2015 IEEE International Conference on Computer Vision Workshop

# Video Event Recognition by Combining HDP and Gaussian Process

Wentong Liao, Bodo Rosenhahn
Institute for Information Processing (TNT),
Leibniz University Hannover
{liao, rosenhahn}@tnt.uni-hannover.de

Michael Ying Yang
Computer Vision Lab Dresden, TU Dresden
Ying.Yang1@tu-dresden.de

## Abstract

*In this paper, we present a framework for automatically analyzing activities and interactions, and recognizing traffic states from surveillance video. Activities and interactions are firstly learned by Hierarchical Dirichlet Process (HDP) models based on low-level visual features. Based on the learning results, a Gaussian Process (GP) classifier is trained to classify the traffic states in online video. Furthermore, the temporal dependencies between video events learned by HDP-Hidden Markov Models (HMM) are effectively integrated into GP classifier to enhance the accuracy of the classification. Our framework couples the benefits of the generative models-HDP with the discriminant models-GP. We validate the proposed model by applying it to the analysis of the three standard video datasets over crowded traffic scenes and compare it with other baseline models. Experimental results demonstrate that our model is effective and efficient.*

## 1. Introduction

Video event classification is an important issue in computer vision and have attracted great attention in recent years [11] due to their significant practical values such as security monitoring, traffic controlling, etc. Most existing approaches focused on recognition of an individual activity [21], or a collective activity [2] in clean backgrounds. This task remains challenging in a crowded public scene due to a large number of agents with different activities at the same time, and complicated interactions such as traffic flows at a busy junction. Moreover, a surveillance video captured from a crowded scene is usually low-quality.

Discriminant models such as GP models and SVM are the most popular approaches to classify video event [1, 4] due to their advantage in terms of classification accuracy. However, they are supervised models and training dataset with manual labels is necessary in advance. Besides, they are feature-based approaches. They have high requirement in the applicability and the preciseness of features to ensure their performance. The most widely used features include HOG feature, optic flow based features, etc.

The generative topic models such as LDA [7] and HDP [22, 13] have shown great promise in exploring motion patterns for dynamic scenes. They effectively learn activities and interactions from non-labeled video by analyzing semantic relationships instead. However, they have serious limitations: consuming computation and work in batch. Besides, most existing methods neglect the temporal dependencies between activities and interactions [22].

Inspired by the benefit of generative and discriminative models, in this paper, we propose a framework to combine the HDP model and the GP model for analyzing and classifying video events. The first step is unsupervised learning the activities using HDP model and traffic states using HDP-HMM, respectively. Based on the learning results, a GP classifier is trained to recognize the traffic states. In addition, the temporal dependencies between two states are integrated into our GP model to enhance classification accuracy.

**Contributions.** First, we effectively combine unsupervised generative model HDP with discriminant model GP, to realize classification of traffic states. Second, we integrate transition information between two states with GP model to enhance the accuracy of classification. Third, we provide detailed experiments and analysis showing that our framework enjoys favorable performance in video event classification in a crowded traffic scene.

### 1.1. Related Work

Topic models have received increasing attention to analyze activity in surveillance video [7, 9, 13, 22, 19, 6]. However, [22] are offline and batch procedures and temporal dependencies are neglected. [7, 9] use latent Dirichlet allocation (LDA) models to infer activities in a video, which requires predefined number of clusters. It is hard to give a proper number of possible activities that may occur in a video from a crowded scene. Besides, their models perform Gibbs sampling in each newly captured video clip to

978-1-4673-9711-7/15 $31.00 © 2015 IEEE
DOI 10.1109/ICCVW.2015.31

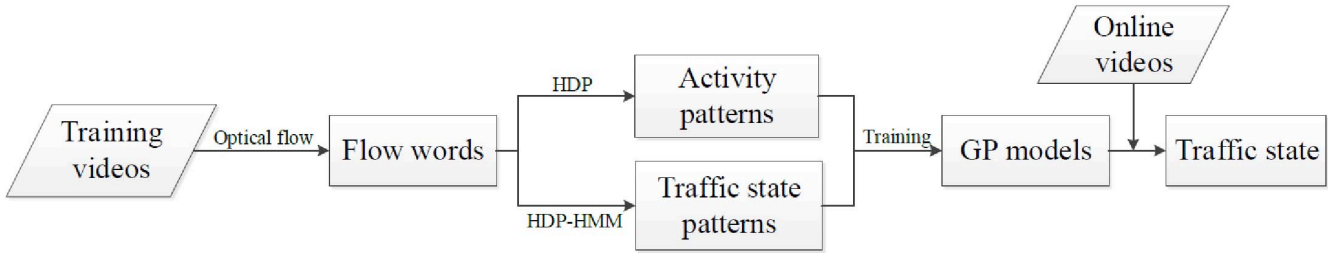



Figure 1. The flow chart of proposed framework

estimate the joint distribution. It is time consuming and especially inefficient in an online model. [13] proposed a new method to infer activities and interactions from video. But this model is computation consuming. Moreover, it is not clear whether the model could work online.

GP models have been often applied for human motion analysis and activities recognition [1, 20] because of its robustness and high accuracy in classification. However, GP models are supervised. They must be fed with manual labeled data set. In the other hand, GP models require proper features to model events such as the most widely used trajectories [12, 5]. However, tracking-based methods depend crucially on the performance of detection and tracking which is costly or unreliable in our complex and crowded scene. [17] proposed an alternative method to combine features for complex event recognition. However, this method is unfeasible in a surveillance video because of the low quality and too many objects in small size. [10] proposed a method to combined the HDP model with One-Class SVM by using Fisher kernel. This method needs to compute the gradient of the log likelihood with respect to each parameters of the model.

## 2. Video Representation

Firstly, some key notations are given in Tab. 1. The application data are surveillance videos from complex and crowded traffic scenes and captured by a fixed camera. They contain many activities and interactions. Some unavoidable problems such as occlusions, a variety object types, small size of objects challenge detection and tracking based methods. In such case, using the local motions as low-level features is a reliable way. First, optical flow features for each pixel between each pair of successive frames are extracted using the method proposed in [14]. A proper threshold (0.85 in our experiments) is necessary to reduce noise. Similar to the related works [22, 13, 8] the camera scene is uniformly divided into square cells of $8 \times 8$ pixels to get rough position features. The optical flow features of each cell are the mean of its pixel memberships and quantized into one of the 8 directions (see Fig. 7(r)) as local motion features. Finally, a vocabulary is constructed, in which each word is characterized by its position and motion direction. A vocabulary with $N$ total flow words is denoted as $\mathbf{V} = \{1, 2, \cdots, N\}$.

The input videos are uniformly segmented into non-overlapping clips for 75 frames each (3 seconds). Each clip is an accumulation of flow words over its frames and represented as a word vector $\mathbf{w} = (w_1, \cdots, w_{|I|})$, where $I$ is set of the word indexes and $|I|$ denotes the total number of occurring words in this clip. The entries of the vector are unique, i.e. we only care about if a word occurs during this clip instead its frequency of occurrence.

Table 1. Notations of variables

| Notations | Descriptions |
|---|---|
| $t = 1, \cdots, T$ | index of video clips |
| $\mathbf{V} = \{1, \cdots, N\}$ | codebook with $N$ total words |
| $w_{ti}$ | $i^{th}$ flow word in clip $t$ |
| $m_{i,j}$ | transition probability from state $i$ to $j$ |
| $p_{kw_i}$ | probability of word $i$ in activity $k$ |
| $\mathbf{a}_k$ | pattern dictionary of activity $k$ |
| $\mathbf{I}_t$ | set of word indexes in clip $t$ |
| $c_{ti}$ | intensity of activity $i$ in clip $t$ |

## 3. Method

The whole framework is illustrated as Fig. 1. First, the local motion are extracted to form a vocabulary. Second, a HDP model and a HDP-HMM are applied to automatically learn the activities and traffic states, respectively. The HDP-HMM also provide the transition information between two states. Next, we propose a method to represent activities based on low-level features and states with mixture of activities. Then, a GP classifier is trained with the representation. The transition information is integrated into the classifier to enhance the classification accuracy. Finally, the trained classifier performs to classify the traffic states in online surveillance video sequence.

We model activity and traffic state patterns as spatial distributions of flow words that have high co-occurrence frequencies within a clip. They are topic basis: a distribution over vocabulary $\mathbf{V}$. Then HDP [18] is applied to learn the latent topics, i.e. the activities and states.

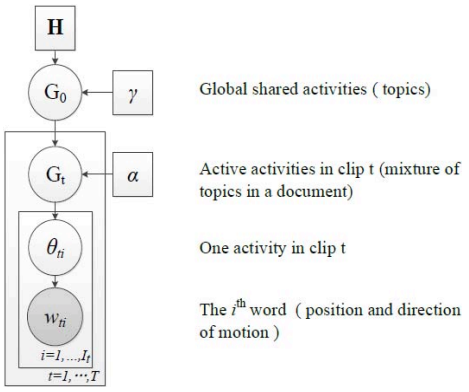



Figure 2. A graphical representation of HDP model

### 3.1. Learning Activities Using HDP

The graphical representation of HDP model is shown in Fig. 2. The global random measure $G_0$ is the global topics (activities) set that is shared by all clips. Its distribution is a Dirichlet Process with concentration parameter $\lambda$ and Dirichlet prior $H$:

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

$G_0$ can be expressed using the stick-breaking formulation [18]:

$$G_0 = \sum_{k=1}^{\infty} \pi_k \delta_{\phi_k}, \ \pi_k = \pi'_k \prod_{l=1}^{k-1}(1-\pi'_l),$$
$$\pi'_k \sim Beta(1,\lambda), \ \phi_k|\gamma, H \sim H,$$

where $\{\phi_k\}$ are parameters of multinomial distributions over words in codebook corresponding to topic $\theta_k$, i.e. word probability vector and the sum of its entries equals 1. $\delta_{\phi_k}$ is the Delta function at point $\phi_k$. $\{\pi_k\}$ are random probability measures (mixtures over topics) and $\Sigma_{k=1}^{\infty}\pi_k = 1$. It is also known as $\pi_k \sim GEM(\gamma)$ in stick-breaking process. The multinomial distribution $\phi_k$ over words in the codebook is generated from $H$. Therefore, $H$ is interpreted as a distribution over multinomial distributions and thus can be defined as a Dirichlet distribution:

$$H = Dir(D_0), \ \ \phi_k|\gamma, H \sim Dir(D_0).$$

$G_t$ is a random measure and drawn from the second DP with concentration parameter $\alpha$ and Dirichlet prior $G_0$:

$$G_t|\alpha, G_0 \sim DP(\alpha, G_0),$$

where $G_0$ itself is drawn from the first DP as demonstrated above. Thus, $G_0$ is a prior distribution over the whole corpus and a sample $G_t$ is its subset. In our case $G_t$ describes the multinomial distribution of active topics in clip $t$. We express it using the stick-breaking representation again:

$$G_t = \sum_{k=1}^{\infty} \pi_{tk} \delta_{\phi_k}, \ \pi_{tk} = \pi'_{tk} \prod_{l=1}^{k-1}(1-\pi'_{tl}),$$
$$\pi'_{tk} \sim Beta(1,\alpha), \ \phi_k|\alpha, G_0 \sim G_0.$$

For the $i^{th}$ word in document $t$, a topic $\theta_{ti}$ is first drawn from $G_t$ and then the word $w_{ti}$ is drawn from multinomial distribution $Multi(w_{ti}; \phi_{\theta_{ti}})$ (i.e. the multinomial distribution over words in codebook corresponding to topic $\theta_{ti}$ ). Different $G_t$ has the same $\phi_k$ as $G_0$, i.e. different clips share the same set of topics and statistical strength. We apply Gibbs sampling schemes to do inference under an HDP model [18].

The hyperparameters $\gamma$ and $\alpha$ are empirically predefined. They are priors on the concentration of the word distribution within topics. They influence the number of activities in $G_0$ and $G_t$. The parameter $D_0$ for the Dirichlet distribution is also set empirically.

Although HDP models decide the number of topics automatically, some of the explored activities are non-semantic (Fig. 3(a)). Some activities are generated because some rare motions need to be explained individually. There motions could be caused by noise or rare events. These activities could lead to ambiguous or even misleading analysis of interactions. Therefore, the typical normal activities are necessary to be selected from all the learned activities. This step is executed as follows. The total number of words that are assigned to activity $k$ throughout the training video is denoted as $n_k$. The ratio of activity $k$ is computed as

$$r_k = \frac{n_k}{n_1 + \cdots + n_K}. \quad (1)$$

We rank $\{r_1, \cdots, r_K\}$ in a decreasing order as $\{r'_1 \geq \cdots \geq r'_K\}$ and calculate the accumulated sum as $R'_j = \sum_{i=1}^{j} r'_i$ Then typical activity set is denoted as

$$\mathbf{A_{typical}} \triangleq \{A_j | R'_j \leq 0.99\}, \ \ 1 \leq j \leq K, \quad (2)$$

where $A_j$ is the activity $j$ with ratio $r'_j$.

(a) (b)

Figure 3. Examples of activity (a) and traffic state (b) patterns without explicit semantics

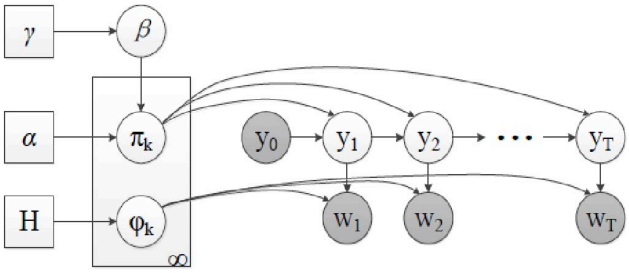



Figure 4. A graphical representation of the HDP-HMM model [18]

(a) All possible words (b) activity pattern words in the dictionary

Figure 5. An example of activity pattern

### 3.2. Learning States using HDP-HMM

A busy traffic junction is normally regulated by traffic lights: different traffic states occur sequentially and circulatory in a certain order. Hidden Markov model (HMM) [3] hat inherent advantages to explore the latent states and their transition information. HMM can be explained as a doubly stochastic Markov chain and is essentially a dynamic variant of a finite mixture model. [18] replaced the finite mixture with a Dirichlet process and proposed the HDP-HMM model as shown in Fig. 4. Therefore, the number of states of HMM is automatically decided by the HDP model instead of given in advance. Its stick-breaking formalism is:

$$\beta \sim GEM(\gamma),\ \ \pi_k \sim DP(\alpha,\beta),\ \ \phi_k \sim H$$
$$y_t|y_{t-1} \sim Mult(\pi_{y_{t-1}}),\ \ \mathbf{w}_t|y_t = s_i \sim Multi(\phi_{y_t=s_i})$$

where $y_t \in \mathbf{S} = \{s_1, \cdots, s_{N_s}\}$ is the state label of the $t^{th}$ clip and $\mathbf{S}$ is the set of possible states and $N_s$ is the total number of states. In this case, each vector $\pi_k = \{\pi_{kl}\}_{l=1,\cdots,N_s}$ is one row of the Markov chain's transition matrix from state $k$ to the other states. For clear explanation, we denote these transition matrix as $\mathbf{M} = \{m_{i,j}\}_{i,j=1\cdots L}$ throughout the paper. Gibbs sampling schemes are applied to do inference under this HDP-HMM. Fig. 8 shows the typical traffic states learned by HDP-HMM for QMUL Junction Dataset [7].

The same as in the activity learning results, the typical traffic states also need to be selected from the results given by HDP-HMM. Fig. 3(a) shows an example of learned states caused by anomaly. This process is the same as described in Sec.3.1.

### 3.3. Activity and State Patterns

Both the learned activities and traffic states are characterized by a multinomial distribution over the words in codebook. These statistic information cannot be directly applied by our classifier. We propose here a new way to represent activity and traffic states based on the learning results.

**Activity patterns** The probability of $i^{th}$ word in activity $\theta_k$ is denoted as $p_{kw_i}$ and $\mathbf{p}_{k\mathbf{w}} = \{p_{kw_i}\}_{i=1}^{N}$, $\Sigma_{i=1}^{N} p_{kw_i} = 1$. We sort $\mathbf{p}_{k\mathbf{w}}$ in descending order $\mathbf{p}'_{k\mathbf{w}} = \{p'_{kw_1} \geq \cdots \geq p'_{kw_N}\}$ and calculate the accumulated sum of probability as:

$$P'_{kj} = \sum_{i=1}^{j} p'_{kw_i}. \tag{3}$$

The pattern dictionary of activity $k$ is denoted as:

$$\mathbf{a}_k = \{w_j | P'_{kj} \leq 0.9\} \tag{4}$$

The words which fall into the rest $10\%$ are viewed as noise or rare motion. Fig. 5 shows a comparison between all possible co-occurring visual words and the selected representative words in the activity of vehicles driving downward.

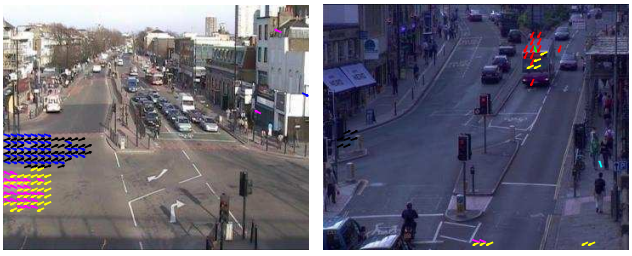
**State patterns** The traffic state of a clip is represented as a mixture of activities $\mathbf{c} = \{c_1, \cdots, c_{N_a}\}$, where $N_a$ is the total number of typical activities. For clip $t$ each component of the vector is computed as:

$$c_{ti} = \frac{|\mathbf{a}_i \cap \mathbf{w}_t|}{|\mathbf{I}_t|}. \tag{5}$$

The vector explains the intensity of each activity in this clip, as shown in Fig. 8.

### 3.4. Gaussian Process for State Classification

The HDP-HMM has mined the main traffic states from training video sequence and labeled the training video clips $\mathbf{y} = \{y_1, \cdots, y_T\}$, $y_t \in \mathbf{S}$ and $T$ is the total number of clips for training. $\mathbf{C} = \{\mathbf{c}_1, \cdots, \mathbf{c}_T\}$ is the set of feature vectors for corresponding clips, where $\mathbf{c}_t$ is the feature vector of clip $t$ given by Eq. (5). Now the training data set $(\mathbf{C}, \mathbf{y})$ is constructed to train our discriminative model (GP). Our task is labeling a new coming video clip $\mathbf{c}^*$ to a traffic state with the highest probability $P(y^*|\mathbf{C}, \mathbf{y}, \mathbf{c}^*)$. For simple illustration the binary classification with two traffic states $y_t \in \{-1, +1\}$ is considered here. The binary classification can be easily extended to multiple classification by using the one-against-all strategy.



(a) imperfect clip segmentation (b) too few motions

Figure 6. Examples of confused traffic states

The general formulation of probability prediction for a new test sample given the training data $(\mathbf{C}, \mathbf{y})$ under a GP model is:

$$p(y^* = +1|\mathbf{C}, \mathbf{y}, \mathbf{c}^*) = \int p(y^*|f^*)p(f^*|\mathbf{C}, \mathbf{y}, \mathbf{c}^*)df^*, \tag{6}$$

where $p(f^*|\mathbf{C}, \mathbf{y}, \mathbf{c}^*)$ is the distribution of latent variable $f^*$ corresponding to sample $\mathbf{c}^*$. It is obtained by integrating over the latent variable $\mathbf{f} = (f_1, \ldots, f_T)$:

$$p(f^*|\mathbf{C}, \mathbf{y}, \mathbf{c}^*) = \int p(f^*|\mathbf{C}, \mathbf{y}, \mathbf{c}^*, \mathbf{f})p(\mathbf{f}|\mathbf{C}, \mathbf{y})d\mathbf{f} \tag{7}$$

where $p(\mathbf{f}|\mathbf{C}, \mathbf{y}) = p(\mathbf{f}|\mathbf{y})p(\mathbf{f}|\mathbf{C}) \;/\; p(\mathbf{y}|\mathbf{C})$ is the posterior over the latent variables. $p(\mathbf{y}|\mathbf{C})$ is the marginal likelihood (evidence), $p(\mathbf{f}|\mathbf{C})$ is the GP prior over the latent function, which in GP model is a jointly zero mean Gaussian distribution and with the covariance given by the kernel K.

The non-Gaussian likelihood in Eq. (7) makes the integral analytically intractable. We have to resort to either analytical approximation of integrals or Monte Carlo methods. We use the *Laplace* method [23] in this paper. As introduced in [16] the mean and variance of $f^*$ are obtained as follows:

$$\begin{aligned} & p(f^*|\mathbf{C}, \mathbf{y}, \mathbf{c}^*) = \mathcal{N}(\mu^*, \sigma^*), \\ with \quad & \mu^* = \mathbf{k}(\mathbf{C})^T\mathbf{K}^-\widetilde{\mathbf{f}}, \\ & \sigma^{*2} = \mathbf{k}(\mathbf{c}^*, \mathbf{c}^*) - \mathbf{k}(\mathbf{C}, \mathbf{c}^*)^T(\mathbf{K} + \mathbf{W}^-)\mathbf{k}(\mathbf{c}, \mathbf{c}^*), \end{aligned}$$

where $\mathbf{W} \triangleq -\nabla\nabla log p(\mathbf{y}|\widetilde{\mathbf{f}})$ is diagonal. $\mathbf{K}$ denotes a $T \times T$ covariance matrix between $T$ training points. $\mathbf{k}(\mathbf{C}, \mathbf{c}^*)$ is a covariance vector between T training video clips $\mathbf{C}$ and test clip $\mathbf{c}^*$, while $\mathbf{k}(\mathbf{c}^*, \mathbf{c}^*)$ is covariance for test clip $\mathbf{c}^*$, and $\widetilde{\mathbf{f}} = \arg\max_{\mathbf{f}} p(\mathbf{f}|\mathbf{C}, \mathbf{y})$. Given the mean and variance of latent variable $f^*$ for test clip $\mathbf{c}^*$, we compute the prediction probability using Eq. (6).

The covariance function and its hyperparameters $\Theta$ crucially affect GP models performance. The Gaussian radial basis function (RBF) is one of the most widely used kernels due to its robustness for different types of data and is given as:

$$K_{RBF}(\mathbf{c}_i, \mathbf{c}_j) = \sigma^2 \exp(-\frac{\|\mathbf{c}_i - \mathbf{c}_j\|^2}{2l^2}). \tag{8}$$

$\Theta = [\sigma, l]$ is the hyperparameter set for RBF. We optimize the hyperparameters using Conjugate Gradient method [15].

### 3.5. Integration of Transition Information into GP Classifier

The input video is segmented into clips along time and some clips maybe span two traffic states, as shown Fig. 6(a). In practice, the traffic volume varies. Sometimes the scene is silent within one or several consecutive clips, as shown in Fig. 6(b). In these two cases, the GP classifier is hard to exactly classify the states of the clips. The clips are classified as the states with highest probability. To enhance the classify accuracy, the transition information between two traffic states is worth considering. For example, the transition from state in Fig. 8(a) to state in Fig. 8(c) is impossible. To exploit the transition information, we define a new state energy function for clip $t$ as:

$$\begin{aligned} & E(y_t = s_i|y_{t-1} = s_j) \\ & = -log\{p(y_t|\mathbf{c}_t)\} + \beta log\{m_{s_i,s_j}\}(1 - \delta(y_t, y_{t-1})) \end{aligned} \tag{9}$$

$$y_t = \arg\min_{y_t = s_i} E(y_t|y_{t-1}) \tag{10}$$

where $p(y_t|\mathbf{c}_t)$ is the likelihood of the $t^{th}$ clip labeled as state $s_i$ given by Eq. (6). $m_{s_i,s_j}$ is the transition probability from state $s_j$ (the state of previous clip) to $s_i$ (the state of current clip). $\delta(y_t, y_{t-1}) = 1,\ if\ y_t = y_{t-1},\ else\ 0$. $\beta$ is the weight of transition energy and is set experimentally as 0.1. It means that, if the state does not change, we do not need to care about the transition problem. If the state changes in two consecutive clips, the transition information is taken into account and the current clip is labeled with the state which has minimal energy.

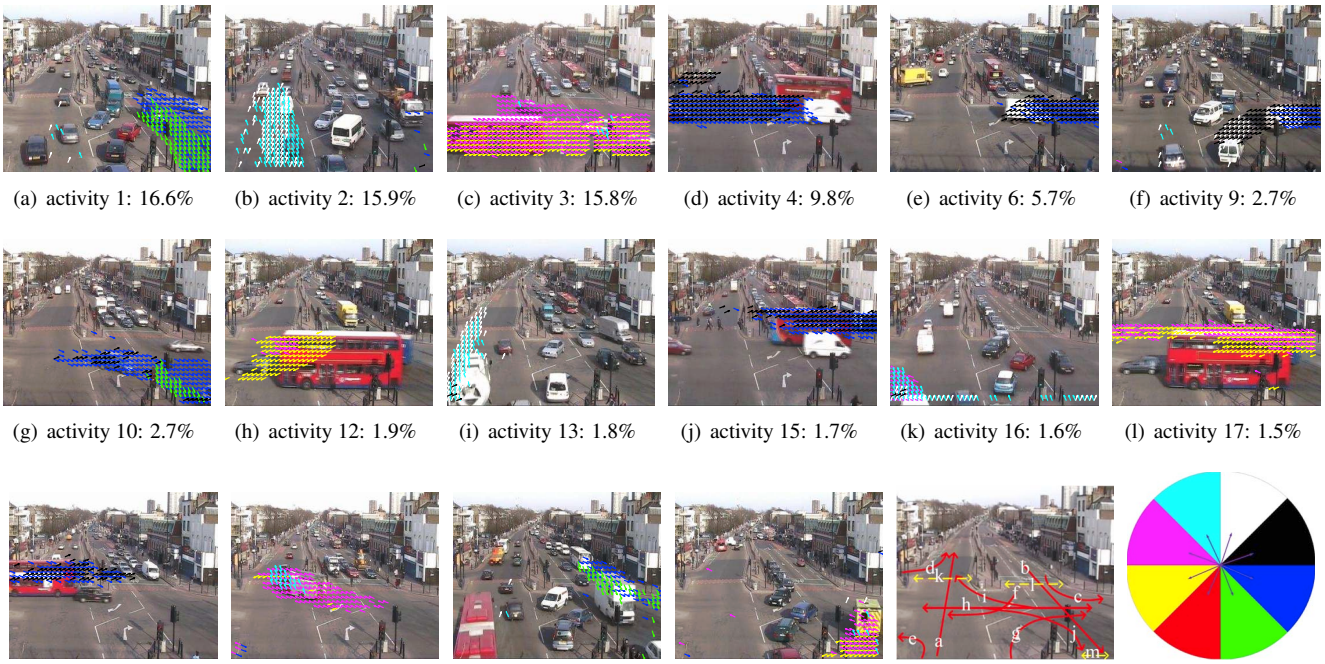
## 4. Experiments

### 4.1. Dataset

Experiments were carried out in real video data from three complex and crowded traffic scenes regulated by the traffic lights. **QMUL Junction Dataset** [7] contains 1 hour of 25 fps video (90000 frames) with frame size $360 \times 288$. The video covers a busy traffic junction containing three major flows in different directions. **QMUL Junction Dataset 2** [7] is 52 minutes of video with 25 fps and frame size is $360 \times 288$. This video covers a busy street with particularly busy pedestrian activity. **MIT Dataset** [22] is 1.5 hour of video with 30fps (162000 frames) and frame size is $720 \times 480$. It covers a far-field traffic scene.

### 4.2. Parameter Setting

For each dataset, the first 500 video clips (about 25 minutes' length) were employed to learn the typical activities

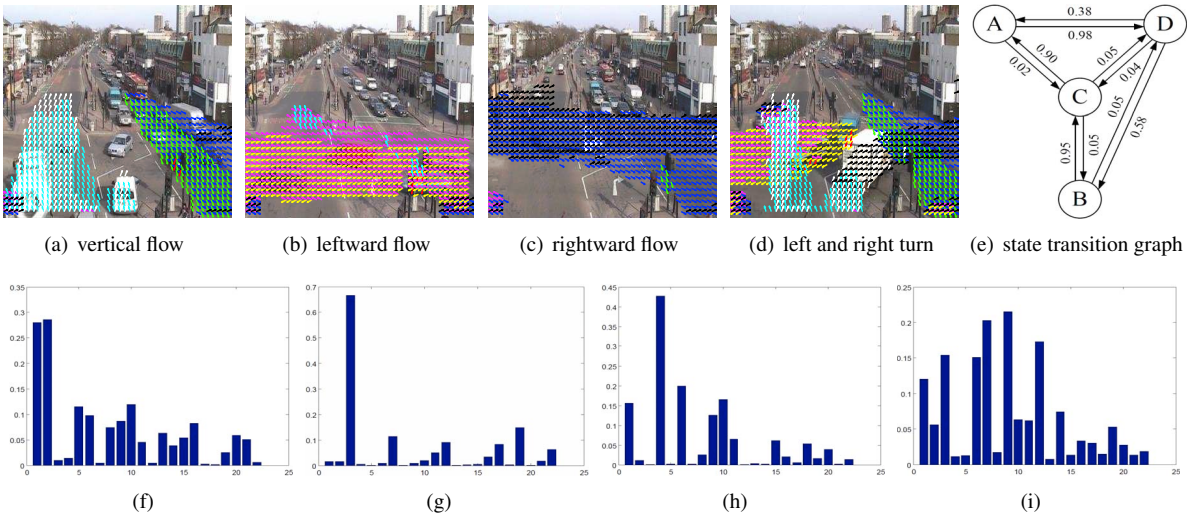


(a) activity 1: 16.6% (b) activity 2: 15.9% (c) activity 3: 15.8% (d) activity 4: 9.8% (e) activity 6: 5.7% (f) activity 9: 2.7%

(g) activity 10: 2.7% (h) activity 12: 1.9% (i) activity 13: 1.8% (j) activity 15: 1.7% (k) activity 16: 1.6% (l) activity 17: 1.5%

(m) activity 18: 1.2% (n) activity 19: 1% (o) activity 20: 1% (p) activity 22: 0.8% (q) all possible lanes (r) quantized directions

Figure 7. (a)-(p) 16 of the 22 most important activity patterns discovered by HDP model in QMUL Junction dataset, shown in order of decreasing importance. (q) Manually labeled legal vehicle driving lanes (red lines) and pedestrian walking lanes (yellow dash lines). (r) quantized directions.



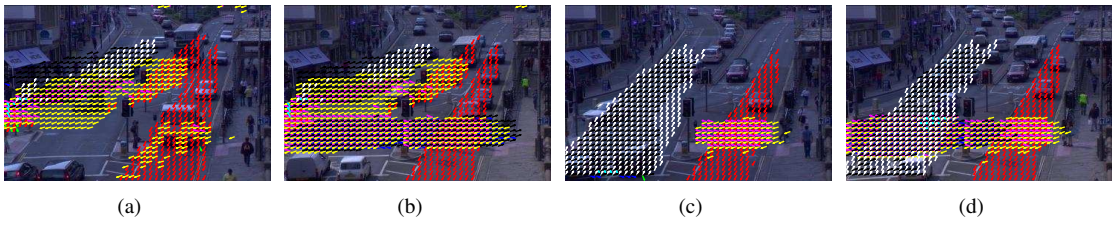

(a) vertical flow (b) leftward flow (c) rightward flow (d) left and right turn (e) state transition graph

(f) (g) (h) (i)

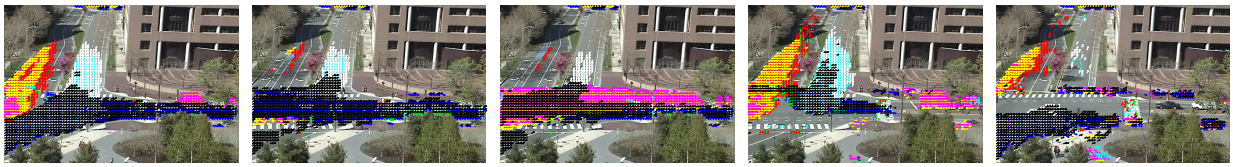
Figure 8. (a)-(b) typical traffic state patterns learned by HDP-HMM model and their corresponding average components of typical activities (f)-(i) throughout the training video sequence. (e) is the state transition graph noted with transition probabilities and directions.

and traffic states. The rest of the video sequences were employed to simulate online video for testing, i.e. 699 clips for QMUL Junction Dataset, 539 clips for QMUL Junction Dataset 2 and 1711 clips for MIT Dataset.

To infer the latent variables under the HDP and HDP-HMM, 1000 sweeps of the Gibbs sampler were executed and the first 500 were used as burn-in. To find the best hyperparameters $(\beta, \alpha)$ for our task, a grid search has been performed on $\beta, \alpha \in \{0.1, 0.5, 1.0, 1.52.0\}$. We analyzed the results with different hyperparameter set. Even though the number of clusters increased with larger $\beta$ and $\alpha$, the numbers of typical activities and states always converged when about least $90\%$ of the total motions were explained. These numbers kept consistent when $\beta$ and $\alpha$ were both



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



(e) (f) (g) (h) (i)

Figure 9. Typical traffic state patterns learned by HDP-HMM model for QMUL Junction Dataset 2 (a)-(d) and MIT Dataset (e)-(i)

Table 2. Comparison of classification results between our method and baseline methods for QMUL Junction Dataset

| State | MCTM [7] | | | | LDA [8] | | | | HMM [8] | | | | Ours | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | L | R | V | VT | L | R | V | VT | L | R | V | VT | L | R | V | VT |
| Left | **.99** | .00 | .00 | .01 | **.49** | .44 | .00 | .06 | **.98** | .00 | .01 | .01 | **1.0** | .00 | .00 | .00 |
| Right | .00 | **.94** | .01 | .05 | .00 | **1.0** | .00 | .00 | .00 | **.92** | .08 | .00 | .00 | **.99** | .00 | .01 |
| Vertical | .00 | .00 | **.77** | .22 | .01 | .17 | **.82** | .00 | .02 | .01 | **.69** | .28 | .00 | .00 | **0.98** | .02 |
| Vertical-Turn | .31 | .05 | .20 | **.43** | .01 | .21 | .30 | **.46** | .49 | .04 | .32 | **.15** | .05 | .00 | .00 | **.95** |
| Average Accuracy | **.78** | | | | **.69** | | | | **.69** | | | | **.98** | | | |

larger than 0.5. The selected typical activities and states look similar. The additional activities and were generated to explain very rare motions. In this paper, we are only interested in typical activities and states and we did not use topic models to estimate likelihood or posterior. Therefore, we did not need precise hyperparameters for our models and the hyperparameters were fixed at $\beta = 2, \alpha = 0.5$ for all experiments.

### 4.3. Learning Typical Activities and States

Due to limited space and for better understanding, we will only analyze the experimental result in QMUL Junction Dataset in detail. The HDP models automatically learned 32 activities in this traffic scene, among which 22 were automatically selected as typical activities (some of them are shown in Fig. 7). For a better illustration, all possible motion flows for vehicles and pedestrians are manually painted and marked with alphabetic letters in Fig. 7(q). For example, the vehicle flow "a" consists of activity 1 and 13.

The HDP-HMM automatically learned 9 traffic states. 4 of them were automatically selected as typical states which have the highest percentage among all training clips, as illustrated in Fig. 8(a)-8(d). For instance, the main components of state vertical flow are activity 1, 2, 13, 20. The corresponding average feature vectors of each typical state through the training video are shown in Fig. 8(f)-8(i). Fig. 8(e) is the state transition graph with transition probabilities and directions. We can see that, the transition from state vertical flow to state rightward flow is very rare.

### 4.4. Traffic State Classification

The learned typical traffic states in QMUL Junction Dataset 2 are shown in Fig. 9(a)-9(d) and the states of MIT Dataset are shown in Fig. 9(e)-9(i).

QMUL Junction Dataset 2 has two main flows and 4 typical states: vehicles driving vertical without (Fig. 9(c)) or with (Fig. 9(d)) pedestrian; vehicles making a turn at the crossing without (Fig. 9(a)) or with (Fig. 9(b)). The MIT Dataset has 5 typical traffic states: Fig. 9(e) explains a busy vertical flow; Fig. 9(e) shows a rightward flow; Fig. 9(g) explains a horizontal flow in two directions; Fig. 9(h) explains vehicles driving downward from top and pedestrian crossing the road; Fig. 9(i) illustrates that vehicles stop behind the crosswalk and pedestrian cross the road.

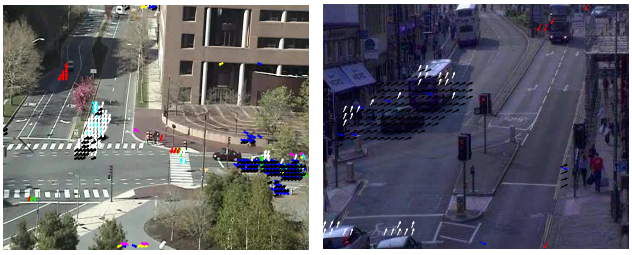
The online screened video sequence was also segmented into clips of 75 frames each. Our experimental results are compared with the baseline methods: Dual-HDP model [22], Markov Clustering Topic Models (MCTM) [7], LDA and HMM. They adopted diverse length of video clip ranging from 1 second to 10 seconds. The experimental

Table 3. Classification performance for the MIT Dataset

| State | | Dual-HDP [22] | | | | | Ours | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | a | b | c | d | e | a | b | c | d | e |
| Manually label | a | 149 | 0 | 2 | 0 | 0 | 610 | 4 | 5 | 0 | 3 |
| | b | 8 | 74 | 4 | 2 | 11 | 3 | 402 | 0 | 2 | 0 |
| | c | 10 | 3 | 60 | 1 | 2 | 3 | 2 | 304 | 2 | 0 |
| | d | 4 | 0 | 2 | 88 | 11 | 7 | 8 | 10 | 222 | 0 |
| | e | 4 | 2 | 6 | 5 | 92 | 6 | 5 | 4 | 8 | 102 |

Table 4. Classification performance for QMUL Dataset 2

| Manually label | Our Classification | | | |
|---|---|---|---|---|
| | a | b | c | d |
| a | 86 | 2 | 1 | 2 |
| b | 2 | 264 | 0 | 4 |
| c | 0 | 0 | 188 | 2 |
| d | 0 | 2 | 0 | 76 |



(a) (b)

Figure 10. Two examples of false classification using GP classifier only

results are directly cited from [8] (for QMUL dataset) and [22] (for MIT dataset). From the comparison in Tab. 2 we see that our model outperforms other three popular methods in terms of classification results in the QMUL dataset. In contrast to the Dual-HDP model in the MIT dataset as listed in Tab. 3, our methods also achieved better classification results. To validate our method, we have executed one more experiment in the QMUL Junction Dataset 2. The results is listed in Tab. 4.

It is worth pointing out that some clips were falsely recognized by traditional GP classifier and corrected by our model. For example, it is ambiguous to determine whether the state in Fig. 10 belongs to state Fig. 9(e) or Fig. 9(f) only based on its appearance. It was falsely classified as the second one with higher probability by GP classifier. Because its previous clip is in the state as Fig. 9(e), it is successfully corrected by using transition information.

The false classification happens when two situations arise. First, the appearance of a clip is ambiguous between two states. Second, the transition probability between these two states is not low. Take the QMUL Junction dataset for example: the transition from state D (left and right turn) to state B (leftward flow) is very normal in this scene. When a clip, in which is state D, contains a clear left turn flow (as shown in Fig. 7(h)) but the other activities are not so clear, this clip is easily classified as state B.

### 4.5. Computation Cost

Since computing the optical flow from each consecutive frames is the most time-consuming process, all optical flow features are computed in advanced. The rest computation cost consists of three parts: 1. off-line learning activities and states using HDP and HDP-HMM models. 2. training GP classifier involving optimization of the hyperparameters. 3. testing the data in the online phase. Our experiments run on a 4 cores CPU of 3.4 GHz with 7 GB RAM with Matlab 2013b. The first part takes about 4 hours on 25 minutes of video clips at a resolution of $360 \times 288$ pixels and the second parts takes about 2 minutes. For testing 35 minutes of video clips, it takes only about 1 second.

## 5. Conclusion

In this paper, we have presented a novel framework for automatic video event analysis and recognition by combining HDP and GP. We employ the unsupervised non-parametric model-HDP to learn the typical activities and states from training video. We propose to represent activities using local motion and states using activity patterns. Then a training dataset is generated to train the GP classifier. Furthermore, the transition information learned by HDP-HMM has been effectively integrated into GP classifier. Our model is validated in three real-world datasets and the experimental results are compared with the baseline methods. In the future, we will extend our model to recognize abnormal activities in online video.

## ACKNOWLEDGEMENTS

The work is partially funded by DFG (German Research Foundation) YA 351/2-1. The authors gratefully acknowledge the support.

## References


[1] Y. Altun, T. Hofmann, and A. J. Smola. Gaussian process classification for segmenting and annotating sequences. In *ICML*, page 4, 2004.

[2] M. R. Amer, P. Lei, and S. Todorovic. Hirf: Hierarchical random field for collective activity recognition in videos. In *ECCV*, pages 572–585, 2014.

[3] C. M. Bishop. *Pattern recognition and machine learning*. springer, 2006.

[4] K. M. Chathuramali and R. Rodrigo. Faster human activity recognition with svm. In *2012 international Conference on Advances in iCT for Emerging Regions (iCTer)*, pages 197–203, 2012.

[5] D. Ellis, E. Sommerlade, and I. Reid. Modelling pedestrian trajectory patterns with gaussian processes. In *ICCV Workshop*, pages 1229–1234, 2009.

[6] R. Emonet, J. Varadarajan, and J.-M. Odobez. Temporal analysis of motif mixtures using dirichlet processes. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 36(1):140–156, 2014.

[7] T. Hospedales, S. Gong, and T. Xiang. A markov clustering topic model for mining behaviour in video. In *ICCV*, pages 1165–1172, 2009.

[8] T. Hospedales, S. Gong, and T. Xiang. Video behaviour mining using a dynamic topic model. *International journal of computer vision*, 98(3):303–323, 2012.

[9] T. M. Hospedales, J. Li, S. Gong, and T. Xiang. Identifying rare and subtle behaviors: A weakly supervised joint topic model. *PAMI*, 33(12):2451–2464, 2011.

[10] D. H. Hu, X.-X. Zhang, J. Yin, V. W. Zheng, and Q. Yang. Abnormal activity recognition based on hdp-hmm models. In *IJCAI*, pages 1715–1720, 2009.

[11] Y.-G. Jiang, S. Bhattacharya, S.-F. Chang, and M. Shah. High-level event recognition in unconstrained videos. *International Journal of Multimedia Information Retrieval*, 2(2):73–101, 2013.

[12] K. Kim, D. Lee, and I. Essa. Gaussian process regression flow for analysis of motion trajectories. In *ICCV*, pages 1164–1171, 2011.

[13] D. Kuettel, M. D. Breitenstein, L. Van Gool, and V. Ferrari. What's going on? discovering spatio-temporal dependencies in dynamic scenes. In *CVPR*, pages 1951–1958, 2010.

[14] C. Liu. Beyond pixels: Exploring new representations and applications for motion analysis. *PhD Thesis., MIT*, 2009.

[15] J. Nocedal and S. Wright. Numerical optimization, series in operations research and financial engineering. *Springer, New York, USA*, 2006.

[16] C. E. Rasmussen. *Gaussian processes for machine learning*. The MIT Press, 2006.

[17] K. Tang, B. Yao, L. Fei-Fei, and D. Koller. Combining the right features for complex event recognition. In *ICCV*, pages 2696–2703, 2013.

[18] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei. Hierarchical dirichlet processes. *Journal of the American Statistical Association*, 101(476), 2006.

[19] J. Wang, W. Fu, H. Lu, and S. Ma. Bilayer sparse topic model for scene analysis in imbalanced surveillance videos. *Image Processing, IEEE Transactions on*, 23(12):5198–5208, 2014.

[20] J. M. Wang, D. J. Fleet, and A. Hertzmann. Gaussian process dynamical models for human motion. *PAMI*, 30(2):283–298, 2008.

[21] X. Wang and Q. Ji. A hierarchical context model for event recognition in surveillance video. In *CVPR*, pages 2561–2568, 2014.

[22] X. Wang, X. Ma, and W. E. L. Grimson. Unsupervised activity perception in crowded and complicated scenes using hierarchical bayesian models. *PAMI*, 31(3):539–555, 2009.

[23] C. K. Williams and D. Barber. Bayesian classification with gaussian processes. *PAMI*, 20(12):1342–1351, 1998.