different topics in different video clips) instead of representing motion directions as in Figure 7.

from $g$ can make a right turn. Cluster 3 is "pedestrians walk on the crosswalks while there is not much traffic". Several topics (21, 13, 17) related to pedestrian walking are much higher that their average distributions on during the whole video sequence. Topics 10 and 15 are also high because they explain that vehicles on road $e$ stop behind the stop line. Cluster 4 is "vehicles on road $d$ make a left turn to road $f$". Topics 5 11 and 12 related to this activity are high. Topics 1 and 4 are also high since horizontal traffic from left to right is allowed at this time. However topics 8, 16 and 20 are very low, because traffic from right to left conflicts with this left turn activity. Cluster 5 is horizontal traffic. During this interaction, topics 13, 17 and 21 are also relatively high, since pedestrians are allowed to walk on $a$. In the second row of Figure 11, we show an example video clip for each type of interaction. In each video clip, we choose the five largest topics and mark motions belonging to different topics by different colors.

### C. Video Segmentation

Given a long video sequence, we can segment it based on different types of interactions. Our models provide a natural way to complete this task in an unsupervised manner since video clips are automatically separated into clusters (interactions) in our model. To evaluate the clustering performance, we create a ground truth by manually labeling the 540 video clips into five typical interactions in this scene as described in Section IV-B. The confusion matrix between our



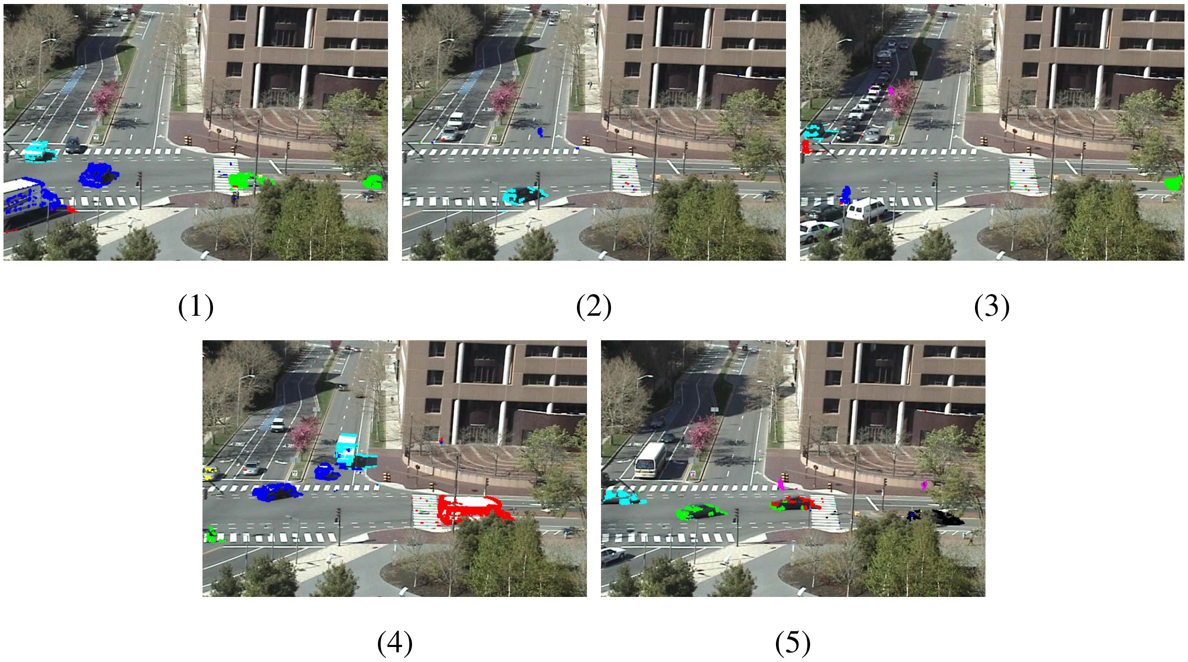
|  |  |  |  |  |
|---|---|---|---|---|
| 149 | 0 | 2 | 0 | 0 |
| 8 | 74 | 4 | 2 | 11 |
| 10 | 3 | 60 | 1 | 2 |
| 4 | 0 | 2 | 88 | 11 |
| 4 | 2 | 6 | 5 | 92 |

Fig. 12. Results of video segmentation. (a) The snapshot of our video result; (b) the confusion matrix; (c) the segmentation result of one and half hours video; (d) zoom in the segmentation result of the last 20 minutes video. In (c) and (d), the x-axis is the index of video clips in temporal order, and the y-axis is the label of five interactions shown in Figure 11.

clustering result and the ground truth is shown in Figure 12 (b). The average accuracy of video segmentation is $85.74\%$. Figure 12 shows the labels of video clips in the whole one and half hours video and in the last 20 minutes. We can observe that each traffic cycle lasts around $85$ seconds. We encourage readers to view our video results in the supplementary results.

### *D. Activity Detection*

We also want to localize different types of atomic activities happening in the video. Since in our hierarchical Bayesian models, each moving pixel is labeled as one of the atomic activities, activity detection becomes straightforward. In Figure 13, we choose five ten seconds long video clips as examples of the five different interactions, and show the activity detection results on them. We encourage readers to view our video results in the supplementary material. As an extension of activity detection, we can detect vehicles and pedestrians based on motions. It is observed that the vehicle motions and pedestrian motions are well separated among atomic activities. However, the user first needs to label each of the discovered atomic activities as



(1) (2) (3)

(4) (5)

Fig. 13. Activity detection. Five video clips are chosen as examples of the five interactions shown in Figure 11. We show one key frame of each video clip. The motions are clustered into different activities marked by different colors. However since there are so many atomic activities, we cannot use a uniform color scheme to represent all of them. In this Figure, the same color in different video clips may indicate different activities. Clip 1 has atomic activities 1 (green), 3 (cyan), 6 (blue) (see these atomic activities in Figure 7). Clip 2 has atomic activities 2 (cyan), 13 (blue). Clip 3 has atomic activities 15 (cyan), 7 (blue), 21 (red). Clip 4 has atomic activities 1 (red), 5 (blue), 7(green), 12 (cyan), 15 (yellow). Clip 5 has atomic activities 8 (red), 16 (cyan), 17 (magenta), 20 (green). Please view our video results in the supplementary material for the five video clips.

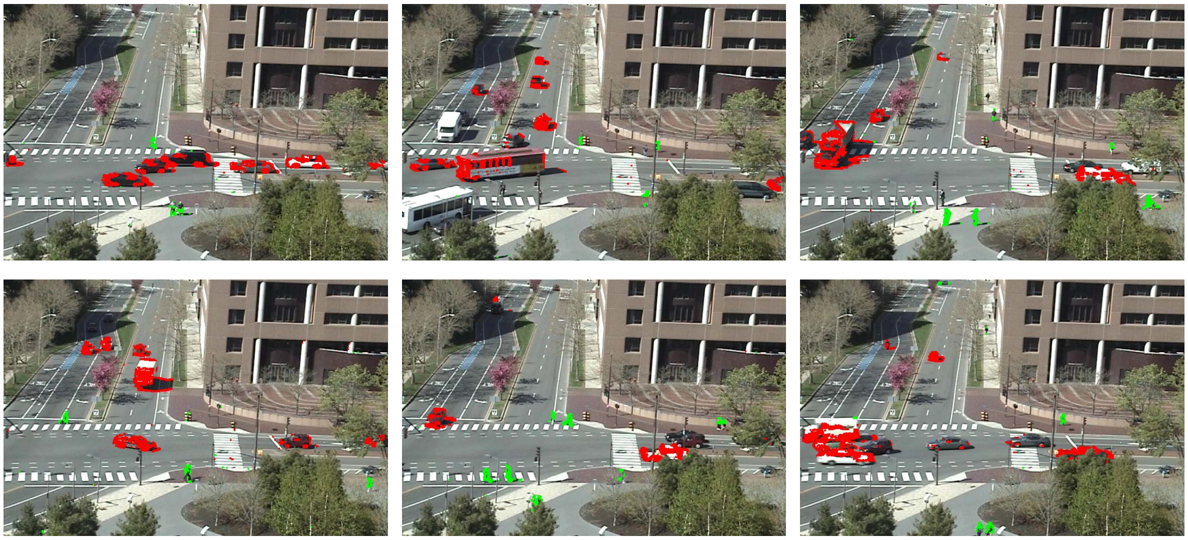
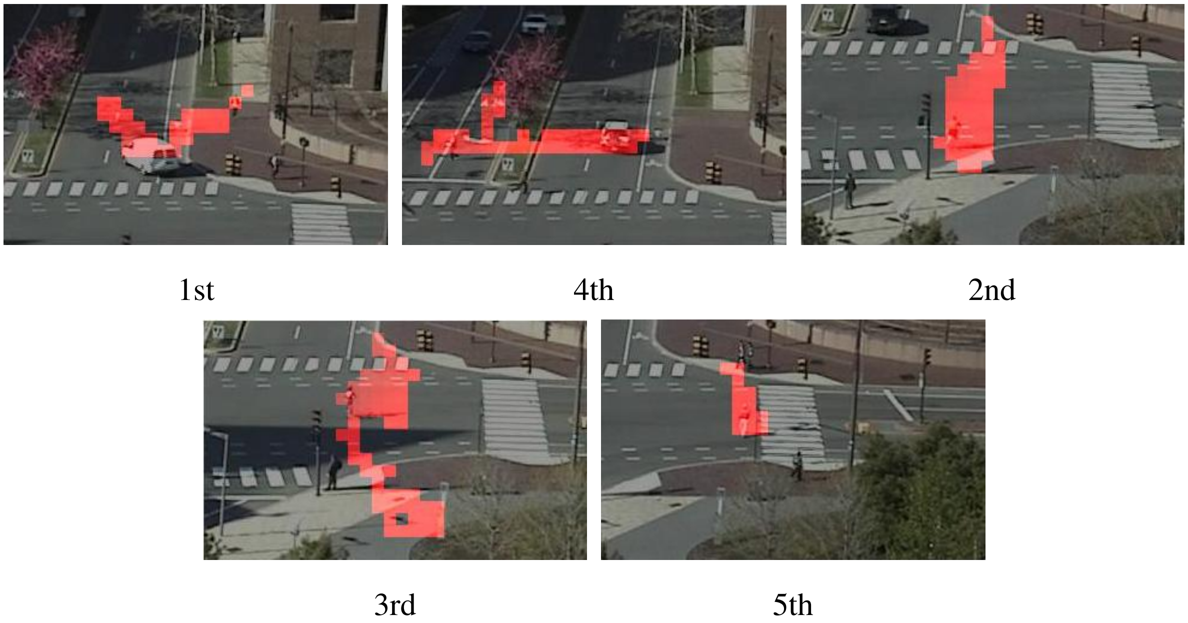


Fig. 14. Vehicle and pedestrian detection. Vehicle motions are marked by red color and pedestrian motions are marked by green color.



1st

4th

2nd

3rd

5th

Fig. 15. Results of abnormality detection. We show the top five video clips with the highest abnormality (lowest likelihood). In each video clip, we highlight the regions with motions of high abnormality.

being related to vehicles or pedestrians. Then we can classify the moving pixels into vehicles and pedestrians based on their atomic activity labels. Figure 14 shows some detection results. More video results can be found in the supplementary material. This approach cannot detect static vehicles and pedestrians. It is complementary to appearance based vehicle and pedestrian detectors, since these two approaches are using very different features (appearance vs. motion) for detection.

### *E. Abnormality Detection*

In visual surveillance, detecting abnormal video clips and localizing abnormal activities in the video clip are of great interest. Under the Bayesian models, abnormality detection has a nice probabilistic explanation by the marginal likelihood of every video clip or motion rather than by comparing similarities between samples. Computing the likelihoods of documents and words under *LDA* mixture has been described in Section III-A (see Eq 5). Computing the likelihood under *HDP* mixture model and *2D-HDP* model is not straightforward. We need to compute the likelihood of document $j$ given other documents, $p(\mathbf{x}_j|\mathbf{x}^{-j})$, where $\mathbf{x}^{-j}$ represents the whole corpus excluding document $j$. For example, in the *HDP* mixture model, since we have already drawn $M$ samples $\{\mathbf{z}^{-j(m)}, \{\pi_c^{(m)}\}, \pi_0^{(m)}\}_{m=1}^M$ from $p(\mathbf{z}^{-j}, \{\pi_c\}, \pi_0|\mathbf{x})$ which is very close to $p(\mathbf{z}^{-j}, \{\pi_c\}, \pi_0|\mathbf{x}^{-j})$, we approximate $p(\mathbf{x}_j|\mathbf{x}^{-j})$ as

$$p(\mathbf{x}_j|\mathbf{x}^{-j}) = \frac{1}{M}\sum_m \sum_{c_j} \int_{\omega_j} \sum_{\mathbf{z}_j} \sum_i p(x_{ji}|z_{ji}, \mathbf{z}^{-j(m)}, \mathbf{x}^{-j}) p(\mathbf{z}_j|\omega_j) p(\omega_j|\pi_{c_j}^{(m)}) \eta_{cj} d\omega_j \tag{15}$$

$p(\omega_j|\pi_{c_j}^{(m)})$ is a Dirichlet distribution. If $(u_1, \ldots, u_T)$ is the Dirichlet prior on $\phi_k$,

$$p(x_{ji}|z_{ji}, \mathbf{z}^{-j(m)}, \mathbf{x}^{-j}) = (u_{x_{ji}} + n_{x_{ji}})/(\sum_{t=1}^{T}(u_t + n_t))$$

is a multinomial distribution, where $n_t$ is the number of words in $\mathbf{x}^{-j}$ with value $t$ assigned to topic $z_{ji}$(see [2]). The computation of $\int_{\omega_j} \sum_{\mathbf{z}_j} p(x_{ji}|z_{ji}, \mathbf{z}^{-j(m)}, \mathbf{x}^{-j}) p(\mathbf{z}_j|\omega_j) p(\omega_j|\pi_{c_j}^{(m)})$ is intractable, but can be approximated with a variational inference algorithm as in [1]. The likelihood computation in *2D-HDP* model is very similar to that in the *HDP* mixture model. The only difference is to replace $\eta_{c_j}$ with $\epsilon_{c_j}^{(m)}$ in Eq 15.

Figure 15 shows the top five detected abnormal video clips. The red color highlights the regions with abnormal motions in the video clips. There are two abnormal activities in the first video. A vehicle is making a right-turn from road *d* to road *f*. This is uncommon in this scene because of the layout of the city. Actually there is no topic explaining this kind of activity in our data (topics are summaries of typical activities). A person is simultaneously approaching road *f*, causing abnormal motion. In the successive video clip, we find that the person is actually crossing road *f* outside the crosswalk region. This video clip ranked No.4 in abnormality. In the second and third videos, bicycles are crossing the road abnormally. The fifth video is another example of a pedestrian crossing the road outside the crosswalk.

### *F. High-Level Semantic Query*

In our framework, it is convenient to use atomic activities as tools to query for interactions of interest. For example, suppose a user wants to detect jay-walking. This is not automatically discovered by the system as a typical interaction. Thus, the user simply picks topics involved in the interaction, e.g. topic 6 and 13, i.e. "pedestrians walk on crosswalk *a* from right to left (topic 13) while vehicles are approaching in vertical direction (topic 6)", and specifies the query distribution $q$ ($q(6) = q(13) = 0.5$ and other mixtures are zeros). The topic distributions $\{p_j\}$

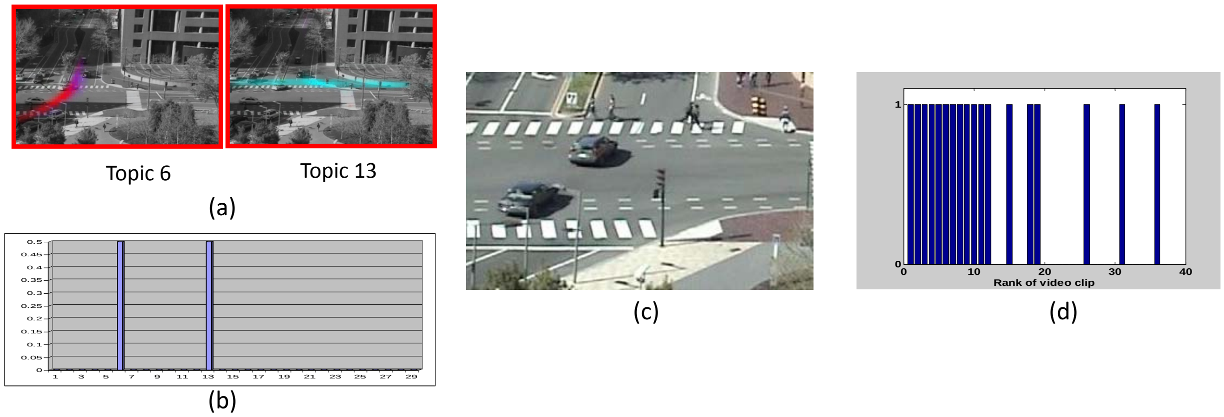


Fig. 16. Query result of jay-walking. (a) We pick two atomic activities (topic 6 and 13) involved in the interaction jay-working. (b) A query distribution is drawn with large weights on topic 6 and 13 and zeros weights on other topics. (c) An example of jay-walk retrieval. We have more video examples in the supplementary results. (d) shows the top 40 retrieval results. If the video clip is correct, it is labeled as 1 otherwise 0.

of video clips in the data set match with the query distribution using relative entropy between $q$ and $p_j$,

$$D(q||p_j) = \sum_{k=1}^{K} q(k) log \frac{q(k)}{p_j(k)} \tag{16}$$

Figure 16 (d) shows the result of querying examples of "pedestrians walk on crosswalk *a* from right to left while vehicles are approaching in vertical direction". All the video clips are sorted by matching similarity. The true instance will be labeled 1, otherwise it is labeled as 0. There are in total 18 jay-walking instances in this data set, and they are all found among the top 37 examples out of the 540 clips in the whole video sequence. The top 12 retrieval results are all correct.

## *G. Comparison with Other Methods*

Another option to model interactions is to first use the original *LDA* in Figure 3 (a) or *HDP* in Figure 4 (b) as a feature reduction step. A distribution $p_j$ over topics or a posterior Dirichlet parameter ($\gamma_j$ in Eq 2) is associated with each document. Then one can cluster documents based on $\{p_j\}$ or $\{\gamma_j\}$ as feature vectors. [1] used this strategy for classification. K-means on $\{p_j\}$ only has $55.6\%$ accuracy of video segmentation on this data set (KL divergence is the distance measure), while the accuracy of our *2D-HDP* model is $85.74\%$. It is hard to define a proper distance for Dirichlet parameters. We cannot get meaningful clusters using $\{\gamma_j\}$.

We also evaluate the algorithm proposed in [3], which used global motion to describe each frame, on this data set. [3] also adopted word-document analysis and used spectral graph partitioning. However, it did not model local atomic activities and the interactions or activities were directly modeled as a distribution over global motion instead of atomic activities. Although their method worked well on simple data sets in [3], where usually there was only one kind of activity in each video clip, it failed on our complicated scene with many activities co-occurring. We did not find meaningful interactions from the discovered clusters using their approach on our data. The formation of clusters is dominated by the amount of traffic flow instead of the types of traffic. The detected abnormal examples are videos with relatively small amounts of motion and do not really include interesting activities.

### *H. Results on the Train Station Scene*

We also test our models on a train station scene. Figure 17 shows the 22 discovered atomic activities from a one hour video sequence. These atomic activities explain people going up or coming down the escalators, or passing by in different ways. Activity detection results are shown in Figure 18. However, we do not see interesting interactions and abnormal activities in this scene. Those results are not shown here.

## V. Limitations and Possible Extensions of This Work

In this framework, we adopt the positions and moving directions of moving pixels as low-level visual features since they are more reliable in a crowded scene. While we have demonstrated the effectiveness of this model in a variety of visual surveillance tasks, including more complicated features is expected to further boost the model's discrimination power. For example, if a pedestrian is walking along the path of vehicles, just based on positions and moving detections his motions cannot be distinguished from those of vehicles and this activity will not be detected as abnormality. If a car drives extremely fast, it will not be detected as abnormal either. Other features, such as appearance and speed, are useful in these scenarios.

The information on the co-occurrence of moving pixels is critical for our methods to separate atomic activities. One moving pixel tends to be labeled as the same atomic activity as other

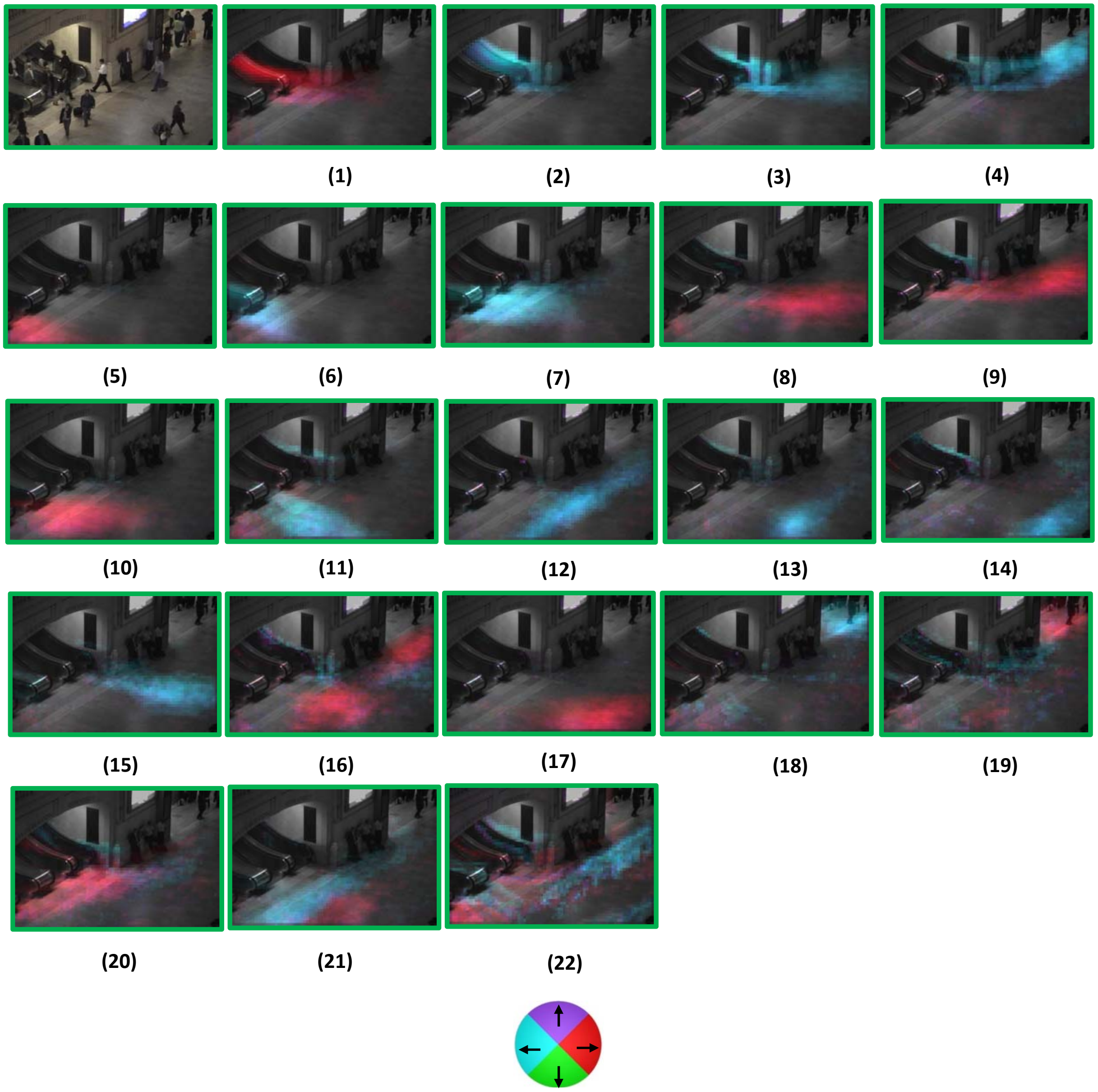
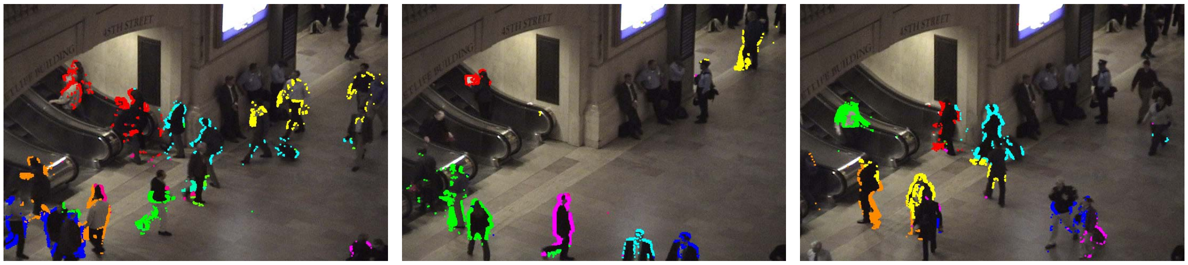


Fig. 17. Motion distributions of discovered atomic activities on a train station scene. The motion is quantized into four directions represented by four colors: red (→), magenta (↑), cyan (←), and green (↓).



(1) (2) (3)

Fig. 18. Activity detection in the train station scene. The motions are clustered into different atomic activities marked by different colors. We choose three video clips as examples. Again, because there are not enough colors to represent 22 atomic activities, we only mark several major activities by colors in each video clips. The same color may represent different activities in different video clips. Video clip 1 has atomic activities 2 (red), 3 (cyan), 4 (yellow), 5 (bule), 6 (orange), and 10 (green). Clip 2 has atomic activities 1 (red), 6 (blue), 13 (blue), 14 (cyan), 15 (green), 18 (yellow). Clip 3 has atomic activities 1 (green), 2 (red), 3 (cyan), 6 (orange), 7 (yellow), 13 (blue), and 14 (magenta).

moving pixels happening around the same time. This information is encoded into the design of video clips as documents. We divide the long video sequence into short video clips. This "hard" division may cause some problems. The moving pixels happening in two successive frames might be divided into two different documents. By intuition, one moving pixel should receive more influence from those moving pixels closer in time. However, in our models, moving pixels that fall into the same video clip are treated in the same way, no matter how close they are. In [43], we proposed a model allowing random assignment of words to documents according to some prior which encodes temporal information. If two moving pixels are temporally closer in space, they have a higher probability to be assigned to the same documents.

We are not utilizing any tracking information in this work. However, in some cases when tracking is doable or objects can be partially tracked (i.e. whenever there is ambiguity caused by occlusion or clutter, stop tracking and initialize a new track later), tracks provide useful information on atomic activities. Motions on the same track are likely to be caused by the same atomic activity. Thus a possible extension of this work is to incorporate both co-occurrence and tracking information.

In this work, we do not model activities and interactions with complicated temporal logic. However the atomic activities and interactions learnt by our framework can be used as units to model more complicated activities and interactions.

## VI. Conclusion

We have proposed an unsupervised framework adopting hierarchical Bayesian models to model activities and interactions in crowded and complicated scenes. Three hierarchical Bayesian models, *LDA* mixture model, *HDP* mixture model, and *2D-HDP* model are proposed. Without tracking and human labeling, our system is able to summarize typical activities and interactions in the scene, segment the video sequences, detect typical and abnormal activities and support high-level semantic queries on activities and interactions. These surveillance tasks are formulated in an integral probabilistic way.

## Appendix

In the appendix, we will explain how to do Gibbs sampling in the *2D-HDP* model as described in Section III-C. The sampling procedure is implemented in two steps. In the first step, given the cluster assignment $\{c_j\}$ of documents fixed, we sample the word topic assignment **z**, mixtures $\pi_0$ and $\pi_c$ on topics. It follows the Chinese Restaurant Process (*CRP*) Gibbs sampling scheme as described in [2], but adding more hierarchical levels. In *CPR*, restaurants are documents, customers are words and dishes are topics. All the restaurants share a common menu. The process can be briefly described as following (see more details in [2]).

- When a customer $i$ comes to restaurant $j$, he sits at one of the existing tables $t$ and eat the dishes served on table $t$, or take a new table $t_{new}$.
- If a new table $t_{new}$ is added to restaurant $j$, it orders a dish from the menu.

Since we are modeling clusters of documents, we introduce “big restaurants”, which are clusters of documents. The label of document cluster $c_j$ associates restaurant $j$ to big restaurant $c_j$. The *CRP* is modified as following.

- If a new table $t_{new}$ needs to be added in restaurant $j$, we go to the big restaurant $c_j$ and choose one of the existing big tables $r$ in $c_j$. $t_{new}$ is associated with $r$, and serve the same dish as $r$.
- Alternatively, the new table $t_{new}$ may take a new big table $r_{new}$ in the big restaurant $c_j$. If that happens, $r_{new}$ orders a dish from the menu. This dish will be served on both $r_{new}$ and $t_{new}$.

Following this modified $CRP$, given $\{c_j\}$, **k**, $\pi_0$ and $\{\pi_c\}$ can be sampled. It is a straightforward extension of the sampling scheme in [2] to more hierarchical levels.

In order to sample $\{c_j\}$ and generate the clusters of documents, given **z**, $\pi_0$, and $\{\pi_c\}$, we add an extra process.

- When a new restaurant $j$ is built, it needs to be associated with one of the existing big restaurants or a new big restaurant needs to be built and associated with $j$. It is assumed that we already know how many tables in restaurant $j$ and dishes served at every table.

Let $m_{jk}^t$ be the number of tables in restaurant $j$ serving dish $z$ and $m_{j\cdot}^t$ be the number of tables in restaurant $j$. To sample $c_j$, we need to compute the posterior,

$$\begin{aligned} & p(c_j|\{m_{jk}^t\}, \mathbf{c}^{-j}, \{\pi_c\}, \pi_0) \\ \propto \; & p(\{m_{jk}^t\}|c_j, \mathbf{c}^{-j}, \{\pi_c\}, \pi_0) p(c_j|\mathbf{c}^{-j}, \{\pi_c\}, \pi_0) \end{aligned} \tag{17}$$

where $\mathbf{c}_{-j}$ is the cluster labels of documents excluding document $j$. $c_j$ could be one of the existing clusters generated at the current stage, i.e. $c_j \in \mathbf{c}^{old}$. In this case,

$$\begin{aligned} & p(m_{jk}^t|c_j, \mathbf{c}^{-j}, \{\pi_c\}, \pi_0) \\ = \; & p(m_{jk}^t|\pi_{c_j}) \\ = \; & \begin{pmatrix} & m_{j\cdot}^t & \\ m_{j1}^t & \cdots & m_{jK}^t \end{pmatrix} \prod_{k=1}^{K} \pi_{c_j k}^{m_{jk}^t} \end{aligned} \tag{18}$$

where $K$ is the number of word topics allocated at the current stage. And,

$$\begin{aligned} & p(c_j|\{\pi_c\}, \mathbf{c}^{-j}, \pi_0) \\ = \; & \frac{n_{c_j}}{M - 1 + \mu} \end{aligned} \tag{19}$$

where $n_{c_j}$ is the number of documents assigned to cluster $c_j$.

$c_j$ could also be a new cluster, i.e. $c_j = c^{new}$. In this case,

$$\begin{aligned} & p(\{m_{jk}^t\}|c_j = c^{new}, \mathbf{c}^{-j}, \{\pi_c\}, \pi_0) \\ = \; & \int p(\{m_{jk}^t\}|\pi_{new}) p(\pi_{new}|\pi_0) d\pi_{\pi_{new}} \\ = \; & \begin{pmatrix} & m_{j\cdot}^t & \\ m_{j1}^t & \cdots & m_{jK}^t \end{pmatrix} \int \prod_{k=1}^{K} \pi_{new,k}^{m_{jk}^t} \frac{\Gamma(\pi_{0u} + \sum_{k=1}^{K} \pi_{0k})}{\pi_{0u} \prod_{k=1}^{K} \pi_{0k}} \pi_{new,u}^{\pi_{0,u}-1} \prod_{k=1}^{K} \pi_{new,k}^{\pi_{0k-1}} d\pi_{new} \\ = \; & \begin{pmatrix} & m_{j\cdot}^t & \\ m_{j1}^t & \cdots & m_{jK}^t \end{pmatrix} \frac{\Gamma(\alpha)}{\prod_{k=1}^{K} \Gamma(\alpha \cdot \pi_{0k})} \cdot \frac{\prod_{k=1}^{K} \Gamma(\alpha \cdot \pi_{0k} + m_{jk}^t)}{\Gamma(\alpha + m_{j\cdot}^t)} \end{aligned} \tag{20}$$

And,

$$p(c_j = c^{new} | \{\pi_c\}, \mathbf{c}^{-j}, \pi_0) = \frac{\mu}{M - 1 + \mu} \tag{21}$$

So we have,

$$\begin{aligned} & p(c_j = c | \{m_{jk}^t\}, \mathbf{c}^{-j}, \{\pi_l\}, \pi_0) \\ \propto \quad & \frac{u_c}{u. + \mu} \prod_{k=1}^{K} \pi_{ck}^{m_{jk}^t}, c \in \mathbf{c}^{old} \\ & \frac{\mu}{u. + \mu} \frac{\Gamma(\alpha)}{\prod_{k=1}^{K} \Gamma(\alpha \cdot \pi_{0k})} \cdot \frac{\prod_{k=1}^{K} \Gamma(\alpha \cdot \pi_{0k} + m_{jk}^t)}{\Gamma(\alpha + m_{j\cdot}^t)}, c = c^{new} \end{aligned} \tag{22}$$

## REFERENCES

[1] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[2] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei. Hierarchical dirichlet process. *Journal of the American Statistical Association*, 2006.

[3] H. Zhong, J. Shi, and M. Visontai. Detecting unusual activity in video. In *Proc. CVPR*, 2004.

[4] L. Zelnik-Manor and M. Irani. Event-based analysis of video. In *Proc. CVPR*, 2001.

[5] A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin. *Bayesian Data Analysis*. Chapman & Hall/CRC, 2004.

[6] F. Li and P. Perona. A bayesian hierarchical model for learning natural scene categories. In *Proc. CVPR*, 2005.

[7] C. Stauffer and E. Grimson. Learning patterns of activity using real-time tracking. *IEEE Trans. on PAMI*, 22:747–757, 2000.

[8] N. Oliver, B. Rosario, and A. Pentland. A bayesian computer vision system for modeling human interactions. *IEEE Trans. on PAMI*, 22:831–843, 2000.

[9] X. Wang, K. Tieu, and E. Grimson. Learning semantic scene models by trajectory analysis. In *Proc. ECCV*, 2006.

[10] Z. Zhang, K. Huang, T. Tan, and L. Wang. Trajectory series analysis based event rule induction for visual surveillance. In *Proc. CVPR*, 2007.

[11] S. Honggeng and R. Nevatia. Multi-agent event recognition. In *Proc. ICCV*, 2001.

[12] S. S. Intille and A. F. Bobick. A framework for recognizing multi-agent action from visual evidence. In *Proc. AAAI*, 1999.

[13] I. Haritaoglu, D. Harwood, and L. S. Davis. W4: Real-time surveillance of people and their activities. *IEEE Trans. on PAMI*, 22:809–830, 2000.

[14] G. Medioni, I. Cohen, F. BreAmond, S. Hongeng, and R. Nevatia. Event detection and analysis from video streams. *IEEE Trans. on PAMI*, 23:873–889, 2001.

[15] M. Brand and V. Kettnaker. Discovery and segmentation of activities in video. *IEEE Trans. on PAMI*, 22:844–851, 2000.

[16] J. Fernyhough, A. G. Cohn, and D. C. Hogg. Constructing qualitative event models automatically from video input. *Image and Vision Computing*, 18:81–103, 2000.

[17] N. Johnson and D. Hogg. Learning the distribution of object trajectories for event recognition. In *Proc. BMVC*, 1995.

[18] T. T. Truyen, D. Q. Phung, H. H. Bui, and S. Venkatesh. Adaboost.mrf: Boosted markov random forests and application to multilevel activity recognition. In *Proc. CVPR*, 2006.

[19] T. Xiang and S. Gong. Beyond tracking: Modelling activity and understanding behaviour. *International Journal of Computer Vision*, 67:21–51, 2006.

[20] N. Ghanem, D. Dementhon, D. Doermann, and L. Davis. Representation and recognition of events in surveillance video using petri net. In *CVPR Workshop*, 2004.

[21] P. Smith, N. V. Lobo, and M. Shah. Temporalboost for event recognition. In *Proc. ICCV*, 2005.

[22] J. W. Davis and A. F. Bobick. The representation and recognition of action using temporal templates. In *Proc. CVPR*, 1997.

[23] T. Xiang and S. Gong. Video behaviour profiling and abnormality detection without manual labelling. In *Proc. ICCV*, 2005.

[24] Y. Wang, T. Jiang, M. S. Drew, Z. Li, and G. Mori. Unsupervised discovery of action classes. In *Proc. CVPR*, 2006.

[25] C. Rao, A. Yilmaz, and M. Shah. View-invariant reprerepresentation and recognition of actions. *International Journal of Computer Vision*, 50:203–226, 2002.

[26] M. Blank, L. Gorelick, Shechtman, M. Irani, and R. Basri. Actions as space-time shapes. In *Proc. ICCV*, 2005.

[27] J. C. Niebles, H. Wang, and F. Li. Unsupervised learning of human action categories using spatial-temporal words. In *Proc. BMVC*, 2006.

[28] E. Shechtman and M. Irani. Space-time behavior based correlation. In *Proc. CVPR*, 2005.

[29] I. Laptev and T. Lindeberg. Space-time interest points. In *Proc. ICCV*, 2003.

[30] J. Sivic, B. C. Russell, A. A. Efros, A. Zisserman, and W. T. Freeman. Discovering object categories in image collections. In *Proc. ICCV*, 2005.

[31] 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 *Proc. CVPR*, 2006.

[32] E. B. Sudderth, A. Torralba, Freeman W. T., and A. S. Willsky. Learning hierarchical models of scenes, objects, and parts. In *Proc. ICCV*, 2005.

[33] E. B. Sudderth, A. Torralba, W. T. Freeman, and A. S. Willsky. Describing visual scenes using transformed dirichlet processes. In *Proc. NIPS*, 2005.

[34] B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. *JCAI*, pages 674–680, 1981.

[35] T. S. Ferguson. A bayesian analysis of some nonparametric problems. *The Annals of Statistics*, 1:209–230m, 1973.

[36] S. MacEachern, A. Kottas, and A. Gelfand. Spatial nonparametric bayesian models. Technical report, Institute of Statistics and Decision Sciences, Duke University, 2001.

[37] T. Hofmann. Probabilistic latent semantic analysis. In *Proc. of Uncertainty in Artificial Intelligence*, 1999.

[38] H. Schfitze and C. Silverstein. Projections for efficient document clustering. In *Proc. of ACM SIGIR*, 1997.

[39] I. S. Dhillon and D. S. Modha. Concept decompositions for large sparse text data using clustering. 42:143–157, 2001.

[40] J. Zhang, Z. Ghahramani, and Y. Yang. A probabilistic model for online document clustering with application to novelty detection. In *Proc. NIPS*, 2004.

[41] I. S. Dhillon. Co-clustering documents and words using bipartite spectral graph partitioning. 2001.

[42] T. L. Griffiths and M. Steyvers. Finding scientific topics. In *Proc. of the National Academy of Sciences of the United States of America*, 2004.

[43] X. Wang and E. Grimson. Spatial latent dirichlet allocation. In *Proc. NIPS*, 2007.