

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

Xiaogang Wang, *Student Member, IEEE*, Xiaoxu Ma, *Student Member, IEEE*, and W. Eric L. Grimson, *Fellow, IEEE*

**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 multiagent interactions are modeled as distributions over atomic activities. These models are learned 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: the *Latent Dirichlet Allocation (LDA)* mixture model, the *Hierarchical Dirichlet Processes (HDP)* mixture model, and the *Dual Hierarchical Dirichlet Processes (Dual-HDP)* model. They advance existing topic 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 into interactions. The *LDA* mixture model assumes that it is already known how many different types of atomic activities and interactions occur in the scene. The *HDP* mixture model automatically decides the number of categories of atomic activities. The *Dual-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 broad interest such as: 1) discovering and providing a summary of typical atomic activities and interactions occurring 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.

## 1 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 Fig. 1). In such scenes, it is often not easy to track individual objects because of frequent occlusions among objects and because 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 multiagent interactions (e.g., vehicles stopped 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 crossing the road outside the crosswalk, and abnormal interactions, e.g., jaywalking (people crossing the road while vehicles are passing by); and
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 1.1).

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

• The authors are with the Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, 32-D524, 32 Vassar Street, Cambridge, MA 02139.  
E-mail: {xgwang, welg}@csail.mit.edu, xiaoxuma@mit.edu.

Manuscript received 15 Aug. 2007; revised 31 Jan. 2008; accepted 14 Mar. 2008; published online 20 Apr. 2008.

Recommended for acceptance by L. Van Gool.

For information on obtaining reprints of this article, please send e-mail to: [tpami@computer.org](mailto:tpami@computer.org), and reference IEEECS Log Number TPAMI-2007-08-0509.

Digital Object Identifier no. 10.1109/TPAMI.2008.87.



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

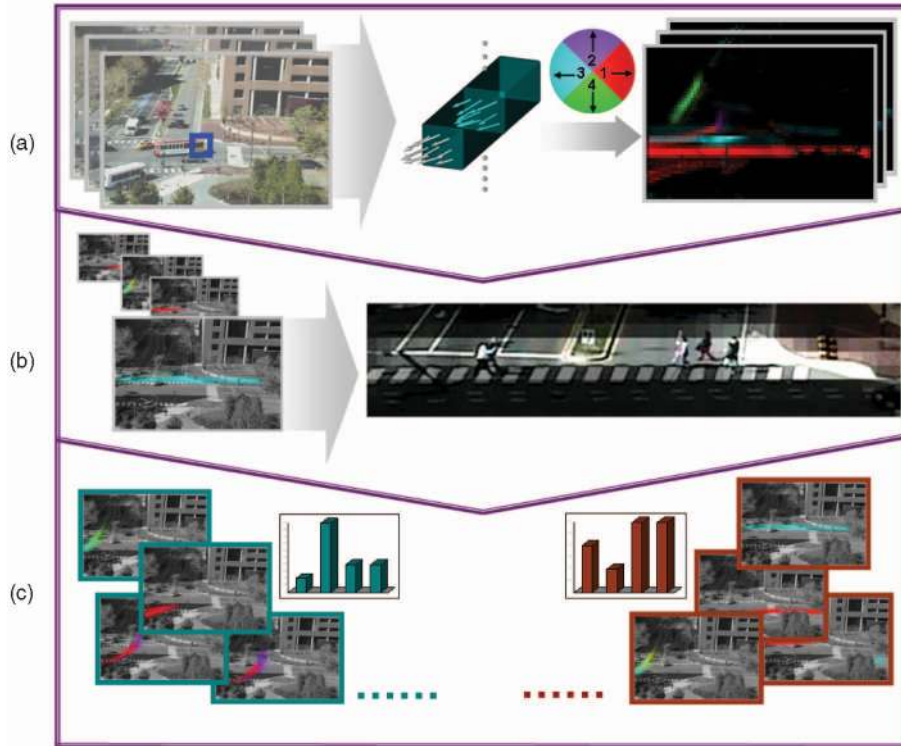


Fig. 2. Our framework connects: low-level visual features, atomic activities, and interactions. (a) The video sequence is divided into short clips as documents. In each clip, local motions are quantized into visual words based on location and motion direction. The four quantized directions are represented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

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, we just model co-occurrences of atomic activities. Atomic activities and interactions are modeled using hierarchical Bayesian models under our framework.

Our system diagram is shown in Fig. 2. We compute local motions (moving pixels) as our low-level visual features. This avoids difficult tracking problems in crowded scenes. We 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. Each moving pixel is labeled by its 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  $\rightarrow$  short video clips  $\rightarrow$  moving pixels) and visual surveillance tasks (interactions  $\rightarrow$  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 learned 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 types 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 learned

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 *Latent Dirichlet Allocation (LDA)* [1] and *Hierarchical Dirichlet Processes (HDP)* [2], from which we can borrow. Under *LDA* and *HDP* models, words, such as “professor” and “university,” often coexisting in the same documents are clustered into the same topic, such as “education.” *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 and [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 *Dual Hierarchical Dirichlet Processes (Dual-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 of 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. *Dual-HDP* automatically decides the numbers of categories of both atomic activities and interactions.

## 1.1 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]. 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 et al. [9] used the modified Hausdorff distance to compare the distance between two tracks and clustered tracks into activities. Oliver et al. [8] used a coupled HMM to model the interaction between two tracks. Intille and Bobick [11] 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 in 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 [10], [19]. However, these primitive events were learned 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 relearned.

The second kind of approaches [3], [4], [20], [21], [22], [23] 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 multiresolution 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 cannot 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 these approaches to model both single-agent activities and multiagent interactions. Although there are actions/events modeling approaches [24], [25], [26], [27], [28], 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 [29], [30], [31], and human action recognition [26]. References [6], [31], [32], and [26] are supervised learning frameworks in the sense that they need to manually label the documents. The video clip in [26] usually contains a single activity and [26] did not model interactions among multiple objects. References [29] and [30], which directly applied an *LDA* model, were 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 follows: Section 2 describes how to compute the low-level visual features. Three novel hierarchical Bayesian models are proposed in Section 3. Section 4 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 5, we discuss the limitations and possible extensions of this work.

## 2 LOW-LEVEL VISUAL FEATURES

Our data sets are video sequences from far-field traffic scenes (Fig. 1a) and train station scenes (Fig. 1c) 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 [33]. 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  $10 \times 10$ . The motion of a moving pixel is quantized into four directions as shown in Fig. 2a. 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. Deciding the size of the codebook is a balance between the descriptive capability and complexity of the model. The whole video sequence is uniformly divided into nonoverlapping 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 3.

## 3 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: the

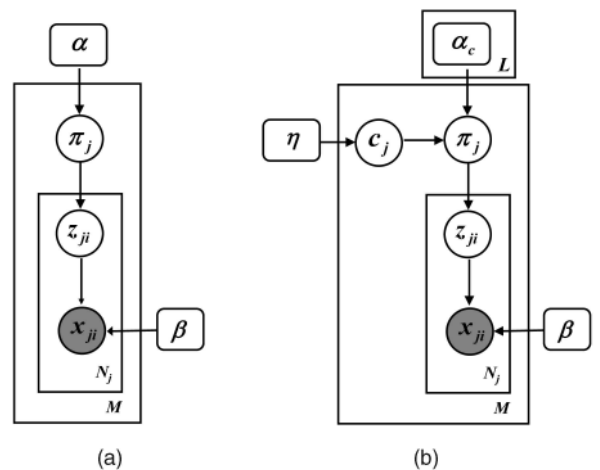


Fig. 3. (a) The LDA model proposed in [1]. (b) Our LDA mixture model.

LDA mixture model, the HDP mixture model, and the Dual-HDP model.

### 3.1 LDA Mixture Model

Fig. 3a 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  $\beta_k = [\beta_{k1}, \dots, \beta_{kW}]$  over a word vocabulary of size  $W$ .  $\beta = \{\beta_k\}$ .  $\alpha = [\alpha_1, \dots, \alpha_K]$  is a Dirichlet prior on the corpus. For each document  $j$ , a parameter  $\pi_j = [\pi_{j1}, \dots, \pi_{jK}]$  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

$$\begin{aligned} p(\mathbf{x}_j, \mathbf{z}_j, \pi_j | \alpha, \beta) &= p(\pi_j | \alpha) \prod_{i=1}^{N_j} p(z_{ji} | \pi_j) p(x_{ji} | z_{ji}, \beta) \\ &= \frac{\Gamma(\sum_{k=1}^K \alpha_k)}{\prod_{k=1}^K \Gamma(\alpha_k)} \pi_{j1}^{\alpha_1 - 1} \dots \pi_{jK}^{\alpha_K - 1} \prod_{i=1}^{N_j} \pi_{jz_{ji}} \beta_{z_{ji}x_{ji}}, \end{aligned} \quad (1)$$

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 of 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 mixture model is shown in Fig. 3b. 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

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

The marginal log likelihood of document  $j$  is

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

Our LDA mixture model is relevant to the model proposed in [6]. However, the hidden variable  $c_j$  in our model was observed in [6]. So, [6] required manually labeling documents in the training set, while our framework is totally unsupervised. This causes a different inference algorithm to be proposed for our model. Using VB [1],  $\log p(\mathbf{x}_j|\alpha_{c_j}, \beta)$  can be approximated by a tight lower bound  $L_1(\gamma_{jc_j}, \phi_{jc_j}; \alpha_{c_j}, \beta)$ :

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

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:

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

$L_2$  is maximized when choosing

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

Our EM algorithm for hyperparameter estimation is:

1. For each document  $j$  and cluster  $c_j$ , find the optimal values of the variational parameters  $\{\gamma_{j,c_j}^*, \phi_{j,c_j}^* : j = 1, \dots, M; c_j = 1, \dots, L\}$  to maximize  $L_1$  (using VB [1]).
2. Compute  $q(c_j|\gamma_{j,c_j}^*, \phi_{j,c_j}^*)$  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:

$$\eta_c \propto \sum_{j=1}^M q(c_j = c|\gamma_{jc}^*, \phi_{jc}^*), \quad (8)$$

$$\beta_{kw} \propto \sum_{j=1}^M \sum_{c_j=1}^L q(c_j|\gamma_{jc_j}^*, \phi_{jc_j}^*) \left[ \sum_{i=1}^N \phi_{jc_jik}^* x_{ji}^w \right], \quad (9)$$

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

$$\begin{aligned} \frac{\partial L_2}{\partial \alpha_{ck}} &= \sum_{j=1}^M q(c_j = c|\gamma_{jc}, \phi_{jc}) \left[ \Psi \left( \sum_{k=1}^K \alpha_{ck} \right) - \Psi(\alpha_{ck}) \right. \\ &\quad \left. + \Psi(\gamma_{jck}) - \Psi \left( \sum_{j=1}^k \gamma_{jck} \right) \right], \end{aligned} \quad (10)$$

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

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

$L_2$  monotonously increases after each iteration.

### 3.2 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 Fig. 4a. A global random

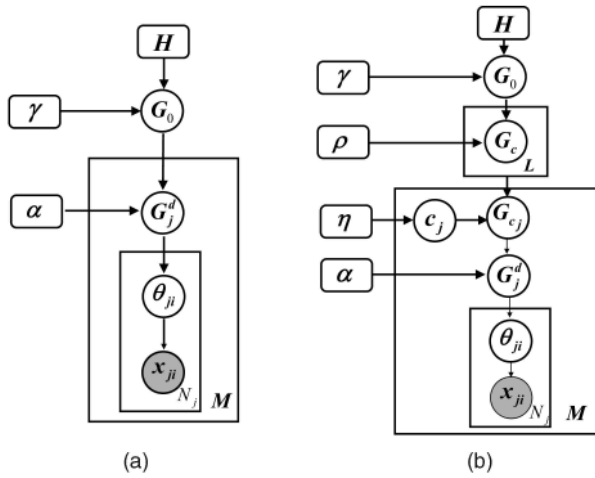


Fig. 4. (a) the *HDP* model proposed in [2]. (b) Our *HDP* mixture model.

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:

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

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.  $\{\phi_k\}$  models 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 of 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 of 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 Fig. 4b, 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  $G_0$  and  $\{G_c\}$ ; and 2) cluster label  $c_j$ . The key issue to be solved in this paper is how to sample  $c_j$ . Given  $c_j$  is 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

$$\begin{aligned} G_0 &= \sum_{k=1}^K \pi_{0k} \delta_{\phi_k} + \pi_{0u} G_{0u}, \\ G_c &= \sum_{k=1}^K \pi_{ck} \delta_{\phi_k} + \pi_{cu} G_{cu}, \\ G_j^d &= \sum_{k=1}^K \omega_{jk} \delta_{\phi_k} + \omega_{ju} G_{ju}^d, \end{aligned}$$

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  $\{\phi_k\}$ . However, they have different mixtures over topics.

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

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

$p(m_{j1}, \dots, m_{jK} | \omega_j^d)$  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

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

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  $\mathbf{z}$ ,  $\pi_0$ , and  $\{\pi_c\}$  using the schemes in [2].
2. Given  $\mathbf{z}$ ,  $\pi_0$ , and  $\{\pi_c\}$ , sample cluster labels  $\{c_j\}$  using posterior (13).

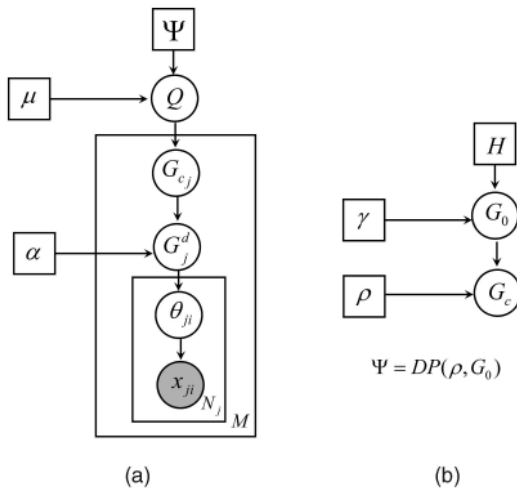


Fig. 5. The graphical model of *Dual-HDP*.  $Q = \sum_{c=1}^{\infty} \epsilon_c \delta_{G_c}$  and  $G_0 = \sum_{k=1}^{\infty} \pi_{0k} \delta_{\phi_k}$  are two infinite mixtures modeling clusters of documents and words, respectively.  $Q$  is generated from  $DDP(\mu, \Psi)$ .  $\Psi = DP(\rho, G_0)$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

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 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.

### 3.3 Dual-HDP

In this section, we propose a *Dual-HDP* model which automatically decides both the number of word topics and the number of document clusters. In addition to the HDP which model the word topics, there is another layer of HDP modeling the clusters of documents. Hence, we call this a *Dual-HDP* model. The graphical model of *Dual-HDP* is shown in Fig. 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 *Dual-HDP* model,  $G_{c_j}$  is drawn from an infinite mixture:

$$Q = \sum_{c=1}^{\infty} \epsilon_c \delta_{G_c}. \quad (14)$$

Notice that  $G_c$  itself is a random distribution with infinite parameters. When a Dirichlet process was first developed by Ferguson [34], the location parameters (such as  $\phi_k$  in (12)) could only be scalars or vectors. MacEachern et al. [35] made an important generalization and proposed the *Dependent Dirichlet Processes (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 Fig. 5a,  $Q$  is sampled from *DDP*( $\mu, \Psi$ ).  $\mu$  is the concentration parameter and  $\epsilon_c = \epsilon'_c \prod_{l=1}^{c-1} (1 - \epsilon'_l)$ ,  $\epsilon'_c \sim \text{Beta}(1, \mu)$ . As shown in Fig. 5b,  $\Psi = DP(\rho, G_0)$  is a Dirichlet process and  $G_c \sim DP(\rho, G_0)$ . Similarly to the *HDP* mixture model in Fig. 4b,  $G_0 \sim DP(\lambda, H)$  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 *Dual-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  $G_{c_j}^d \sim DP(\alpha, G_{c_j})$ . Word  $i$  in document  $j$  samples topic  $\phi_k$  from  $G_j^d$  and samples its word value from  $\text{Discrete}(\phi_k)$ .

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

1. Given the cluster assignment  $\{c_j\}$  of documents is fixed, sample the word topic assignment  $\mathbf{z}$ , masses  $\pi_0$  and  $\pi_c$  on topics using the schemes in [2].
2. Given  $\mathbf{z}$ , 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 to a new cluster. We use the Chinese restaurant franchise for sampling. See details in the Appendix.

### 3.4 Discussion on the Words-Documents Co-Clustering Framework

We propose three words-documents co-clustering models. Readers may ask why do we need a co-clustering framework? Can't 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 the language processing literature, there has been considerable work dealing with word clustering [36], [1], [2] and document clustering [37], [38], [39] separately. Dhillon [40] 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 versa. Thus, a co-clustering framework can solve the two closely related problems in a better way. Dhillon [40] 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. References [36] and [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.

### 3.5 Example of Synthetic Data

We use an example of synthetic data to demonstrate the strength of our hierarchical Bayesian models (see Fig. 6). The toy data is similar to that used in [41]. The word vocabulary is a set of  $5 \times 5$  cells. There are 10 topics with distributions over horizontal bars and vertical bars (Fig. 6a), 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

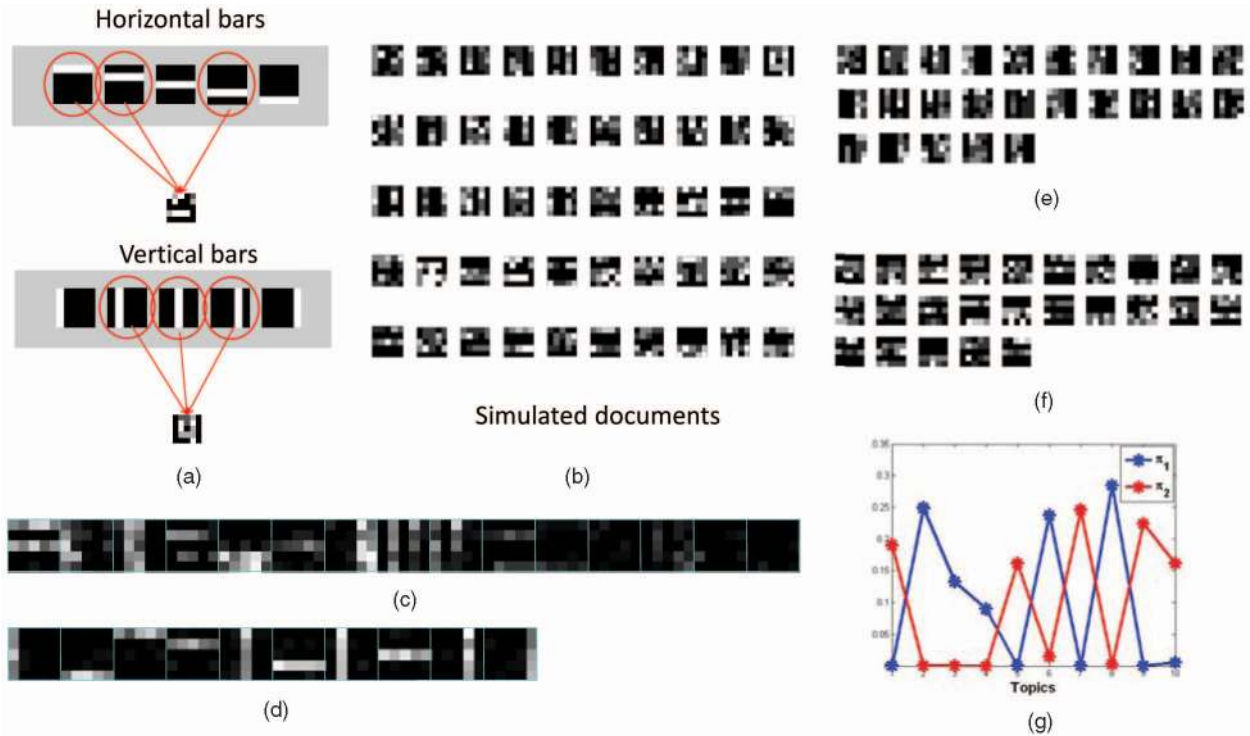


Fig. 6. Experiment on synthetic data. (a) There are 10 topics with distributions along horizontal bars and vertical bars. A synthetic document can be generated in one of the two ways. It randomly combines several vertical bar topics and sample words from them or randomly combines several horizontal bar topics. (b) The simulated documents. (c) Topic distributions learned by the *HDP* model in [2]. (d) Topic distributions learned by the *Dual-HDP* model. Documents are grouped into two clusters shown in (e) and (f). (g) Topic mixtures of two clusters  $\pi_1$  and  $\pi_2$ .

generate documents by randomly choosing several topics from the 10, adding noise to the bar distributions, and sampling 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 Fig. 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 Fig. 6c, *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. Fifteen topics are discovered and many of the topics include more than one bar. Using our *HDP* mixture model and *Dual-HDP* model to co-cluster words and documents, the 10 topics are discovered nearly perfectly as shown in Fig. 6d. Meanwhile, the documents are grouped into two clusters as shown in Figs. 6e and 6f. The topic mixtures  $\pi_1$  and  $\pi_2$  of these two clusters are shown in Fig. 6g.  $\pi_1$  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 *Dual-HDP*, we tried different numbers of document clusters as initialization, and found it always converges to two clusters.

## 4 VISUAL SURVEILLANCE APPLICATIONS AND EXPERIMENTAL RESULTS

After computing the low-level visual features as described in Section 2, 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 3. In this section, we explain how to use the results from hierarchical Bayesian models for activity analysis. We will mainly show results from *Dual-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. Some video examples of our results can be found from our website (<http://groups.csail.mit.edu/vision/app/research/HBM.html>).

### 4.1 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 1, 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 co-occur in the same video clip. Since the moving pixels are treated as words in our hierarchical



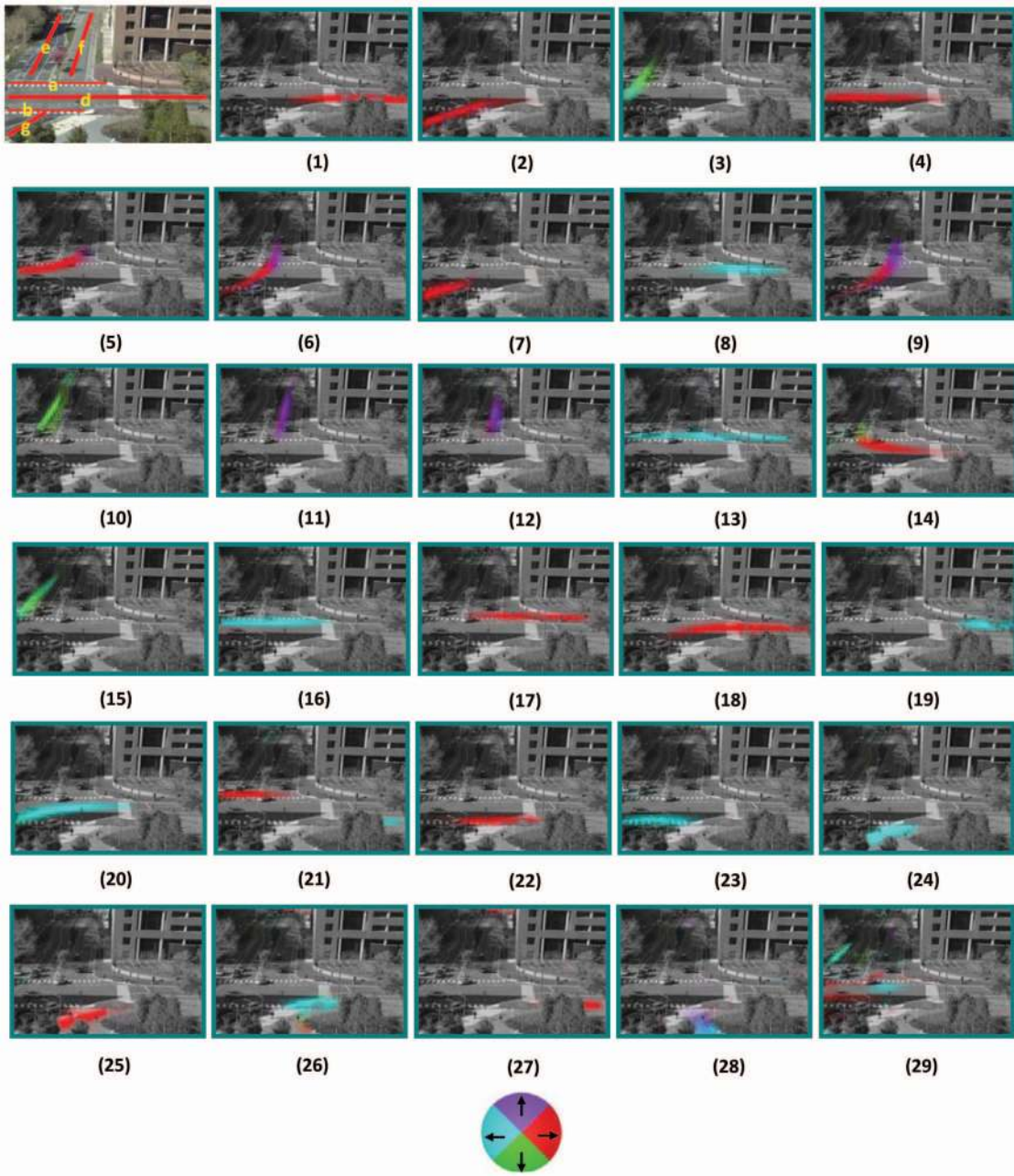


Fig. 7. Motion distributions of some topics discovered by our *HDP* models. The motion is quantized into four directions represented by four colors: red ( $\rightarrow$ ), magenta ( $\uparrow$ ), cyan ( $\leftarrow$ ), and green ( $\downarrow$ ). The topics are sorted according to how many words in the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

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 Fig. 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 Fig. 8. Topic 2 explains vehicles making a right turn. Topics 5, 14, and 20 explain vehicles making left turns.

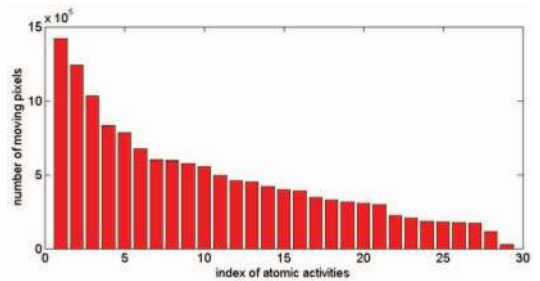


Fig. 8. Histogram of moving pixels assigned to 29 topics in Fig. 7.

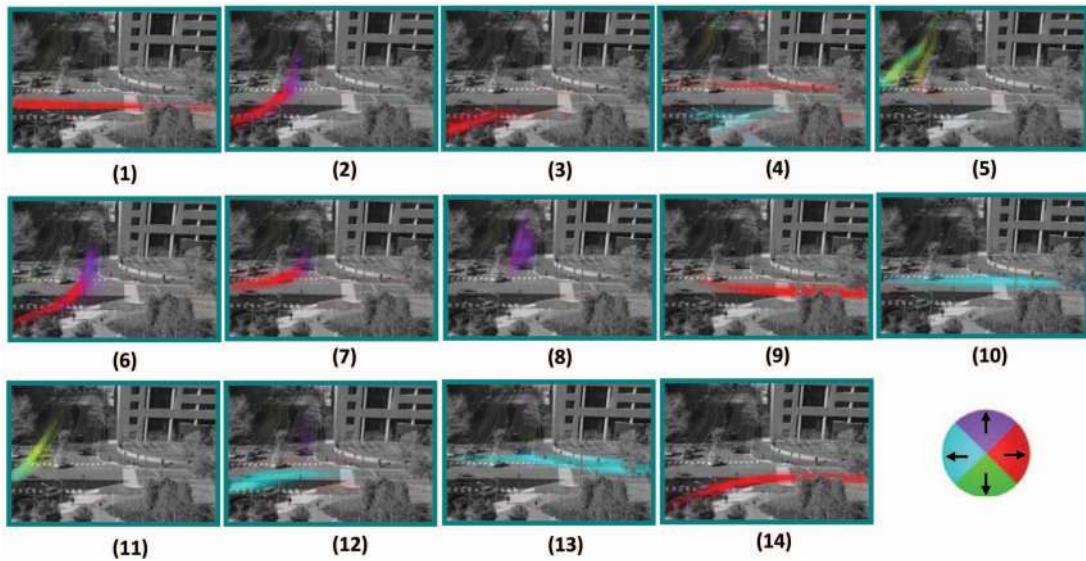


Fig. 9. Motion distributions of topics discovered by our *LDA* model when the topic number is fixed as 14.

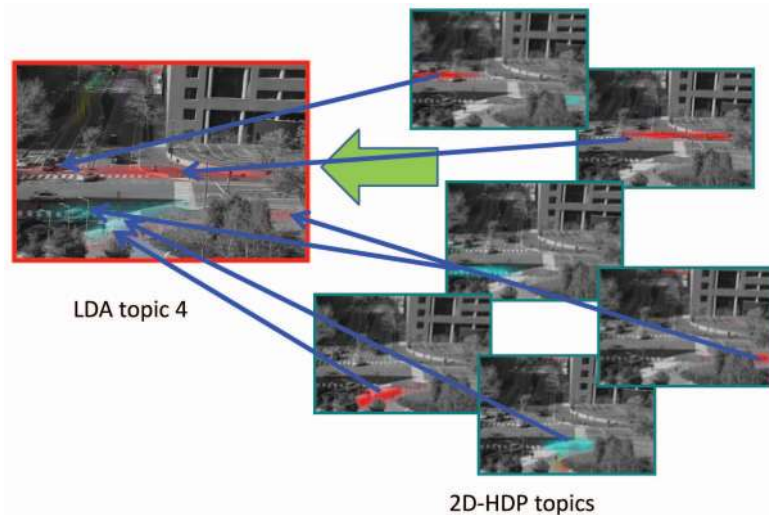


Fig. 10. When the number of word topics is set as 14 in *LDA*, *HDP* topics 17, 21, 23, 24, 25, 26, and 27 related to pedestrian walking are merged into one *LDA* topic 14.

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 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 topics 10 and 19, we find vehicles stopping behind the stop lines during red lights. Topics 13, 17, and 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. When the number of topics is set as 29, *LDA* model provides similar result as *HDP*. In Fig. 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 Fig. 10, *HDP* topics 17, 21, 23, 24, 25, 26, and 27 related to pedestrian walking in Fig. 7 merge into *LDA* topic 4 in Fig. 9. Topics 8, 16, and 19 in Fig. 7 merge into topic 10 in Fig. 9.

## 4.2 Discover Interactions

Multiagent 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. Fig. 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*. Topics 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 from *g* can make a

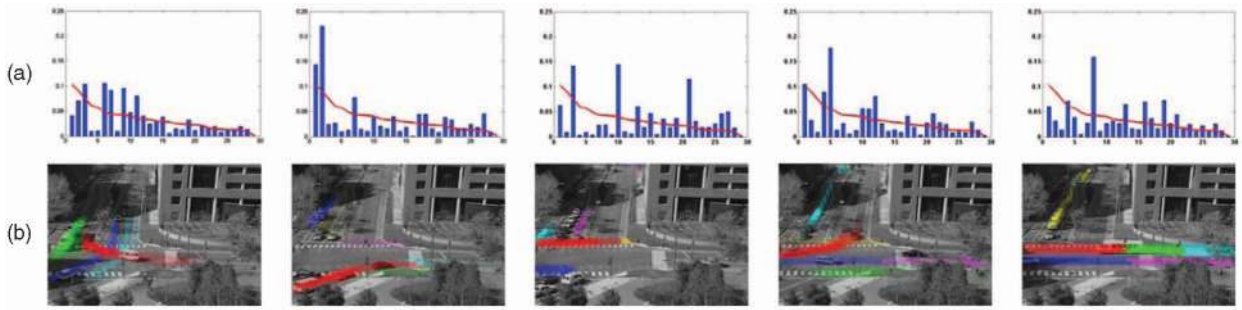


Fig. 11. The short video clips are grouped into five clusters. **In the first row**, we plot the mixtures  $\{\pi_c\}$  over 29 topics as prior of each cluster represented by blue bars. For comparison, the red curve in each plot is the average topic mixture over the whole corpus. The x-axis is the index 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 Fig. 7.

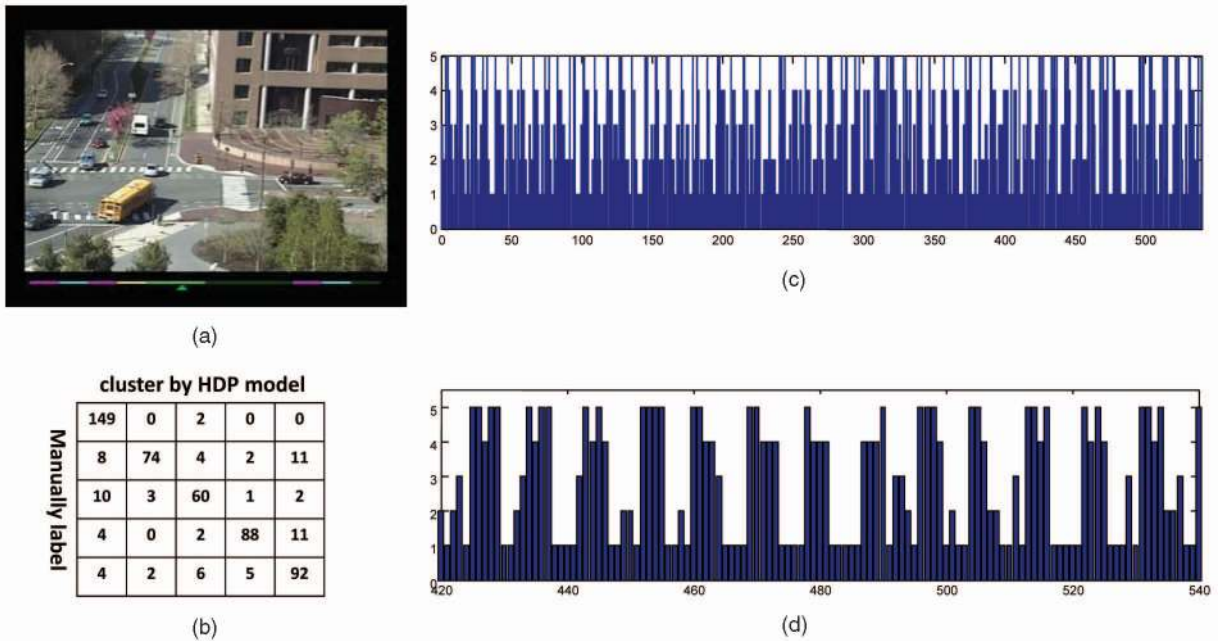


Fig. 12. Results of video segmentation. (a) The snapshot of our video result. (b) The confusion matrix. (c) The segmentation result of one and one-half hours of video. (d) Zoom in of the segmentation result of the last 20 minutes of video. In (c) and (d), the x-axis is the index of video clips in temporal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

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 than 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 Fig. 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.

### 4.3 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 4.2. The confusion matrix between our clustering result and the ground truth is shown in Fig. 12b. The average accuracy of video segmentation is 85.74 percent. Fig. 12 shows the labels of video clips in the entire one and half hours of video and in the last 20 minutes. Note the periodicity of the labels assigned. We can observe that each traffic cycle lasts around 85 seconds.

### 4.4 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

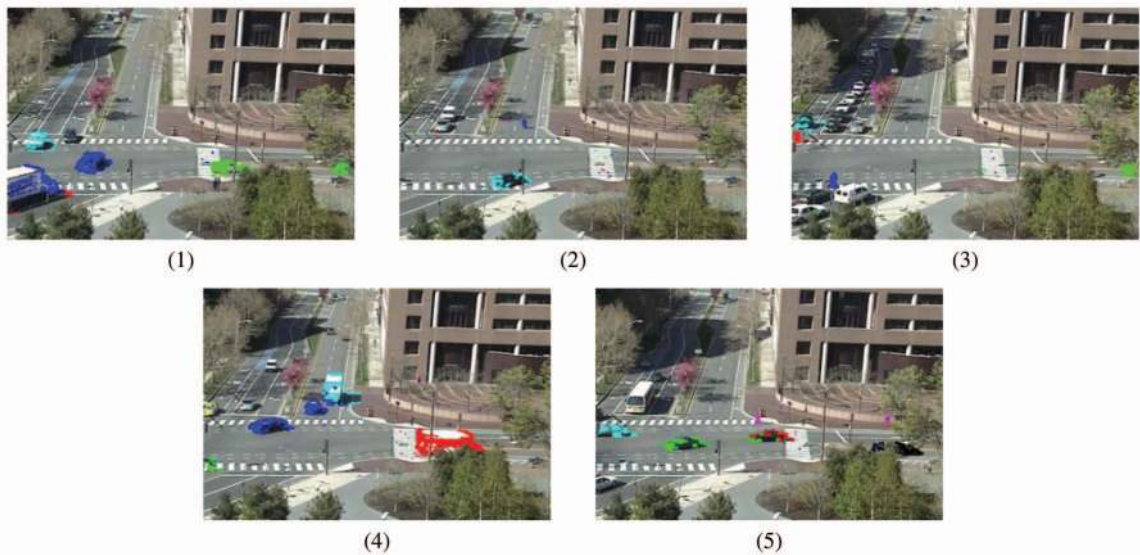


Fig. 13. Activity detection. Five video clips are chosen as examples of the five interactions shown in Fig. 11. We show one key frame of each video clip. The motions are clustered into different activities marked by different colors. However, since there are so many atomic activities, we cannot use a 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), and 6 (blue) (see these atomic activities in Fig. 7). Clip 2 has atomic activities 2 (cyan) and 13 (blue). Clip 3 has atomic activities 15 (cyan), 7 (blue), and 21 (red). Clip 4 has atomic activities 1 (red), 5 (blue), 7 (green), 12 (cyan), and 15 (yellow). Clip 5 has atomic activities 8 (red), 16 (cyan), 17 (magenta), and 20 (green).



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

activities, activity detection becomes straightforward. In Fig. 13, we choose five 10 second long video clips as examples of the five different interactions and show the activity detection results on them. 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 being related to vehicles or pedestrians. Then, we can classify the moving pixels into vehicles and pedestrians based on their atomic activity labels. Fig. 14 shows some detection results. 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 versus motion) for detection.

#### 4.5 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 3.1 (see (5)). Computing the likelihood under *HDP* mixture model and *Dual-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

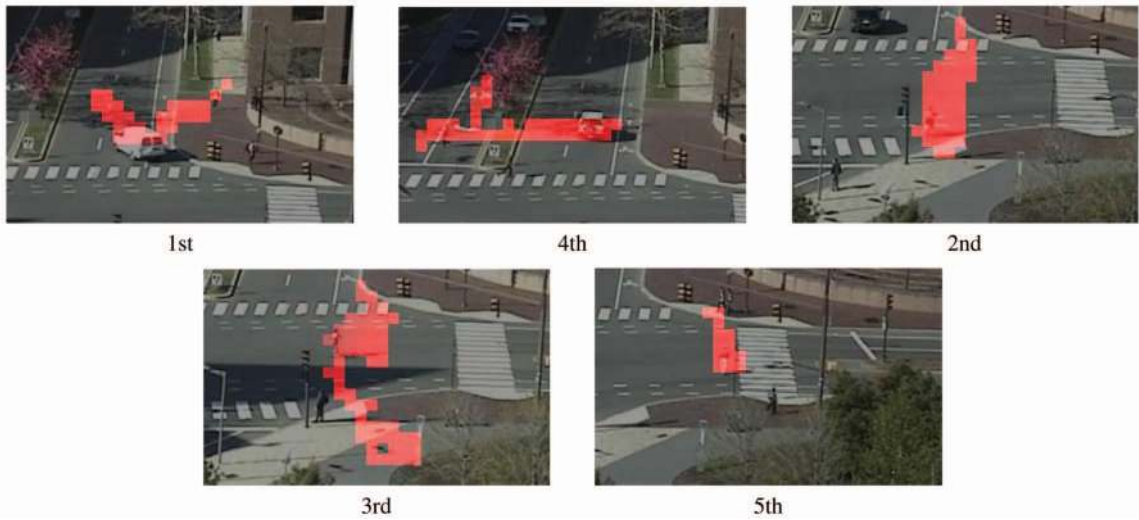


Fig. 15. Results of abnormality detection. We show the top five video clips with the highest abnormality (lowest likelihood). In each video clip, we highlight the regions with motions of high abnormality.

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

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

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

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  $\int_{\omega_j} \sum_{z_j} p(x_{ji}|z_{ji}, \mathbf{z}^{-j(m)}, \mathbf{x}^{-j}) p(\mathbf{z}_j|\omega_j) p(\omega_j|\pi_{c_j}^{(m)})$  is intractable, but can be approximated with a variational inference algorithm as in [1]. The likelihood computation in *Dual-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 (15).

Fig. 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 fourth 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.

#### 4.6 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 jaywalking. This is not automatically discovered by the system as a typical interaction. Thus, the user simply picks topics involved in

the interaction, e.g., topics 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  $\{p_j\}$  of video clips in the data set match with the query distribution using relative entropy between  $q$  and  $p_j$ :

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

Fig. 16d shows the result of querying examples of “pedestrians walk on crosswalk  $a$  from right to left while vehicles are approaching in vertical direction.” All of the video clips are sorted by matching similarity. A true instance will be labeled 1; otherwise, it is labeled as 0. There are 18 jaywalking 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.

#### 4.7 Comparison with Other Methods

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

#### 4.8 Results on the Train Station Scene

We also test our models on a train station scene. Fig. 17 shows the 22 discovered atomic activities from a 1 hour video sequence. These atomic activities explain people going up or coming down the escalators or passing by in

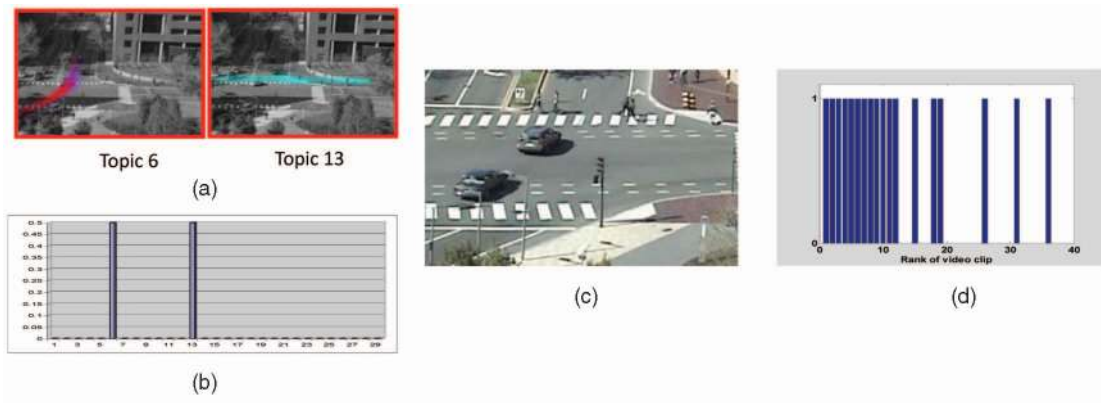


Fig. 16. Query result of jaywalking. (a) We pick two atomic activities (topics 6 and 13) involved in the interaction jaywalking. (b) A query distribution is drawn with large weights on topics 6 and 13, and zero weights on other topics. (c) An example of jaywalk retrieval. (d) shows the top 40 retrieval results. If the video clip is correct, it is labeled as 1; otherwise, 0.

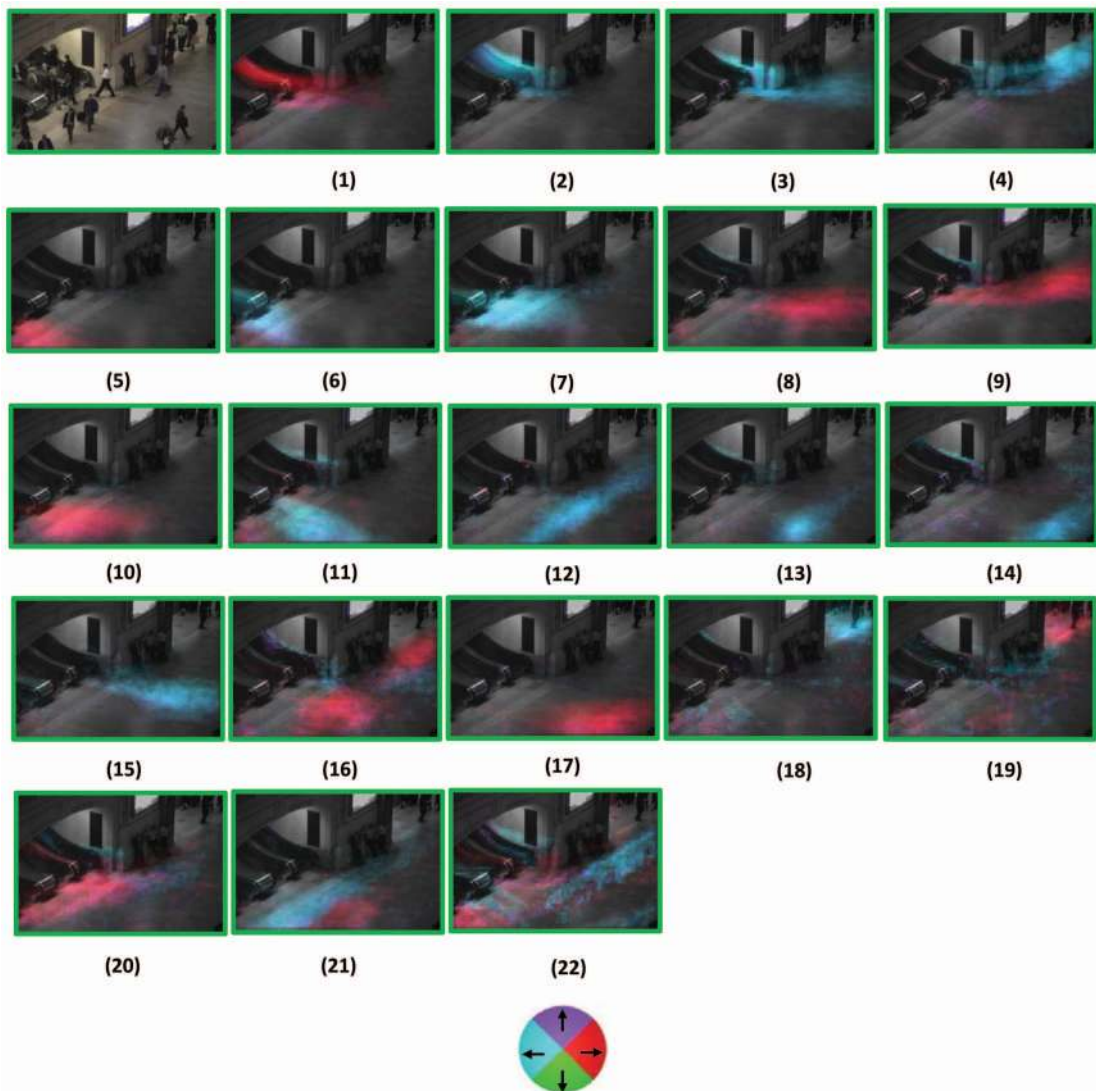


Fig. 17. Motion distributions of discovered atomic activities on a train station scene. The motion is quantized into four directions represented by four colors: red ( $\rightarrow$ ), magenta ( $\uparrow$ ), cyan ( $\leftarrow$ ), and green ( $\downarrow$ ).

different ways. Activity detection results are shown in Fig. 18. However, we do not see interesting interactions and abnormal activities in this scene. Those results are not shown here.

#### 4.9 Discussion

The space complexities of the three proposed models are all  $O(KW) + O(KL) + O(KM) + O(N)$ , where  $K$  is the number of topics,  $W$  is the size of the codebook,  $L$  is the number



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 (blue), 6 (orange), and 10 (green). Clip 2 has atomic activities 1 (red), 6 (blue), 13 (blue), 14 (cyan), 15 (green), and 18 (yellow). Clip 3 has atomic activities 1 (green), 2 (red), 3 (cyan), 6 (orange), 7 (yellow), 13 (blue), and 14 (magenta).

of document clusters,  $M$  is the number of documents, and  $N$  is the total number of words. Using EM and VB, the time complexity of the learning and inference of the LDA mixture model is  $O(ML) + O(NK) + O(LK^2)$ . Running on a computer with 3 GHz CPU, it takes less than 1 hour to process a 1.5 hour video sequence. The Gibbs sampling inference of *HDP* mixture model and *Dual-HDP* model is much slower. The time complexity of each Gibbs sampling iteration is  $O(NK) + O(ML)$ . It is difficult to provide theoretical analysis on the convergence of Gibbs sampling. It takes around 12 hours to process a 1.5 hour video sequence. In recent years, variational inference was proposed for *HDP* [42] and it is faster than Gibbs sampling. A possible extension of this work is to explore variational inference algorithms under *HDP* mixture model and *Dual-HDP* model. Currently, our algorithm is running in a batch mode. However, once the model has been learned from a training video sequence and fixed, it can be used to do motion/video segmentation and abnormality detection on new video stream in an online mode.

## 5 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 an 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 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 learned by our framework can be used as units to model more complicated activities and interactions.

## 6 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: the *LDA* mixture model, the *HDP* mixture model, and the *Dual-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

Here, we will explain how to do Gibbs sampling in the *Dual-HDP* model as described in Section 3.3. The sampling procedure is implemented in two steps. In the first step, given the cluster assignment  $\{c_j\}$  of documents is fixed, we

sample the word topic assignment  $\mathbf{z}$ , 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 eats the dishes served on table  $t$  or takes 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 follows:

- 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 serves 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  $\{c_j\}$ ,  $\mathbf{k}$ ,  $\pi_0$ , 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  $\mathbf{z}$ ,  $\pi_0$ , 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  $m_j^t$  be the number of tables in restaurant  $j$ . To sample  $c_j$ , we need to compute the posterior

$$p(c_j | \{m_{jk}^t\}, \mathbf{c}^{-j}, \{\pi_c\}, \pi_0) \propto p(\{m_{jk}^t\} | c_j, \mathbf{c}^{-j}, \{\pi_c\}, \pi_0) p(c_j | \mathbf{c}^{-j}, \{\pi_c\}, \pi_0), \quad (17)$$

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,

$$p(m_{jk}^t | c_j, \mathbf{c}^{-j}, \{\pi_c\}, \pi_0) = p(m_{jk}^t | \pi_{c_j}) = \binom{m_j^t}{m_{j1}^t \cdots m_{jK}^t} \prod_{k=1}^K \pi_{c_j k}^{m_{jk}^t}, \quad (18)$$

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

$$p(c_j | \{\pi_c\}, \mathbf{c}^{-j}, \pi_0) = \frac{n_{c_j}}{M-1+\mu}, \quad (19)$$

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,

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

In addition,

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

So, we have

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

## ACKNOWLEDGMENTS

The authors wish to acknowledge the US Defense Advanced Research Projects Agency and DSO National Laboratories (Singapore) for partially supporting this research.

## REFERENCES

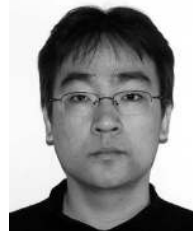
- [1] D.M. Blei, A.Y. Ng, and M.I. Jordan, “Latent Dirichlet Allocation,” *J. Machine Learning Research*, vol. 3, pp. 993-1022, 2003.
- [2] Y.W. Teh, M.I. Jordan, M.J. Beal, and D.M. Blei, “Hierarchical Dirichlet Process,” *J. Am. Statistical Assoc.*, 2006.
- [3] H. Zhong, J. Shi, and M. Visontai, “Detecting Unusual Activity in Video,” *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2004.
- [4] L. Zelink-Manor and M. Irani, “Event-Based Analysis of Video,” *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2001.
- [5] A. Gelman, J.B. Carlin, H.S. Stern, and D.B. Rubin, *Bayesian Data Analysis*. Chapman and Hall/CRC, 2004.
- [6] L. Fei-Fei and P. Perona, “A Bayesian Hierarchical Model for Learning Natural Scene Categories,” *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2005.
- [7] C. Stauffer and E. Grimson, “Learning Patterns of Activity Using Real-Time Tracking,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, pp. 747-757, 2000.
- [8] N. Oliver, B. Rosario, and A. Pentland, “A Bayesian Computer Vision System for Modeling Human Interactions,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, pp. 831-843, 2000.
- [9] X. Wang, K. Tieu, and E. Grimson, “Learning Semantic Scene Models by Trajectory Analysis,” *Proc. Ninth European Conf. Computer Vision*, 2006.



- [10] S. Honggeng and R. Nevatia, "Multi-Agent Event Recognition," *Proc. Int'l Conf. Computer Vision*, 2001.
- [11] S.S. Intille and A.F. Bobick, "A Framework for Recognizing Multi-Agent Action from Visual Evidence," *Proc. 16th Nat'l Conf. Artificial Intelligence*, 1999.
- [12] I. Haritaoglu, D. Harwood, and L.S. Davis, "W4: Real-Time Surveillance of People and Their Activities," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, pp. 809-830, 2000.
- [13] G. Medioni, I. Cohen, F. BreAmond, S. Honggeng, and R. Nevatia, "Event Detection and Analysis from Video Streams," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 23, pp. 873-889, 2001.
- [14] M. Brand and V. Kettner, "Discovery and Segmentation of Activities in Video," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, pp. 844-851, 2000.
- [15] J. Fernyhough, A.G. Cohn, and D.C. Hogg, "Constructing Qualitative Event Models Automatically from Video Input," *Image and Vision Computing*, vol. 18, pp. 81-103, 2000.
- [16] N. Johnson and D. Hogg, "Learning the Distribution of Object Trajectories for Event Recognition," *Proc. Sixth British Machine Vision Conf.*, 1995.
- [17] T.T. Truyen, D.Q. Phung, H.H. Bui, and S. Venkatesh, "Ada-boost.mrf: Boosted Markov Random Forests and Application to Multilevel Activity Recognition," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2006.
- [18] T. Xiang and S. Gong, "Beyond Tracking: Modelling Activity and Understanding Behaviour," *Int'l J. Computer Vision*, vol. 67, pp. 21-51, 2006.
- [19] N. Ghanem, D. Dementhon, D. Doermann, and L. Davis, "Representation and Recognition of Events in Surveillance Video Using Petri Net," *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition Workshops*, 2004.
- [20] P. Smith, N.V. Lobo, and M. Shah, "Temporalboost for Event Recognition," *Proc. Int'l Conf. Computer Vision*, 2005.
- [21] J.W. Davis and A.F. Bobick, "The Representation and Recognition of Action Using Temporal Templates," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 1997.
- [22] T. Xiang and S. Gong, "Video Behaviour Profiling and Abnormality Detection without Manual Labelling," *Proc. Int'l Conf. Computer Vision*, 2005.
- [23] Y. Wang, T. Jiang, M.S. Drew, Z. Li, and G. Mori, "Unsupervised Discovery of Action Classes," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2006.
- [24] C. Rao, A. Yilmaz, and M. Shah, "View-Invariant Representation and Recognition of Actions," *Int'l J. Computer Vision*, vol. 50, pp. 203-226, 2002.
- [25] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri, "Actions as Space-Time Shapes," *Proc. Int'l Conf. Computer Vision*, 2005.
- [26] J.C. Niebles, H. Wang, and F. Li, "Unsupervised Learning of Human Action Categories Using Spatial-Temporal Words," *Proc. 16th British Machine Vision Conf.*, 2006.
- [27] E. Shechtman and M. Irani, "Space-Time Behavior Based Correlation," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2005.
- [28] I. Laptev and T. Lindeberg, "Space-Time Interest Points," *Proc. Int'l Conf. Computer Vision*, 2003.
- [29] J. Sivic, B.C. Russell, A.A. Efros, A. Zisserman, and W.T. Freeman, "Discovering Object Categories in Image Collections," *Proc. Int'l Conf. Computer Vision*, 2005.
- [30] 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," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, 2006.
- [31] E.B. Sudderth, A. Torralba, W.T. Freeman, and A.S. Willsky, "Learning Hierarchical Models of Scenes, Objects, and Parts," *Proc. Int'l Conf. Computer Vision*, 2005.
- [32] E.B. Sudderth, A. Torralba, W.T. Freeman, and A.S. Willsky, "Describing Visual Scenes Using Transformed Dirichlet Processes," *Proc. Conf. Neural Information Processing Systems*, 2005.
- [33] B.D. Lucas and T. Kanade, "An Iterative Image Registration Technique with an Application to Stereo Vision," *Proc. Int'l Joint Conf. Artificial Intelligence*, pp. 674-680, 1981.
- [34] T.S. Ferguson, "A Bayesian Analysis of Some Nonparametric Problems," *The Annals of Statistics*, vol. 1, pp. 209-230, 1973.
- [35] S. MacEachern, A. Kottas, and A. Gelfand, "Spatial Nonparametric Bayesian Models," technical report, Inst. of Statistics and Decision Sciences, Duke Univ., 2001.
- [36] T. Hofmann, "Probabilistic Latent Semantic Analysis," *Proc. 15th Conf. Uncertainty in Artificial Intelligence*, 1999.
- [37] H. Schfütze and C. Silverstein, "Projections for Efficient Document Clustering," *Proc. ACM Special Interest Group on Information Retrieval*, 1997.
- [38] I.S. Dhillon and D.S. Modha, *Concept Decompositions for Large Sparse Text Data Using Clustering*, vol. 42, pp. 143-157, 2001.
- [39] J. Zhang, Z. Ghahramani, and Y. Yang, "A Probabilistic Model for Online Document Clustering with Application to Novelty Detection," *Proc. Conf. Neural Information Processing Systems*, 2004.
- [40] I.S. Dhillon, "Co-Clustering Documents and Words Using Bipartite Spectral Graph Partitioning," *Proc. ACM Special Interest Group on Knowledge Discovery and Data Mining*, 2001.
- [41] T.L. Griffiths and M. Steyvers, "Finding Scientific Topics," *Proc. Nat'l Academy of Sciences USA*, 2004.
- [42] Y.W. Teh, K. Kurihara, and M. Welling, "Collapsed Variational Inference for HDP," *Proc. Conf. Neural Information Processing Systems*, 2007.
- [43] X. Wang and E. Grimson, "Spatial Latent Dirichlet Allocation," *Proc. Conf. Neural Information Processing Systems*, 2007.



learning. He is a student member of the IEEE and the IEEE Computer Society.



learning, especially in the areas of image modeling, image understanding, and object recognition. He is a student member of the IEEE and the IEEE Computer Society.



W. Eric L. Grimson received the BSc (Hons) degree in mathematics and physics from the University of Regina, in 1975 and the PhD degree in mathematics from the Massachusetts Institute of Technology (MIT), in 1980. He is the Bernard Gordon Professor of Medical Engineering in MIT's Department of Electrical Engineering and Computer Science. He is a member of MIT's Computer Science and Artificial Intelligence Laboratory and the head of its Computer Vision Group. He also holds a joint appointment as a lecturer on radiology at Harvard Medical School and at Brigham and Women's Hospital. He is currently the head of the Department of Electrical Engineering and Computer Science at MIT. Prior to this position, he served as an associate department head for computer science and as an education officer for the department. He is a recipient of the Bose Award for Undergraduate Teaching at MIT. His research interests include computer vision and medical image analysis. Since 1975, he and his research group have pioneered state-of-the-art methods for activity and behavior recognition, object and person recognition, image database indexing, site modeling, stereo vision, and many other areas of computer vision. Since the early 1990s, his group has been applying vision techniques in medicine for image-guided surgery, disease analysis, and computational anatomy. He is a fellow of the IEEE and the American Association for Artificial Intelligence (AAAI) and a member of the IEEE Computer Society.

Xiaogang Wang received the BS degree in electrical engineering and information science from the University of Science and Technology of China, in 2001 and the MS degree in information engineering from the Chinese University of Hong Kong, in 2004. He is currently a PhD student in the Computer Science and Artificial Intelligence Laboratory at the Massachusetts Institute of Technology. His research interests include computer vision and machine

Xiaoxu Ma received the BS and MS degrees in electrical engineering from Tsinghua University, Beijing, in 1997 and 2000, respectively. He is currently a PhD student in the Computer Science and Artificial Intelligence Laboratory at the Massachusetts Institute of Technology. From 2000 to 2003, he worked as an assistant researcher in the Multimodal User Interface group at Microsoft Research Asia. His research interests include computer vision and statistical

W. Eric L. Grimson received the BSc (Hons) degree in mathematics and physics from the University of Regina, in 1975 and the PhD degree in mathematics from the Massachusetts Institute of Technology (MIT), in 1980. He is the Bernard Gordon Professor of Medical Engineering in MIT's Department of Electrical Engineering and Computer Science. He is a member of MIT's Computer Science and Artificial Intelligence Laboratory and the head of its Computer Vision Group. He also holds a joint appointment as a lecturer on radiology at Harvard Medical School and at Brigham and Women's Hospital. He is currently the head of the Department of Electrical Engineering and Computer Science at MIT. Prior to this position, he served as an associate department head for computer science and as an education officer for the department. He is a recipient of the Bose Award for Undergraduate Teaching at MIT. His research interests include computer vision and medical image analysis. Since 1975, he and his research group have pioneered state-of-the-art methods for activity and behavior recognition, object and person recognition, image database indexing, site modeling, stereo vision, and many other areas of computer vision. Since the early 1990s, his group has been applying vision techniques in medicine for image-guided surgery, disease analysis, and computational anatomy. He is a fellow of the IEEE and the American Association for Artificial Intelligence (AAAI) and a member of the IEEE Computer Society.