creative commons
COMMONS DEED

저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:

저작자표시. 귀하는 원저작자를 표시하여야 합니다.

비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.

변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 이용허락규약(Legal Code)을 이해하기 쉽게 요약한 것입니다.

Disclaimer

dCollection

Ph.D. DISSERTATION

Online Inference Model for Traffic Pattern Analysis and Anomaly Detection

교통 패턴 분석과 비정상 탐지를 위한 온라인 추론 모델

BY

Hawook Jeong

FEBRUARY 2015

DEPARTMENT OF ELECTRICAL ENGINEERING AND
COMPUTER SCIENCE
COLLEGE OF ENGINEERING
SEOUL NATIONAL UNIVERSITY

Ph.D. DISSERTATION

Online Inference Model for Traffic Pattern Analysis and Anomaly Detection

교통 패턴 분석과 비정상 탐지를 위한 온라인 추론 모델

BY

Hawook Jeong

FEBRUARY 2015

DEPARTMENT OF ELECTRICAL ENGINEERING AND
COMPUTER SCIENCE
COLLEGE OF ENGINEERING
SEOUL NATIONAL UNIVERSITY

Online Inference Model for Traffic Pattern Analysis and Anomaly Detection

교통 패턴 분석과 비정상 탐지를 위한 온라인 추론 모델

지도교수 최 진 영

이 논문을 공학박사 학위논문으로 제출함

2014 년 11 월

서울대학교 대학원

전기 컴퓨터 공학부

정하욱

정하욱의 공학박사 학위논문을 인준함

2014 년 12 월

위 원 장	조 남 익	
부위원장	최 진 영	
위 원	오 성 회	
위 원	곽 노 준	
위 원	강 훈	

Abstract

In this thesis, we propose a method for modeling trajectory patterns with both regional and velocity observations through the probabilistic inference model. By embedding Gaussian models into the discrete topic model framework, our method uses continuous velocity as well as regional observations unlike existing approaches. In addition, the proposed framework combined with Hidden Markov Model can cover the temporal transition of the scene state, which is useful in checking a violation of the rule that some conflict topics (e.g. two cross-traffic patterns) should not occur at the same time. To achieve online learning even with the complexity of the proposed model, we suggest a novel learning scheme instead of collapsed Gibbs sampling. The proposed two-stage greedy learning scheme is not only efficient at reducing the search space but also accurate in a way that the accuracy of online learning becomes not worse than that of the batch learning. To validate the performance of our method, experiments were conducted on various datasets. Experimental results show that our model explains satisfactorily the trajectory patterns with respect to scene understanding, anomaly detection, and prediction.

Keywords: trajectory analysis, topic model, latent Dirichlet allocation, surveillance
Student Number: 2011-30975

Contents

List of Figures

Chapter 1

Introduction

1.1 Statement of Problem

The number of surveillance cameras is increasing all around the world for safety and security in both public and private environments, such as airports, train stations, highways, parking lots, markets, offices, and so on. Because of the large number of cameras, it is very important to develop intelligent visual surveillance systems to process a large amount of data obtained from the cameras in real-time and fully automatically. For this reason, intelligent visual surveillance has been one of the most active research issues in computer vision recently. The intelligent visual surveillance includes various tasks: 1) to detect and recognize objects of interest (Stauffer & Grimson, 1999; Chang et al., 2012; Cui et al., 2012; Dalal & Triggs, 2005; Dollar et al., 2012), 2) to track the moving objects in surveillance scenes (Rodriguez et al., 2009; Kuo et al., 2010; Yang & Nevatia, 2014; Benfold & Reid, 2011; Qin & Shelton, 2012), and 3) to understand and describe the activity patterns of the moving objects (Basharat et al., 2008; Hospedales et al., 2009; Hu et al., 2006; Morris & Trivedi, 2008; Piciarelli &

Figure 1.1 Examples of various traffic scenes. They have various perpective angles, crowd densities, sizes of moving agents, and rules of normal patterns.

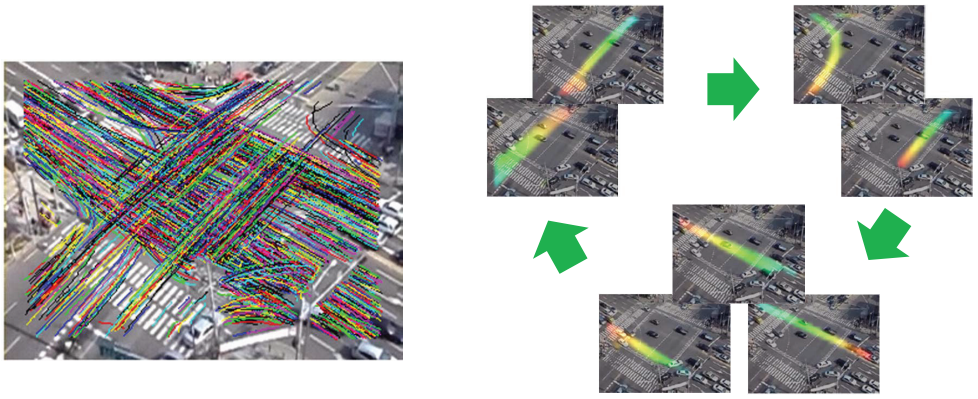
Foresti, 2006; Wang et al., 2009, 2006). Among the tasks, understanding the activity patterns can have a wide variety of applications, especially in traffic scenes such as accident prediction and detection, traffic control, scene structures estimation, and traffic violation detection.

Figure 1.1 shows examples of various traffic scenes. As shown in the figure, modeling activity patterns in realistic traffic surveillance scenes is very challenging because they have various perspective angles, crowd densities, sizes of moving agents, and rules of normal patterns. This variety of scenes makes it difficult to generalize typical path patterns of moving objects without considering the scene specific properties, so activity patterns should depend on each scene. However, it is very expensive and impractical to obtain labeled motion data (e.g. trajectories) by human labor whenever

new camera is installed in a specific traffic scene. In the case of realistic traffic videos, annotating activities is especially difficult because multiple other activities happen simultaneously. Therefore, for the sake of understanding the traffic scenes, unsupervised analysis of motion patterns without prior knowledge or manual efforts is essentially required.

In most cases, moving objects follow specific motion patterns; for example, most cars and pedestrians move according to specific traffic rules. The goal of motion pattern analysis algorithms is to learn the implicit traffic rules of the surveillance scene in an unsupervised way from a large amount of crude data as shown in Figure. 1.2. Using a data-driven perspective, the term "anomaly" and "abnormal events" are defined as outliers that are far from the typical patterns (e.g. go straight, U-turn, turn right, etc.) explained using the training data following the traffic rules. Hence, the terminology "anomaly detection" in this thesis becomes a process of finding motions which do not obey these rules. In other words, applications in traffic scenes such as accident and traffic violation detection can be fulfilled by anomaly detection. Many researchers have proposed various learning models to discover the typical normal motion patterns from raw data in surveillance video (Basharat et al., 2008; Emonet et al., 2011; Hospedales et al., 2009; Hu et al., 2006; Kuettel et al., 2010; Morris & Trivedi, 2008; Piciarelli & Foresti, 2006; Wang et al., 2009, 2006).

Through analyzing strength and weakness of the existing works on unsupervised learning of motion patterns, we establish the following five requirements that the learning model should satisfy to work well in actual environments. First, the model should recognize regions showing normal movement patterns. The regions should be categorized into semantic regions representing typical activities (e.g. go straight upward, turn right, walk across the street, etc.). This is important for explaining the activities in an intersection, detecting intrusions of restricted areas, and detecting illegal U-turns. Second, the model should include not only direction information but also speed in-



(a) (b)

Figure 1.2 An example of motion pattern analysis. (a) Crude motion data (unlabeled trajectories) in a surveillance scene. Note that a large number of trajectories are broken. (b) Results of learning typical activities. The typical patterns are denoted with red and blue coloring, where objects move from red to blue. Some typical patterns occur at the same time, and their occurrences have temporal rules. (best viewed in color)

formation for each activity regions. This would increase the discrimination ability of the model to detect abnormal patterns such as pedestrians walking along the path of vehicles, bikes running in pedestrian road, cars driving with over speed, cars stopping in a railroad crossing, and so on. Third, spatio-temporal relationship between typical activity patterns needs to be considered. For instance, it is impossible for two straight movements, “moving from left to right” and “moving from top to bottom,” to occur in an intersection at the same time. The model also needs to recognize the temporal order of activities such as governed by a traffic signal. Fourth, the algorithm should be robust to crowded scenes. In crowded scenes, it is hard to extract motions of individual objects. Even the current state-of-the-art methods for multi-object tracking (Qin & Shelton, 2012; Walk et al., 2010) are still limited for applying to the crowded scenes.

Fifth, the model should be able to adapt itself to temporal changes of the scene (e.g. reversible lane, traffic volume changes). Online learning approach will not only enable the adaptation but also save memory and computational load because the model does not need to keep old data. A surveillance system running over months or even years, for example, would require an online model if it needs to keep running.

According to the authors' survey, there is no existing work satisfying all of the aforementioned requirements until now, the details on this issue will be described in related works of Section 1.2 and here we would give a brief mention. Object tracking based approach (Wang et al., 2006; Hu et al., 2006; Piciarelli & Foresti, 2006; Morris & Trivedi, 2008; Basharat et al., 2008), whose observations are actual velocity from trajectories, can satisfy the first and second requirements but hardly fulfill the third and fourth requirements. On the other hand, the topic model based approach (Hospedales et al., 2009; Kuettel et al., 2010; Emonet et al., 2011; Wang et al., 2009), whose observations are quantized directions in a local region, are particularly useful for the first, third and fourth requirements. This kind of observations, however, cannot deal with precise velocities (second requirement). Furthermore, most of the motion learning methods are restricted to offline learning not allowing to adapt to the changing situations (fifth requirement). The crowd motion approach (Kratz & Nishino, 2009; Rodriguez et al., 2009; Wang et al., 2012) does not fulfill the first and third requirements since it is designed to understand only the crowd motion rather than typical motion paths.

1.2 Related Works

1.2.1 Motion Pattern Analysis Using Trajectory

One of the conventional approaches used for unsupervised activity analysis is to learn trajectory patterns through measuring pairwise distances of trajectories and clustering.

This approach utilizes distance measure between two different trajectories and groups similar ones together. The existing trajectory distance measures include Euclidean distance (Fu et al., 2005), Hausdorff distance and its variations (Junejo et al., 2004; Wang et al., 2006), hidden Markov model (Porikli & Porikli, 2004), Dynamic Time Warping (DTW) (Keogh & Pazzani, 2000), and so on. When computing the trajectory distance, some methods require two trajectories to be temporally aligned for long common subsequence (LCSS) analysis (Vlachos et al., 2002; Buzan et al., 2004). On the other hand, (Piciarelli & Foresti, 2006) have proposed a distance measure matching only a part of the trajectory (only an overlapped part), instead of matching all points on a trajectory. Based on the computed similarity matrix among trajectories, standard clustering algorithms such as spectral clustering (Wang et al., 2006; Ng et al., 2001), graph-cuts (Shi & Malik, 1997), agglomerative and divisive hierarchical clustering (Li et al., 2006; Antonini & Member, 2006), and fuzzy c-means (Hu et al., 2006) were used to categorize trajectories into different activity patterns. A comparison of various distance measurements and clustering methods can be found in (Morris & Trivedi, 2008, 2009).

Since these methods using distance measures to group similar trajectories can model trajectories in a whole path, they can deal with the long term characteristics of trajectories. However, these distance-based approach has several drawbacks. First, these methods suffer from errors due to a perspective projection distortion which is caused when three-dimensional space is projected on a two-dimensional surface. Because of the distortion, similar trajectories in 3-D space can be considered relatively different in the 2-D video, whereas different trajectories in 3-D space can looks like similar in the 2-D video as shown in Figure 1.3. Second, these methods are vulnerable to fragmentation of trajectories. Due to inevitable tracking failure, there exist broken trajectories which do not overlapped at all but belong to the same activity pattern. Thus, it is very difficult to define distance measures that make these broken trajectories to be close without losing generality and objectiveness. Third, the computation to

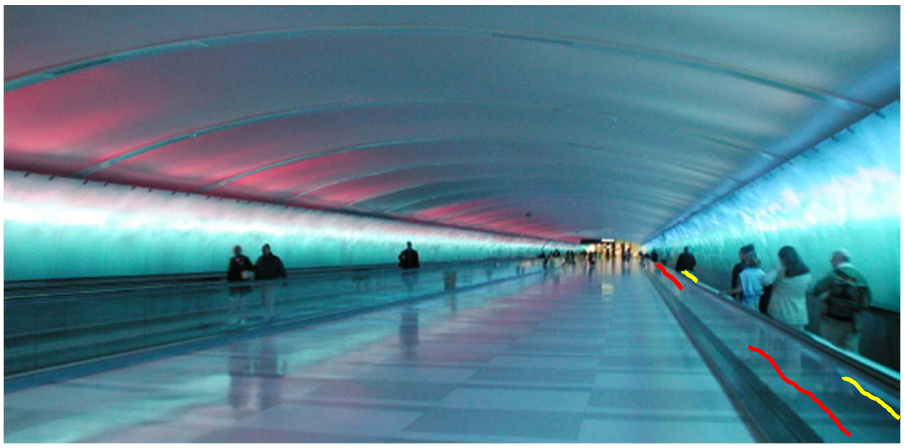


Figure 1.3 An example of perspective projection distortion. In this scene, parallel lines appear to converge, so similar pairs of trajectory in 3-D space looks different in 2-D surface.

obtain the distance for every pair of trajectories is heavy, with complexity of $O(N^2)$ in both time and space, where N is the number of trajectories. Moreover, some clustering algorithms such as spectral clustering need to compute the eigenvectors and eigenvalues of the similarity matrix, and their computational cost will be even high. When it comes to a memory issue, since visual surveillance systems often require processing data collected over weeks or even months, it is impossible to load such a huge similarity matrix into memory of a common personal computer. Fourth, this approach lacks a probabilistic explanation of activity patterns happening in the scene. Abnormal trajectories in this approach are simply detected if those have larger distance to all trajectory clusters, so spatio-temporal relationship among trajectory patterns does not considered.

Another kind of approach converts trajectories into feature vectors instead of computing pairwise distances for clustering. Since the trajectories have various length, it is difficult to directly use them as feature vectors. Therefore, sub-sampling can be ap-

plied to make all the trajectories have the same length (Makris & Ellis, 2002; Liao et al., 2006; Hu et al., 2007). Then, the feature vectors of trajectories were clustered using algorithms such as k-means (MacQueen, 1967) and neural networks (Sumpter & Bulpitt, 1998; Hu et al., 2004). However, these methods are also vulnerable to fragmentation of trajectories and perspective projection distortion.

Alternatively, some methods learn the transition probabilities of each pixel to its nearby pixels using Gaussian mixture models (GMM) (Basharat et al., 2008) or kernel density estimation (KDE) (Saleemi et al., 2009). In this methods, transitions of the state (previous location, size, and passing time) of an object on a trajectory are represented as feature vectors. Thus, these methods enable to statistically learn the velocities and the sizes of moving objects at each position. They are more invariant to scene variation and more robust to trajectory fragmentation and perspective projection distortion than distance-based approach. However, these methods may fail to detect anomalies in regions where movements are diverse, such as the center of an intersection. In such situations, the trained model would count all patterns as normal because they are not fully aware of mutual dependence among trajectories; that is, they cannot handle spatio-temporal relationship among typical activity patterns (*i.e.*, they do not fulfill the third requirement).

1.2.2 Motion Pattern Analysis Using Local Motions

Local motion based methods have been proposed recently to overcome the problem of object tracking failure in a crowded scene. These methods adopt mixture of Gaussians (Saleemi et al., 2010), sparse coding (Zhao et al., 2011), Markov random field (Benezeth et al., 2011), dynamic textures (Mahadevan et al., 2010), probabilistic topic models (Wang et al., 2009; Hospedales et al., 2009; Kuettel et al., 2010; Emonet et al., 2011; Varadarajan et al., 2012), and so on. In particular, the topic models have been prevalently employed to learn motion patterns because they can well discover typ-

ical activities using co-occurrence property. The Dual Hierarchical Dirichlet Process (Dual-HDP) (Wang et al., 2009) discovers typical activity patterns by modeling spatial relation of activities. Markov Clustering Topic Model (MCTM) (Hospedales et al., 2009) additionally considers temporal relationships between activities, and Dependent Dirichlet Process Hidden Markov Model (DDP-HMM) (Kuettel et al., 2010) solves the same problem in a non-parametric manner. However, the above methods ignore the temporal order of low-level motion features, which leads to incomplete modeling of long-term path. This approach has been extended by considering the temporal information inside the topic (Emonet et al., 2011; Varadarajan et al., 2012). Nevertheless, all of these topic model based approaches cannot completely address the precise velocity of a whole trajectory since they only use quantized directions obtained from optical flows in a local cell (*i.e.*, it does not fulfill the second requirements). Moreover, the collapsed Gibbs sampling, which is commonly utilized for learning of the topic models, is not only ineffective in dealing with a large solution space of a complex model but also restricted to offline learning making it unable to adapt to a changing situation (*i.e.*, it does not fulfill the fifth requirements).

Crowd motion analysis (Kratz & Nishino, 2009; Rodriguez et al., 2009; Wang et al., 2012) has also been conducted to detect strange motion patterns in an extremely crowded scene. Probabilistic Crowd Model (Rodriguez et al., 2009) allows objects to be tracked even in extremely crowded scenes, and local spatio-temporal motion pattern (Kratz & Nishino, 2009; Wang et al., 2012) is modeled in the dense crowded scenes. These methods, however, allow their model to understand only the crowd motion rather than typical motion paths (*i.e.*, it does not fulfill the first and the third requirements). Hence, this approach is not suitable for the task of deducing traffic rules though it gives good performance on anomaly detection in the crowded scene.

1.3 Contributions

In this thesis, we propose an approach to meet all of the aforementioned requirements for motion pattern analysis. This purpose is achieved through embedding the precise velocity pattern model, spatio-temporal pattern transition model, and the topic model into a probabilistic graphical framework. In particular, the newly defined continuous velocity model is distinctive from the existing models (Wang et al., 2009; Hospedales et al., 2009; Kuettel et al., 2010; Emonet et al., 2011; Varadarajan et al., 2012; Wang et al., 2011), which do not provide satisfactory performance on the second requirement. In addition, to achieve online and real-time learning even with the enormous complexity of the proposed model, we suggest an efficient two-stage greedy learning method. The learning method of collapsed Gibbs sampling (Griffiths & Steyvers, 2004) restricts the existing models to offline learning. On the other hand, our learning method is designed to infer latent variables step by step in a greedy manner to reduce the search space. Moreover, the sub-model in each step is learned in a way that the online learning should not lose the learning capabilities shown in the offline learning. The whole learning process allows online adaptation of the model quickly and accurately. We evaluate our method on six datasets for activity pattern modeling and anomaly detection, showing that our method outperforms the state-of-the-art methods.

1.4 Thesis Organization

We provide an organization and overview, which are considered by subsequent thesis chapters. In chapter 2, as for the preliminaries, we briefly review the Latent Dirichlet Allocation (LDA) approach and explain how the LDA wards and can be applied to computer vision applications. Then, we will address two representative approximate inference methods for LDA (variational inference and collapsed Gibbs sampling). Chapter 3 addresses the proposed probabilistic inference model to analyze tra-

jectory patterns in traffic scene and to detect abnormal activities. The proposed inference model is formulated in a probabilistic graphical framework including trajectory pattern model, spatio-temporal relation of trajectories, and velocity model of each trajectory pattern. In addition, we suggest a approximate learning scheme instead of collapsed Gibbs sampling that is conventionally utilized in the existing methods. Lastly, the detection procedure is described for the recently observed scene to be tested by the trained model to detect anomalies in the current scene. Chapter 4 presents experimental details (both quantitatively and qualitatively). In chapter 5, we conclude by summarizing the contributions of this thesis, and briefly mention directions for the future research.

Chapter 2

Preliminaries

In this chapter, we present the theoretical background of Latent Dirichlet Allocation (LDA) which is a baseline of the proposed model and is helpful to understand the rest of the thesis. If the reader is already familiar with this field, this chapter can be skipped. For details and theoretical proofs, refer to the cited literatures.

2.1 Latent Dirichlet Allocation (LDA)

2.1.1 Probabilistic Graphical Model

Before addressing Latent Dirichlet Allocation (LDA), we explain the foundation of probabilistic graphical models to describe the notations, the independence assumptions of the models, the principle of maximum a posteriori (MAP), and Bayesian inference through a simple example (Griffiths et al., 2008). A probabilistic graphical model can provide an efficient and intuitive framework for describing high-dimensional probability distributions: nodes denote random variables and directed edges denote possible dependence between the random variables, and plates denote replication of a substruc-

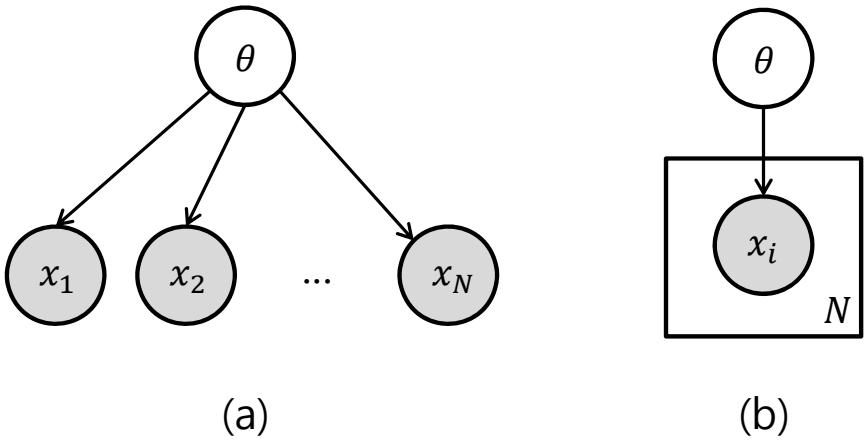


Figure 2.1 (a) An example of graphical models. Nodes denote random variables and directed edges denote possible dependence between the random variables. Observed random variables are shaded, and latent random variables are unshaded. (b) The equivalent graphical model with (a) using the plate notation.

ture inside the plates. Also, the probabilistic graphical models can be used to describe latent variable models (Blei, 2014) which is a method of developing complicated structured probability distributions, where the observed (known) variables interact with latent random variables. In the conventional notations of the latent variable models, observed random variables are shaded, and latent random variables are unshaded.

Figure 2.1 shows an example of a graphical model that could generate a flip sequence of a biased coin. In the figure, observed variables $x_1, x_2, ..., x_N$ are binary random variables that stand for the outcomes of N number of successive tosses ($x_i = 1$ if the coin produces head; $x_i = 0$ otherwise.), and θ is a latent random variable with range of 0 to 1 which represents the bias of a coin (i.e. if the coin is fair, then $\theta = 0.5$). The latent variable θ can be considered a model parameter that needs to be estimated as well. The edges express the probabilistic dependencies between the variables; in other words, conditioned on the parent θ, each variable x_i is independent with all other vari-

ables. Thus, since heads of coin occurs with probability of θ and tails occurs with $1-\theta$ on each flip, the probability of a particular flip sequence of a biased coin with N_H heads and N_T tails given θ is

$$p(x_1, x_2, ..., x_N|\theta) = \prod_{i=1}^{N} p(x_i|\theta) \tag{2.1}$$

$$= \theta^{N_H}(1-\theta)^{N_T}, \tag{2.2}$$

which is regarded as a likelihood. Also, applying a consequence of probabilistic dependencies of the graphical model, the full joint probability distribution can be factorized as follows:

$$p(x_1, x_2, ..., x_N, \theta) = p(x_1, x_2, ..., x_N|\theta)p(\theta) \tag{2.3}$$

$$= p(\theta)\prod_{i=1}^{N} p(x_i|\theta) \tag{2.4}$$

$$= p(\theta)\theta^{N_H}(1-\theta)^{N_T}. \tag{2.5}$$

In order to estimate the best θ given a flip sequence of a biased coin $\{x_1, x_2, ..., x_N\}$, the principle of maximum a posteriori (MAP) is applied as follows:

$$\hat{\theta} = \arg\max_{\theta} p(\theta|x_1, x_2, ..., x_N). \tag{2.6}$$

As given by the Eq. 2.6, MAP maximizes the posterior probability. According to the posterior probability $p(\theta|x_1, x_2, ..., x_N)$, we can apply Bayes' rule to obtain

$$\underbrace{p(\theta|x_1, x_2, ., x_N)}_{posterior} = \frac{\overbrace{p(x_1, x_2, ., x_N|\theta)}^{likelihood}\overbrace{p(\theta)}^{prior}}{\underbrace{p(x_1, x_2, ., x_N)}_{evidence}}, \tag{2.7}$$

where

$$p(x_1, x_2, ..., x_N) = \int_0^1 p(x_1, x_2, ..., x_N|\theta)p(\theta)d\theta. \tag{2.8}$$

As given by the Eq. 2.7, different choices of the prior $p(\theta)$ will lead to different inference results about the value of θ. In this example, two types of prior will be addressed: uniform prior and beta distribution.

Uniform prior If a prior $p(\theta)$ is assumed to be uniform, $p(\theta)$ is equal for all range of 0 to 1, so $p(\theta) = 1$ if $\theta \in [0, 1]$. Therefore, the posterior probability $p(\theta|x_1, x_2, ..., x_N)$ can be rewritten by substituting Eq. 2.2 and Eq. 2.8 as follows:

$$p(\theta|x_1, x_2, ..., x_N) = \frac{p(x_1, x_2, ..., x_N|\theta)}{p(x_1, x_2, ..., x_N)} \tag{2.9}$$

$$= \frac{\theta^{N_H}(1-\theta)^{N_T}}{\int_0^1 \theta^{N_H}(1-\theta)^{N_T} d\theta}. \tag{2.10}$$

The denominator can be calculated using a little calculus of integral, which lead to a constant value,

$$\int_0^1 \theta^{N_H}(1-\theta)^{N_T} d\theta = \frac{(N_H!N_T!)}{(N_H + N_T + 1)!}. \tag{2.11}$$

Thus, the optimal $\hat{\theta}$ is determined by finding θ that maximizes the likelihood function $p(x_1, ..., x_N|\theta) = \theta^{N_H}(1-\theta)^{N_T}$. Then, we can find the analytic solution for this problem by differentiating the likelihood function as follows:

$$\frac{dp(x_1, ..., x_N|\theta)}{d\theta} = \{N_H - (N_H + N_T)\theta\} \left\{\theta^{N_H - 1}(1-\theta)^{N_T - 1}\right\}. \tag{2.12}$$

From the above equation, we can conclude that the optimal $\hat{\theta}$ is $\frac{N_H}{N_H + N_T}$. For example, if a coin flip sequence "HHHHHHHHHH" is observed, the optimal $\hat{\theta}$ will be 1; on the other hands, if a coin flip sequence "HTHTHHTTHH" is observed, the optimal $\hat{\theta}$ will be 0.6. However, the estimated $\hat{\theta}$ is not reliable if we observe only a few flips such as "HH" or "TH". To deal with the above problem, we can consider better intuition that we might have about θ, rather than using the prior of uniform $p(\theta)$.

Beta distribution prior In this case, we will use a beta distribution as a prior which can give stronger intuition about the value of θ. Beta distribution is a family of continuous probability distributions defined on the interval [0, 1] parametrized by two positive shape parameters that control the shape of the distribution. The probability density function of the beta distribution given parameter α, β is defined as follows:

$$p(\theta) = Beta(\theta|\alpha, \beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\theta^{\alpha-1}(1-\theta)^{\beta-1}, \tag{2.13}$$

where $\Gamma(\alpha) = \int_0^\infty x^{\alpha-1}e^{-x}dx$ is gamma function which satisfies the following property: $\Gamma(\alpha+1) = \alpha\Gamma(\alpha)$; that is, $\Gamma(\alpha)$ is equivalent to $(\alpha-1)!$ when α is a positive integer. As shown in Figure 2.2, an estimation of θ is influenced by not only an observed flip sequence $\{x_1, x_2, ..., x_N\}$ but also the prior distribution determined by a selection of α, β.

The posterior probability can be written by substituting for the likelihood $p(x_1, ..., x_N|\theta)$ and beta distribution prior $p(\theta)$ as follows:

$$\begin{aligned} p(\theta|x_1, x_2, ..., x_N) &= \frac{p(x_1, x_2, ..., x_N|\theta)p(\theta)}{p(x_1, x_2, ..., x_N)} \\ &= \frac{\left\{\theta^{N_H}(1-\theta)^{N_T}\right\}\left\{\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\theta^{\alpha-1}(1-\theta)^{\beta-1}\right\}}{p(x_1, x_2, ..., x_N)} \\ &= \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\frac{\theta^{N_H+\alpha-1}(1-\theta)^{N_T+\beta-1}}{p(x_1, x_2, ..., x_N)}. \end{aligned} \tag{2.14}$$

Since the denominator $p(x_1, x_2, ..., x_N) = \int_0^1 p(x_1, x_2, ..., x_N|\theta)p(\theta)d\theta$ and $\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}$ is a constant with the variation of θ, the MAP problem can be summarized as follows:

$$\begin{aligned} \hat{\theta} &= \arg\max_\theta p(\theta|x_1, x_2, ..., x_N) \\ &= \arg\max_\theta \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\frac{\theta^{N_H+\alpha-1}(1-\theta)^{N_T+\beta-1}}{p(x_1, x_2, ..., x_N)} \\ &= \arg\max_\theta \theta^{N_H+\alpha-1}(1-\theta)^{N_T+\beta-1}. \end{aligned} \tag{2.15}$$

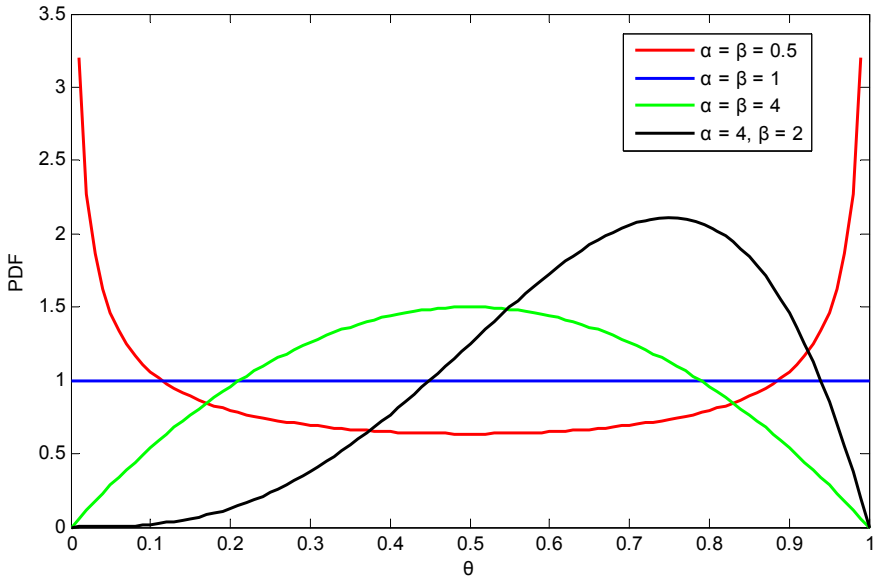


Figure 2.2 Probability density function of beta distribution. By setting α, β with prior assumption, we can control the estimation of θ. If $\alpha = \beta = 1$, $p(\theta)$ is equivalent to the uniform prior.

Then, we can find the analytic solution for this problem by computing derivative of only the $\theta^{N_H+\alpha-1}(1-\theta)^{N_T+\beta-1}$ with respect to θ:

$$\frac{dp(\theta|x_1,...,x_N)}{d\theta} \propto \frac{d\left\{\theta^{N_H+\alpha-1}(1-\theta)^{N_T+\beta-1}\right\}}{d\theta} \tag{2.16}$$

$$= \{N_H+\alpha-1-(N_H+N_T+\alpha+\beta-2)\theta\} \times \left\{\theta^{N_H+\alpha-2}(1-\theta)^{N_T+\beta-2}\right\}. \tag{2.17}$$

Therefore, we can conclude that the optimal $\hat{\theta}$ is $\frac{N_H+\alpha-1}{N_H+N_T+\alpha+\beta-2}$ $(0 < \theta < 1)$. Due to the effect of the prior, we obtain different estimation of the optimal $\hat{\theta}$ with the same observation sequence. For instance, if we set $\alpha = \beta = 100$ with confidence that the coin is fair, the estimated $\hat{\theta}$ is $\frac{102}{201} \approx 0.507$ when observing the coin sequence "HTH". This result is totally different from the case of assuming the uniform prior,

$\hat{\theta} = \frac{2}{3} \approx 0.67$. When applying the beta distribution prior, estimation of the parameter θ is affected not only by observation data but also by prior knowledge (user-setting of α, β). Similarly, if we set $\alpha = \beta = 0.1$ with confidence that the coin is highly biased to one side but we do not know which side it is, the estimated $\hat{\theta}$ is $\frac{1.1}{1.2} \approx 0.92$ when observing the same coin sequence "HTH". Consequently, a prior plays a role of smoothing or regularizing the observed data, preventing the estimated latent variables from over-fitting when the data are far from the prior knowledge which is presumed.

The basic principles of probabilistic graphical models (notations, independence assumptions, and Bayesian inference) explained in the above example can help to understand LDA model that will be described in the subsequent subsection.

2.1.2 LDA Property & Formulation

Latent Dirichlet Allocation (LDA) (Blei et al., 2003) is a hierarchical probabilistic graphical model which is widely used for a natural language processing. LDA is also known as a *topic model* that is used to analyze relationships between a set of documents and words composing the documents. The documents and words are observations of LDA, and the relationships are demonstrated by topics (latent thematic random variables for a document). The topic model is a type of statistical model that discovers a distribution of topics in a document given a set of documents consisting of words, where a topic can be explained by a probability distribution over words. The model assumes a probabilistic generative process that specifies how words in documents can be generated on the basis of latent variables of LDA. In order to generate words in a document, a distribution over topics is chosen; then, a topic is generated according to this distribution, and a word in the document is generated from the topic.

The generative process for LDA can be easily explained by the example of Figure 2.3 with an assumption that all the latent variables and distribution parameters

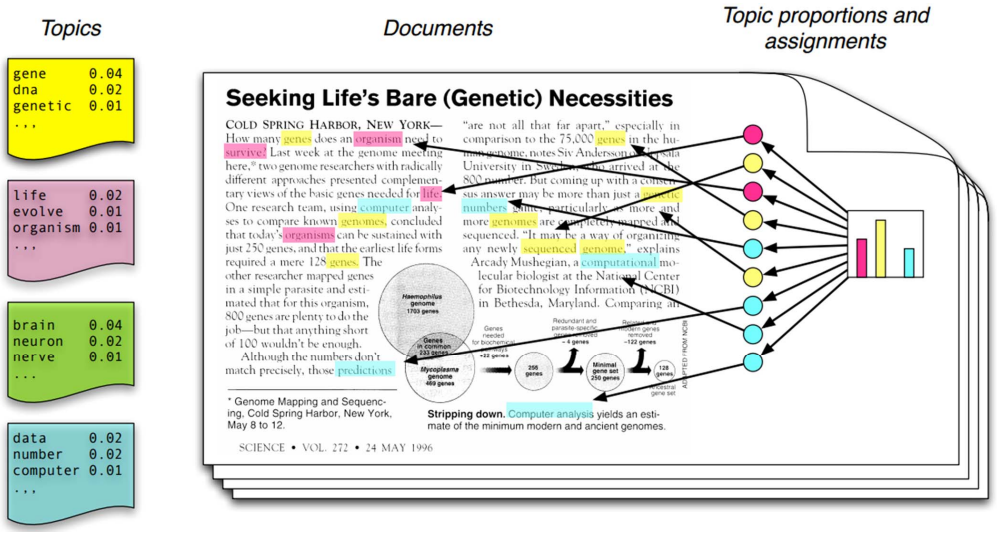


Figure 2.3 An example of probabilistic generative process for LDA. Topics have information about distributions over words (far left). Words in a document are modeled to be generated as shown in the figure (from right to left). This example figure is captured from (Blei, 2012).

are already known. [1] At the far left of the figure, four *topics* are shown with different colors (yellow, pink, green, blue) and are described by a probability distribution over words. The words on the left are sorted in a descending order of probability to show the top handful of words, and this is usually enough to give a rough understanding about the topics. Yellow topic is related to genetics, which contains words such as "gene", "dna", and "genetic". Pink topic is related to evolutionary biology, which contains words such as "life", "evolve", and "organism". Green topic is related to neurobiology, which contains words such as "brain", "neuron", and "nerve". Blue topic is related to data analysis, which contains words such as "computer", "number", and

[1] Distinction between latent variables and parameters is somewhat arbitrary. According to literatures in this field, if dimensionality of an unobserved variable does not increase with the number of observations, it is usually referred to as a parameter; otherwise a latent (hidden) variable.

"data". For the generative process shown at the right of the figure, a *topic proportion* (distribution over the topics represented by the colored histogram with pink, yellow, and blue) is chosen at first. From the topic proportion, we can conclude that the article of this example consists of words which are not related to neurobiology but related to mixture of topics: genetics, evolutionary biology, and data analysis. Then, *topic assignments* (shown in the colored coins) are generated with respect to the topic proportion (distribution over the topics). Finally, using the topic assignments, a word (highlighted with color shading in a document) is generated for each topic assignment from the corresponding topic (probability distribution over words). This generative process using LDA is not completely same as the generation mechanism of words and documents by human, but LDA has useful analysis about words which are close in meaning. The strong point of LDA is that it can be learned without any prior annotations or labeling of the documents, so it enables us to organize and summarize a large amount of text that would be impossible by human annotation.

LDA can be mathematically formulated with the following notations:

- A *word* $w \in \{1, 2, ..., V\}$ is the basic unit of discrete data [2], where V is vocabulary size. The *vocabulary* is determined by finding unique words from all words to be analyzed and mapping the unique word into a positive integer. Hence, vocabulary size V is the number of unique words.

- A *document* consists of N_d words, where d is an index of a document. In other words, d-th document is denoted by $\{w_{d1}, w_{d2}, ..., w_{dN_d}\}$. For the input of LDA, a collection of M documents is used, which are denoted by $\{w_{di} | d = 1, 2, ..., M, i = 1, 2, ..., N_d\}$.

[2] In the original LDA paper (Blei et al., 2003), w is represented using a V-dimentional unit-basis vector that has a single component equal to one and all other components eqult to zero, but in this thesis, w is represented as an non-zero index of the unit-basis vector for facilitating explanation. This change of notation for explaining LDA does not impede the use of equation or actual implementation.

- A *topic assignment* $z_{di} \in \{1, 2, ..., K\}$ for each word w_{di} is a latent random variable to be estimated, where K is a design parameter which stands for the number of topics. In case of the above example in Figure 2.3, $K = 4$ is equivalent to the number of unique colors (yellow, pink, green, blue), and z_{di} is shown as a colored coin.

- *Topics* are denoted by $\{\phi_1, \phi_2, ..., \phi_K\}$, where k-th topic $\phi_k \in \mathbb{R}^V$ is represented as a distribution over the vocabulary (at the far left of Figure 2.3). Since ϕ_k is a distribution parameter, component-wise summation of ϕ_k must be 1 (i.e. $\sum_{v=1}^{V} \phi_k(v) = 1$), and each component must be positive (i.e. $\phi_k(v) \geq 0$ for all v). The parameter ϕ_k indicates which words are important for the topic k.

- A *topic proportion* for the d-th document $\theta_d \in \mathbb{R}^K$ (the colored histogram in Figure 2.3) is a distribution parameter to be estimated. The parameter θ_d contains knowledge about which topics are important for the d-th document. Also, component-wise summation of θ_d must be 1, and each component must be positive.

- Design hyperparameters $\alpha = [\alpha(1), \alpha(2), ..., \alpha(K)]^T \in \mathbb{R}^K, \beta = [\beta(1), \beta(2), ..., \beta(V)]^T \in \mathbb{R}^V$ are used as prior information to generate distribution parameters θ_d, ϕ_k, respectively. For the sake of convenience, LDA uses symmetric values α, β such that $\alpha(1) = \alpha(2) = ... = \alpha(K)$ and $\beta(1) = \beta(2) = ... = \beta(V)$; that is, each hyperparameter has only a single degree of freedom. Strictly speaking, the expression $\alpha = 1$ is not mathematically proper because α is a K-dimensional vector, but we use this expression that means $\alpha = [1, 1, ..., 1]^T$ for the brevity of notations.

Using the above variables, probabilistic relations among the variables are defined. First, a topic proportion for the d-th document θ_d is generated by following equa-

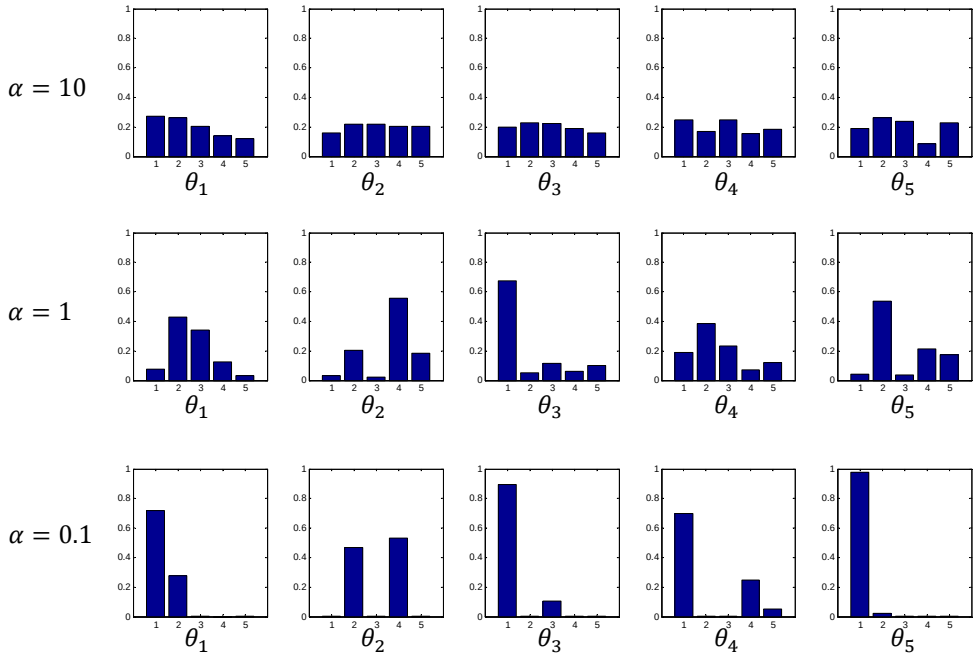


Figure 2.4 Examples of θ_d drawn by Dirichlet distributions for various settings of the parameter α. The smaller α is, the more sparse components of θ_d is generated, where $\theta_d \sim Dir(\alpha)$.

tion:

$$\theta_d \mid \alpha \sim Dir(\theta_d|\alpha), \tag{2.18}$$

where $Dir(\theta_d|\alpha)$ is Dirichlet distribution which is the multivariate version of the beta distribution. Figure 2.4 shows examples of θ_d drawn by Dirichlet distributions for various settings of the parameter α. If α is relatively small (usually under 1), the Dirichlet distribution prefers to generate sparse histograms where only a few components of θ_d have a non-zero weight. On the other hand, when α is larger, all components of θ_d tend to be distributed evenly. Therefore, based on our intuition that a document should have a small number of topics rather than mixture of almost all topics, α should be set not to much large in practice. Since LDA assumes that D topic proportions $\theta_1, \theta_2, ..., \theta_D$ is

dependent only on the hyperparameter α, the random variables $\theta_1, \theta_2, ..., \theta_D$ are conditionally independent to each other given α, and joint probability of θ_d is factorized as follows:

$$p(\theta_1, ..., \theta_D|\alpha) = \prod_{d=1}^{D} p(\theta_d \mid \alpha). \tag{2.19}$$

Second, N_d topic assignments are independent and identically distributed (i.i.d) random variables given the d-th topic proportion θ_d for the d-th document, and each topic assignment z_{di} is drawn as follows:

$$p(z_{d1}, z_{d2}, ..., z_{dN_d} \mid \theta_d) = \prod_{i=1}^{N_d} p(z_{di} \mid \theta_d), \tag{2.20}$$

$$z_{di}|\theta_d \sim Multi(z_{di}|\theta_d), \tag{2.21}$$

where $Multi(z_{di}|\theta_d)$ is Multinomial distribution. For instance, if $\theta_d = [0.1, 0.2, 0.3, 0.4]^T$, the probability of z_{di} to be generated is determined as $p(z_{di} = 1|\theta_d) = 0.1$, $p(z_{di} = 2|\theta_d) = 0.2$, $p(z_{di} = 3|\theta_d) = 0.3$, and, $p(z_{di} = 4|\theta_d) = 0.4$, respectively. Intuitively, this probability definition of $p(z_{di} \mid \theta_d)$ penalizes documents for having too many possible topics. That is because making a parameter θ_d concentrate on sparse components will increase the probability when drawing the same number of topic assignments (e.g. $\theta_d = [0.6, 0.4, 0, 0]^T$ is better than $\theta_d = [0.25, 0.25, 0.25, 0.25]^T$).

Third, K topics $\{\phi_k\}_{k=1}^{K}$ are defined as following equation given the parameter β.

$$p(\phi_1, \phi_2, ..., \phi_K|\beta) = \prod_{k=1}^{K} p(\phi_k|\beta), \tag{2.22}$$

where

$$\phi_k|\beta \sim Dir(\phi_k|\beta). \tag{2.23}$$

The parameter β is related to the prior count on the frequency of words generated from a topic, which affects a bias towards sparsity of ϕ_k. Thus, the parameter β should be

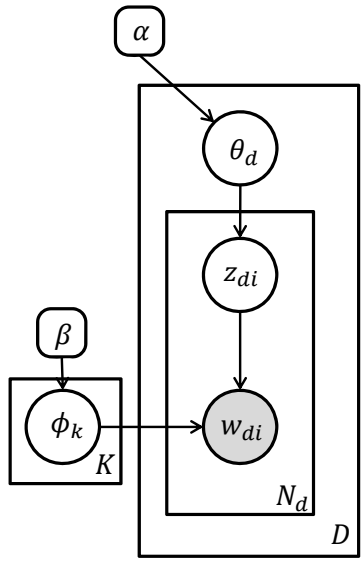
designed to a small value from the assumption that a large part of the entire vocabulary are nothing to with the specific topic k.

Fourth, for each of the N_d words in the d-th document, a word w_{di} is generated from a multinomial probability conditioned on the topic assignment z_{di} and topics $\phi_1, \phi_2, ..., \phi_K$,

$$w_{di}|z_{di,}\phi_1, \phi_2, ..., \phi_K \sim Multi(w_{di}|\phi_{z_{di}}). \tag{2.24}$$

Thus, given a knowledge that a document is about a particular topic, we can expect particular words to appear in the document more or less frequently. This definition implied that having sparsely distributed topics ϕ_k can result in a high probability for a set of words. Also, to increase the probability, the topic distributions $\phi_1, \phi_2, ..., \phi_K$ should have non-zero components to be non-overlapped as many as possible, since the sum of components in each topic distribution must be 1 and the topic distributions need to cover every vocabulary V (the dimension of ϕ_k).

Using variables and their dependence defined in the above, the overall model is graphically represented as shown in Figure. 2.5. The figure can be interpreted in a top-down order through the generative process, where the nodes denote random variables, and the arrows denote possible dependence among random variables. As mentioned earlier, the user-defined hyperparameters α, β for the Dirichlet distribution are treated as constant values in the model. The words in all documents $\{w_{di}|d = 1, ..., D, i = 1, ..., N_d\}$ are the only observations for LDA, while probability parameters ϕ_k, θ_d, and the topic assignment z_{di} are latent (i.e. unobserved) variables that we would like to infer. Hence, the variable w_{di} is shaded and the other variables are unshaded. Plates (the boxes in the figure) denote repetition of sampling, and the constant variable in the bottom-right corner referring to the number of repetitions. The inner plate containing z_{di} and w_{di} illustrates the repeated sampling of topic assignments and words until N_d words have been generated for the d-th document. The outer plate surrounding θ_d illus-



Notations
D: the number of documents.
N_d: the number of words in d-th document.
K: the number of topics.
α: Dirichlet prior on the per-document topic distributions.
β: Dirichlet prior on the per-topic word distribution.
θ_d: topic distribution for d-th document.
ϕ_k: word distribution for topic k.
z_{di}: the topic for the i-th word in d-th document.
w_{di}: the specific word.

Mathematical description
Choose $\theta_d \sim Dir(\alpha)$.
Choose $\phi_k \sim Dir(\beta)$.
Choose a topic $z_{ji}|\theta_d \sim Multi(\theta_d)$.
Choose a word $w_{ji}|\phi_k, z_{di} \sim Multi(\phi_{z_{di}})$.

Figure 2.5 Graphical representation for latent Dirichlet allocation and summary of notations and formulations. The latent variables are unshaded and the observed variables are shaded. Arrows indicate conditional dependencies between two variables. The rectangles are plate notation which denotes replication.

trates the generation of D samples of a topic proportion (distribution over topics) for each document d. The plate surrounding ϕ_k illustrates the repeated sampling of topics (distribution over words) for each topic index k until T topics have been generated.

With the notations and dependencies defined above, the generative process for LDA corresponds to the following joint probability distribution of the latent and observed variables given the hyperparameter α, β:

$$p(\phi, \theta, z, w|\alpha, \beta) = \left(\prod_{k=1}^{K} p(\phi_k|\beta) \right) \prod_{d=1}^{D} p(\theta_d \mid \alpha) \prod_{i=1}^{N_d} p(z_{di} \mid \theta_d) p(w_{di} \mid z_{di}, \phi), \tag{2.25}$$

where the variables without indices imply that they contain all possible indices in order to concisely represent notations; in other words, $\phi = \{\phi_k|k = 1, 2, ..., K\}$,

$\theta = \{\theta_d | d = 1, 2, ..., D\}$, $z = \{z_{di} | d = 1, 2, ..., D, i = 1, 2, ..., N_d\}$, $w = \{w_{di} | d = 1, 2, ..., D, i = 1, 2, ..., N_d\}$.

Statistical methods for inference of LDA can be used to invert of the generative process, inferring the set of topic related variables that were responsible for generating a collection of documents. The details about inference methods for LDA will be addressed in Section 2.2.

2.2 Inference of LDA

We have described the motivation, property, notation, and formulation of LDA with an example and graphical representation. In this section, we turn our attention to procedures for model inference and parameter estimation under LDA. Since LDA is hierarchical Bayesian model, we first describe Bayesian inference: to reason about the posterior distribution over the parameters and latent variables conditioned on the observation. This task can be done by finding the configuration of all latent variables ϕ, θ, z that maximize the posterior probability (MAP) given the observations w and hyper-parameters α, β:

$$\hat{\phi}, \hat{\theta}, \hat{z} = \arg\max_{\phi,\theta,z} p(\phi, \theta, z | w, \alpha, \beta), \tag{2.26}$$

where the posterior probability of LDA $p(\phi, \theta, z | w, \alpha, \beta)$ is given by Bayes's rule with the joint probability of LDA in Eq. 2.25:

$$p(\phi, \theta, z | w, \alpha, \beta) = \frac{p(\phi, \theta, z, w | \alpha, \beta)}{p(w | \alpha, \beta)}, \tag{2.27}$$

$$= \frac{p(\phi, \theta, z, w | \alpha, \beta)}{\int_\phi \int_\theta \sum_z p(\phi, \theta, z, w | \alpha, \beta) d\theta d\phi}. \tag{2.28}$$

The analytic solution of Eq.2.26, which is also referred to as a closed-form solution, is not available because symbolic integration of the denominator that finds antiderivative of the joint probability is impossible according to (Dickey, 1983). Of course,

despite the intractable calculation of the denominator, we can regard the denominator as an unknown constant value by assuming that ϕ, θ, z are integrated out and can only consider the nominator $p(\phi, \theta, z, w|\alpha, \beta)$ to find the optimal $\hat{\phi}, \hat{\theta}, \hat{z}$. However, the probability density function (pdf) of the joint distribution has very high dimensionality, non-convexity, and a lot of saddle points, so we cannot analytically calculate global maximum of the pdf.

The other option for exact inference of the MAP problem is a numerical method. The joint probability $p(\phi, \theta, z, w|\alpha, \beta)$ can be easily computed under the one specific setting of the hidden variables, parameters, and given observations. However, recalling the fact that the number of random variables in LDA is extremely large and thereby configuration complexity of these variables is enormous (e.g. complexity of topic assignment z is $O(K^M), M = \sum_{d=1}^{D} N_d$), we cannot numerically compute the joint probability of all possible instantiations of the hidden random variables to find the best case. For this reason, the exact inference of Eq.2.26 with the numerical method is also intractable.

Although the MAP problem of LDA is intractable for exact inference, approximate inference algorithms can be considered for LDA, such as collapsed Gibbs sampling (Griffiths & Steyvers, 2004) and variational inference (Blei et al., 2003). These algorithms approximate the posterior in Eq. 2.27 by forming an alternative distribution over the latent variables and parameters related to topic that is adapted to be close to the true posterior. In the subsequent subsections, we will introduce two main approximate inference methods for LDA and give discussion about the both methods.

2.2.1 Collapsed Gibbs Sampling

Gibbs sampling is one of a family of sampling methods known as the Markov Chain Monte Carlo (MCMC) framework (Andrieu et al., 2003), which is an approximate iterative technique designed to sample variables from complex and high-dimensional dis-

tributions. In other words, after a number of iterations through a Markov chain which is a sampling sequence of random variables, the samples from stationary distribution of the Markov chain converges to the desired probability distribution (i.e. posterior of LDA in this case). Each state of the Markov chain is an assignment of values to the variables being sampled, and transitions between states follow a simple rule: the next state is reached by sequentially sampling of all variables given conditional distributions of subsets of variables where each subset is conditioned on the value of all variables.

For example, consider the joint distribution $p(z) = p(z_1, z_2, ..., z_N)$ from which we want to sample z, and suppose there is no closed-form solution for $p(z)$, but a representation for the conditional distributions is available. Thus z_i is replaced by a new value drawn from the distribution $p(z_i|z_{-i})$, where z_{-i} is a set $\{z_i\}_{i=1}^N$ with z_i omitted, (i.e. $z_{-i} = \{z_1, ..., z_{i-1}, z_{i+1}, ..., z_N\}$). This procedure is repeated by choosing the variable to be updated at each step from some distribution randomly in the following (Bishop, 2006).

1. Randomly initialize each $z_i^1 \in \{1, 2, ..., K\}$, where $i = 1, 2, ..., N$,

2. For each step $t = 1, 2, ..., T$:

   - Replace z_1^t by a new value z_1^{t+1}, sampling $z_i^{t+1} \sim p(z_1|z_2^t, z_3^t, ..., z_N^t)$.
   - Replace z_2^t by a new value z_2^{t+1}, sampling $z_i^{t+1} \sim p(z_2|z_1^{t+1}, z_3^t, ..., z_N^t)$.
   - ...
   - Replace z_j^t by a new value z_j^{t+1},
     sampling $z_j^{t+1} \sim p(z_j|z_1^{t+1}, ..., z_{j-1}^{t+1}, z_{j+1}^t, ...z_N^t)$.
   - Replace z_N^t by a new value z_N^{t+1}, sampling $z_N^{t+1} \sim p(z_N|z_1^{t+1}, ..., z_{N-1}^{t+1})$.

From the above procedure, the samples begin to converge to what would be sampled from the true distribution, and the convergence of Gibbs sampling is theoretically guar-

anteed. Although diagnosing convergence is a minor problem when Gibbs sampling inference method is used, Gibbs sampling is quite powerful and has fairly good performance in practice. Typically, an acceptable estimation of convergence can be obtained by calculating the log-likelihood.

To derive the learning algorithm of LDA, we are interested in the latent topic proportion of each document θ, the topics ϕ, and the topic assignments for each word z. However, we do not need to include the parameter sets θ and ϕ for the inference of LDA, because they can be interpreted as statistics of the associations between the observed words w and the corresponding topic assignments z. In other words, z is a sufficient statistic (Kay, 1998) [3] for estimating and calculating both the parameter θ and ϕ which can be integrated out. This strategy of integrating out the parameters for model inference is referred to as collapsed sampling (Neal, 2000). Therefore, a simpler algorithm can be used if we integrate out the multinomial parameters θ and ϕ, and simply sample z, which is called a *collapsed Gibbs sampling*.

The collapsed Gibbs sampling for LDA should compute the probability of a topic assignment z_{di} corresponding to a word w_{di}, given all other topic assignments to all other words except w_{di}. Thus, we are interested in computing the following conditional posterior distribution for z_{di} given by:

$$p(z_{di}|z_{-di}, w, \alpha, \beta), \tag{2.29}$$

where z_{-di} denotes a simple description of a set of all topic assignments except for z_{di}, and words w not having an index is concise notation version of a set with all possible

[3] For example, if $x_1, x_2, ..., x_N$ are independent, identically and normally distributed samples with the population mean μ (a parameter) and known variance σ^2, then the sample mean function $T(x_1, x_2, ..., x_N) = \frac{1}{N}\sum_{i=1}^{N} x_i$ is a *sufficient statistic* for μ, where the population mean is distinguished from the sample mean from the fact that the population mean considers every member of the population. Once the sample mean is known, no further information about μ can be obtained from the sample itself. On the other hand, the median is not sufficient for the mean: even if the median of the sample is known, knowing the all samples itself would provide further information about the population mean μ.

indices, i.e. $\{w_{di}|d = 1, ..., D, i = 1, ..., N_d\}$. Then, we apply Bayes' rule to obtain the joint probability of z and w given the hyperparameter α and β.

$$p(z_{di}|z_{-di}, w, \alpha, \beta) = \frac{p(z_{di}, z_{-di}, w|\alpha, \beta)}{p(z_{-di}, w|\alpha, \beta)}. \tag{2.30}$$

From the above equation, the denominator can be considered a constant value since Gibbs sampling assumes all variables are known except for z_{di}, so we can derive as follows:

$$p(z_{di}|z_{-di}, w, \alpha, \beta) \propto p(z_{di}, z_{-di}, w|\alpha, \beta) \tag{2.31}$$

$$= p(z, w|\alpha, \beta). \tag{2.32}$$

From the definition of LDA described in the previous section, the joint distribution of z and w can be factorized:

$$p(z, w|\alpha, \beta) = \int\int p(z, w, \theta, \phi|\alpha, \beta)d\phi d\theta \tag{2.33}$$

$$= \int\int p(\phi|\beta)p(\theta \mid \alpha)p(z|\theta)p(w|z, \phi)d\phi d\theta \tag{2.34}$$

$$= \int p(\phi|\beta)p(w|z, \phi)d\phi \int p(\theta \mid \alpha)p(z|\theta)d\theta \tag{2.35}$$

$$= \int p(w, \phi|z, \beta)d\phi \int p(z, \theta|\alpha)d\theta \tag{2.36}$$

$$= p(w|z, \beta)p(z|\alpha). \tag{2.37}$$

The first term $p(w|z, \beta)$ can be derived by substituting Dirichlet and multinomial probability into $p(\phi|\beta)$ and $p(w|z, \phi)$, respectively:

$$p(w|z, \beta) = \int p(\phi|\beta)p(w|z, \phi)d\phi \tag{2.38}$$

$$= \int \left\{ \prod_{k=1}^{K} p(\phi_k|\beta) \right\} \prod_{d=1}^{D} \prod_{i=1}^{N_d} p(w_{di}|z_{di}, \phi)d\phi \tag{2.39}$$

$$= \int \left\{ \prod_{k=1}^{K} \frac{1}{B(\beta)} \prod_{v=1}^{V} \phi_k(v)^{\beta-1} \right\} \prod_{d=1}^{D} \prod_{i=1}^{N_d} \phi_{z_{di}}(w_{di})d\phi, \tag{2.40}$$

where $\phi_k(v)$ is the v-th component of the vector $\phi_k \in \mathbb{R}^V$, and $B(\beta) = \frac{\prod_{v=1}^{V} \Gamma(\beta(v))}{\Gamma\left(\sum_{v=1}^{V} \beta(v)\right)}$ is Beta function which is used to normalize the Dirichlet distribution $p(\phi_k|\beta)$. Since $x^a x^b = x^{a+b}$, we can replace the innermost products $\prod_{d=1}^{D} \prod_{i=1}^{N_d} \phi_{z_{di}}(w_{di})$ by counting the number of times that the word $w_{di} = v$ is assigned to the topic $z_{di} = k$ and by exponentiating to the counts.

$$\prod_{d=1}^{D} \prod_{i=1}^{N_d} \phi_{z_{di}}(w_{di}) = \prod_{k=1}^{K} \prod_{v=1}^{V} \{\phi_k(v)\}^{h(k,v)}, \tag{2.41}$$

where $h_\phi \in \mathbb{N}^{K \times V}$ denotes the histogram matrix which counts the number of times the word $w_{di} = v$ is assigned to the topic $z_{di} = k$ given by:

$$h_\phi(k, v) = \sum_{d=1}^{D} \sum_{i=1}^{N_d} \delta\left[w_{di} - v\right] \delta\left[z_{di} - k\right], \tag{2.42}$$

in which δ is the Kronecker delta function. Therefore, Eq. 2.40 can be rewritten as follows:

$$p(w|z, \beta) = \int \left\{ \prod_{k=1}^{K} \frac{1}{B(\beta)} \prod_{v=1}^{V} \phi_k(v)^{\beta - 1} \right\} \prod_{d=1}^{D} \prod_{i=1}^{N_d} \phi_{z_{di}}(w_{di}) d\phi \tag{2.43}$$

$$= \int \left\{ \prod_{k=1}^{K} \frac{1}{B(\beta)} \prod_{v=1}^{V} \phi_k(v)^{\beta - 1} \right\} \prod_{k=1}^{K} \prod_{v=1}^{V} \{\phi_k(v)\}^{h_\phi(k,v)} d\phi \tag{2.44}$$

$$= \prod_{k=1}^{K} \frac{1}{B(\beta)} \int \prod_{v=1}^{V} \{\phi_k(v)\}^{h_\phi(k,v) + \beta - 1} d\phi. \tag{2.45}$$

Then, using the trick that multiplies the Beta function $B(h_\phi(k, \cdot) + \beta)$ to both the nominator and denominator, we can integrate out ϕ, since integrals of Dirichlet distribution is 1, where dot notation $h_\phi(k, \cdot)$ is a V-dimensional vector that contains all

indices for $v = \{1, 2, ..., V\}$.

$$p(w|z,\beta) = \prod_{k=1}^{K} \frac{1}{B(\beta)} \int \prod_{v=1}^{V} \{\phi_k(v)\}^{h_\phi(k,v)+\beta-1} d\phi \tag{2.46}$$

$$= \prod_{k=1}^{K} \frac{B(h_\phi(k,\cdot)+\beta)}{B(\beta)} \underbrace{\int \frac{1}{B(h_\phi(k,\cdot)+\beta)} \prod_{v=1}^{V} \{\phi_k(v)\}^{h_\phi(k,v)+\beta-1} d\phi}_{=1\ (Integral\ of\ pdf)} \tag{2.47}$$

$$= \prod_{k=1}^{K} \frac{B(h_\phi(k,\cdot)+\beta)}{B(\beta)}. \tag{2.48}$$

In a similar way, the second term $p(z|\alpha)$ in Eq. 2.37 can be calculated as follows:

$$p(z|\alpha) = \int p(\theta \mid \alpha) p(z|\theta) d\theta \tag{2.49}$$

$$= \int \prod_{d=1}^{D} p(\theta_d \mid \alpha) \prod_{i=1}^{N_d} p(z_{di}|\theta_d) d\theta \tag{2.50}$$

$$= \int \prod_{d=1}^{D} \left\{ \frac{1}{B(\alpha)} \prod_{k=1}^{K} \theta_d(k)^{\alpha-1} \right\} \prod_{i=1}^{N_d} \theta_d(z_{di}) d\theta. \tag{2.51}$$

Then, by counting the duplicated terms (i.e. $h_\theta(d,k) = \sum_{i=1}^{N_d} \delta\left[z_{di} - k\right]$), the equation

can be simplified as follows:

$$p(z|\alpha) = \int \prod_{d=1}^{D} \left\{ \frac{1}{B(\alpha)} \prod_{k=1}^{K} \theta_d(k)^{\alpha-1} \right\} \prod_{k=1}^{K} \theta_d(k)^{h_\theta(d,k)} d\theta \tag{2.52}$$

$$= \int \prod_{d=1}^{D} \frac{1}{B(\alpha)} \prod_{k=1}^{K} \theta_d(k)^{\alpha-1} \theta_d(k)^{h_\theta(d,k)} d\theta \tag{2.53}$$

$$= \int \prod_{d=1}^{D} \frac{1}{B(\alpha)} \prod_{k=1}^{K} \theta_d(k)^{h_\theta(d,k)+\alpha-1} d\theta \tag{2.54}$$

$$= \prod_{d=1}^{D} \frac{B(h_\theta(d,\cdot)+\alpha)}{B(\alpha)} \underbrace{\int \frac{1}{B(h_\theta(d,\cdot)+\alpha)} \prod_{k=1}^{K} \theta_d(k)^{h_\theta(d,k)+\alpha-1} d\theta}_{=1\ (Integral\ of\ pdf)} \tag{2.55}$$

$$= \prod_{d=1}^{D} \frac{B(h_\theta(d,\cdot)+\alpha)}{B(\alpha)}. \tag{2.56}$$

Using the derivation results of both terms in Eq. 2.37, the joint distribution of words w and topic assignments z becomes:

$$p(z, w|\alpha, \beta) = p(w|z, \beta)p(z|\alpha) \tag{2.57}$$

$$= \left\{ \prod_{k=1}^{K} \frac{B(h_\phi(k,\cdot)+\beta)}{B(\beta)} \right\} \left\{ \prod_{d=1}^{D} \frac{B(h_\theta(d,\cdot)+\alpha)}{B(\alpha)} \right\}. \tag{2.58}$$

From the joint distribution, the Gibbs sampling equation in Eq. 2.32 for LDA can be derived using the Bayes' rule, chain rule, and definition of independence among variables:

$$p(z_{di}|z_{-di}, w, \alpha, \beta) = \frac{p(z_{di}, z_{-di}, w|\alpha, \beta)}{p(z_{-di}, w|\alpha, \beta)} \tag{2.59}$$

$$= \frac{p(z, w|\alpha, \beta)}{p(z_{-di}, w|\alpha, \beta)} \tag{2.60}$$

$$= \frac{p(z|\alpha, \beta)p(w|z, \alpha, \beta)}{p(z_{-di}|\alpha, \beta)p(w_{di}, w_{-di}|z_{-di}, \alpha, \beta)} \tag{2.61}$$

$$= \frac{p(z|\alpha)p(w|z, \beta)}{p(z_{-di}|\alpha)p(w_{-di}|z_{-di}, \beta)p(w_{di}|\alpha, \beta)}. \tag{2.62}$$

Then, the nominator can be replaced by the derivation result of Eq. 2.58 which is composed of Beta functions. Also, the denominator can be represented by the Beta functions, since $p(z_{-di}|\alpha)$ and $p(w_{-di}|z_{-di},\beta)$ is almost equivalent to the Eq. 2.58 except for omitting z_{di} and w_{di} and $p(w_{di}|\alpha,\beta)$ is a constant. In other words, by excluding a count for z_{di} and w_{di} from the original histogram $h_\phi(k,\cdot)$ and $h_\theta(d,\cdot)$ to obtain the histograms $h_\phi(k,-di)$ and $h_\theta(d,-di)$ [4] , the conditional distribution for Gibbs sampling can be derived using Beta functions given as follows:

$$p(z_{di}|z_{-di},w,\alpha,\beta) \propto \frac{\left\{\prod_{k=1}^{K}\frac{B(h_\phi(k,\cdot)+\beta)}{B(\beta)}\right\}\left\{\prod_{d=1}^{D}\frac{B(h_\theta(d,\cdot)+\alpha)}{B(\alpha)}\right\}}{\left\{\prod_{k=1}^{K}\frac{B(h_\phi(k,-di)+\beta)}{B(\beta)}\right\}\left\{\prod_{d=1}^{D}\frac{B(h_\theta(d,-di)+\alpha)}{B(\alpha)}\right\}} \tag{2.63}$$

$$= \prod_{k=1}^{K}\frac{B(h_\phi(k,\cdot)+\beta)}{B(h_\phi(k,-di)+\beta)} \times \prod_{d=1}^{D}\frac{B(h_\theta(d,\cdot)+\alpha)}{B(h_\theta(d,-di)+\alpha)}. \tag{2.64}$$

From the definition of the Beta function expressed by Gamma functions and the property of the Gamma function that $\frac{\Gamma(x)}{\Gamma(x-1)} = x-1$, we can reduce a fraction by eliminating duplicated terms. The nominator of the first term is given by:

$$B(h_\phi(k,\cdot)+\beta) = \frac{\prod_{v=1}^{V}\Gamma(h_\phi(k,v)+\beta(v))}{\Gamma\left(\sum_{v=1}^{V}\left[h_\phi(k,v)+\beta(v)\right]\right)}, \tag{2.65}$$

and the denominator of the first term is given by:

$$B(h_\phi(k,-di)+\beta) = \frac{\prod_{v=1}^{V}\Gamma(h_\phi(k,v)-\delta\left[w_{di}-v\right]\delta\left[z_{di}-k\right]+\beta(v))}{\Gamma\left(\sum_{v=1}^{V}\left[h_\phi(k,v)-\delta\left[w_{di}-v\right]\delta\left[z_{di}-k\right]+\beta(v)\right]\right)}. \tag{2.66}$$

Thus, for the terms of $\delta\left[w_{di}-v\right]\delta\left[z_{di}-k\right] = 0$, we can neglect them since two corresponding Gamma functions of Eq. 2.65 and Eq. 2.66 become equal and cancelled.

[4] $h_\phi(k,-di) = h_\phi(k,v) - \delta\left[w_{di}-v\right]\delta\left[z_{di}-k\right], (v=1,...,V)$ and
$h_\theta(d,-di) = h_\theta(d,k) - \delta\left[z_{di}-k\right], (k=1,...,K)$

On the other hands, if $\delta\left[w_{di}-v\right]\delta\left[z_{di}-k\right]=1$ (i.e. $w_{di}=v$ and $z_{di}=k$), we can apply the Gamma function property $\frac{\Gamma(x)}{\Gamma(x-1)}=x-1$, so the first term of Eq. 2.64 is simplified as follows:

$$\prod_{k=1}^{K}\frac{B(h_{\phi}(k,\cdot)+\beta)}{B(h_{\phi}(k,-di)+\beta)}\propto\frac{h_{\phi}(k,v)-\delta\left[w_{di}-v\right]\delta\left[z_{di}-k\right]+\beta(v)}{\sum_{v=1}^{V}\left[h_{\phi}(k,v)-\delta\left[w_{di}-v\right]\delta\left[z_{di}-k\right]+\beta(v)\right]}. \quad (2.67)$$

In the similar way, the second term of Eq. 2.64 can be also simplified as follows:

$$\prod_{d=1}^{D}\frac{B(h_{\theta}(d,\cdot)+\alpha)}{B(h_{\theta}(d,-di)+\alpha)}\propto\frac{h_{\theta}(d,k)-\delta\left[z_{di}-k\right]+\alpha(k)}{\sum_{k=1}^{K}\left[h_{\theta}(d,k)-\delta\left[z_{di}-k\right]+\alpha(k)\right]} \quad (2.68)$$

$$=\frac{h_{\theta}(d,k)-\delta\left[z_{di}-k\right]+\alpha(k)}{\sum_{k=1}^{K}\left[h_{\theta}(d,k)+\alpha(k)\right]-1}. \quad (2.69)$$

As a results, the Gibbs sampling equation for LDA is proportional to Eq. 2.67 and Eq. 2.69.

$$p(z_{di}|z_{-di},w,\alpha,\beta)\propto\frac{h_{\phi}(k,v)-\delta\left[w_{di}-v\right]\delta\left[z_{di}-k\right]+\beta(v)}{\sum_{v=1}^{V}\left[h_{\phi}(k,v)-\delta\left[w_{di}-v\right]\delta\left[z_{di}-k\right]+\beta(v)\right]}\times\left\{h_{\theta}(d,k)-\delta\left[z_{di}-k\right]+\alpha(k)\right\}, \quad (2.70)$$

where the denominator of Eq. 2.69 is ignored since it is constant to the variation of z_{di}.

For the final step, we need to obtain the multinomial parameters θ and ϕ which can be calculated by using posterior estimates of z. According to the conjugacy property (Diaconis & Ylvisaker, 1979) between the Dirichlet distribution and the multinomial distribution, if a random variable has multinomial distribution and the prior distribution of the random variable's parameter is a Dirichlet distribution, then the posterior distribution of the parameter is also a Dirichlet distribution. This means that we can successively update our knowledge of a parameter by combining new observations, one after another, without running into mathematical difficulties. In other words,

the posterior distribution of the parameter θ and ϕ are given as follows:

$$p(\theta_d|z,\alpha) = \frac{\overbrace{p(z|\theta_d)}^{Multinomial}\overbrace{p(\theta_d|\alpha)}^{Dirichlet}}{p(z|\alpha)}$$

$$= Dir(\theta_d|h_\theta(d,\cdot)+\alpha), \tag{2.71}$$

$$p(\phi_k|z,w,\beta) = Dir(\phi_k|h_\phi(k,\cdot)+\beta). \tag{2.72}$$

Therefore, using the expectation formula of the Dirichlet distribution with a prior α, $E[\theta_d|\alpha] = \frac{\alpha(k)}{\sum_{k=1}^{K}\alpha(k)}$, we can estimate the parameter θ and ϕ:

$$\hat{\theta}_d(k) = E[\theta_d(k)|h_\theta(d,\cdot)+\alpha] = \frac{h_\theta(d,k)+\alpha(k)}{\sum_{k=1}^{K}[h_\theta(d,k)+\alpha(k)]}, \tag{2.73}$$

$$\hat{\phi}_k(v) = E[\phi_k(v)|h_\phi(k,\cdot)+\beta] = \frac{h_\phi(k,v)+\beta(v)}{\sum_{v=1}^{V}[h_\phi(k,v)+\beta(v)]}. \tag{2.74}$$

Implementation

Implementation of LDA using the collapsed Gibbs sampling is straightforward when using the derivation results of Eq. 2.70, Eq. 2.73, and Eq. 2.74. The procedure of Gibbs sampling is summarized in Algorithm 1. In this procedure, three main data structures are used, the counting histogram $h_\phi(k,v)$ and $h_\theta(d,k)$ which have dimension $K \times V$ and $D \times K$ respectively, and the last one is topic assignments z_{di} which can be represented an array whose length is $\sum_{d=1}^{D} N_d$, where N_d is the number of words for the d-th document. The collapsed Gibbs sampling algorithm runs over the three steps: initialization, sampling iteration, and model parameter estimation. In the initialization step, the counting histograms are filled with Dirichlet prior to pre-calculate the summing of $\alpha(k)$ and $\beta(v)$ in Eq. 2.70, and a topic for each word is assigned at random. In the sampling step, we must decrement a count for the current topic assignment z_{di} before building a distribution from Eq. 2.70. Then we can obtain posterior distribution of each topic assignment using Eq. 2.70. After that, this discrete distribution is utilized to draw a new topic assignment z_{di} for the word w_{di}. The drawing processing can be

implemented by calculating cumulative distribution function (CDF) from the discrete posterior distribution, and then using inverse transform sampling (Vogel, 2002) which generates a random number from the uniform distribution in the interval $[0, 1]$ and takes the result of inverse of CDF from the random number. Finally, the multinomial parameters θ and ϕ are calculated by using posterior estimates of z according to the Eq. 2.73, and Eq. 2.74.

Algorithm 1 Collapsed Gibbs sampling algorithm for Latent Dirichlet Allocation

Input: A documnet set $\{w_{di}|d = 1, ..., D, i = 1, ..., N_d\}$, where d-th document consists of a set of N_d words and each $w_{di} \in \{1, ..., V\}$.
Hyperparameters α, β, and the number of topic K

Output: LDA model parameters θ, ϕ, and topic assignments z_{di} corresponding to each word w_{di}.

Initialization
For the histogram matrix, $h_\phi(k, v) = \beta$ and $h_\theta(d, k) = \alpha$, where $\forall\ k \in \{1, ..., K\},\ \forall v \in \{1, ..., V\},\ \forall d \in \{1, ..., D\}$.
for all document indices $d \leftarrow 1, \ldots, D$ **do**
  for all word indices $i \leftarrow 1, \ldots, N_d$ **do**
    Draw a topic for each word $z_{di} \sim Multi(z_{di}|[\frac{1}{K}, \frac{1}{K}, ..., \frac{1}{K}]^T)$.
    Increment document-topic count: $h_\theta(d, z_{di}) \leftarrow h_\theta(d, z_{di}) + 1$.
    Increment topic-vocabulary count: $h_\phi(z_{di}, w_{di}) \leftarrow h_\phi(z_{di}, w_{di}) + 1$.
  end for
end for

Collapsed Gibbs sampling
while not converge **do**
  for all document indices $d \leftarrow 1, \ldots, D$ **do**
    for all word indices $i \leftarrow 1, \ldots, N_d$ **do**
      Decrement document-topic count: $h_\theta(d, z_{di}) \leftarrow h_\theta(d, z_{di}) - 1$.
      Decrement topic-vocabulary count: $h_\phi(z_{di}, w_{di}) \leftarrow h_\phi(z_{di}, w_{di}) - 1$.
      Update posterior distribution of z_{di}, $p \in \mathbb{R}^K$ as follows:
      for topic indices $k \leftarrow 1, \ldots, K$ **do**
        $p(k) \leftarrow \frac{h_\phi(k, w_{di})}{\sum_{k=1}^{K} h_\phi(k, w_{di})} h_\theta(d, k)$
      end for
      Normalize $p(\cdot)$ that sums to 1: $p(k) \leftarrow \frac{p(k)}{\sum_{k=1}^{K} p(k)}$.
      Draw a new topic assignment using the posterior $z_{di} \sim Multi(z_{di}|p)$.
      Increment document-topic count: $h_\theta(d, z_{di}) \leftarrow h_\theta(d, z_{di}) + 1$.
      Increment topic-vocabulary count: $h_\phi(z_{di}, w_{di}) \leftarrow h_\phi(z_{di}, w_{di}) + 1$.
    end for
  end for
end while

Estimate model parameters
Calculate the parameters θ, ϕ according to Eq. 2.73, and Eq. 2.74.

2.2.2 Variational Inference

Variational inference is a deterministic methodology (unlike sampling methods which are based on stochastic inference) for approximating posteriors in an intractable probabilistic model (Jordan et al., 1999). This method is used in complex statistical models consisting of observed random variables and unobserved random variables (which we want to estimate), with various conditional dependency among the random variables. We will begin with deriving how variational methods can be applied to approximate Bayesian inference, then the detail process for LDA will be explained.

The basic idea of the variational inference is to use variational distribution that makes a complex model into simpler models by neglecting some dependency of the complex model. Thus, we can make an assumption that certain latent variables can be approximately independent conditioned on the observed data; for example, the posterior distribution over the latent variables z given the observation x can be approximated by a variational distribution $q(\cdot)$ as follows:

$$p(z|x) \approx q(z). \tag{2.75}$$

The variational distribution $q(z)$ should belong to a family of distributions of simpler form than $p(z|x)$. This family is selected with the intention of making $q(z)$ be similar to the true posterior $p(z|x)$. The dissimilarity between $q(z)$ and $p(z|x)$ is measured by the Kullback–Leibler divergence (Kullback & Leibler, 1951) that is a non-symmetric measure of the difference between two probability distributions, so inference is performed by selecting the distribution $q(z)$ that minimize the dissimilarity.

The Kullback–Leibler divergence (KL-divergence) is defined as

$$D[q(z)||p(z|x)] \triangleq \int_z q(z) \log \frac{q(z)}{p(z|x)} dz, \tag{2.76}$$

where $\log x$ is the natural logarithm. The property of KL-divergence is that if $q(z)$ is equal to $p(z|x)$, the dissimilarity measure $D[q(z) \, \| p(z|x)]$ becomes zero; otherwise a

positive value. Then, we can make the substitution $p(z|x) = \frac{p(z,x)}{p(x)}$ by the conditional probability:

$$D\left[q(z)||p(z|x)\right] = \int_z q(z) \log \frac{q(z)p(x)}{p(z,x)} dz \tag{2.77}$$

$$= \int_z q(z) \log \frac{q(z)}{p(z,x)} dz + \int_z q(z) \log p(x) dz \tag{2.78}$$

$$= \int_z q(z) \log \frac{q(z)}{p(z,x)} dz + \log p(x) \underbrace{\int_z q(z) dz}_{=1} \tag{2.79}$$

$$= \int_z q(z) \log \frac{q(z)}{p(z,x)} dz + \log p(x). \tag{2.80}$$

Using the derivation result of the above equation, we can decompose the log marginal probability $\log p(x)$ as follows:

$$\log p(x) = D\left[q(z)||p(z|x)\right] - \int_z q(z) \log \frac{q(z)}{p(z,x)} dz \tag{2.81}$$

$$= D\left[q(z)||p(z|x)\right] + \int_z q(z) \log \frac{p(z,x)}{q(z)} dz \tag{2.82}$$

$$= D\left[q(z)||p(z|x)\right] + \mathcal{L}(q), \tag{2.83}$$

where the last term of this equation is defined as a lower-bound

$$\mathcal{L}(q) \triangleq \int_z q(z) \log \frac{p(z,x)}{q(z)} dz \tag{2.84}$$

$$= \int_z q(z) \log p(z,x) dz - \int_z q(z) \log q(z) dz \tag{2.85}$$

$$= E_q\left[\log p(z,x)\right] + H\left[q(z)\right]. \tag{2.86}$$

Here, the notation $E_q\left[\cdot\right]$ is an expectation with respect to the distribution $q(\cdot)$, and $H\left[q(z)\right] = -\int_z q(z) \log q(z) dz$ is defined as the entropy of $q(z)$. Since the log evidence $\log p(x)$ is not related to $q(\cdot)$ and is constant given the observation x, optimizing (maximizing) this lower-bound $\mathcal{L}(q)$ is equivalent to minimizing the KL divergence between $q(z)$ and the true posterior $p(z|x)$ as illustrated in Figure 2.6.

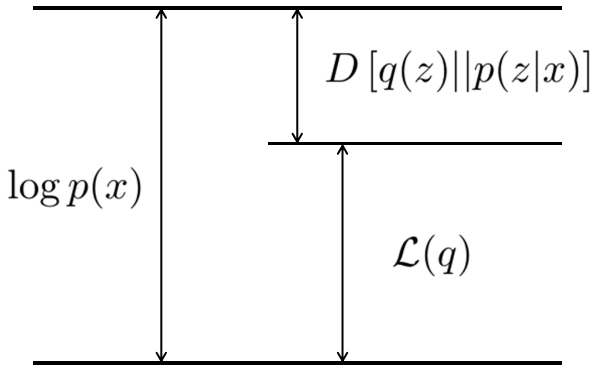


Figure 2.6 Relation among the log marginal probability $\log p(x)$, KL-divergence $D\left[q(z)||p(z|x)\right]$, and the lower-bound $\mathcal{L}(q)$. The quantity $\mathcal{L}(q)$ provides a lower bound on the log marginal probability $\log p(x)$ with difference given by the KL divergence $D\left[q(z)||p(z|x)\right]$. By maximizing $\mathcal{L}(q)$, we can minimize the KL divergence since the log marginal probability is constant with respect to $q(z)$. (Bishop, 2006)

As mentioned earlier, we need to choose a variational distribution $q(z)$ that has a simpler dependency structure than that of the exact (non-approximated) model, which enables the calculation of the lower bound $\mathcal{L}(q)$ to be tractable. The mean field approximation (Parisi, 1988) is a popular way to simplify the dependency structure by partitioning the elements of z into disjoint groups z_i where $i = 1, 2, ..., N$. This approximation makes it possible to convert a complex model into simpler models by partitioning the original complex model. This partitioning can be achieved by the addition of extra parameters that is called *variational parameters* (Winn, 2004). In other words, the variational parameters are applied to approximate a probability distribution of the model so that it can has a simpler dependency structure than that of the exact (non-approximated) model. Thus, we assume that the variational distribution $q(z)$ can be factorized as follows:

$$q(z) = \prod_{i=1}^{N} q_i(z_i) \tag{2.87}$$

In this assumption, designing the variational distribution $q(z)$ for approximating the

true posterior $p(z|x)$ depends on designing each factor $q_i(z_i)$. In practice, instead of selecting $q_i(z_i)$ from all possible distribution forms, we can choose $q_i(z_i)$ to be in a particular parameterized distribution family:

$$\prod_{i=1}^{N} q_i(z_i) = \prod_{i=1}^{N} q(z_i|\lambda_i), \tag{2.88}$$

where λ_i is a variational parameter for each hidden variable z_i. For example, $q(\cdot)$ is fixed with Gaussian distribution, and $q_i(z_i)$ is changed by adjusting the parameters for the mean and variance.

Then, we should find all of the distribution $q_i(z_i)$ for the lower bound $\mathcal{L}(q)$ to be largest. To achieve this, substituting Eq.2.87 into the definition of lower bound $\mathcal{L}(q)$ in Eq.2.85, and then it is dissected by the each factor $q_i(z_i)$ as follows:

$$\mathcal{L}(q) = \int_z q(z) \log p(z,x) dz - \int_z q(z) \log q(z) dz \tag{2.89}$$

$$= \int_z \left\{ \prod_{i=1}^{N} q_i(z_i) \right\} \log p(z,x) dz - \int_z \left\{ \prod_{j=1}^{N} q_j(z_j) \right\} \log \left\{ \prod_{i=1}^{N} q_i(z_i) \right\} dz \tag{2.90}$$

$$= \int_z \left\{ \prod_{i=1}^{N} q_i(z_i) \right\} \log p(z,x) dz - \int_{z_1} ... \int_{z_N} \left\{ \prod_{j=1}^{N} q_j(z_j) \right\} \sum_{i=1}^{N} \log q_i(z_i) dz_1 ... dz_N \tag{2.91}$$

$$= \int_z \left\{ \prod_{i=1}^{N} q_i(z_i) \right\} \log p(z,x) dz - \sum_{i=1}^{N} \int_{z_i} q_i(z_i) \log q_i(z_i) dz_i \tag{2.92}$$

$$= \int_z \left\{ \prod_{i=1}^{N} q_i(z_i) \right\} \log p(z,x) dz + \sum_{i=1}^{N} H\left[q_i(z_i)\right]. \tag{2.93}$$

Then, terms of the above equation are separated in a specific factor $q_j(z_j)$, using the notation $(-j)$ that denotes all indices except j; that is, $\int_{z_{(-j)}} = \int_{z_1} ... \int_{z_{j-1}} \int_{z_{j+1}} ... \int_{z_N}$

and the notation $E_{q_{(-j)}}[\cdot]$ is an expectation with respect to the distribution $\prod_{i \neq j} q_i(z_i)$:

$$\mathcal{L}(q) = \int_{z_j} q_j(z_j) \int_{z_{(-j)}} \left\{ \prod_{i \neq j} q_i(z_i) \right\} \log p(z, x) dz_{(-j)} dz_j + H\left[q_j(z_j)\right] + \sum_{i \neq j} H\left[q_i(z_i)\right] \tag{2.94}$$

$$= \int_{z_j} q_j(z_j) E_{q_{(-j)}}\left[\log p(z, x)\right] dz_j + \underbrace{H\left[q_j(z_j)\right]}_{= -\int_{z_j} q_j(z_j) \log q_j(z_j) dz_j} + \sum_{i \neq j} H\left[q_i(z_i)\right] \tag{2.95}$$

$$= \int_{z_j} q_j(z_j) \left\{ E_{q_{(-j)}}\left[\log p(z, x)\right] - \log q_j(z_j) \right\} dz_j + \sum_{i \neq j} H\left[q_i(z_i)\right]. \tag{2.96}$$

From the above result, we suppose that the $\{q_i(z_i) | i \neq j\}$ is fixed, and then we can maximize $\mathcal{L}(q)$ with respect to all possible forms of the distribution $q_j(z_j)$. In order to obtain the optimal solution for $q_j(z_j)$, we define the distribution $q_j^*(z_j)$ by normalizing $E_{q_{(-j)}}\left[\log p(z, x)\right]$ for the $q_j^*(z_j)$ to be a valid probability distribution, which is given by

$$q_j^*(z_j) = \frac{\exp\left(E_{q_{(-j)}}\left[\log p(z, x)\right] \right)}{\int_{z_j} \exp\left(E_{q_{(-j)}}\left[\log p(z, x)\right] \right) dz_j} = \frac{1}{C} \exp\left(E_{q_{(-j)}}\left[\log p(z, x)\right] \right), \tag{2.97}$$

where $C = \int_{z_j} \exp\left(E_{q_{(-j)}}\left[\log p(z, x)\right] \right) dz_j$ is the constant normalization factor. Using the notation $q_j^*(z_j)$ defined above, we can derive $\mathcal{L}(q)$ to be maximized by minimizing the negative KL divergence as follows:

$$\mathcal{L}(q) = \int_{z_j} q_j(z_j) \left\{ \underbrace{E_{q_{(-j)}}\left[\log p(z, x)\right]}_{= \log q_j^*(z_j) + \log C} - \log q_j(z_j) \right\} dz_j + \sum_{i \neq j} H\left[q_i(z_i)\right] \tag{2.98}$$

$$= \int_{z_j} q_j(z_j) \log \frac{q_j^*(z_j)}{q_j(z_j)} dz_j + \log C + \sum_{i \neq j} H\left[q_i(z_i)\right] \tag{2.99}$$

$$= -D\left[q_j(z_j) || q_j^*(z_j)\right] + \log C + \sum_{i \neq j} H\left[q_i(z_i)\right]. \tag{2.100}$$

Because the last two terms $\log C$ and $\sum_{i \neq j} H\left[q_i(z_i)\right]$ do not depend on $q_j(z_j)$, only the KL divergence between $q_j(z_j)$ and $q_j^*(z_j)$ influences the lower-bound $\mathcal{L}(q)$. Therefore, the lower-bound can be maximized by setting $q_j(z_j) = q_j^*(z_j)$, in which $q_j^*(z_j)$ is obtained easily by taking the expectation with respect to all other hidden variables and variational distributions $\{q_i(z_i)|i \neq j\}$. In other words, by picking each factor $q_j(z_j)$ and replacing the optimal value one by one, $\mathcal{L}(q)$ can increase gradually until convergence. The convergence is guaranteed according to (Boyd & Vandenberghe, 2004) because each factor for the variational distribution $q_j(z_j)$ can be designed to be convex. This scheme is similar to the case of Gibbs sampling which samples z_j from the distribution given all hidden and observed variables except z_j. The difference is that sampling z_j is a stochastic approach (i.e. it has randomness), whereas taking the expectation is a deterministic approach.

For the variational inference of LDA, we recall the objective function and joint probability of LDA given by:

$$\phi^*, \theta^*, z^* = \arg\max_{\phi,\theta,z} p(\phi, \theta, z|w, \alpha, \beta) \tag{2.101}$$

$$= \arg\max_{\phi,\theta,z} \frac{p(\phi, \theta, z, w|\alpha, \beta)}{p(w|\alpha, \beta)} \tag{2.102}$$

$$= \arg\max_{\phi,\theta,z} p(\phi, \theta, z, w|\alpha, \beta), \tag{2.103}$$

where

$$p(\phi, \theta, z, w|\alpha, \beta) = \left(\prod_{k=1}^{K} p(\phi_k|\beta)\right) \prod_{d=1}^{D} p(\theta_d \mid \alpha) \prod_{i=1}^{N_d} p(z_{di} \mid \theta_d) p(w_{di} \mid z_{di}, \phi). \tag{2.104}$$

To approximate the posterior related to the joint distribution of LDA in Eq.(2.25), a simpler variational distribution $q(\phi, \theta, z|\lambda, \gamma, \varphi)$ that can be factorized for easier

computation is utilized in VI as follows (Blei et al., 2003):

$$q(\phi, \theta, z|\lambda, \varphi, \gamma) = \left(\prod_{k=1}^{K} q(\phi_k|\lambda_k)\right)\left(\prod_{d=1}^{D} q(\theta_d|\gamma_d)\right)\left(\prod_{d,i}^{D,N_d} q(z_{di}|\varphi_{di})\right), \tag{2.105}$$

where λ, γ, and φ are the variational parameters used for approximate inference of ϕ, θ, and z respectively. Here, the forms of each factorized variational distribution $q(\phi_k|\lambda_k)$, $q(\theta_d|\gamma_d)$, and $q(z_{di}|\varphi_{di})$ are chosen to be Dirichlet, Dirichlet, and multinomial distribution, respectively:

$$\phi_k|\lambda_k \sim Dirichlet(\phi_k|\lambda_k) \tag{2.106}$$

$$\theta_d|\gamma_d \sim Dirichlet(\theta_d|\gamma_d) \tag{2.107}$$

$$z_{di}|\varphi_{di} \sim Multi(z_{di}|\varphi_{di}). \tag{2.108}$$

Hence, instead of solving optimization of the objective function in Eq. 2.103, the optimal values of the variational parameters are found as follow:

$$\begin{aligned} \lambda^*, \gamma^*, \varphi^* &= \arg\min_{\lambda,\gamma,\varphi} D\left[q(\cdot)\,\|p(\cdot)\right], \\ where, \quad q(\cdot) &= q(\phi, \theta, z|\lambda, \gamma, \varphi), \\ p(\cdot) &= p(\phi, \theta, z|w, \alpha, \beta). \end{aligned} \tag{2.109}$$

The optimal variational parameters are founded by minimizing the Kullback-Leibler (KL) divergence $D\left[q(\cdot)\,\|p(\cdot)\right]$ between the variational distribution and the true posterior $p(\phi, \theta, z|w, \alpha, \beta)$ as shown in Figure 2.7.

As in case of the relation among the evidence $\log(w|\alpha, \beta)$, KL-divergence $D\left[q(\cdot)\,\|p(\cdot)\right]$, and the lower-bound $\mathcal{L}(q)$ described in Figure 2.6, minimizing the KL divergence is equivalent to maximizing the lower-bound with respect to the variational parameters

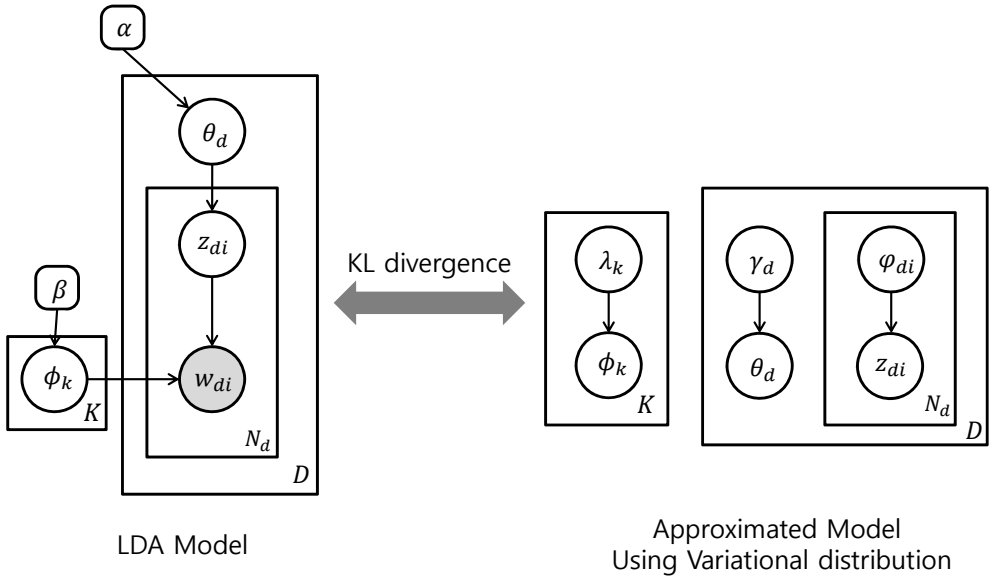


Figure 2.7 Graphical representation of the original LDA model and approximated model using variational distribution. The goal of variational inference is to optimize the variational parameters λ, γ, φ so that they can make the variational distribution close in Kullback-Leibler (KL) divergence to the posterior of LDA. (Blei, 2014)

λ, γ, φ. Thus, The lower-bound is given as follows:

$$\mathcal{L}(q) = p(w|\alpha,\beta) - D\left[q(\phi,\theta,z|\lambda,\gamma,\varphi)\,\|p(\phi,\theta,z|w,\alpha,\beta)\right] \tag{2.110}$$

$$= p(w|\alpha,\beta)\underbrace{\iint \sum_z q(\phi,\theta,z|\lambda,\gamma,\varphi)d\phi d\theta}_{=1}$$

$$- \iint \sum_z q(\phi,\theta,z|\lambda,\gamma,\varphi)\log\frac{q(\phi,\theta,z|\lambda,\gamma,\varphi)}{p(\phi,\theta,z|w,\alpha,\beta)}d\phi d\theta \tag{2.111}$$

$$= \iint \sum_z q(\phi,\theta,z|\lambda,\gamma,\varphi)\log\frac{p(w|\alpha,\beta)p(\phi,\theta,z|w,\alpha,\beta)}{q(\phi,\theta,z|\lambda,\gamma,\varphi)}d\phi d\theta \tag{2.112}$$

$$= \iint \sum_z q(\phi,\theta,z|\lambda,\gamma,\varphi)\log\frac{p(\phi,\theta,z,w|\alpha,\beta)}{q(\phi,\theta,z|\lambda,\gamma,\varphi)}d\phi d\theta. \tag{2.113}$$

Then, the lower-bound can be expanded by using the factorization of $q(\cdot)$ and $p(\cdot)$:

$$\begin{aligned}\mathcal{L}(q) &= \iint \sum_{z} q(\phi, \theta, z|\lambda, \gamma, \varphi) \log p(\phi, \theta, z, w|\alpha, \beta) d\phi d\theta \\ &\quad - \iint \sum_{z} q(\phi, \theta, z|\lambda, \gamma, \varphi) \log q(\phi, \theta, z|\lambda, \gamma, \varphi) d\phi d\theta \qquad (2.114) \\ &= E_q \left[\log p(\phi, \theta, z, w|\alpha, \beta)\right] - E_q \left[\log q(\phi, \theta, z|\lambda, \gamma, \varphi)\right] \qquad (2.115) \\ &= E_q \left[\log \left\{ \left(\prod_{k=1}^{K} p(\phi_k|\beta) \right) \prod_{d=1}^{D} p(\theta_d \mid \alpha) \prod_{i=1}^{N_d} p(z_{di} \mid \theta_d) p(w_{di} \mid z_{di}, \phi) \right\} \right] \\ &\quad - E_q \left[\log \left\{ \left(\prod_{k=1}^{K} q(\phi_k|\lambda_k) \right) \left(\prod_{d=1}^{D} q(\theta_d|\gamma_d) \right) \left(\prod_{d,i}^{D,N_d} q(z_{di}|\varphi_{di}) \right) \right\} \right] \qquad (2.116) \\ &= \sum_{k=1}^{K} E_q \left[\log p(\phi_k|\beta)\right] + \sum_{d=1}^{D} E_q \left[\log p(\theta_d \mid \alpha)\right] + \sum_{d=1}^{D} \sum_{i=1}^{N_d} E_q \left[\log p(z_{di} \mid \theta_d)\right] \\ &\quad + \sum_{d=1}^{D} \sum_{i=1}^{N_d} E_q \left[\log p(w_{di} \mid z_{di}, \phi)\right] \\ &\quad - \sum_{k=1}^{K} E_q \left[\log q(\phi_k|\lambda_k)\right] - \sum_{d=1}^{D} E_q \left[\log q(\theta_d|\gamma_d)\right] - \sum_{d=1}^{D} \sum_{i=1}^{N_d} E_q \left[\log q(z_{di}|\varphi_{di})\right]. \qquad (2.117)\end{aligned}$$

According to the above derivation, the objective function (lower-bound) $\mathcal{L}(q)$ turns out to be the sum of the expectation of the log probabilities of the posterior under the variational parameters minus the log probabilities of the variational distributions. Taking each expectation of the above equations can be analytically calculated. For the first term, recalling the definition of $p(\phi_k|\beta)$ that it is the Dirichlet distribution

$$p(\phi_k|\beta) = \frac{\Gamma\left(\sum_{v=1}^{V} \beta(v)\right)}{\prod_{v=1}^{V} \Gamma(\beta(v))} \prod_{v=1}^{V} \phi_k(v)^{\beta(v)-1}, \qquad (2.118)$$

the expectation of log probability with respect to the variational distribution q is de-

rived as follows:

$$E_q\left[\log p(\phi_k|\beta)\right] = E_q\left[\log\Gamma\left(\sum_{v=1}^{V}\beta(v)\right) - \sum_{v=1}^{V}\log\Gamma(\beta(v)) + \sum_{v=1}^{V}\left(\beta(v)-1\right)\log\phi_k(v)\right], \tag{2.119}$$

and note that $q(\phi, \theta, z|\lambda, \gamma, \varphi)$ is a function of only ϕ, θ, and z. Thus, we can get

$$E_q\left[\log p(\phi_k|\beta)\right] = \log\Gamma\left(\sum_{v=1}^{V}\beta(v)\right) - \sum_{v=1}^{V}\log\Gamma(\beta(v)) + \sum_{v=1}^{V}\left(\beta(v)-1\right)E_q\left[\log\phi_k(v)\right]. \tag{2.120}$$

In the similar way, we can also obtain the results of other expectations:

$$E_q\left[\log p(\theta_d \mid \alpha)\right] = \log\Gamma\left(\sum_{k=1}^{K}\alpha(k)\right) - \sum_{k=1}^{K}\log\Gamma(\alpha(k)) + \sum_{k=1}^{K}\left(\alpha(k)-1\right)E_q\left[\log\theta_d(k)\right] \tag{2.121}$$

$$E_q\left[\log p(z_{di} \mid \theta_d)\right] = \sum_{i=1}^{N_d}\sum_{k=1}^{K}\varphi_{di}(k)E_q\left[\log\theta_d(k)\right] \tag{2.122}$$

$$E_q\left[\log p(w_{di} \mid z_{di}, \phi)\right] = \sum_{i=1}^{N_d}\sum_{k=1}^{K}\sum_{v=1}^{V}\varphi_{di}(k)\delta\left[v - w_{di}\right]\log\phi_k(v) \tag{2.123}$$

$$E_q\left[\log q(\phi_k|\lambda_k)\right] = \log\Gamma\left(\sum_{v=1}^{V}\lambda_k(v)\right) - \sum_{v=1}^{V}\log\Gamma(\lambda_k(v)) + \sum_{v=1}^{V}\left(\lambda_k(v)-1\right)E_q\left[\log\phi_k(v)\right] \tag{2.124}$$

$$E_q\left[\log q(\theta_d|\gamma_d)\right] = \log\Gamma\left(\sum_{k=1}^{K}\gamma_d(k)\right) - \sum_{k=1}^{K}\log\Gamma(\gamma_d(k)) + \sum_{k=1}^{K}\left(\gamma_d(k)-1\right)E_q\left[\log\theta_d(k)\right] \tag{2.125}$$

$$E_q\left[\log q(z_{di}|\varphi_{di})\right] = \sum_{i=1}^{N_d}\sum_{k=1}^{K}\varphi_{di}(k)\log\varphi_{di}(k). \tag{2.126}$$

Then, we can allow the lower-bound $\mathcal{L}(q)$ to be a function with respect to variational parameters λ, γ, φ, observed variables w, and hyperparameters α, β by substituting these expectations into in Eq.2.117. In order to maximize the lower-bound $\mathcal{L}(q)$ with

respect to variational parameters λ, γ, φ, we take derivatives w.r.t these parameters, and then set this derivative to zero for yielding the optimal value of each variational parameter; that is,

$$\frac{\partial \mathcal{L}(q)}{\partial \varphi_{di}} = 0 \tag{2.127}$$

$$\frac{\partial \mathcal{L}(q)}{\partial \gamma_{d}} = 0 \tag{2.128}$$

$$\frac{\partial \mathcal{L}(q)}{\partial \lambda_{k}} = 0. \tag{2.129}$$

As a result, we can optimize variational parameters using coordinate ascent over the variational parameters as follows:

$$\varphi_{di}(k) \propto \exp\left\{E_q\left[\log \theta_d(k)\right] + E_q\left[\log \phi_k(w_{di})\right]\right\} \tag{2.130}$$

$$\gamma_d(k) = \alpha(k) + \sum_{i=1}^{N_d} \varphi_{di}(k) \tag{2.131}$$

$$\lambda_k(v) = \beta(v) + \sum_{d=1}^{D}\sum_{i=1}^{N_d} \varphi_{di}(k)\delta[w_{di} - v]. \tag{2.132}$$

Here, the expectations under q of $\log \theta_d(k)$ and $\log \phi_k(w_{di})$ are given by

$$E_q\left[\log \theta_d(k)\right] = \Psi(\gamma_d(k)) - \Psi(\sum_{k=1}^{K} \gamma_d(k)) \tag{2.133}$$

$$E_q\left[\log \phi_k(w_{di})\right] = \Psi(\lambda_k(w_{di})) - \Psi(\sum_{v=1}^{V} \lambda_k(v)), \tag{2.134}$$

where $\Psi(x) = \frac{d}{dx}\log\Gamma(x) = \frac{\Gamma'(x)}{\Gamma(x)}$ is the digamma function (the logarithmic derivative of the gamma function) whose detailed derivation is in Appendix A.1 of (Blei et al., 2003).

The iterative updates of the variational parameters in Eq. 2.130-Eq.2.132 are guaranteed to converge into a stationary point of the lower-bound. For the iteration, φ and γ are updated with λ fixed, and λ is updated given the fixed φ and γ. The iteration algorithm is finished after relative improvement of the lower-bound $\mathcal{L}$ is less than a preset

threshold or after the maximum number of iterations. After the algorithm converges, the parameters γ_d is used to obtain the topic proportion θ_d for the d-th document, and λ_k is used to calculate the topic-word distribution ϕ_k for the k-th topic. The final distribution results ϕ, θ are obtained by calculating an expectation of the approximate distribution $q(\cdot)$ given each optimal parameters λ, γ:

$$\hat{\phi}_k(v) = E_{\underbrace{q(\phi_k|\lambda_k)}_{Dirichlet}}[\phi_k(v)|\lambda_k] = \frac{\lambda_k(v)}{\sum_{v=1}^{V}\lambda_k(v)} \tag{2.135}$$

$$\hat{\theta}_d(k) = E_{\underbrace{q(\theta_d|\gamma_d)}_{Dirichlet}}[\theta_d(k)|\gamma_d] = \frac{\gamma_d(k)}{\sum_{k=1}^{K}\gamma_d(k)}. \tag{2.136}$$

The overall procedure of variational inference of LDA is summarized in Algorithm 2.

Algorithm 2 Variational inference algorithm for Latent Dirichlet Allocation

Input: A documnet set $\{w_{di}|d = 1, ..., D, i = 1, ..., N_d\}$, where d-th document consists of a set of N_d words and each $w_{di} \in \{1, ..., V\}$.
Hyperparameters α, β, and the number of topic K

Output: Variatinal parameters $\lambda_k \in \mathbb{R}^V, \gamma_d \in \mathbb{R}^K, \varphi_{di} \in \mathbb{R}^K$.

Initialize λ randomly.
while $\mathcal{L}(q)$ not converge **do**
  for all document indices $d \leftarrow 1, \ldots, D$ **do**
    Initialize $\gamma_d = 1$ (The constant 1 is arbitrary).
    while γ_d not converge **do**
      for all word indices $i \leftarrow 1, \ldots, N_d$ **do**
        for all topic indices $k \leftarrow 1, \ldots, K$ **do**
          Set $\varphi_{di}(k) \propto \exp\{E_q[\log \theta_d(k)] + E_q[\log \phi_k(w_{di})]\}$.
        end for
      end for
      for all topic indices $k \leftarrow 1, \ldots, K$ **do**
        Set $\gamma_d(k) = \alpha(k) + \sum_{i=1}^{N_d} \varphi_{di}(k)$.
      end for
    end while
  end for
  for all topic indices $k \leftarrow 1, \ldots, K$ **do**
    Set $\lambda_k(v) = \beta(v) + \sum_{d=1}^{D} \sum_{i=1}^{N_d} \varphi_{di}(k)\delta[w_{di} - v]$.
  end for
  Set $\mathcal{L}(q)$ by Eq.2.117.
end while

Chapter 3

Proposed Approach

Figure. 3.1 shows the schematic diagram of the proposed framework. We first apply a simple background subtraction (Stauffer & Grimson, 1999) to extract foreground map and detect corner points on the foreground pixels. We perform KLT (Tomasi & Kanade, 1991) on these corner point to extract trajectories. By using the KLT trajectories, we can reduce the tracking error in a crowded scene because KLT tracks corner points, which are relatively easier to track than each object in a crowded scene. Of course, the tracking of corner points under the far-field view may generate broken trajectories. Despite the broken trajectories, our framework can cope with this problem by considering co-occurrence property of many corner point trajectories. After KLT tracking, consequent trajectories are collected during a time interval. The trajectories in the same time interval compose a *collection* that is a mixture of diverse activities. The dozens of trajectory collections are piled as in Figure. 3.1, and a recent set of collections is used as an input to the proposed inference model for online update.

The proposed inference model is formulated in a probabilistic graphical framework including trajectory pattern model, spatio-temporal relation of trajectories, and

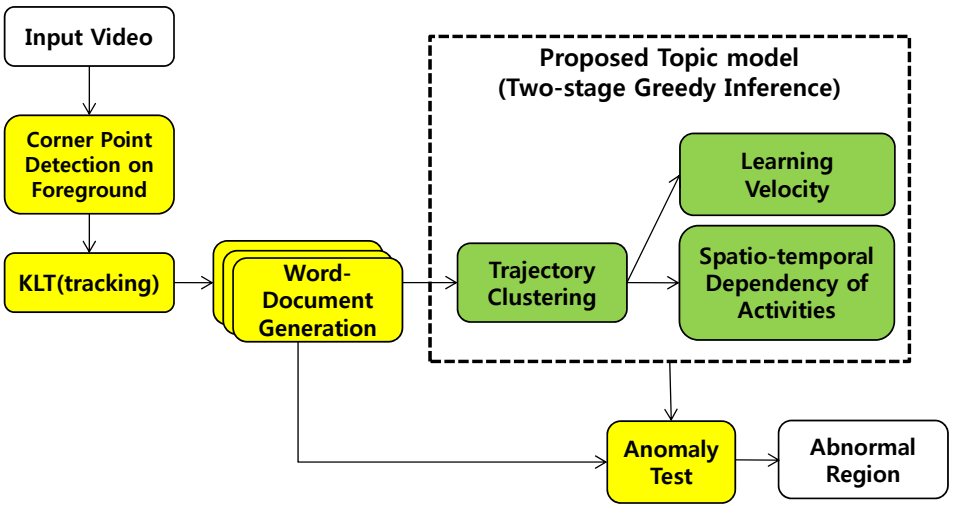


Figure 3.1 Overall scheme of the proposed method.

velocity model of each trajectory. To infer this model in online manner, instead of exact inference, an approximate method is proposed by two-stage greedy inference with three sub-models of trajectory clustering, spatio-temporal dependency modeling, and velocity learning. Lastly, the recently observed scene is tested by the trained model to detect anomalies in the current scene.

3.1 Probabilistic Inference Model

In this section, we describe the proposed model denoted with green in Figure. 3.1. The main frame of our approach is topic model such as Latent Dirichlet Allocation (LDA) (Blei et al., 2003), which is proposed for analysis of relationships between a set of documents and words in the documents. In this approach, the frequency of occurrence of each word in a document is used as a feature to train the model. For example, a word "relativity" tends to *co-occur* with words such as "Einstein", "energy", "gravity", "universe" in each document, so a set of the words is interpreted by the viewer as the physics-related topic. Because of the ability of co-occurrence modeling, LDA

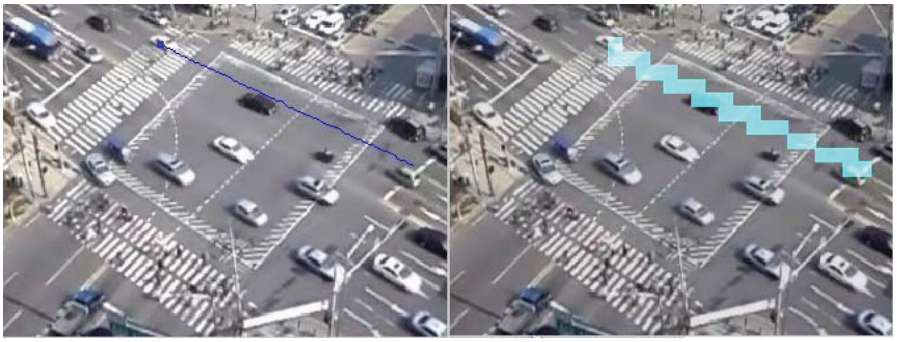


Figure 3.2 Example of a single trajectory corresponding with a set of cells.

is adopted as a baseline of many motion pattern learning frameworks (Emonet et al., 2011; Hospedales et al., 2009; Kuettel et al., 2010; Varadarajan et al., 2012; Wang et al., 2009). In these works, quantized local motions are treated as words, a set of the local motions in a video clip is treated as a document, and the topic can be treated as typical motion patterns.

In our approach, we also have to define variables corresponding to "word", "document", and "topic" in the topic model literatures. We define "words" as grid cells dividing a scene, where all of the cells in a scene have the same height and width. Instead, we newly define the velocity of trajectory (details are defined in the following), which can handle not only quantized direction inside a cell but also long-term actual velocity over dozens of frames. The trajectory is denoted by a set of grid cells as in in Figure. 3.2 and velocity vector defined as in Figure. 3.3. A "document" in the topic model corresponds to a collection of trajectories defined by a set of trajectories collected in a time interval. The trajectories are categorized into multiple typical patterns (topics), referred to as trajectory patterns (e.g. turn left from south to west, go-straight downwards, etc.).

The indexed variables for the proposed model are defined as following. The index of i-th cell of j-th trajectory in t-th collection of trajectories is denoted by $c_{tji} \in$

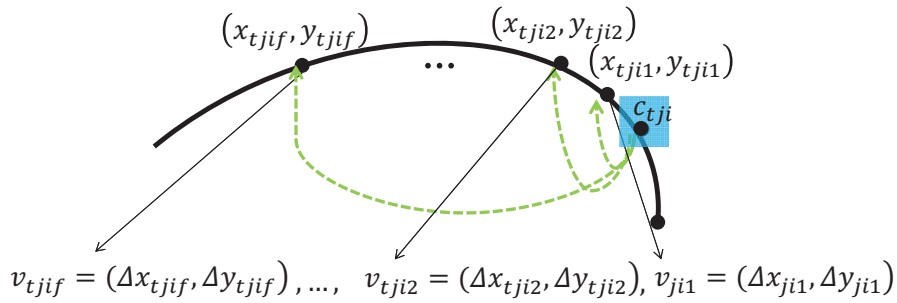


Figure 3.3 Synthetic trajectory with marked points and relative vectors from origin coordinate in cell c_{tji}.

$\{1, 2, ..., C\}$, where C is the number of grid cells in a scene. As depicted in Figure. 3.3, the velocity vector $v_{tjif} \in \mathbb{R}^2$ is defined as a relative vector from a point in the i-th cell on the j-th trajectory to the point at the frame of f-steps ahead. Following the above definition of variables, observed trajectories in the collection of the t-th time interval can be expressed by a set of cells $\{c_{tji}\}_{i=1,j=1}^{N_{tj},M}$ and a set of velocity vectors $\{v_{tjif}\}_{f=1,i=1,j=1}^{F_{tji},N_{tj},M}$, where M is the number of trajectories in the collection, N_{tj} is the number of cells where the j-th trajectory passes, and F_{tji} is the maximum value of f according to the length of the observed trajectory. We also define a design parameter F, acting as the maximum possible value for F_{tji}.

The state of t-th collection $s_t \in \{1, 2, ..., S\}$ is a set of trajectory patterns that can occur at the same time, such as a vertical moving state (a mixture of go-straight upwards and downwards) governed by a traffic light. The sequence of the state s_t is modeled so that it transits from one state to another over time, according to multinomial distribution with transition probability matrix π as follows:

$$p(s_t \mid s_{t-1}, s_{t-2}, ..., s_1) = p(s_t \mid s_{t-1}), \tag{3.1}$$

where,

$$s_t \mid s_{t-1} \sim Multi(s_t|\pi_{s_{t-1}}). \tag{3.2}$$

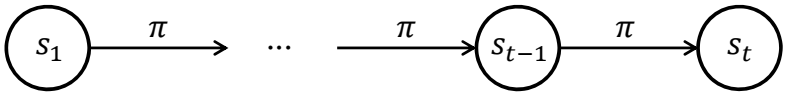


Figure 3.4 Graphical representation of the state transition model.

As in the equation, we assume that the state transition is only dependent on the previous state for the simplicity of the model. The graphical representation of this model is shown in Figure 3.4. For this example, the sequence of states $\{s_t\}$ is formed according to the change of a traffic signal as time passes. The constant S is a design parameter determining the number of states, usually selected to 2 or 3 according to the traffic changes in an intersection case.

If the state s_t is given, the distribution of topic occurrence (topic proportion) in the state can be determined. The topic occurrence probability vector for t-th collection is defined by $\theta_t \in \mathbb{R}^K$, where K is a design parameter that stands for the number of typical trajectory patterns in a scene. The θ_t is represented with a histogram that must sum to 1, and the distribution of θ_t is assumed to be Dirichlet distribution with given parameter α, i.e.,

$$\theta_t \mid s_t, \alpha \sim Dir(\theta_t \mid \alpha_{s_t}). \tag{3.3}$$

The θ_t is used as the parameter of multinomial distribution over the K trajectory patterns (topics) for the t-th collection. For example, if the current state s_t is about vertical movements determined by a traffic signal, the distribution parameter θ_t corresponding to the state s_t would make its components related to topics of horizontal traffic movements have zero or small values.

The trajectory pattern of the j-th trajectory in the t-th collection is denoted with $z_{tj} \in \{1, 2, ..., K\}$, which is defined to follow a multinomial distribution with the

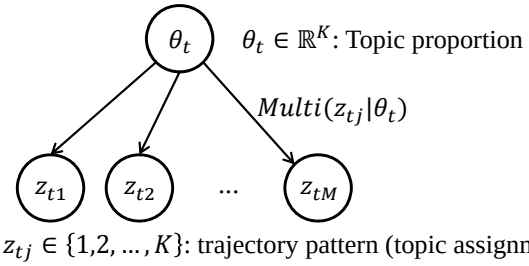


Figure 3.5 Graphical representation of the trajectory pattern (topic) generative model.

parameter θ_t, i.e.,

$$z_{tj} \mid \theta_t \sim Multi(z_{tj} \mid \theta_t). \tag{3.4}$$

Intuitively, this probability definition of $p(z_{tj} \mid \theta_t)$ encourages the t-th trajectory collection to have sparse possible topics (trajectory patterns). Also, we assume that trajectory patterns $z_{t1}, z_{t2}, ..., z_{tM}$ are independent and identically distributed (i.i.d) random variables given the parameter θ_t for the t-th trajectory collection, so the joint probability of z_{tj} is factorized as follows:

$$p(z_{t1}, z_{t2}, ..., z_{tM} \mid \theta_t) = \prod_{j=1}^{M} p(z_{tj} \mid \theta_t), \tag{3.5}$$

and the graphical representation is shown in Figure 3.5. In fact, it is hard to say that trajectory patterns assigned to each trajectory are always independent, but the i.i.d. assumption of the trajectory patterns under the known θ_t is very reasonable. This is because non-zero components of θ_t corresponding to current state s_t are dependent on co-occurring trajectory patterns which are governed by traffic signal. The co-occurring trajectory patterns have no chance of conflicting each other, so the dependency among them is negligible.

The multinomial parameter $\phi_k \in \mathbb{R}^C$, $k \in \{1, 2, ..., K\}$ holds spatial information about which cell has high probability to appear in the k-th trajectory pattern, where C is the number of cells in the scene (i.e. the scene is divided by grid into C cells).

The distribution of ϕ_k defined to be Dirichlet distribution with hyperparameter β as follows:

$$p(\phi_1, \phi_2, ..., \phi_K \mid \beta) = \prod_{k=1}^{K} p(\phi_k \mid \beta), \tag{3.6}$$

where,

$$\phi_k \mid \beta \sim Dir(\beta). \tag{3.7}$$

We define the cell c_{tji} to be generated by a multinomial distribution with the parameters $\phi_{z_{tj}} \in \mathbb{R}^C$ being related to the trajectory pattern z_{tj}, given by:

$$c_{tji} \mid z_{tj}, \phi_1, \phi_2, ..., \phi_K \sim Multi(c_{tji} \mid \phi_{z_{tj}}), \tag{3.8}$$

Even though a value of cell c_{tji} is not only dependent on the topic assignment z_{tj} and topics ϕ but also affected by the previous cell positions $c_{tj1}, c_{tj2}, ..., c_{tj(i-1)}$ in the actual environment, we assume that the generation of cell position c_{tji} is independent with the other cells given the topic assignment of j-th trajectory z_{tj} and topics $\phi_1, \phi_2, ., \phi_K$ for the simplicity of the model:

$$p(c_{tj1}, c_{tj2}, ..., c_{tjN_{tj}} \mid z_{tj}, \phi_1, \phi_2, ., \phi_K) = \prod_{i=1}^{N_{tj}} p(c_{tji} \mid z_{tj}, \phi_1, \phi_2, ., \phi_K). \tag{3.9}$$

Instead of this assumption, we additionally utilize the velocity vector to learn the temporal information of the trajectory pattern.

The velocity vector v_{tjif} is modeled to be drawn from a Gaussian distribution with its mean $\mu_{c_{tji}z_{tj}f}$ and variance $\Sigma_{c_{tji}z_{tj}f}$ as follows:

$$v_{tjif} \mid z_{tj}, c_{tji}, \mu, \Sigma \sim \mathcal{N}(v_{tjif} \mid \mu_{c_{tji}z_{tj}f}, \Sigma_{c_{tji}z_{tj}f}). \tag{3.10}$$

Consequently, the defined variables of the proposed model can deal with not only global-level activities such as spatio-temporal trajectory patterns governed by traffic signal but also micros-level activities such as precise velocities. Using variables and

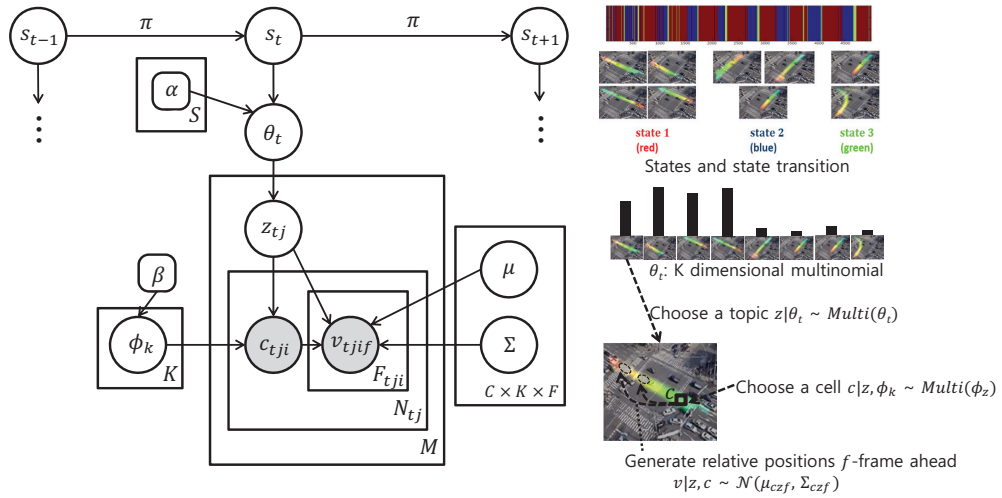


Figure 3.6 Graphical representation of the proposed model. The hidden variables are unshaded and the observed variables are shaded. The rectangles are "plate" notation which denotes replication.

their dependence defined in the above, the overall model to consider trajectory patterns (topics), velocity patterns of the trajectories, and spatio-temporal transition patterns of the states is graphically represented as shown in Figure. 3.6. The figure can be interpreted in a top-down order through the generative process (Blei et al., 2003), where the nodes denote random variables, and the edges denote possible dependence between random variables.

The primary goal of our framework is to infer the latent variables and parameters from the given observations $\{c_{tji}\}$ and $\{v_{tjif}\}$ in a surveillance video through an online unsupervised learning scheme. [1] This task can be done by posterior inference, which can be regarded as a reversal of the generative process that the graphical model illustrates. The posterior inference for all latent variables $s, \phi, \theta, z, \mu, \Sigma$ given

[1] To concisely represent notations, the set notation $\{\cdot\}$ without the range of index is defined as a set of variables containing all possible indices. Also, the variables without indices imply that they deal with all possible indices, such as,

$$c = \{c_{tji}\} = \{c_{tji}\}_{t=1,j=1,i=1}^{T,M,N_j}, p(s) = p\left(\{s_t\}_{t=1}^{T}\right) = \prod_{t=1}^{T} p(s_t).$$

the observations c, v and hyper-parameters α, β is as follows:

$$s^*, \phi^*, \theta^*, z^*, \mu^*, \Sigma^* = \underset{s,\phi,\theta,z,\mu,\Sigma}{\arg\max}\, p(s, \phi, \theta, z, \mu, \Sigma | c, v, \alpha, \beta), \tag{3.11}$$

where,

$$p(s, \phi, \theta, z, \mu, \Sigma | c, v, \alpha, \beta) = \frac{p(s, \phi, \theta, z, \mu, \Sigma, c, v | \alpha, \beta)}{p(c, v | \alpha, \beta)}. \tag{3.12}$$

The numerator on the right-hand side in Eq.(3.12) corresponds to a joint probability distribution represented by the proposed model. Also, using the chain rule and assumptions of independence among variables, the joint probability can be factorized into Eq.(3.13), which consists of the probability distributions defined in Eq. (3.1)-(3.10).

$$\begin{aligned} p(s, \phi, \theta, z, \mu, \Sigma, c, v | \alpha, \beta) = & \left(\prod_{k=1}^{K} p(\phi_k | \beta) \right) \\ & \prod_{t=1}^{T} p(s_t \mid s_{t-1}) p(\theta_t \mid s_t, \alpha) \prod_{j=1}^{M} p(z_{tj} \mid \theta_t) \\ & \prod_{i=1}^{N_{tj}} p(c_{tji} \mid z_{tj}, \phi) \prod_{f=1}^{F_{tji}} p(v_{tjif} \mid z_{tj}, c_{tji}, \mu, \Sigma). \end{aligned} \tag{3.13}$$

The learning of distribution parameters $(\phi, \theta, \mu, \Sigma)$ for the proposed model can be achieved by maximizing the probability $p(s, \phi, \theta, z, \mu, \Sigma, c, v | \alpha, \beta)$ with latent variables s, ϕ, θ, z, μ, Σ to be inferred under the given observations c, v and the hyper-parameters α, β. However, the exact inference is intractable due to non-convexity of the joint probability function and a tremendous search space caused by calculating the joint probability for all possible configurations of the latent variables to find the best case. Instead of exact inference, we propose an approximate inference method that will be presented in the Section 3.2.

As for an application of inference results of the proposed model, anomaly detection can be performed. Using the distribution parameters $\mu, \Sigma, \phi, \theta$ inferred from the

learning phase and the current observations $\{c_{t'ji}\}$, $\{v_{t'jif}\}$ at the current time t',[2] the a state $s^*_{t'}$ and a topic assignment $z^*_{t'j}$ for each trajectory j are estimated by maximizing a posterior:

$$s^*_{t'}, \{z^*_{t'1}, z^*_{t'2}, ., z^*_{t'M}\} = \arg\max_{s_{t'}, \{z_{t'j}\}} \left[p(s_{t'}, \{z_{t'j}\} | \{c_{t'ji}\}, \{v_{t'jif}\}, \mu, \Sigma, \phi, \theta, \alpha, \beta) \right]. \tag{3.14}$$

Here,

$$p(s_{t'}, \{z_{t'j}\} | \{c_{t'ji}\}, \{v_{t'jif}\}, \mu, \Sigma, \phi, \theta, \alpha, \beta) = \frac{p(s_{t'}, \{z_{t'j}\}, \{c_{t'ji}\}, \{v_{t'jif}\}, \mu, \Sigma, \phi, \theta | \alpha, \beta)}{p(\{c_{t'ji}\}, \{v_{t'jif}\}, \mu, \Sigma, \phi, \theta, \alpha, \beta)}. \tag{3.15}$$

The denominator of Eq.(3.15) is constant to the variation of optimization variables s, z, so it is enough to maximize the numerator (joint probability) of Eq.(3.15) to achieve Eq.(3.14). Therefore, the joint probability in Eq.(3.13) can substitute for the posterior in Eq.(3.14) by fixing $t = t'$ and removing $\prod_{t=1}^{T}$. The observations are extracted from trajectories of the current frame and $j \in [1, M]$, $i \in [1, N_j]$, $f \in [1, F_{ji}]$. Indeed, if the joint probability $p(s^*_{t'}, \{z^*_{t'j}\}, \{c_{t'ji}\}, \{v_{t'jif}\}, \mu, \Sigma, \phi, \theta | \alpha, \beta)$ in Eq.(3.13) has low value even with the optimal $s^*_{t'}, \{z^*_{t'j}\}$, the current scene is decided to be abnormal. However, as in case of model learning, exact inference of Eq.(3.14) is intractable. The details for anomaly detection with approximate method are described in Section 3.3.

3.2 Model Learning

An exact learning of the proposed model by maximizing the joint probability Eq.(3.13) is intractable because of the aforementioned reasons in the previous section. Hence, many conventional methods using various topic models (Wang et al., 2009; Hospedales et al., 2009; Kuettel et al., 2010; Emonet et al., 2011; Wang et al., 2011) commonly

[2]Because the anomaly detection task should be performed for every frame, we compose t'-th trajectory collections from the trajectories on the current frame.

employ collapsed Gibbs sampling (CGS) for an approximate learning of the models. CGS is a popular Markov Chain Monte Carlo (MCMC) approach for topic model learning. However, on the results of online MCMC learning for topic models (Canini et al., 2009), the results have shown that online MCMC learning is inferior to the offline learning. According to (Zhai et al., 2012), in case of distributed processing for the learning of the topic models, variational inference (VI)(Blei et al., 2003; Bishop, 2006) gives better results than CGS. To achieve an online learning of the proposed topic model, a large set of the trajectory collections for the offline learning needs to be separated by time. Because each separated set of the collections can be an input to the distributed processing, VI method can be a better option for the online learning of our model than CGS. VI assumes that each variational distribution used to approximate the posterior and to treat each document (in our case, collection of trajectories) is independent. For this reason, it is difficult to apply VI directly to our model because the model has the states for each collection which is dependent on the previous state. Moreover, inferring all latent variables all together is not efficient to real-time computation in terms of a search space.

In our greedy inference approach, in order to directly apply VI to the proposed model in Figure. 3.6, we utilize the fact that the state s_t is hardly changed in a short time for the online inference; thus, θ_t can be inferred without knowing the current state s_t. Also, to reduce the search space for the solution, we assume that each velocity pattern μ, Σ in a cell c of each typical pattern z is inseparable. On the assumption, we can find the typical patterns z based on the cells c at first, and then velocity patterns are mined on the regions of each typical pattern. This assumption is reasonable from the fact that activity regions c are more susceptible to the typical pattern z than precise velocity v. As a result, three simple sub-models are obtained as shown in Figure. 3.7. The first sub-model in Figure. 3.7-(a) is the same graphical model of Latent Dirichlet Allocation (LDA) (Blei et al., 2003), so it is straightforward to adopt online VI (Hoffman

et al., 2010) to the sub-model. If latent variables z and θ are given from the learning of the first sub-model in Figure. 3.7-(a), remaining latent variables $\{s_t\}$ and $\{\mu_{ckf}, \Sigma_{ckf}\}$ in Figure. 3.6 are conditionally independent by d-separation property (Bishop, 2006). In other words, $\{s_t\}$ does not influence $\{\mu_{ckf}, \Sigma_{ckf}\}$ and vice versa for the given z and θ. Therefore, we can reasonably optimize the sub-model of the first stage and then use these results to optimize the remaining two sub-models in Figure. 3.7-(b,c) in a greedy manner.

First, we optimize ϕ, θ, and z of the first sub-model in Figure. 3.7-(a) using LDA. The LDA can be used to cluster trajectories effectively, since it is robust to broken trajectories using the co-occurrence property. To be specific, because the collection is composed of concurrent trajectories in short time duration, the LDA can cluster co-occurring cells (words) in trajectory collections (documents) into the same trajectory patterns (topics). Using the inference result in the first stage, we use $\{\theta_t\}$ as observations to infer hidden variables $\{s_t\}$ and state transition matrix π in Figure. 3.7-(b). In addition, the pattern assignments of each trajectory z inferred in the first stage is also used as observations to infer Gaussian parameters per cell c, typical pattern k, and frame f in Figure. 3.7-(c). By this procedure, the search space to solve the complex model can be reduced effectively. Detailed description for each sub-model is presented in the following.

3.2.1 Online Trajectory Clustering

Leaning of the first sub-model takes a role of online trajectory clustering. For the online processing, the entire T collections of trajectories for the proposed model in Figure. 3.6 should be separated into a small set of collections by time. The small set that consists of the D collections is used as an input for the mini-batch learning whose results allow the model to be updated online. In other words, D is the number of collections for the mini-batch, so $\frac{T}{D}$ times of mini-batches should be performed for the whole video.

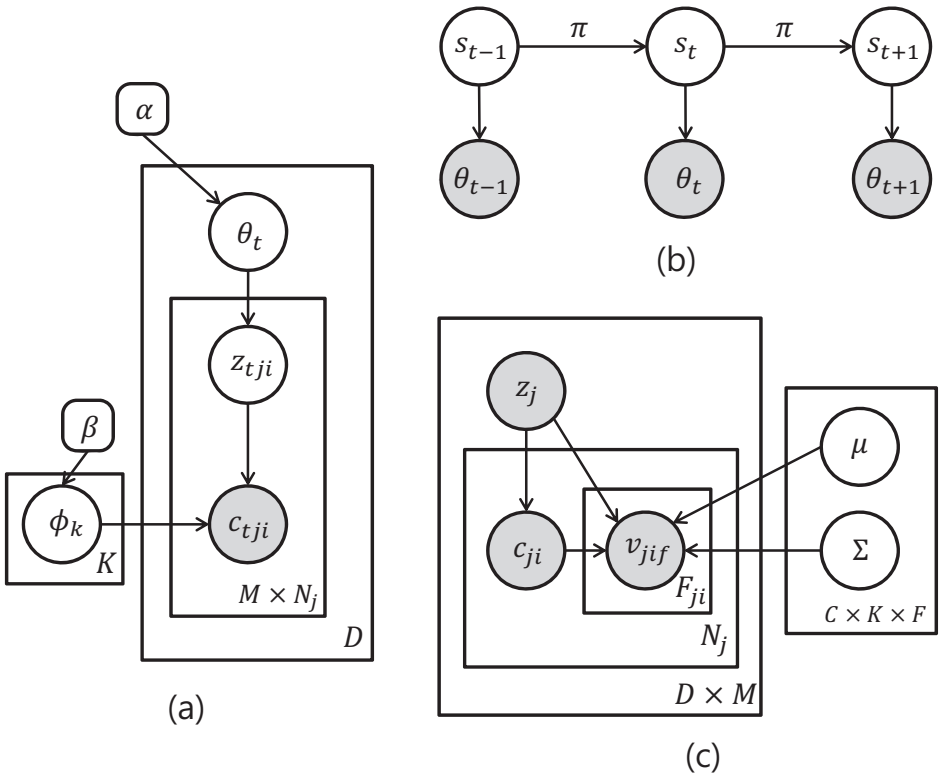


Figure 3.7 Three sub-models for two-stage learning.

Because the proposed model in Figure. 3.6 is assumed to be divided by ignoring the dependence between s and θ and between z and v, the full joint probability of the proposed model in the Eq.(3.13) can ignore $p(s_t \mid s_{t-1})$, $p(v_{tjif} \mid z_{tj}, c_{tji}, \mu, \Sigma)$ and can approximate $p(\theta_t \mid s_t, \alpha) \approx p(\theta_t \mid \alpha)$. Thus, the objective function of each mini-batch and joint probability of the first sub-model for the D collections is given by:

$$\phi^*, \theta^*, z^* = \arg\max_{\phi,\theta,z} p(\phi, \theta, z|c, \alpha, \beta), \tag{3.16}$$

where,

$$p(\phi, \theta, z, c|\alpha, \beta) = \left(\prod_{k=1}^{K} p(\phi_k|\beta)\right) \prod_{t=1}^{D} p(\theta_t \mid \alpha) \prod_{j=1}^{M} p(z_{tj} \mid \theta_t) \prod_{i=1}^{N_{tj}} p(c_{tji} \mid z_{tj}, \phi). \tag{3.17}$$

Also, in order to make Eq.(3.17) to be the same as the joint probability of LDA, the topic assignment z_{tj} for each trajectory is changed to be assigned for each cell (*i.e.* z_{tji}), and then z_{tj} is obtained by post-inference using z_{tji}.

Therefore, we can solve the problem with LDA in Figure. 3.7-(a). By changing the topic assignment z_{tj} into z_{tji} from the Eq.(3.17), the joint distribution of LDA is given by:

$$p(\phi, \theta, z, c|\alpha, \beta) = \left(\prod_{k=1}^{K} p(\phi_k|\beta)\right) \prod_{t=1}^{D} p(\theta_t \mid \alpha) \prod_{j=1}^{M} \prod_{i=1}^{N_{tj}} p(z_{tji} \mid \theta_t) p(c_{tji} \mid z_{tji}, \phi), \tag{3.18}$$

where $j \in \{1, 2, ..., M\}$ is the trajectory index in the collections, and $i \in \{1, 2, ..., N_{tj}\}$ is the cell index in a trajectory. To approximate the posterior related to the joint distribution of LDA in Eq.(3.18), a simpler variational distribution $q(\{\phi_k\}, \{\theta_t\}, \{z_{tji}\}|\lambda, \gamma, \varphi)$

that can be factorized for easier computation is utilized in VI as follows (Blei et al., 2003):

$$q(\{\phi_k\},\{z_{tji}\},\{\theta_t\}|\lambda,\varphi,\gamma) = \left(\prod_{k=1}^{K} q(\phi_k|\lambda_k)\right)\left(\prod_{t=1}^{D} q(\theta_t|\gamma_t)\right)\left(\prod_{t,j,i}^{D,M,N_{tj}} q(z_{tji}|\varphi_{tji})\right), \tag{3.19}$$

where λ, γ, and φ are the variational parameters used for approximate inference of ϕ, θ, and z respectively. Hence, instead of solving optimization of Eq.(3.16), the optimal values of the variational parameters are found as follow:

$$\begin{aligned} \widetilde{\lambda}^*, \gamma^*, \varphi^* &= \arg\min_{\lambda,\gamma,\varphi} D_{KL}\left(q(\cdot)\,\|p(\cdot)\right), \\ where \quad q(\cdot) &= q(\phi,\theta,z|\lambda,\gamma,\varphi), \\ p(\cdot) &= p(\phi,\theta,z|c,\alpha,\beta). \end{aligned} \tag{3.20}$$

The optimal variational parameters are founded by minimizing the Kullback-Leibler (KL) divergence D_{KL} between the variational distribution and the true posterior $p(\phi,\theta,z|c,\alpha,\beta)$ via an iterative fixed-point method (Blei et al., 2003). For online VI, mini-batch LDA in Eq.(3.20) is executed using the small set of D collections coming in as time goes on. Because the parameter of multinomial distribution ϕ_k is learned regardless of time index t, it should be updated for every mini-batch. For online inference of ϕ_k, we update the variational parameters λ for the ϕ as follows (Hoffman et al., 2010):

$$\lambda^* = (1-\rho_\tau)\,\lambda^* + \rho_\tau \widetilde{\lambda}^*. \tag{3.21}$$

where, ρ_τ is a decaying factor decreasing over time and $\widetilde{\lambda}^*$ is an optimized parameter from the mini-batch in Eq.(3.20). The updated parameter λ^* in Eq.(3.21) is utilized as an initial value in the next mini-batch. This initialization allows ϕ to be influenced by all collections in the past by only observing the recent collections for the mini-batch. The (Hoffman et al., 2010) has shown that the λ^* updated by Eq.(3.21) for online LDA

converges to a stationary point of the variational objective function and experimentally has verified that it could perform not worse than the offline LDA. The ϕ^*, θ^*, z^* are obtained by calculating an expectation of the approximate distribution $q(\cdot)$ given each optimal parameters $\lambda^*, \gamma^*, \varphi^*$. For more details, refer to (Blei et al., 2003; Hoffman et al., 2010).

After the optimization process for LDA, we get z^*_{tji} indicating the topic assignment of each cell as shown in Figure. 3.7-(a). This result cannot be directly used in the next stage because the inference result of the full model (of Figure. 3.6) is the latent variable z^*_{tj} indicating the most typical pattern of the j-th trajectory among the K clustered patterns. To resolve the incompatibility, we consider the mode of the inference results of the first sub-model as the results of the original model. For example, if we have $\{z^*_{tj1}, z^*_{tj2}, z^*_{tj3}, ..., z^*_{tjN_j}\}$ and $\{c_{tj1}, c_{tj2}, c_{tj3}, ..., c_{tjN_j}\}$ for a j-th trajectory in t-th collection, then we assign z^*_{tj} as

$$z^*_{tj} = Mode\{z^*_{tji}\}_{i=1}^{N_{tj}}. \tag{3.22}$$

This is a reasonable assignment since choosing the mode would give least error with respect to maximum likelihood estimation (Duda et al., 2000).

3.2.2 Spatio-Temporal Dependency of Activities

The spatio-temporal relationship among the typical patterns is represented in Figure. 3.7-(b). From the set $\{z^*_{tj}\}_{j=1}^{M}$ obtained in the first stage inference, θ^*_t per trajectory collection is also obtained. Given a set of histogram $\{\theta^*_t\}_{t=1}^{D}$, where D is the number of collections, we partition the D observations into S sets $\{\Theta_1, \Theta_2, ..., \Theta_S\}$. The objective function to minimize is the within-cluster sum of squares:

$$\underset{\{\Theta_n\}_{n=1}^{S}}{\arg\min} \sum_{n=1}^{S} \sum_{\overline{\theta}^*{}_t \in \Theta_n} \left\| \overline{\theta}^*{}_t - m_n \right\|^2, \tag{3.23}$$

where $m_n \in \mathbb{R}^K$ is the mean of vectors in a set Θ_n and $\{\overline{\theta^*}_t\}$ is the dimension-wise normalized version of $\{\theta_t^*\}$. In the normalization, different observation frequencies in topics are set to the same scale. To minimize the objective function, we perform K-means clustering with $K = S$. Then with the clustering results, we obtain the cluster indices $\{s_t^*\}_{t=1}^D$ for all $\{\theta_t^*\}_{t=1}^D$, where $s_t^* \in \{1, 2, ..., S\}$ corresponds to cluster index of θ_t^*. The state transition matrix π also can be obtained by counting the frequency of transition in the cluster indices. The parameter m_n implies general patterns about spatial co-occurrences of trajectory patterns, such as cars are moving horizontally (m_1) or cars are moving vertically (m_2). The m_n is also used to estimate a current state at the anomaly test phase.

In the online process, only $\{\theta_t^*\}_{t=1}^D$ residing inside a sliding time window is kept so that the model adapts to the changes in time. A size of the sliding window is designed to be bigger than the size of mini-batch for online-LDA in order to increase the clustering performance. As K-means performance depends much on initialization, we perform this multiple times with random initial conditions and use the best result. As the K-means algorithm is very fast, it scarcely affects entire computational time of the proposed method.

3.2.3 Velocity Learning

As in Figure. 3.7-(c), given clustered trajectory information $\{z_{tj}^*\}$ and the observations $\{c_{tji}\}$ and $\{v_{tjif}\}$, Gaussian models learn velocities of the trajectory. The velocities can be modeled for each pixel in the scene, but it is a waste of memory and needs extremely large amount of data. Assuming adjacent pixels in the scene have similar motions, we learn these motions based on each cell. In our modeling scheme, Gaussian models exist not only for each cell but also for each typical pattern. Therefore, since multiple typical patterns may exist for the same cell, multiple Gaussian models may exist to describe the complex motions of a single cell. An example of this case

would be a cell in the center of an intersection. The Gaussian model learns the statistical information about the position of a trajectory at f frame before. Figure. 3.3 is an illustration of obtaining the relative vector $v_{tjif} \in \mathbb{R}^2$ for a trajectory. Then for each Gaussian model, we update the Gaussian parameters $\mu \in \mathbb{R}^2$ and $\Sigma \in \mathbb{R}^{2\times 2}$ with each trajectory.

The update equation for μ is given by:

$$\mu_{c_{tji}z^*_{tj}f} = (1 - \rho_{c_{tji}z^*_{tj}f})\mu_{c_{tji}z^*_{tj}f} + \rho_{c_{tji}z^*_{tj}f} v_{tjif}, \tag{3.24}$$

where $\rho_{c_{tji}z^*_{tj}f}$ is the learning rate. For the online update of the covariance matrix Σ, we keep $Z \in \mathbb{R}^{2\times 2}$ as a second moment of v such that:

$$Z_{c_{tji}z^*_{tj}f} = (1 - \rho_{c_{tji}z^*_{tj}f})Z_{c_{tji}z^*_{tj}f} + \rho_{c_{tji}z^*_{tj}f} v_{tjif} v^T_{tjif}, \tag{3.25}$$

and the covariance matrix Σ is calculated by

$$\Sigma_{c_{tji}z^*_{tj}f} = Z_{c_{tji}z^*_{tj}f} - \mu_{tjif}\mu^T_{tjif}. \tag{3.26}$$

ρ_{ckf} is determined to be inversely proportional to the number of times that the model has been updated. To avoid from the model being overly stiff, we keep lower bound for ρ_{ckf}.

3.3 Anomaly Detection

The optimization problem of Eq.(3.14) for anomaly detection is related to find the most appropriate $s_{t'}$, $z_{t'j}$ from the observations $\{c_{t'ji}\}$, $\{v_{t'jif}\}$ and the distribution parameters obtained through learning procedure in Section 3.2. The distribution parameters are assumed to be fixed in the anomaly detection phase. Since the computational complexity for exact inference for Eq.(3.14) is heavy with complexity of $O(SK^M)$, we present approximate inference method. For the approximation, we make two assumptions: 1) the typical pattern (topic) of each trajectory is independent from others in a

state; 2) activity regions c are more dominant to determine the typical pattern than precise velocity v. Using the first assumption, we can estimate the topic assignment $z_{t'j}$ of j-th trajectory without knowing the current state $s_{t'}$; thus, $z_{t'j}$ is not dependent on $s_{t'}$, $\theta_{t'}$. The second assumption make the dependence between z and v to be ignored; thus μ and Σ can be also ignored. Using the assumptions, a posterior of topic assignment $z_{t'j}$ can be approximately computed by only given regional observations c and the learned multinomial parameters ϕ as follows:

$$p(z_{t'j}|\{c_{t'ji}\}_{i=1}^{N_j}, \{v_{t'jif}\}_{i=1,j=1}^{N_{t'j},F_{t'ji}}, \mu, \Sigma, \phi, \theta, \alpha, \beta) \approx p(z_{t'j}|\{c_{t'ji}\}_{i=1}^{N_{t'j}}, \phi). \tag{3.27}$$

Also, the approximate posterior can be factorized into likelihood and a prior by Bayes' rule,

$$p(z_{t'j}|\{c_{t'ji}\}_{i=1}^{N_{t'j}}, \phi) \propto p(\{c_{t'ji}\}_{i=1}^{N_{t'j}}|z_{t'j}, \phi)p(z_{t'j}|\phi). \tag{3.28}$$

Because the likelihood $p(\{c_{t'ji}\}_{i=1}^{N_{t'j}}|z_{t'j}, \phi)$ follows multinomial distribution defined as in Eq.(3.8) and the prior is uniform, we can find the proper topic assignment $z^*_{t'j}$ given by:

$$z^*_{t'j} = \underset{k\in\{1,...,K\}}{\arg\max} \left[p(\{c_{t'ji}\}_{i=1}^{N_{t'j}}|z_{t'j}, \phi_k)\right]. \tag{3.29}$$

Likewise, the state $s^*_{t'}$ is estimated by utilizing $\{m_n\}_{n=1}^{S}$ obtained in Eq.(3.23) and the K-dimensional histogram $\theta^*_{t'}$ calculated from the frequency of $\{z^*_{t'j}\}_{j=1}^{M}$ as follows:

$$s^*_{t'} = \underset{s\in\{1,...,S\}}{\arg\min} \|\theta^*_{t'} - m_s\|. \tag{3.30}$$

As a result, the computational complexity of the posterior optimization in Eq.(3.14) can be reduced from $O(SK^M)$ into $O(KM)+O(S)$ via the proposed approximation.

By using the estimated $s^*_{t'}$ and $\{z^*_{t'1}, z^*_{t'2}, ..., z^*_{t'M}\}$, we can assume all latent variables are given, so the observations $\{c_{t'ji}\}$ and $\{v_{t'jif}\}$ are tested based on the trained

model in reverse:

$$p(\{c_{t'ji}\},\{v_{t'jif}\}|s^*_{t'},\{z^*_{t'j}\},\mu,\Sigma,\phi,\theta,\alpha,\beta) \propto$$
$$p(\{c_{t'ji}\},\{v_{t'jif}\},s^*_{t'},\{z^*_{t'j}\},\mu,\Sigma,\phi,\theta|\alpha,\beta). \quad (3.31)$$

The right-hand side of Eq.(3.31) can be factorized into the six pre-defined distributions Eq.(3.1-3.10) by conditional independence as in case of Eq.(3.13). In fact, the probability of learning parameters $p(\phi_k|\beta)$, $p(\theta^*_{t'} \mid s^*_{t'},\alpha)$ do not have influence on the Eq.(3.31). Thus, we check the remaining four conditions in Eq.(3.1,3.4,3.8,3.10) to decide whether the current state or each trajectory is normal or not:

(a) For the current state, $p(s^*_{t'} \mid s^*_{t'-1})$ defined in Eq.(3.1) is tested using the state transition matrix π and the given the previous state $s^*_{t'-1}$. It is to examine the temporal relation among the typical patterns of trajectories.

(b) For the topic assignment $z^*_{t'j}$ of j-th trajectory in the current scene, $p(z^*_{t'j}|m_{s^*_{t'}})$ defined in Eq.(3.4) is tested. Even though each trajectory is assumed to be independent of others when the inference of Eq.(3.14) is approximated, after estimating the dominent current state $s^*_{t'}$, an abnormal trajectory violating the current state can be detected. It can consider the spatial relation among the typical patterns of trajectories.

(c) For a set of cells $\left\{c_{t'ji}\right\}_{i=1}^{N_{t'j}}$ passed by j-th trajectory, $p\left(\left\{c_{t'ji}\right\}_{i=1}^{N_{t'j}}|z^*_{t'j},\phi\right)$ defined in Eq.(3.8) is tested given the topic assignment $z^*_{t'j}$. It is to examine the overall path of the trajectory.

(d) For a set of velocities $\left\{v_{t'jif}\right\}_{i=1,f=1}^{N_{t'j},F_{t'ji}}$ obtained by calculating relative vectors as described in Figure. 3.3, $p\left(\left\{v_{t'jif}\right\}_{i=1,f=1}^{N_{t'j},F_{t'ji}}\middle|z^*_{t'j},\left\{c_{t'ji}\right\}_{i=1}^{N_{t'j}},\mu,\Sigma\right)$ defined in Eq.(3.10) is tested. It is to detect an trajectory with abnormal speed although its overall path is similar to one of the typical patterns.

If the current state has low probability on the condition (a), the state of the current frame is decided to be abnormal. Also, a trajectory that has low probability under at least one of the conditions (b)$\sim$(d) is determined to be abnormal; thus, a cell contain-

ing current position of the abnormal trajectory is regarded as an abnormal region.

3.4 Summary of the Proposed Method

The proposed method uses the two-stage greedy inference to learn the proposed probabilistic model. The latent variables in the proposed model in Figure. 3.6 have a knowledge about the overall path of typical patterns, their spatio-temporal dependency, and their precise velocities. Given the observations defined in the Section. 3.1, the proposed inference method can be summarized as **Algorithm 3** and **Algorithm 4**.

Algorithm 3 Two-stage Greedy Inference (Model Learning)

Input: $\{c_{tji}\}_{i=1,j=1,t=1}^{N_{tj},M,T}$, $\{v_{tjif}\}_{f=1,i=1,j=1,t=1}^{F_{tji},N_{tj},M,T}$ ▷ T is the total number of trajectory collections in the video.

Output: $\{s_t\}, \{\phi_k\}, \{\theta_t\}, \{z_{tji}\}, \{\mu_{ckf}\}, \{\Sigma_{ckf}\}, \{m_n\}$ for all indices.

for $\tau \leftarrow 1, \ldots, \frac{T}{D}$ **do** ▷ D is the number of collections for the mini-batch. (In our case, D = 10)

For each set of collection for the mini-batch, optimize

$\phi^*, \theta^*, z^* = \arg\max_{\phi,\theta,z} p(\phi, \theta, z|c, \alpha, \beta)$ in Eq.(3.16)

Find a topic assignment, $z_{tj}^* = Mode\{z_{tji}^*\}_{i=1}^{N_{tj}}$ by Eq.(3.22).

Using the given θ^*, optimize

$\arg\min_{\{\Theta_n\}_{n=1}^S} \sum_{n=1}^{S} \sum_{\overline{\theta}^*{}_t \in \Theta_n} \left\| \overline{\theta}^*{}_t - m_n \right\|^2$ using K-means.

Then we obtain $\{s_t^*\}$ and $\{m_n\}$.

Using the given z^* from Eq.(3.22) and observations c and v,

update Gaussian parameters by Eq.(3.24)-(3.26)

$\mu_{c_{tji}z_{tj}^*f} = (1 - \rho_{c_{tji}z_{tj}^*f})\mu_{c_{tji}z_{tj}^*f} + \rho_{c_{tji}z_{tj}^*f} v_{tjif}$

$\Sigma_{c_{tji}z_{tj}^*f} = Z_{c_{tji}z_{tj}^*f} - \mu_{tjif}\mu_{tjif}^T$

where,

$Z_{c_{tji}z_{tj}^*f} = (1 - \rho_{c_{tji}z_{tj}^*f}) Z_{c_{tji}z_{tj}^*f} + \rho_{c_{tji}z_{tj}^*f} v_{tjif} v_{tjif}^T$

end for

Algorithm 4 Anomaly test

Input: Observations $\{c_{t'ji}\}_{i=1,j=1}^{N_{tj},M}$, $\{v_{t'jif}\}_{f=1,i=1,j=1}^{F_{tji},N_{tj},M}$ and distribution parameters $\{\phi_k\}, \{\mu_{ckf}\}, \{\Sigma_{ckf}\}, \{m_n\}$.

Output: Indices of abnormal trajectory $j \in \{1, ..., M\}$.

for every current frame t' **do**
  for $j \leftarrow 1, \ldots, M$ **do**
    $z^*_{t'j} = \underset{k\in\{1,\ldots,K\}}{\arg\max} \left[p(\{c_{t'ji}\}_{i=1}^{N_{t'j}} | z_{t'j}, \phi_k)\right]$
  end for
  Calculate, $\theta^*_{t'} = histogram\left(\{z^*_{t'j}\}_{j=1}^{M}\right)$
  $s^*_{t'} = \underset{s\in\{1,\ldots,S\}}{\arg\min} \left\|\theta^*_{t'} - m_s\right\|$
  Using the estimated $s^*_{t'}$ and $\{z^*_{t'j}\}_{j=1}^{M}$,
  Test $p(s^*_{t'} \mid s^*_{t'-1})$ defined in Eq.(3.1)
  for $j \leftarrow 1, \ldots, M$ **do**
    Following three probabilities are calculated and compare with the threshold:
    $p(z^*_{t'j} | m_{s^*_{t'}})$ defined in Eq.(3.4)
    $p\left(\left\{c_{t'ji}\right\}_{i=1}^{N_{t'j}} | z^*_{t'j}, \phi\right)$ defined in Eq.(3.8)
    $p\left(\left\{v_{t'jif}\right\}_{i=1,f=1}^{N_{t'j},F_{t'ji}} \middle| z^*_{t'j}, \left\{c_{t'ji}\right\}_{i=1}^{N_{t'j}}, \mu, \Sigma\right)$ defined in Eq. (3.10)
  end for
end for

Chapter 4

Experiments

We have done experiments on six different videos to analyze motion patterns and to detect abnormal activities. The MIT dataset is from (Wang et al., 2009), the QMUL Junction dataset is from (Hospedales et al., 2009), Wide Intersection (WI) video is our own dataset of an eight-lane road with heavy traffic, the UCSD dataset is from (UCSD, 2010), the UMN dataset is from (UMN, 2009), and the level crossing is from (Machy et al., 2007). The first three datasets are from intersections and used to evaluate the validities of the unsupervised modeling results of our method. In these videos, traffic flows are governed by a trafic signal which has been modeled with state transition in our model. The other three datasets were used to detect abnormal activities in scenes. These videos contain abnormal activities which are hard to detect in case of using quantized directions and conventional topic modeling methods (Wang et al., 2009; Hospedales et al., 2009; Kuettel et al., 2010; Emonet et al., 2011; Varadarajan et al., 2012; Wang et al., 2011).

The cell size of each video was identically fixed to 10×10 and the mini-batch size D was fixed to 10 in the all experiments. We equally set the number of topic K to 12

for three intersection videos and K to 3 for other videos. This is because, unlike intersection datasets, the latter three datasets are in narrow field-of-view situations where moving objects have only a few typical patterns. Furthermore, we experimented with different K on the state estimation and the prediction task to be described in Section 4.1 and 4.3, but the variation of K did not have a significant impact on the performance as long as K was not significantly far from the actual number of typical patterns. The experiments were conducted on a computer with Intel i5 2500, 3.3GHz CPU. In spite of non-optimized C++ implementation and single core processing, the proposed method could run on almost real-time (18-20fps), including motion extracting, model learning, and anomaly testing tasks.

4.1 Result of Traffic Pattern Understanding

WI dataset: Modeling results for the WI dataset are shown in Figure. 4.1. The number of state S is set to 3, and each state are represented in red, blue, and green. The latent variable set $\{s_t\}_{t=1}^{D}$ inferred by the Eq.(3.23) is graphically represented with the colored bar on the top of the figure. The horizontal axis of the bar, namely, represents time interval index t of the collection of trajectories. In this bar, we can find that each state changes regularly depending on time. The change of states coincides with the traffic lights which controls movement of vehicles and pedestrians. The state transitions are not well learned at first, but as a result of online learning, the model well describes the state and the transition of states as more data comes in. Our online learning correctly updates the model as more data are observed.

The transition matrix π is shown on the right of the bar. The probability for a transition from state i to state j is π_{ij}. Higher probability is denoted as white, whereas black denotes low probability. The matrix shows that the most probable state transition occurs in the order of $1 \rightarrow 2 \rightarrow 3 \rightarrow 1$, except staying on the same state. Each state is represented by a mixture of co-occurring typical activity patterns. Since the

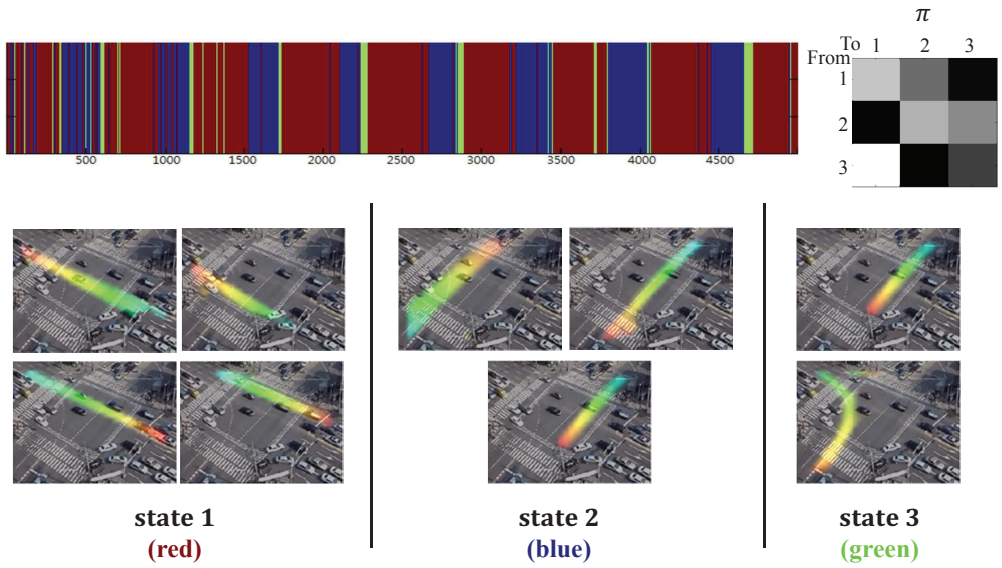


Figure 4.1 Typical patterns and their spatio-temporal relationship for the WI video sequence. The colored bar on the top shows state estimation. The transition matrix is shown on the top-right, where higher probability is denoted as white. The typical moving patterns are denoted with red and blue coloring, where objects move from red to blue. (best viewed in color)

width of the road in WI video is wide (eight-lanes), each pattern appears per single or double lane. Typical patterns are shown on the bottom three subfigures in Figure. 4.1. The patterns are denoted with red and blue coloring, where objects move from red to blue. State 1 is composed of four typical activity patterns: cars coming and going from northwest to southeast. In state 2, cars are coming and going from northeast to southwest, which cannot happen at the same time with state 1. State 3 is a mixture of turning left and going-straight from southwest. During left turn signal, which is state 3, there is no activity going from northeast to southwest. We can also find left turn signal is very short compared to other states as shown in the bar.

In the typical patterns results in Figure 4.1, we can see that the number of typical patterns shown the figure is only nine even though we designed the parameter K to be

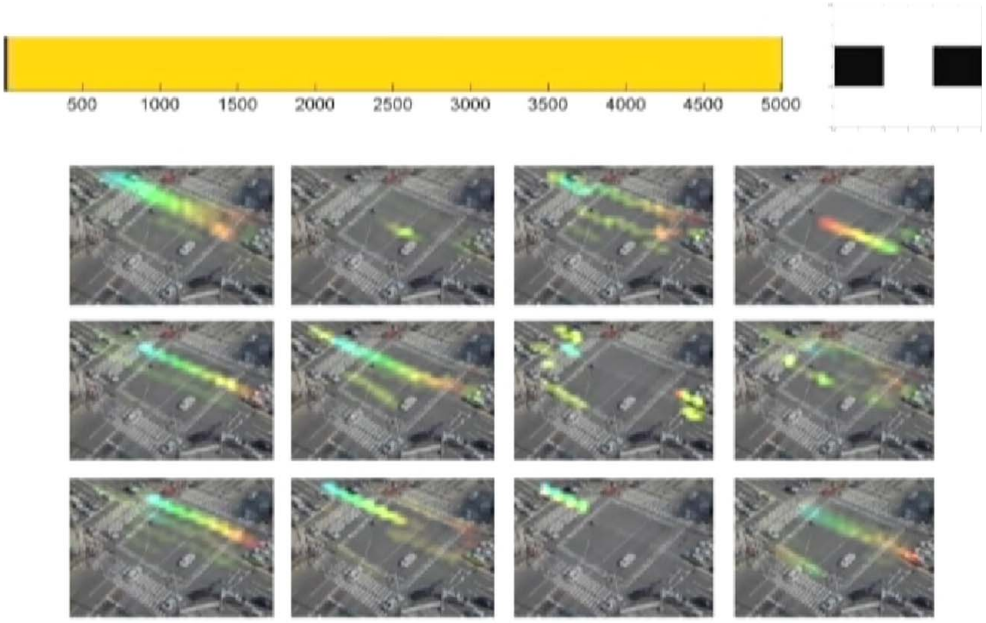
(a) (b) (c)

Figure 4.2 Omitted typical patterns to facilitate display of trajectory patterns. (best viewed in color)

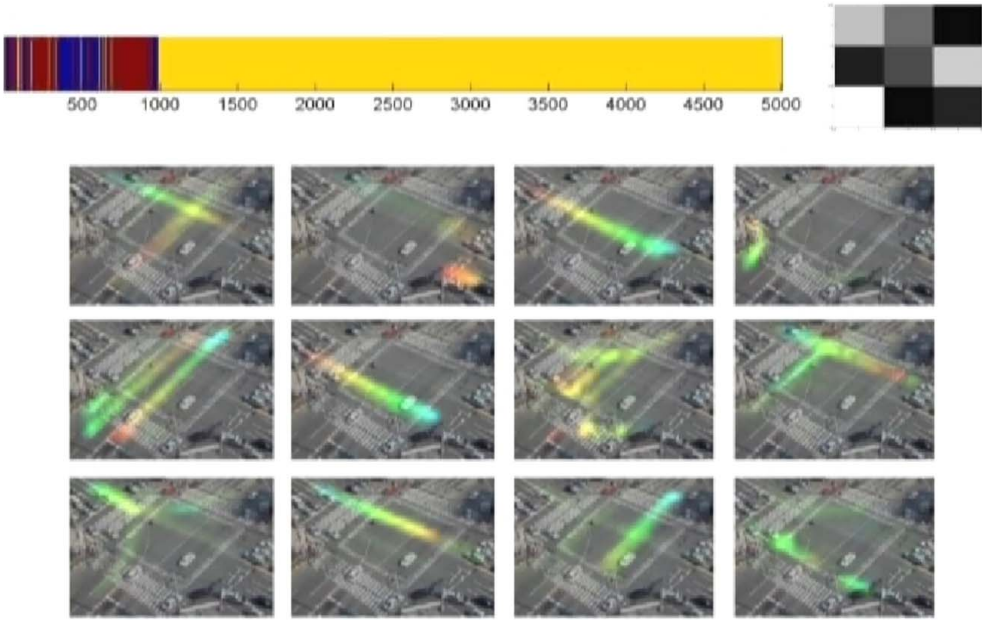
12. The omitted typical patterns are shown in Figure 4.2. The first omitted pattern has to be assigned to the state 1, but we did not put it in the figure on the paper due to a lack of space. The reason why the state 1 contains more patterns than other states is that the volume of traffic going straight from bottom-right to up-left and the reverse is much heavier than others. We also omitted the right turn patterns in the bottom right of the scene (red box shown in Figure 4.2-(b,c)) because the right turn is always permitted; so it should be assigned to all of the states.

Figure 4.3 and 4.4 show the process of online inference. At the first mini-batch, since the number of observed trajectories are small, the modeling result is very crude, and motion patterns are only straight moving from north-west to south-east and the reverse moving. However, as time goes on and as the number of observed trajectories increases, trajectory patterns begin to converge.

QMUL Junction dataset: QMUL Junction Dataset is the footage of objects crossing an intersection which has four-lane and right turn signals. Three states are used for this experiment. Results are shown in Figure. 4.5. In the figure, state 1 describes activities with right turn signal. State 2 includes activities corresponding to vertical movements. Similarly, state 3 captures horizontal movements of cars. As shown in the col-

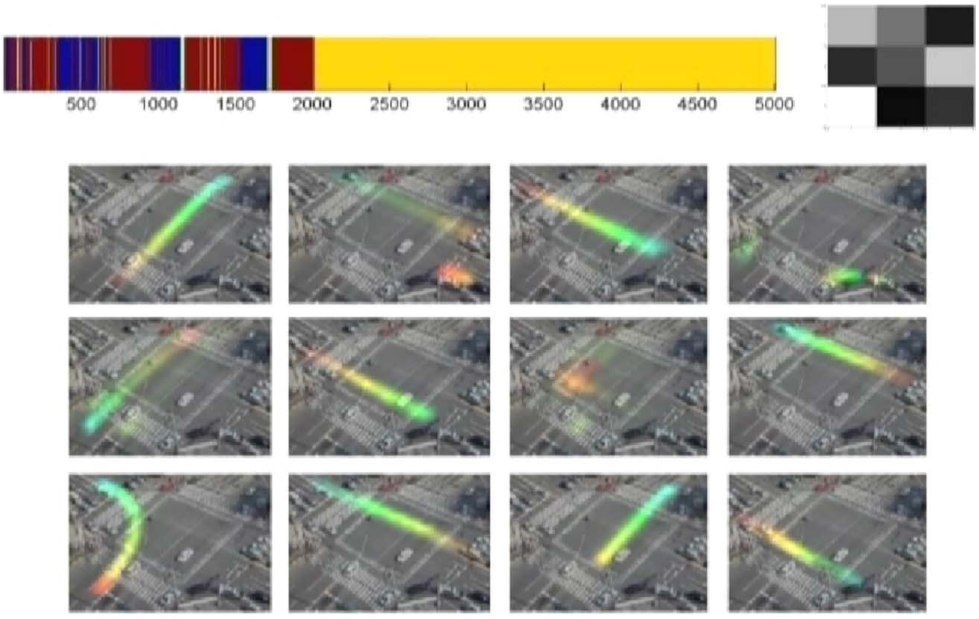


(a) Results after the first mini-batch.

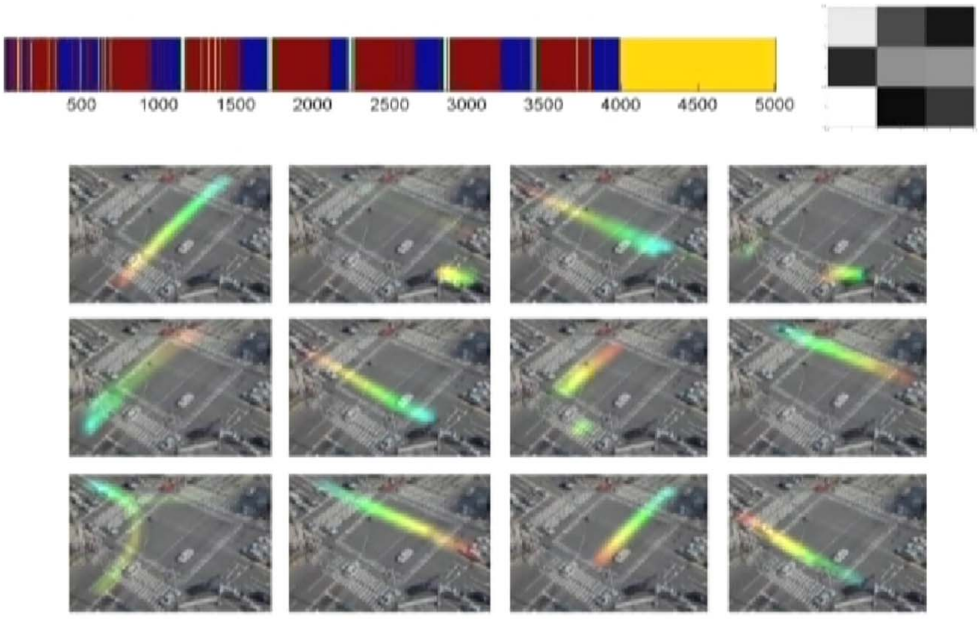


(b) Results after obtaining 1000 trajectory collections (100 mini-batches).

Figure 4.3 The process of online inference-(1).



(a) Results after obtaining 2000 trajectory collections (200 mini-batches).



(b) Results after obtaining 4000 trajectory collections (400 mini-batches).

Figure 4.4 The process of online inference-(2).

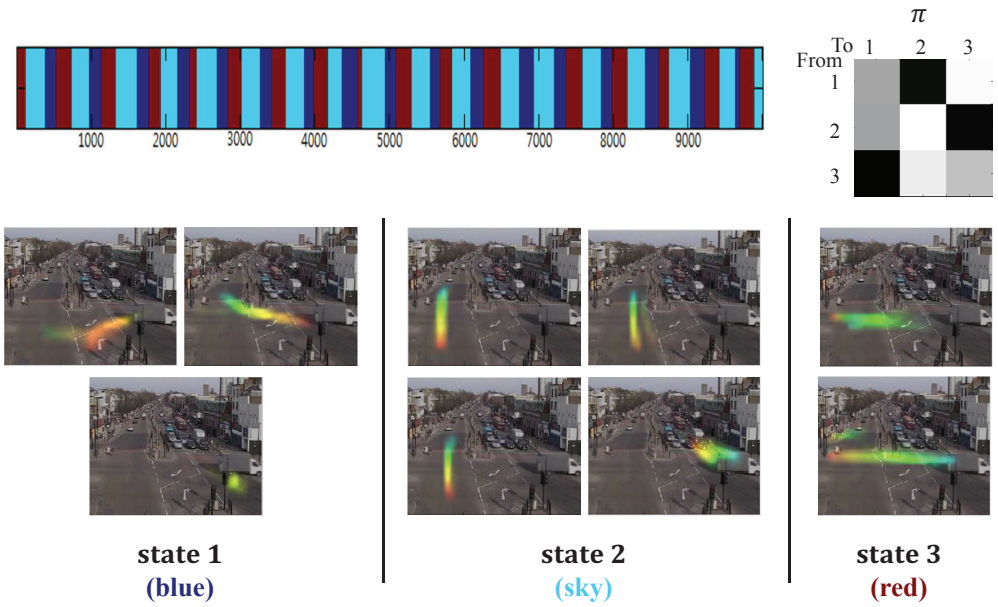


Figure 4.5 Typical patterns and their spatio-temporal relationship for the QMUL video sequence. The colored bar on the top shows state estimation. The transition matrix is shown on the top-right, where higher probability is denoted as white. The typical moving patterns are denoted with red and blue coloring, where objects move from red to blue. (best viewed in color)

ored bar and the transition matrix π, states repeatedly change in order of 1→3→2→1. This transition shows well a change of activity controlled by the signal in the scene. Vertical movements of cars appear when right turn signal is finished, and the horizontal straight signal starts after the vertical straight patterns.

MIT dataset: We applied two-stage greedy learning to extract two global states from MIT junction dataset. Figure. 4.6 shows the results. Unlike the above two datasets (WI and QMUL videos), strict state classification caused by a traffic signal is impossible in MIT video because turning and crossing movements are not protected by traffic signals. Hence, we set $S = 2$ for the MIT data so that only rough state assignments (vertical and horizontal moving) could be done. State 1 represents vertical activities

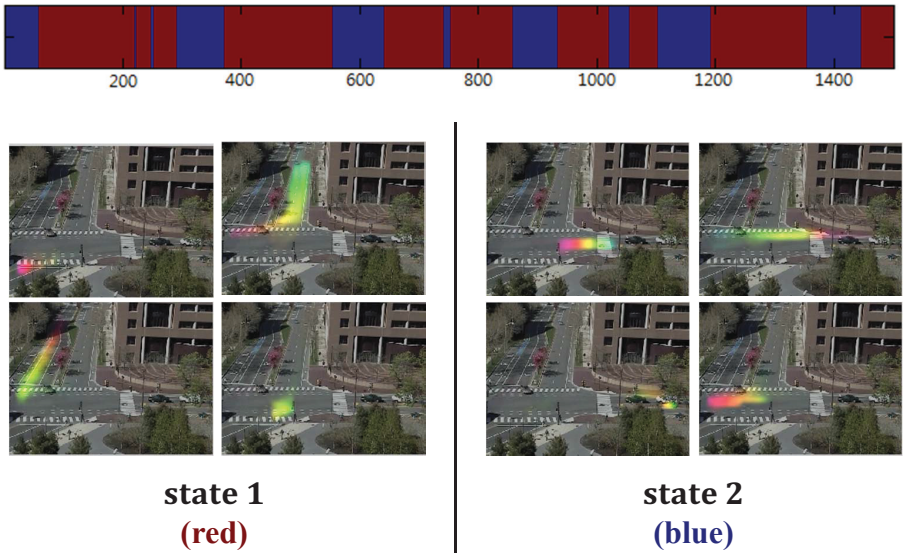
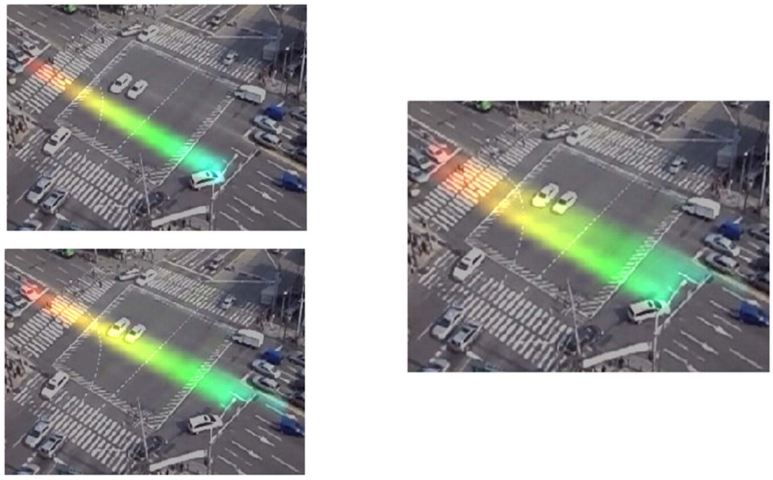


Figure 4.6 Typical patterns and their spatio-temporal relationship for the MIT video sequence. The colored bar on the top shows state estimation. The typical moving patterns are denoted with red and blue coloring, where objects move from red to blue. (best viewed in color)

and state 2 describes horizontal car movements. These two states are alternately repeated, closely relates to the traffic rules in the dataset. In this case, however, KLT tracker performs poorly for objects turning right, which come from bottom and go to right, because they are occluded by the traffic light pole. Although the proposed model can deal with general cases of broken trajectories by co-occurrence property, it still has a limitation in the case that trajectories are always broken at the same position. For this reason, a collection cannot often include the trajectories in both sides of the breaking position (e.g. fixed occlusion) at the same time because the collection just covers short duration. Hence, it is difficult to apply co-occurrence property to the consistently broken tracks. Performance improvement is expected if a more robust feature tracker such as (Rodriguez et al., 2009) is used.

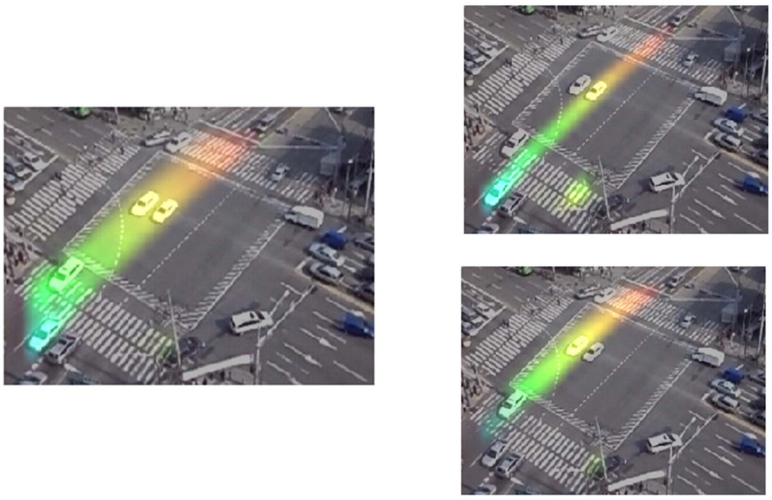
Variation of design parameters: As shown in the above results on the three



(a) K=12 (b) K=8

Figure 4.7 The example of merging two typical patterns. Adjacent two patterns (each pattern exist per lane) are merged into one typical pattern under the setting of a small K.

datasets, the proposed method gives an interpretation of activities in the scene (e.g. finding typical activities in unsupervised way, learning spatio-temporal relation among the typical activities), which are essential tasks of the topic model based approach (Wang et al., 2009; Hospedales et al., 2009; Kuettel et al., 2010; Emonet et al., 2011; Varadarajan et al., 2012). According to the qualitative results of our traffic pattern understanding, the precise parameter design for the number of topic K and the number of state S seems to be critical. However, even if $K = 12$ is not exactly the same as the actual number of typical patterns, scene understanding performance of the proposed method is not critically affected. For instance, when K is designed to be smaller than the actual number of typical patterns, co-occurring similar two typical patterns are sometimes merged into one as shown in Figure 4.7. On the other hand, with a large K, as shown in Figure 4.8, a typical pattern (e.g. go straight) can be split into multiple subpatterns (e.g. go straight in each lane) as long as K is not significantly far from the actual number. However, if K is set to very small value as shown in Figure 4.9, the



(a) K=12 (b) K=16

Figure 4.8 The example of splitting two typical patterns. One typical pattern is split into adjacent two typical patterns (each pattern exist per lane).

proposed method cannot detect a certain anomaly. The first trajectory pattern model in the figure is merged from left-turn and going straight, and the merged pattern is not a normal activity which can be dangerous in the real situation. Therefore, K should be set to a larger value than the number of inherent traffic patterns that can not be merged into the other patterns in the scene.

Likewise, the result of variation of S is similar to the case of K. As shown in Figure 4.10, if S is designed to bigger or smaller than the actual number of states, the trained results of parameter $\{m_n | n = 1, .., S\}$ can be split or merged. Practically, the case of $S = 4$ is not a problem. However, when $S = 2$, the proposed model cannot detect some abnormal events related to the requirement 3 suggested in Section 1.1 because a set of trajectory patterns in the state 2 is composed of activities with different traffic signal. In other words, setting S to too low value can cause under-modeling and false negatives. Thus, as we set S to a larger value than the actual number of states according to the kinds of traffic signals, the number of S does not affect the

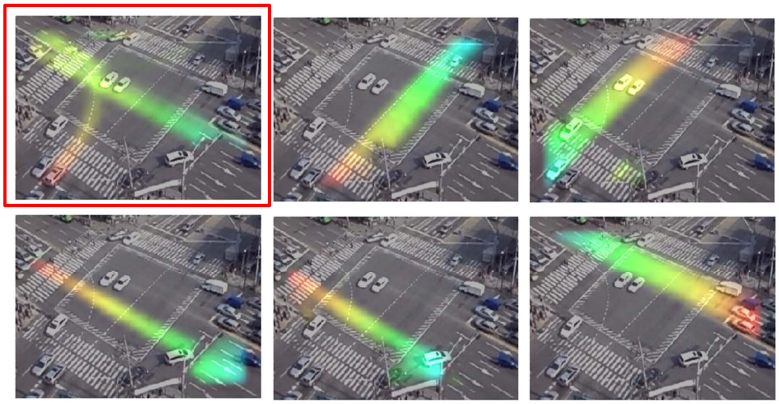


Figure 4.9 Trajectory patterns when $K = 6$ (highly under-designed).

performance much.

Although K and S are the fixed design parameters in our method, the proposed model can adapt properly the changing environment. The example can be a reversible lane, where cars go upwards in the morning and go downwards in the evening. In this case, even the model with the same number of topics K can adapt to the change of direction of the pattern. These cases do not disturb the automatic understanding of traffic patterns, and the simulation results on this matter will be described in Section 4.2. In addition, we conducted additional quantitative evaluation by measuring the state estimation error explained earlier and by evaluating the prediction task with different K, which will be covered in detail in Section 4.3.

Discussion: Although the qualitative results of our traffic pattern understanding in Figure 4.1, 4.5, and 4.6 are not so different from the results of the existing methods (Wang et al., 2009; Hospedales et al., 2009; Kuettel et al., 2010; Emonet et al., 2011; Varadarajan et al., 2012), there are two main distinctions between the proposed model and the existing methods. First, the proposed method incrementally takes trajectory data with online learning, which is differentiated from the batch learning methods. For example, an existing method such as (Hospedales et al., 2009) estimates state as-

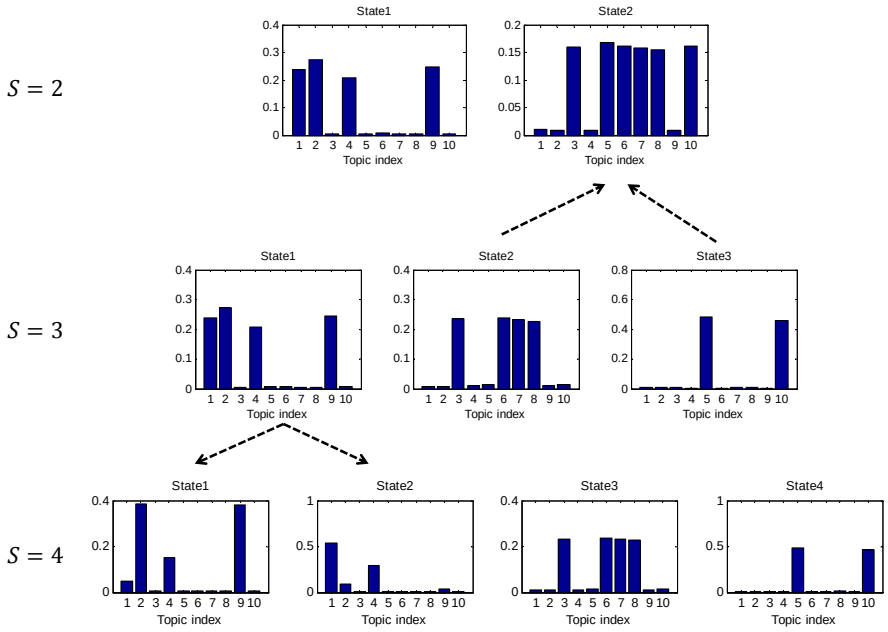


Figure 4.10 The result of parameters $\{m_n | n = 1, .., S\}$ according to variation of S.

signments at once using all data from beginning to end; on the other hand, our method lengthen the state estimation bar as time goes on. Figure. 4.11 shows error rates of state estimation in the WI dataset. In the figure, the state estimation of each trajectory collection is compared to the ground truth, and then error rates for each set of 500 collections are displayed. Because the MCTM (Hospedales et al., 2009) takes 5000 collections of the trajectories at once, the state estimation error rate consistently remains near 5%. The proposed method, on the contrary, receives input data by the 10 collections in online fashion. Therefore, the error rate is over 20% at the beginning due to lacks of data, but soon afterwards, the error rate decreases and becomes similar to the results of MCTM. (*i.e.* $T = 5000$ and $D = 10$ in terms of the notation of this paper.) In addition, the experiments are conducted with different K, and the results of our method shows the stability with respect to the variations of K. Our online learning method not only enables the adaptation of scene changes but also saves memory be-

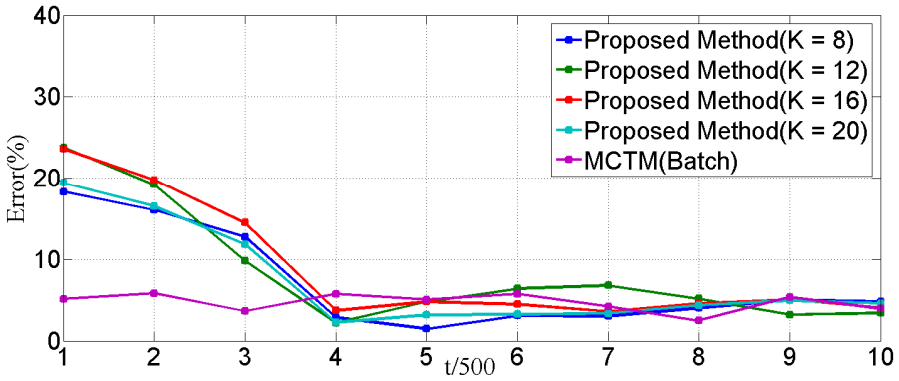


Figure 4.11 Error rate of state estimation in the WI dataset and comparison with the batch learning method.

cause our model does not need to keep old trajectory collections. Second, our model utilizes a precise velocity as an observation beyond quantized direction. As the merit of adding precise velocity to the model is difficult to display on the scene understanding results, subsequent sections will show the effect of using velocity observations.

4.2 Applications in Anomaly Detection

This section provides anomaly detection results using the proposed model. To confirm that the proposed method can satisfy the five essential requirements for the traffic pattern modeling suggested in Section 1.1, we will show the example of anomaly detection results according to each requirement.

Requirement 1: The first requirement is that the entire region in the surveillance scene should be categorized into semantic regions representing typical activities (e.g. go straight upward, turn right, walk across the street, etc.). This makes it possible to detect intrusion of restricted areas, jaywalking, lane violation, and illegal U-turn. Detected abnormal events related to the first requirement are shown in Figure. 4.12-(a-d). Figure. 4.12-(a) illustrates a detection of an illegal U-turn action which is captured from the MIT dataset. In Figure. 4.12-(b), two jaywalking activities (one is going by

Figure 4.12 Examples of anomaly detections related to the first requirement (semantic regions of normal pattern). (a) illegal U-turn; (b) jaywalking; (c) intrusion of restricted areas; (d) driving on the wrong direction. (best viewed in color)

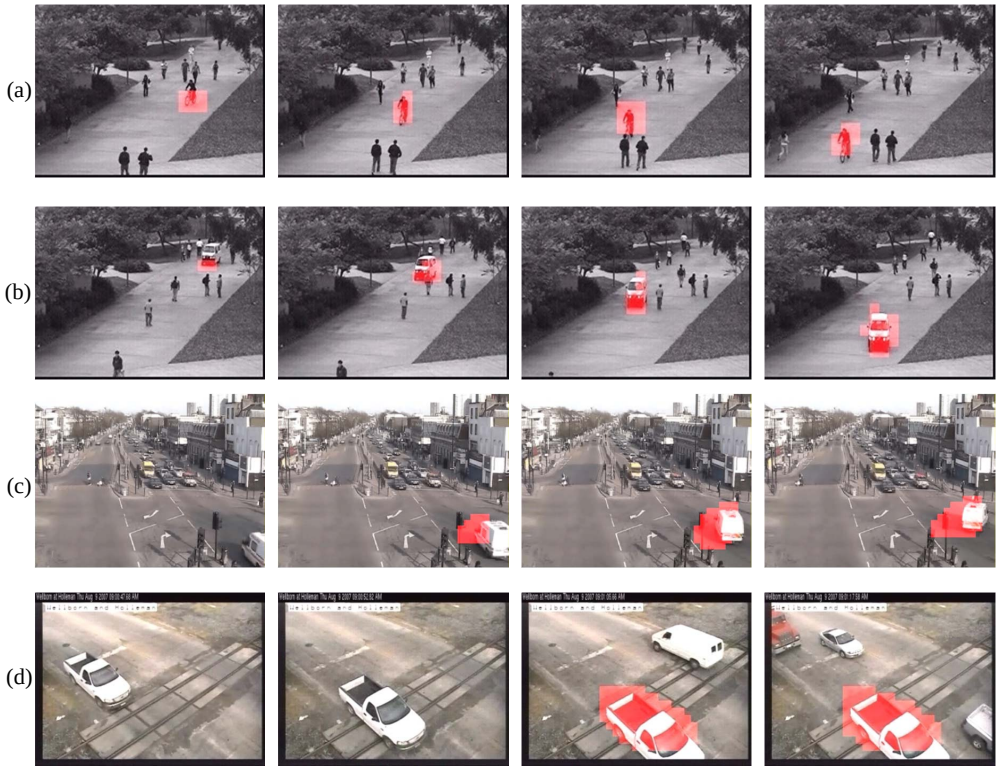


Figure 4.13 Examples of anomaly detections related to the second requirement (Speed information). (a,b) over speed on a pavement; (c) going on the opposite direction; (d) a car stops on a railway. (best viewed in color)

bicycle and the other is on walk) are detected. Figure. 4.12-(c) shows intrusion of restricted areas (the lawn). Also, motorbike driving on the wrong direction is detected in Figure. 4.12-(d) captured from the WI dataset. Our method can detect these events as abnormal because the regions of these abnormal activities are not matched with the regions of typical trajectory patterns represented by the trained parameter ϕ.

Requirement 2: The second requirement is that the model should include not only direction information but also precise speed information for each activity regions. This gives the model the discrimination ability to detect pedestrians walking along the path of vehicles, bikes running in pedestrian road, cars driving with over speed, cars stop-

ping in a railroad crossing, and so on. Abnormal events violating the normal speed patterns from the second requirement are shown in Figure. 4.13-(a-d). Since abnormal activities caused by violating speed rules hardly occur in intersection datasets, we additionally conducted anomaly test for UCSD and level crossing datasets to confirm the performance of our model. These datasets contain abnormal activities that are difficult to detect when using methods based on the conventional topic models with quantized directions (*e.g.* over-speeding objects, cars stopping on a railroad crossing for a long time, and so on). UCSD dataset captures people, cars, and bicycles showing various velocity patterns. The scene is usually crowded with pedestrians, but bikes and cars drive on pavements rarely. Our method shows good performances by the proposed model with the precise velocity observations. Figure. 4.13-(a-b) illustrates detection of a bike and a car driving on pavement. Since these objects have much faster velocity than other normal pedestrians, they are detected as abnormal. On the other hand, because the quantized directions have no information about speed, the methods based on the quantized direction feature cannot detect an object moving with over-speed. In Figure. 4.12-(c), an ambulance uses improper lanes and goes on the opposite direction. The result in Figure. 4.13-(d) captured form level crossing dataset shows detection of a potentially dangerous region, where a car stops on a railway for a long time. Note that other cars stopping before railroad are determined as normal. On the contrary, the conventional topic models have difficulty in understanding long-term motion of a single object because they are based on local motions extracted between two frames.

For further analysis of the strength of the velocity observations, we look into the likelihood of trajectories in the scene of Figure. 4.13-(b) from the UCSD dataset. In this example, we examine six trajectories (two abnormal trajectories and four normal trajectories), and each trajectory is depicted in a color different from others. The first trajectory (blue) and the second trajectory (green) are extracted from a car going from top to bottom, which are faster than usual motions of pedestrians. The third and fourth

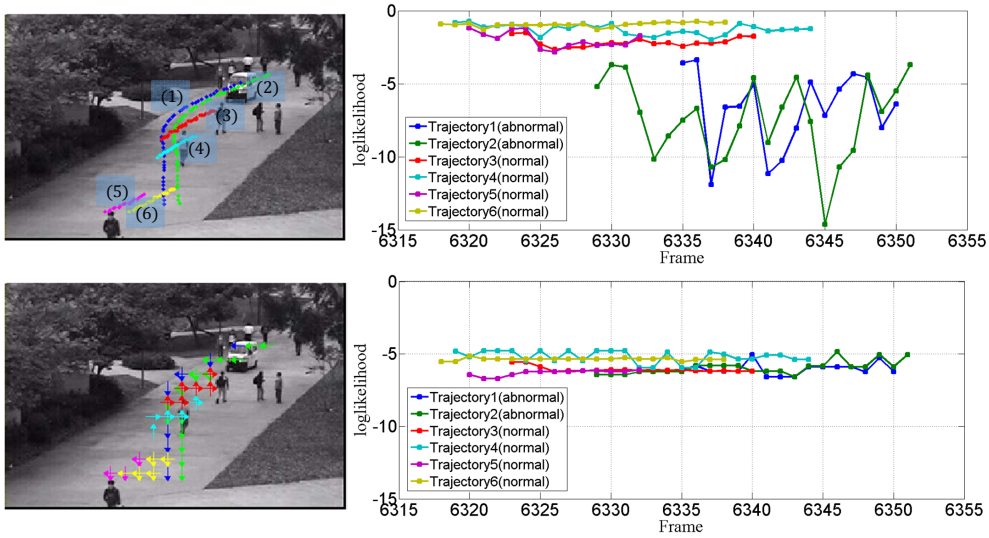


Figure 4.14 Comparison of motion likelihoods between the proposed model (actual velocity of trajectories) and MCTM (quantized direction) (Hospedales et al., 2009). The first row (result of the proposed model): actual trajectories in the UCSD dataset (left) and motion likelihood of each trajectory (right). The second row (result of MCTM): quantized direction converted from each trajectory denoted with different color (left) and their motion likelihood (right). (best viewed in color)

trajectories (red and sky) are extracted from pedestrians walking from bottom to top, and the fifth and sixth trajectories (purple and yellow) are from a pedestrian walking from top to bottom. In case of the proposed method, which utilizes actual velocities of the trajectories and trains them with Gaussian models, the log-likelihood of trajectory 1 and 2 is lower than that of another trajectories as shown in the first row of Figure. 4.14. On the other hand, other topic model based methods such as MCTM (Hospedales et al., 2009) covert the actual motions between two frames into quantized directions at a grid position. Each quantized direction is depicted as one of the four directions (up, down, right, left) at the grid position as shown in left-bottom of Figure. 4.14, where the same colored arrows denote that they are extracted from the same object. This motion representation method, however, cannot distinguish over-speed from walking speed. Therefore, all trajectories have similar likelihoods as shown in the lower graph of Figure. 4.14 because overall paths of the trajectories without velocity information are likely to occur in the scene.

Requirement 3: The third requirment is that spatio-temporal relationship among typical activity patterns should be considered. The spatial relation modeling among trajectory patterns can deal with a potential risk of car crash. The temporal order of activities such as governed by traffic signals can detect a trouble of a traffic control system. Since the abnormal event related to this requirement is very dangerous, it is hard to obtain a sufficient quantity of actual video datasets. Thus, we made video animation which could simulate the trouble of a traffic control system and a car crash event, and we also synthesized and edited the actual videos to have such an abnormal event. The detected abnormal events related to the third requirement using the actual and synthetic video are shown in Figure. 4.15-(a-d). Figure. 4.15-(a) shows a vehicle ignoring the traffic signal and turning right, causing an almost car crash. Even though this vehicle would be considered normal in state 2 (as Figure. 4.5), it is detected as abnormal since the activity occurs when state 3 is dominant. In the animation of Fig-

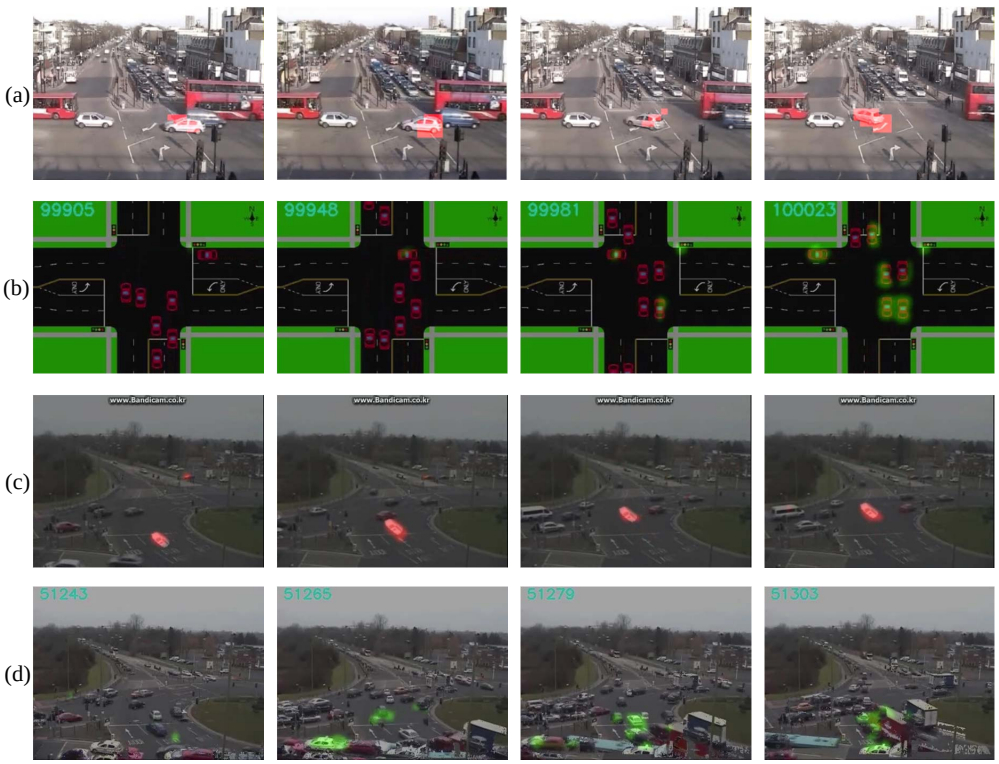


Figure 4.15 Examples of anomaly detections related to the third requirement (spatial interaction of trajectory patterns). (best viewed in color)

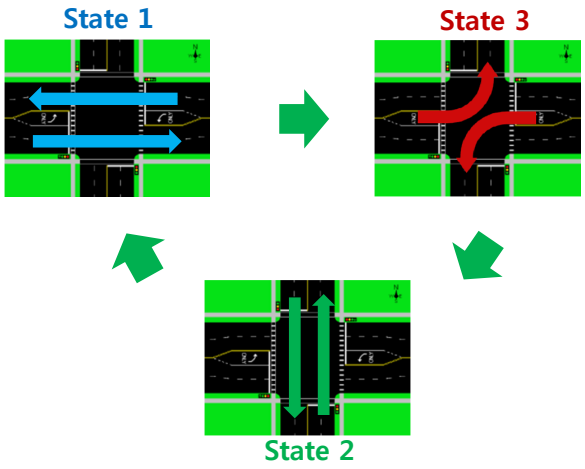


Figure 4.16 Scenario of a traffic animation to simulate a trouble of a traffic control system.. (best viewed in color)

ure 4.15-(b), the car moving from right to left is detected as abnormal because the traffic signal is for the vertical movements. This moving can be normal when the other cars moving upward and downward do not exist under the traffic signal for horizontal movement. Our method can distinguish whether it is normal or not by considering co-occurring trajectory patterns. Likewise, in Figure 4.15-(c), a synthesized car which goes across many cars moving rightward is detected abnormal because the majority of cars are moving upward and downward according to the traffic signal. Figure. 4.15-(d) shows examples of abnormal detections in a synthesized video where various trajectory patterns arising in all traffic signals occurs factitiously at the same time.

In addition, in order to evaluate an abnormal event for a trouble of a traffic control system, we simulate a traffic situation as shown in Figure 4.16. In the simulation, we made a video having three states (vertical movement, horizontal movement, and left turn), and these states moves in a cycle ($1 \rightarrow 3 \rightarrow 2 \rightarrow 1$) shown in the green arrows in the figure. To simulate a trouble of a traffic control system, the cycle is changed to be reverse at the last part of the video (i.e. state 1 changes to state 2 instead of the

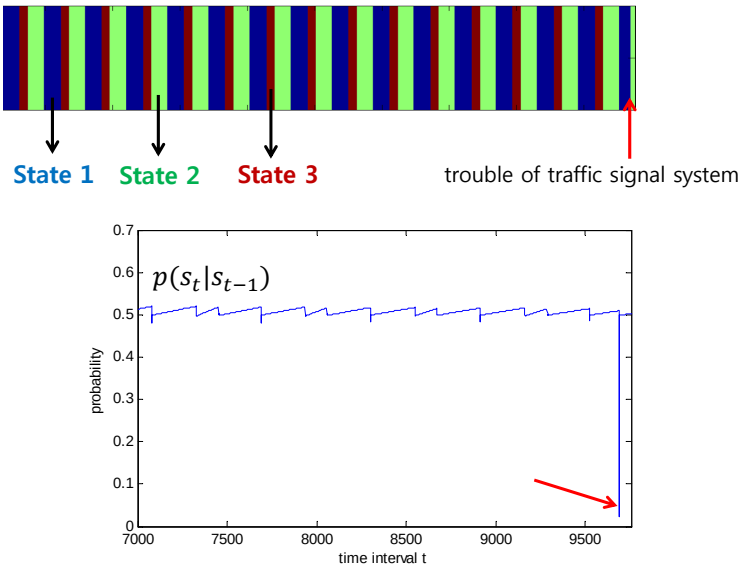


Figure 4.17 State transition probability owing to the trouble of traffic signal. (best viewed in color)

correct state 3). As shown in Figure 4.17, the state transition probability drops rapidly when a trouble occurs in the traffic signal. The estimated state sequence of the video is graphically depicted with the colored bars on the top of the figure (state 1 is blue, state 2 is greed, and state 3 is red). The red arrows indicate the moment in a trouble situation of traffic signal, and the graph in the bottom of the figure shows that the state transition probability decreases dramatically at the trouble moment.

Requirement 4: The fourth requirment is that the proposed method should be

Figure 4.18 Tracking failure case of the object based multi-target tracking method in a crowed scene. (best viewed in color)

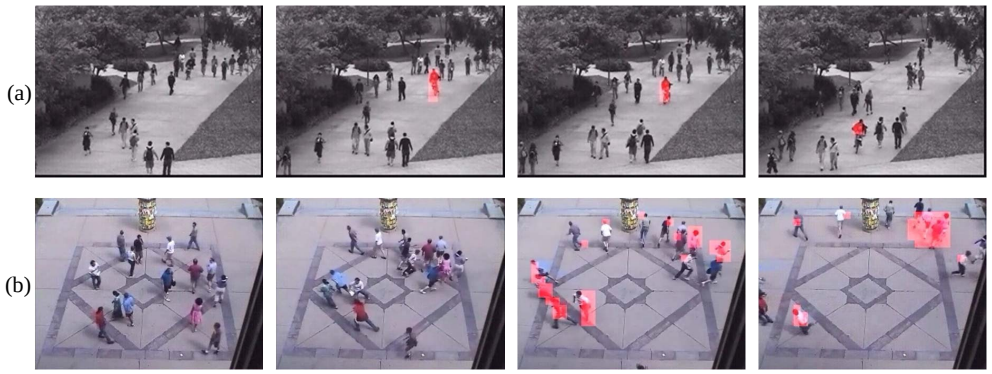
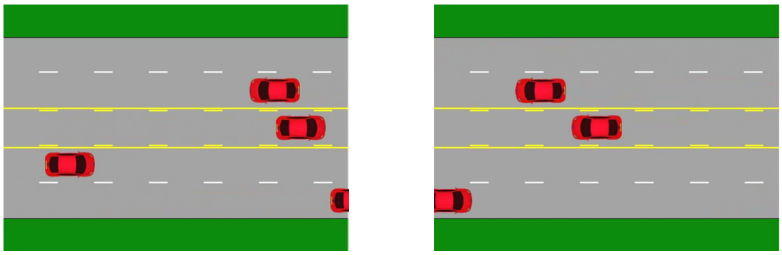


Figure 4.19 Examples of anomaly detections related to the fourth requirement (robust to crowded scenes). (best viewed in color)

robust to crowded scenes. As shown in Figure 4.18, it is hard to extract motions of individual objects in the crowded scenes. In this video, a bicycle is moving faster than the other pedestrian crowds, but the conventional object-based multi-tracking method cannot extract individual motion of the bicycle due to frequent occlusions. However, because our method extracts KLT trajectories based on feature points rather than object-level, motions of moving objects are extracted relatively easier even in the occluded situation. Hence, the proposed method can detect abnormal events related to the fourth requirement as shown in Figure. 4.19-(a-b). Figure. 4.19-(a) is a abnormal detection result of the same case shown in Figure 4.18, which show that our method can detect a fast bicycle in spite of the crowded situation. Figure. 4.19-(b) shows a abnormal detection result for the UMN dataset. In the video, people are loitering slowly in a square, and then suddenly scatter. The proposed method detects the event well.

Requirement 5: The fifth requirement is that the model should be able to adapt itself to temporal changes of the scene (e.g. reversible lane, traffic volume changes). Since the abnormal event related to the fifth requirement is very dangerous and proper dataset for a reversible lane does not exist, we conducted a simulation of the reversible lane using a video animation as shown in Figure 4.20. In the figure, the center lane is



(a) rightward (b) leftward

Figure 4.20 Video animation of a reversible lane.

reversible: cars can move rightward or leftward depending on the time duration. The scenario of this simulation is that cars in the reversible lane go to rightward at the first, and then fifteen minutes later, cars in the reversible lane go to leftward. At the last part of the video, a car goes against the correct direction of the reversible lane (abnormal event). The result of the simulation is shown in Figure 4.21. Figure 4.21-(a) shows an alarm right at the moment when the reversible lane is changed from rightward to leftward. However, after a while, the model adapts the leftward moving pattern, so cars are not detected as abnormal anymore as shown in Figure 4.21-(b). At the last part of the video, a car going against the rule of the reversible lane (moving rightward) is decided to be abnormal as shown in Figure 4.21-(c). For further analysis for an adaptation of the model in this simulation, Figure 4.22 shows a process of trajectory pattern adaptation.

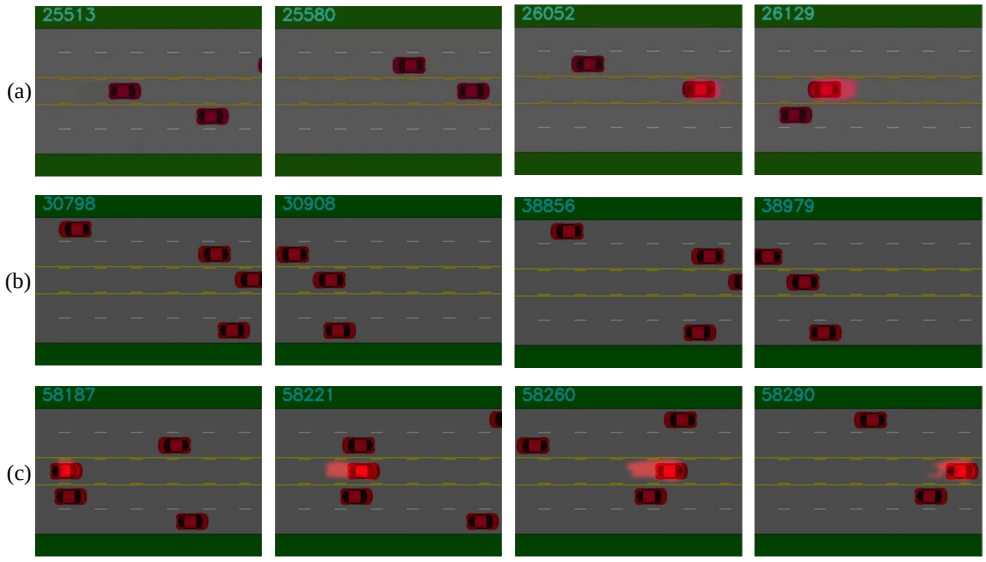


Figure 4.21 Examples of anomaly detections related to the fifth requirement (online adaptation). (best viewed in color)

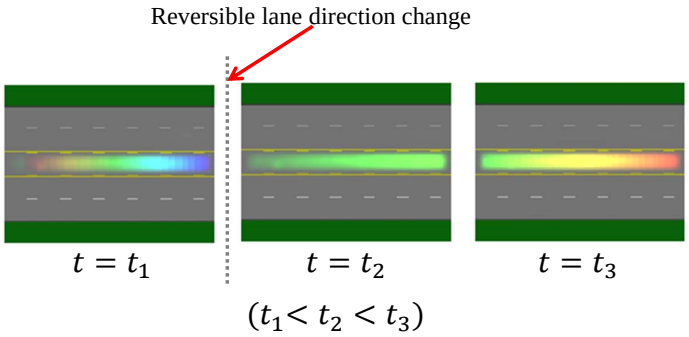


Figure 4.22 Process of trajectory pattern adaptation. The typical moving patterns are denoted with red and blue coloring, where objects move from red to blue. (best viewed in color)

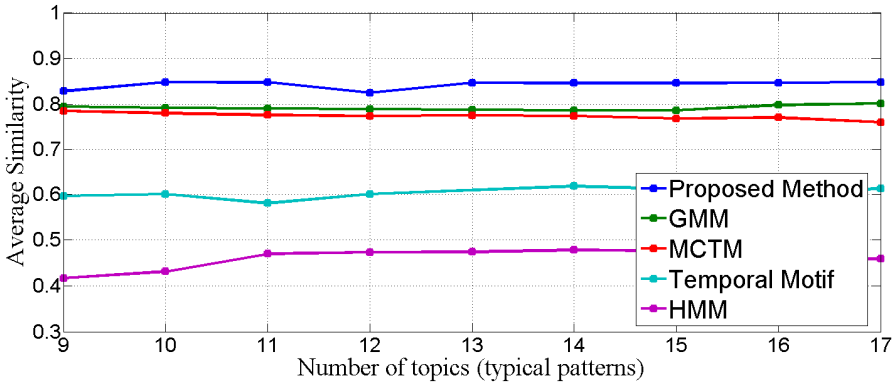


Figure 4.23 Comparison of average accuracy on a prediction. X-axis indicates number of topics denoted as K in our paper. Exceptionally, in case of the GMM based methods, X-axis indicates the number of Gaussian components.

4.3 Prediction Task

The number of abnormal activities in the actual traffic video datasets is not enough to give meaningful quantitative results. This is because the model would prefer overfitting to only a few events, harming the credibility. Therefore, in order to quantitatively compare the performance of our method against other algorithms, we conducted activity prediction tasks presented in (Emonet et al., 2011). The prediction task can test the whole video sequence although abnormal activities are not happened in the video. For this reason, the prediction tasks can be used for a general evaluation of the model's plausibility. For the task, future observations are estimated using given past observations. For example, if the upward motions are observed in the bottom of the scene and the right-turn pattern is learned at the position, future observations (maybe rightward motions in the right-side of the scene) can be estimated based on the trained model. The estimated future observations are represented as a probability histogram whose summation must be 1, and then the similarities to the actual observations are measured using Bhattacharyya coefficient.

MIT dataset was used for the comparison and the existing methods (Emonet et al., 2011; Hospedales et al., 2009) using 29 past time instances (seconds) to estimate the observations of the 30th time instances. Unlike the existing topic models (Emonet et al., 2011; Hospedales et al., 2009), whose observations are represented by quantized local motions between only two frames, the proposed model utilizes trajectories as observations. This type of observation allows our method to do the prediction task with trajectories from the current frame (not observations obtained from 29 past time instances) and the trained model. Also we validated the prediction accuracy on the different design parameter K, representing the number of topics. Comparison results are shown in Figure. 4.23. The figure shows that the proposed method outperforms Temporal Motif (Emonet et al., 2011) and MCTM (Hospedales et al., 2009) even though we conduct the prediction task with observations only in the current frame. This result is caused by the fact that Temporal Motif (Emonet et al., 2011) and MCTM (Hospedales et al., 2009) utilize quantized local motions, but our model mines actual velocity of trajectories. This provides the validity of the use of accurate velocity observations, allowing more plausible scene model and giving precise predictions.

We also provide the result of comparison with GMM-based trajectory modeling (Basharat et al., 2008), whose trajectory representation method is similar to ours (*i.e.* it also uses actual velocity observations). The reason why the proposed method is more accurate than (Basharat et al., 2008) is that we have inter-related multi-Gaussian models based on typical patterns (topics). For example, in the center of intersection, the GMM would estimate a future position of the trajectory based on only the previous path. Thus, in some cases, the GMM model may have difficulty in predicting whether an object will go straight or turn right. On the contrary, the prediction of our method (including other topic model based methods) is based on not only previous path but also mutual dependence among typical activities. Therefore, the proposed method can give a confident prediction whether an object will go straight or turn right.

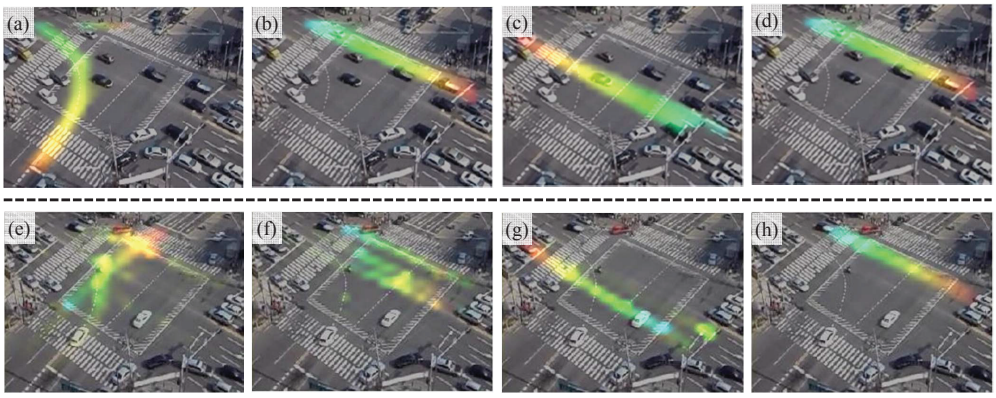


Figure 4.24 Qualitative comparison of proposed method and sampling based learning. The first row: Examples of typical patterns learned by the two-stage learning. The second row: Examples of typical patterns learned by online Gibbs sampling (Canini et al., 2009).

4.4 Comparison with Sampling

Two-stage inference for the proposed model is used to overcome the shortcomings of sampling based inference mentioned in Section 3.2. To conduct comparison with a sampling, we adopt the incremental Gibbs sampler for topic model, which is proposed in (Canini et al., 2009). In this work, incremental update is enabled by occasionally resampling topic variables and rejuvenating old topic assignments by considering new data. Figure. 4.24 shows the qualitative comparison result of the proposed method and the online Gibbs sampling method on the data given incrementally. Activity patterns in the figure are selected from overall typical patterns discovered by each learning method.

Figure. 4.24-(a-d) are the result of the proposed method and Figure. 4.24-(e-f) are results optimized by incremental Gibbs sampler. The activities in Figure. 4.24-(a,e) represent the left turn, going from southwest to northwest. Comparing to the result inferred by the proposed method, the result of incremental Gibbs sampler in Figure. 4.24-(e) is not fully separated from other activities going southwest to north-

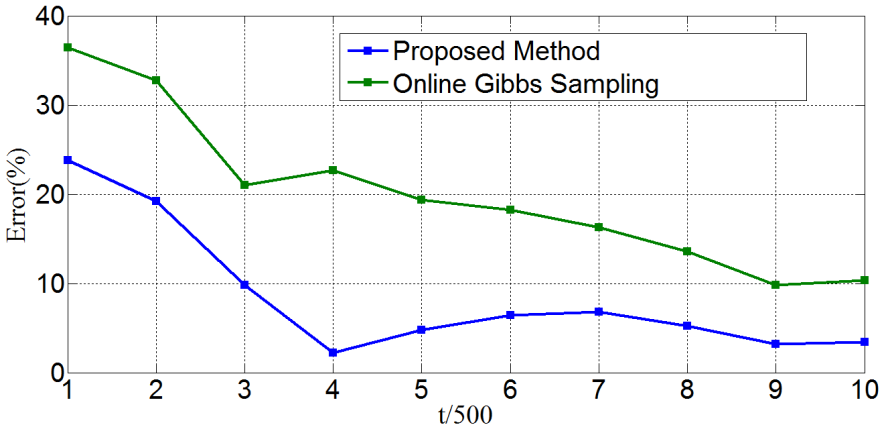


Figure 4.25 Quantitative comparison with online Gibbs Sampling (Canini et al., 2009) on the error rates of the state estimation.

east. Also, another result of Gibbs sampling in Figure. 4.24-(f) does not model the pattern as clearly as in (b). These results show that sampling based method does not guarantee good performance in case of distributed processing for online learning.

For the quantitative comparison, the state estimation is performed by online Gibbs sampling Canini et al. (2009), then it is compared with the ground truth (in the same way of Figure. 4.11). As shown in Figure. 4.25, error rates of each method for each set of 500 collections are displayed. Because the two methods are based on the online inference, the error rate is high at beginning, and then decreases gradually. The error rates of the proposed method are lower than that of online Gibbs sampling over the entire range. Consequently, the proposed method, though it is an approximate inference, gives better performance than the online Gibbs sampling method.

Chapter 5

Conculsion

5.1 Concluding Remarks

This thesis introduced a new method for analyzing a traffic patterns in a scene and detecting anomalies. By investigation on the previous studies we identified the essential requirements for the traffic pattern modeling in actual environments. The proposed method met those requirements by modeling the scene with a graphical inference model which uses the point trajectories of the scene considering the overall path, their spatio-temporal dependency, and their precise velocities. The problem of high dimensionality of the proposed model was relaxed with the proposed two-stage greedy inference, allowing the solutions to be obtained efficiently. This approximate inference strategy is a meaningful attempt to find an alternative outperforming CGS which is conventionally used to learn topic models for scene understanding.

As shown in the experiments, the effects of the proposed approach are summarized as follows. The scene understanding results showed that the proposed method could automatically discover not only typical patterns but also spatio-temporal rela-

tions among them. Also, the state estimation results of the proposed online inference maintained a comparable performance to the batch learning method. In the experiment on the likelihood evolvement of a trajectory over a time, the proposed method was able to distinguish the speed of moving objects, which was impossible with the quantized directions. Using the proposed velocity model with regard to typical patterns, our method also gave outstanding accuracy on the prediction task. On the comparison to the online sampling method, the two-stage online inference guaranteed more robust results than the sampling based learning.

5.2 Future Works

The sub-model optimization strategy presented in Section 3.2 introduced the several independence assumptions for online inference. Although we could not find miss-detection cases caused by the assumptions in our experiments of the six video datasets, the miss-detection cases might occur when a rigorous validation with more various video is performed. As for the future work, we will validate our sub-model optimization strategy and pursue a relaxation of the assumptions.

Another future work can be an issue to expand our model into a non-parametric model. If the parameters K and S in our model are estimated automatically, the performance of adjusting to the changing environment would be enhanced. However, in order to estimate S and K automatically, model selection problem should be included in the proposed inference framework, which is not straightforward. Although simple heuristics can be applied to the model update, it might harm the convergence of the online learning. Due to the characteristics of surveillance systems, a large amount of data is continuously obtained; thus, the long-time stability for 24 hours and 7 day is very important. For this reason, it is essential to prove the stability and convergence of the online learning method that determines K and S automatically. This problem can be a good topic for the future works.

Bibliography

Andrieu, C., de Freitas, N., Doucet, A., & Jordan, M. I. (2003). An introduction to MCMC for machine learning. *Machine Learning*, *50*(1-2), 5–43. URL http://dx.doi.org/10.1023/A:1020281327116

Antonini, G., & Member, S. (2006). Counting pedestrians in video sequences using trajectory clustering. *IEEE Transactions on Circuits and Systems for Video Technology, Vol.16, Issue*, (pp. 1008–1020).

Basharat, A., Gritai, A., & Shah, M. (2008). Learning object motion patterns for anomaly detection and improved object detection. In *IEEE Conference on CVPR*.

Benezeth, Y., Jodoin, P.-M., & Saligrama, V. (2011). Abnormality detection using low-level co-occurring events. *Pattern Recognition Letters*, *32*(3), 423–431.

Benfold, B., & Reid, I. (2011). Stable multi-target tracking in real-time surveillance video. In *International Conference on Computer Vision and Pattern Recognition*, (pp. 3457–3464).

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc.

Blei, D. M. (2012). Probabilistic topic models. *Commun. ACM*, *55*(4), 77–84.

Blei, D. M. (2014). Build, compute, critique, repeat: Data analysis with latent variable models. *Annual Review of Statistics and Its Application*, *1*(1), 203–232.

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *JML. Res.*, *3*, 993–1022.

Boyd, S., & Vandenberghe, L. (2004). *Convex Optimization*. New York, NY, USA: Cambridge University Press.

Buzan, D., Sclaroff, S., & Kollios, G. (2004). Extraction and clustering of motion trajectories in video. In *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, vol. 2, (pp. 521–524 Vol.2).

Canini, K. R., Shi, L., Neuroscience, H. W., & Griffiths, T. L. (2009). Online inference of topics with latent dirichlet allocation. In *In AI-STATS*.

Chang, H. J., Jeong, H., & Choi, J. Y. (2012). Active attentional sampling for speed-up of background subtraction. In *Computer Vision and Pattern Recognition*, (pp. 2088–2095). IEEE.

Cui, X., Huang, J., Zhang, S., & Metaxas, D. N. (2012). Background subtraction using low rank and group sparsity constraints. In *Computer Vision - ECCV 2012 - 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part I*, (pp. 612–625).

Dalal, N., & Triggs, B. (2005). Histograms of oriented gradients for human detection. In C. Schmid, S. Soatto, & C. Tomasi (Eds.) *International Conference on Computer Vision & Pattern Recognition*, vol. 2, (pp. 886–893). INRIA Rhône-Alpes, ZIRST-655, av. de l'Europe, Montbonnot-38334.
URL `http://lear.inrialpes.fr/pubs/2005/DT05`

Diaconis, P., & Ylvisaker, D. (1979). Conjugate priors for exponential families. *The Annals of Statistics*, *7*(2), 269–281.
URL http://dx.doi.org/10.1214/aos/1176344611

Dickey, J. M. (1983). Multiple hypergeometric functions: probabilistic interpretations and statistical uses. *Journal of the American Statistical Association*, *78*(383), 628–637.

Dollar, P., Wojek, C., Schiele, B., & Perona, P. (2012). Pedestrian detection: An evaluation of the state of the art. *IEEE Trans. Pattern Anal. Mach. Intell.*, *34*(4), 743–761.

Duda, R. O., Hart, P. E., & Stork, D. G. (2000). *Pattern Classification (2nd Edition)*. Wiley-Interscience.

Emonet, R., Varadarajan, J., & Odobez, J.-M. (2011). Extracting and locating temporal motifs in video scenes using a hierarchical non parametric bayesian model. In *IEEE Conference on CVPR*, (pp. 3233–3240).

Fu, Z., Hu, W., & Tan, T. (2005). Similarity based vehicle trajectory clustering and anomaly detection. In *International Conference on Image Processing*, (pp. 602–605). IEEE.

Griffiths, T. L., Kemp, C., , & Tenenbaum, J. B. (2008). Bayesian models of cognition. *Cambridge Handbook of Computational Cognitive Modeling*.

Griffiths, T. L., & Steyvers, M. (2004). Finding scientific topics. *PNAS*, *101*(suppl. 1), 5228–5235.

Hoffman, M., Blei, D. M., & Bach, F. (2010). Online learning for latent dirichlet allocation. In *NIPS*.

Hospedales, T. M., Gong, S., & Xiang, T. (2009). A markov clustering topic model for mining behaviour in video. In *ICCV*, (pp. 1165–1172). IEEE.

Hu, W., Xiao, X., Fu, Z., Xie, D., Tan, T., & Maybank, S. (2006). A system for learning statistical motion patterns. *IEEE Trans. Pattern Anal. Mach. Intell.*, *28*(9), 1450–1464.

Hu, W., Xie, D., Fu, Z., Zeng, W., & Maybank, S. (2007). Semantic-based surveillance video retrieval. *Image Processing, IEEE Transactions on*, *16*(4), 1168–1181.

Hu, W., Xie, D., Tan, T., & Maybank, S. (2004). Learning activity patterns using fuzzy self-organizing neural network. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, *34*(3), 1618–1626.

Jordan, M. I., Ghahramani, Z., Jaakkola, T. S., & Saul, L. K. (1999). An introduction to variational methods for graphical models. *Mach. Learn.*, *37*(2), 183–233. URL `http://dx.doi.org/10.1023/A:1007665907178`

Junejo, I. N., Javed, O., & Shah, M. (2004). Multi feature path modeling for video surveillance. In *International Conference on Pattern Recognition (ICPR)*, (pp. 716–719).

Kay, S. (1998). *Fundamentals of Statistical Signal Processing: Estimation theory*. No. V. 1 in Fundamentals of Statistical Signal Processing. Prentice-Hall PTR. URL `http://books.google.co.kr/books?id=aFwESQAACAAJ`

Keogh, E. J., & Pazzani, M. J. (2000). Scaling up dynamic time warping for datamining applications. In *In Proc. 6th Int. Conf. on Knowledge Discovery and Data Mining*, (pp. 285–289).

Kratz, L., & Nishino, K. (2009). Anomaly detection in extremely crowded scenes using spatio-temporal motion pattern models. *2013 IEEE Conference on Computer Vision and Pattern Recognition*, *0*, 1446–1453.

Kuettel, D., Breitenstein, M. D., Van Gool, L., & Ferrari, V. (2010). What's going on? discovering spatio-temporal dependencies in dynamic scenes. In *CVPR*, (pp. 1951–1958).

Kullback, S., & Leibler, R. A. (1951). On information and sufficiency. *The Annals of Mathematical Statistics*, *22*(1), 79–86.
URL `http://dx.doi.org/10.1214/aoms/1177729694`

Kuo, C.-H., Huang, C., & Nevatia, R. (2010). Multi-target tracking by on-line learned discriminative appearance models. In *International Conference on Computer Vision and Pattern Recognition*, (pp. 685–692). IEEE.

Li, X., Hu, W., & Hu, W. (2006). A coarse-to-fine strategy for vehicle motion trajectory clustering. *Pattern Recognition, International Conference on*, *1*, 591–594.

Liao, H.-Y. M., Chen, D.-Y., Su, C.-W., & Tyan, H.-R. (2006). Real-time event detection and its application to surveillance systems. In *International Symposium on Circuits and Systems*. IEEE.

Machy, C., Desurmont, X., Delaigle, J.-F., & Bastide, A. (2007). Introduction of cctv at level crossings with automatic detection of potentially dangerous situations.

MacQueen, J. B. (1967). Some methods for classification and analysis of multivariate observations. In L. M. L. Cam, & J. Neyman (Eds.) *Proc. of the fifth Berkeley Symposium on Mathematical Statistics and Probability*, vol. 1, (pp. 281–297). University of California Press.

Mahadevan, V., Li, W., Bhalodia, V., & Vasconcelos, N. (2010). Anomaly detection in crowded scenes. In *IEEE Conference on CVPR*, (pp. 1975 –1981).

Makris, D., & Ellis, T. (2002). Path detection in video surveillance. *Image and Vision Computing*, *20*, 895–903.

Morris, B., & Trivedi, M. (2008). A survey of vision-based trajectory learning and analysis for surveillance. *Circuits and Systems for Video Technology, IEEE Transactions on*, *18*(8), 1114 –1127.

Morris, B., & Trivedi, M. M. (2009). Learning trajectory patterns by clustering: Experimental studies and comparative evaluation. In *CVPR*, (pp. 312–319).

Neal, R. M. (2000). Markov chain sampling methods for dirichlet process mixture models. *JOURNAL OF COMPUTATIONAL AND GRAPHICAL STATISTICS*, *9*(2), 249–265.

Ng, A. Y., Jordan, M. I., & Weiss, Y. (2001). On spectral clustering: Analysis and an algorithm. In *ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS*, (pp. 849–856). MIT Press.

Parisi, G. (1988). *Statistical field theory*. Frontiers in physics. Addison-Wesley Pub. Co.
URL `http://books.google.co.kr/books?id=2Wm5AAAAIAAJ`

Piciarelli, C., & Foresti, G. L. (2006). Online trajectory clustering for anomalous events detection. *Pattern Recognition Letters*, (pp. 1835–1842).

Porikli, F., & Porikli, F. (2004). Trajectory distance metric using hidden markov model based representation. In *In Proc. ECCV PETS Workshop*.

Qin, Z., & Shelton, C. R. (2012). Improving multi-target tracking via social grouping. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Rodriguez, M., Ali, S., & Kanade, T. (2009). Tracking in unstructured crowded scenes. In *International Conference on Computer Vision (ICCV)*, (pp. 1389–1396). IEEE.

Saleemi, I., Hartung, L., & Shah, M. (2010). Scene understanding by statistical modeling of motion patterns. In *CVPR*, (pp. 2069–2076). IEEE.

Saleemi, I., Shafique, K., & Shah, M. (2009). Probabilistic modeling of scene dynamics for applications in visual surveillance. *IEEE Trans. on PAMI*, *31*(8), 1472–1485.

Shi, J., & Malik, J. (1997). Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *22*, 888–905.

Stauffer, C., & Grimson, W. E. L. (1999). Adaptive background mixture models for real-time tracking. In *CVPR*, (pp. 2246–2252).

Sumpter, N., & Bulpitt, A. J. (1998). Learning spatio-temporal patterns for predicting object behaviour. *Image Vision and Computing*, *18*, 2000.

Tomasi, C., & Kanade, T. (1991). Detection and tracking of point features. Tech. rep., IJCV.

UCSD (2010). Anomaly dataset. `http://www.svcl.ucsd.edu/projects/anomaly/dataset.html`.

UMN (2009). Crowd dataset. `http://www.cs.ucf.edu/ramin/`.

Varadarajan, J., Emonet, R., & Odobez, J. (2012). Bridging the past, present and future: Modeling scene activities from event relationships and global rules. In *IEEE Conference on CVPR*, (pp. 2096 –2103).

Vlachos, M., Kollios, G., & Gunopulos, D. (2002). Discovering similar multidimensional trajectories. In *In Proc. IEEE Conf. Data Engineering*, (pp. 673–684).

Vogel, C. R. (2002). *Computational Methods for Inverse Problems*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics.

Walk, S., Majer, N., Schindler, K., & Schiele, B. (2010). New features and insights for pedestrian detection. In *Conference on CVPR*. IEEE, San Francisco: IEEE.

Wang, B., Ye, M., Li, X., Zhao, F., & Ding, J. (2012). Abnormal crowd behavior detection using high-frequency and spatio-temporal features. *Machine Vision and Applications*, *23*(3), 501–511.

Wang, X., Ma, K. T., Ng, G.-W., & Grimson, W. E. (2011). Trajectory analysis and semantic region modeling using nonparametric hierarchical bayesian models. *Int. J. Comput. Vision*, *95*(3), 287–312.

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

Wang, X., Tieu, K., & Grimson, E. (2006). Learning semantic scene models by trajectory analysis. In *Proceedings of the 9th ECCV - Volume Part III*, ECCV'06, (pp. 110–123). Berlin, Heidelberg: Springer-Verlag.

Winn, J. M. (2004). *Variational Message Passing and its Applications*. Ph.D. thesis, University of Cambridge.

Yang, B., & Nevatia, R. (2014). Multi-target tracking by online learning a CRF model of appearance and motion patterns. *International Journal of Computer Vision*, *107*(2), 203–217.

Zhai, K., Boyd-Graber, J., Asadi, N., & Alkhouja, M. (2012). Mr. LDA: A flexible large scale topic modeling package using variational inference in mapreduce. In *ACM International Conference on World Wide Web*.

Zhao, B., Fei-Fei, L., & Xing, E. P. (2011). Online detection of unusual events in videos via dynamic sparse coding. In *IEEE Conference on CVPR*. Colorado Springs, CO.

초록

본 논문은 CCTV의 영상감시 상황에서 교통 패턴 분석과 비정상 탐지를 하는데 있어서 기존 방법들이 갖는 한계점을 극복하기 위한 새로운 온라인 추론모델을 제안하였다. 교통 패턴 분석을 위한 추론 모델은 영상 감시 상황에서 움직이는 물체에 의한 교통의 흐름을 사용자의 의한 사전정보 없이 자동으로 분석해야 한다. 추론 모델을 제안하기에 앞서, 움직임 패턴 분석에 관한 기존 연구들을 조사하고 다양한 종류의 영상 감시 상황을 분석함으로써 교통 패턴을 분석하는 알고리즘이 가져야 할 5가지 필수 요건을 제안하였다. 첫 번째 조건은 각 움직임 패턴의 영역탐지, 두 번째 조건은 영역내의 미세한 속도 모델, 세 번째 조건은 궤적 패턴간의 시공간적 관계 모델링, 네 번째 조건은 혼잡상황에서의 강인성, 마지막 조건은 알고리즘의 온라인 학습 및 실시간 처리이다.

이러한 다섯 가지 요구조건을 충족시키기 위하여 자연어 처리에 활용되는 토픽모델을 변형해 교통 흐름 분석에 적합하도록 새로운 모델을 제안하였다. 기존의 토픽 모델은 미세한 속도 패턴을 분석하지 못한다는 단점을 개선하기 위해 가우시안 모델을 함께 결합하여 궤적패턴이 특정 위치에서 어떠한 속도 분포를 가지는지를 모델링 하였다. 또한 교통신호에 따른 차들의 움직임의 거시적인 변화를 모델링 하기 위해 히든 마르코프 모델 (HMM)을 계층적으로 추론모델의 최상단에 결합하여 교통 신호가 바뀜에 따라 궤적 패턴의 혼합이 어떻게 편하는지를 전이확률 형태로 모델링하였다. 한편 이러한 복잡한 모델을 온라인 및 실시간으로 학습하고 테스트 하기 위해 기존 연구에서 널리 사용되지만 온라인 학습을 할 경우 성능이 많이 저하되는 깁스 샘플링 (Gibbs sampling) 방법을 배제하고, 온라인 학습을 할 경우에도 비교적 강인한 variational inference 방법을 활용해 단계별로 근사 추론을 하는 이단 탐욕 추론 (two-stage greedy inference) 방법을 제안함으로써 모델 학습을 위한 검색 공간을 줄임으로써 모델을 학습

하기 위한 연산량을 절약했다. 그리고 근사 추론을 위한 각 단계에서는 온라인 학습과 오프라인 학습의 성능차이가 없는 방법을 활용해 근사로 인한 정확도 손실을 최소화 하고자 했다. 본 논문에서는 제안한 알고리즘의 성능을 평가하기 위해서 다양한 동영상에서 실험을 진행하였으며, 교통 패턴 분석, 비정상 행동 탐지의 성능이 처음에 제안했던 교통패턴분석 시스템이 가져야할 5가지 필수 조건을 만족시킬 뿐만 아니라, 기존 방법에 비해 우수한 성능을 보임을 정성적, 정량적으로 분석함으로써 제안한 모델의 유효성 및 타당성을 입증하였다.

주요어: 영상감시, 궤적 모델, 비정상행동탐지, 토픽 모델
학번: 2011-30975