Detecting Road Traffic Events by Coupling Multiple Timeseries with Nonparametric Bayesian Method

Shiming Yang, Konstantions Kalpakis, *Senior member,* Alain Biem

***Abstract*—Road traffic sensors provide us with rich multi-variable datastreams about the current traffic conditions. Occasionally, there are unusual traffic events (such as accidents, jams, severe weather, etc) that disrupt the expected road traffic conditions. Detecting the occurrence of such events in an online and real-time manner is useful to drivers in planning their routes and in the management of the transportation infrastructure.**

We propose a new method for detecting traffic events that impact road traffic conditions by extending the Bayesian Robust Principal Component Analysis (RPCA) approach. Our method couples multiple traffic datastreams so that they share a certain sparse structure. This sparse structure is used to localize traffic events in space and time. The traffic datastreams are measurements of different physical quantities (e.g. traffic flow, road occupancy) by different nearby sensors. Our proposed method process datastreams in an incremental way with little computational cost, and hence it is suitable to detect events in an online and real-time manner.

We experimentally analyze the detection performance of the proposed coupled Bayesian RPCA using real data from loop detectors on the Minnesota I-494. We find that our method significantly improves the detection accuracy when compared with the traditional PCA and non-coupled Bayesian RPCA.

I. Introduction

A LARGE number of traffic sensors are continuously deployed to collect data for traffic conditions. Federal and State transportation agencies carry out various programs to collect traffic data by means of inductive loop detectors, video surveillance systems, and microwave radar sensors. Collected traffic data include traffic volume, velocity, density, and vehicle classification. These data serve different purposes of study, such as alerting drivers about congestion and accidents, planning new road pavements to accommodate predicted traffic loads, and so on.

There are two major ways to utilize the automatically and continuously collected sensor data. First, we can detect traffic events in their early stage, and send early warnings to drivers for decision making. Second, we can use data assimilation methods to fuse real observations into forecasting models and produce more accurate near-future traffic condition prediction, which also helps drivers avoid traffic jams and plan better travel routes. In this study, we focus on the first way of using traffic observations. On busy highways, accidents may cause severe and quick accumulation of vehicles. By the time police receives report and issues a warning to other drivers, it may have past several minutes or even longer. In this situation, many drivers may miss the chance to take an exit before entering the traffic jam trap. However, if we utilize continuous, automated road sensor observations, we can promptly detect the incidents and traffic jams in their early stage of formation. Knowing the development of incidents can help choose appropriate prediction models for issuing early warnings, and drivers will have a chance to adapt their routes.

Traffic observations demonstrate strong spatial and temporal patterns, showing periodicity and strong correlation between adjacent upstream and downstream observations. These patterns may vary depending on time in a day, day of the week, seasons, or locations. Figure 1 shows annual average of traffic flow on each weekday with 15-minute time resolution. The traffic flow data were collected from sensors along the southbound I-494 in Minnesota by the Minnesota Department of Transportation. The figure shows different flow patterns in each weekday, e.g. Saturdays and Sundays have smaller flow amount than weekdays. At different hours within a day, traffic flow also shows different patterns, while some of these patterns persist at the same time but different days. Besides, we can observe that neighboring sensors present similar patterns. Occasional incidents or events of long duration show as abnormal events on the background of normal traffic behaviors.

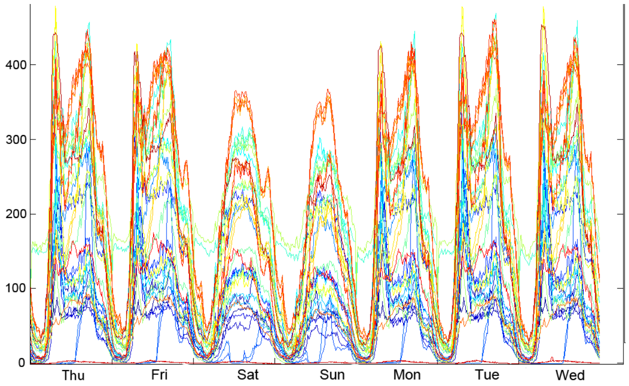


Fig. 1. Patterns for annual average of traffic flows (veh/15min) on each weekday for 38 loop detector sensors on I-494 southbound and eastbound.

To find "abnormal" events, one simple yet feasible way is to define a normal range of traffic measurements based on experience, and use thresholds to identify such special events. However, threshold based methods are not reliable nor adap-

S. Yang is with Department of Computer Science and Electrical Engineering, University of Maryland, Baltimore County, 1000 Hilltop Circle, USA shiming1@umbc.edu.

K. Kalpakis is with Faculty of Computer Science and Electrical Engineering, University of Maryland, Baltimore County, 1000 Hilltop Circle, USA kalpakis@csee.umbc.edu

A. Biem is with IBM T.J. Watson Research Center, NY, USA. biem@us.ibm.com

An earlier version of this work has appeared at the 16th International IEEE Conference on Intelligent Transportation Systems, Hague, Netherlands, Oct.6-9,2013.

tive to changing environment. One approach, namely robust PCA, which is widely used in detecting moving objects in video frames shows promising connections to the traffic event detection. In robust PCA, the normal background data dwell in a low dimensional space, due to the strong correlation among normal observations, and special events dwell in a sparse subspace, which allows an observation set to be decomposed into two subspaces, a normal background subspace specified by a low rank matrix, and an anomaly subspace specified by a sparse matrix [5], [6].

A traffic time series can be viewed as images, if we collect the daily observations as one column of pixels in that 'image' matrix. Special events can be viewed as anomalies in the image. Thus, a 1-dimensional datastream of traffic observations can be converted into a matrix format.

In this paper, we convert 1-dimensional time series into matrix format, and then decompose that matrix into superposition of a low rank, a sparse, and a noise matrix. We use a non-parametric Bayesian method, the Bayesian RPCA to learn the latent low rank and sparse matrices from the observation. Furthermore, we extend the Bayesian RPCA to multiple variables/timeseries/datastreams, which may correspond to different observations that are aligned in time and space or aligned in time but collected by nearby sensors, by coupling them to share sparsity pattern/structure. Our hypothesis is that by sharing a sparsity structure among multiple datastreams affected by the same events, we may be able to improve the detection accuracy of such events from those observation datastreams. We call this approach the coupled Bayesian RPCA. Using real traffic data, we experimentally show that our proposed approach achieves higher accuracy in detecting different types of traffic events.

The remainder of this paper is organized as follows. In Section II, we briefly review related work on traffic event detection, the robust PCA and its typical applications. In Section III, we describe the characteristics of traffic data streams, and show the spatial correlations between sensors. We also demonstrate different types of events of interest. In Section IV, we describe our proposed coupled Bayesian Robust PCA method. The results of the experimental evaluation of our method are given in Section V. Conclusions are in Section VI. In Appendix, we give detailed derivation of Gibbs sampler for Bayesian robust PCA.

II. Related Work

Detecting interesting events is a pervasive problem in many applications, such as video security surveillance, text mining, etc. In road traffic event detection, a large volume of work is based on surveillance video. In some methods, pixel-level features are extracted to represent interesting spatial and/or temporal events. Jiang et el. [15] proposed a dynamic hierarchical clustering method to detect abnormalities in traffic video. Morris and Trivedi [16] compute traffic flow parameters from live video streams, and use speed profiles to categorize traffic motion states. Since image or video based detection is vulnerable to severe weather conditions, many detection algorithms also have been studied based on loop detectors. Guralnik et al. [12] segmented traffic flow time series into piecewise homogeneous regions for a change-point detection. Some methods applied neural networks and fuzzy logic [22]. Methods that compare occupancy or speed to a preset threshold are also seen in the event detection studies, like the California algorithm [19] and the pattern recognition algorithm [9].

Ihler et al. [14] recently use Markov Modulated Poisson Processes (MMPP) to discover events from timeseries of count data. In a probabilistic model framework, they separate the observed timeseries as a superposition of regular and aperiodic processes. They use a non-homogeneous Poisson process to model regular count data. For rare events, they use a Markov chain to model the transition between states including increased, decreased, or unchanged activities. The MMPP method was applied to both highway traffic and building pedestrian counts, and achieved significantly higher accuracy compared to a baseline threshold model.

Principal components analysis (PCA) is well known technique used for dimensionality reduction. PCA also has successful applications in finding outliers. It can be viewed in probabilistic Bayesian way by using a latent variable [4], [21]. With Bayesian PCA, the latent variable can be inferred [4]. Conventional PCA's major drawback is that it is not robust to outliers. Recently, the robust PCA [5], [7], [8] (a special case of matrix completion) has attracted much attention in decomposing matrices into low rank and sparse components matrices. This method finds wide applications in many engineering and statistical modeling problems, where order, dimensionality, or complexity of a model can be evaluated by the rank of an appropriate matrix. The low rank component can be viewed as a "denoised" version of the data. The sparse component is useful in detecting the outliers. For example, in detecting moving objects in a sequence of movie frames with a varying background, one can use the sparse components to locate the moving objects. Becker et. al [3] demonstrated an image processing system that recover images from noisy observations using matrix completion. Candès et. al. [6], [7] provided examples of matrix completion in recovering signals from the magnitude of their Fourier transform, as well a collaborative filtering in online recommendation systems.

Ding et al. [11] proposed a hierarchical Bayesian model to decompose sequential video image matrix into a low rank and a sparse component. They considered that an object will have larger probability to show in the next frame, if it shows up in the previous frame. Hence, they introduce Markov dependence into the Bayesian RPCA. This is a good example of utilizing known structure of sparsity to guide the learning scheme. In our study, we experimentally find that the Bayesian RPCA has good performance for detecting traffic events. Then, we explore how to couple multiple traffic time series by taking advantage of any shared sparsity structure. Though Ding et al. [11] gave the algorithm of Bayesian RPCA, it is not detailed how to derive the distribution of each parameter conditioned on the current values of the other variables. In the Appendix, we provide detail derivation steps of Gibbs sampler for Bayesian RPCA, and clearly point out the factors that encode the Markov dependency among the sparse events.

This work provides a partial answer to Ihler's question [14] *"how those covaring data streams should be combined, and to what degree their parameters can be shared"*.

III. Traffic Data and Events

A. Traffic data

A variety of sensors are used to measure traffic conditions, such as inductive loop detectors, video surveillance cameras, microwave radars and probe vehicles. Nowadays, using GPS-enabled smartphones to collect traffic data is attracting an increasing interest and attention. Typically measured quantities include traffic density, vehicle speed, traffic flow, etc. The traffic flow measures the number of vehicles that pass a single point in a given time interval. Road occupancy is the percentage of time that a detector is active due to the presence of vehicles during a time period. It can be used as a proxy for road density [18]. Given a distribution of vehicle lengths, the average vehicle speed can be estimated from traffic flow rate and road occupancy.

Traffic data have some important characteristics. First, daily traffic data have strong temporal correlation, especially when grouped by weekday.

Second, they contain sparse interesting features. Compared to daily repeated traffic patterns, traffic events are rare and hence manifested as random sparse features in the data that persist for some time. As result, we get structured sparsity with random occurrences in the traffic data streams.

Third, traffic data also present strong spatial correlation. Downstream traffic measurements are influenced by nearby upstream traffic measurements, and vice versa. For example, Figure 2 illustrates strong linear correlation (≥ 0.76) between upstream and downstream sensor measurements (except for sensors at the flyovers) for the Minnesota real dataset described in Section V. Fourth, using different types of traffic measurements allows to better capture the characteristics and effects of traffic events. Hence, we seek to take advantage of the shared sparsity structure between temporarily and spatially correlated variables/measurements of different types.

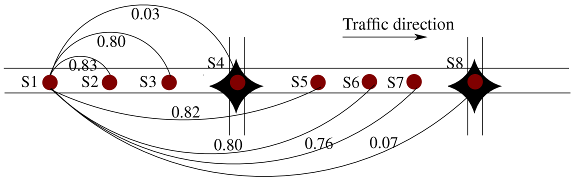


Fig. 2. Linear correlation between upstream and downstream sensor flow rate measurements in a section of I-494.

B. Traffic events

Information about current traffic events is critical to drivers when deciding their traveling routes and schedules, as well as to the management of the transportation infrastructure. Government transportation agencies publish timely traffic incident information, such as the type (crash, roadwork, stall, and hazard), location, start time, and duration of incidents.

Typically, an incident may cause the downstream traffic flow and road density to reduce sharply, while the upstream traffic may temporarily reduce flow and increase road density. However, sensor measurements indicate exceptions are possible. For example, late night roadwork that is announced way ahead of time may have limited impact on traffic measurements, and hence it will not be detectable from the sensor measurements alone.

Traffic events as announced have low spatial and temporal resolution: we often only get the closest intersection where an event happened, while there is latency between the occurrence and reported time of events. Therefore, it is desirable to increase the accuracy of the location and happening time of events from traffic data streams. Moreover, it is also desirable to detect events before they are reported to the relevant transportation agencies, so that it reduces the latency and improves asset utilization when managing incidents.

Furthermore, due to limited transportation management assets, many events go unreported, despite the fact that have noticeable impact on traffic and would be useful to drivers. For example, traffic jams caused by non-accidents can produce low flow rate and high road density; social activities, such as football games and music concerts, can produce unexpected high traffic volumes; reduced visibility due to bad weather (heavy snow, fog, or rain) leads into reduced road capacity, increased road density, and reduced vehicle flow rates. In the experiments presented below, we will study how the algorithm performs on finding events like those as well as reported incidents.

Figure 3 illustrates four events of different types, which are indicated by green shadow areas. Event 1 is a roadwork event lasting 47 minutes and blocking a road lane [1]. We observe reduced flow rate and road occupancy during that time, which normally has higher vehicle flux.

Event 2 corresponds to reduced traffic speed due to limited visibility [2]. Traffic remained free flow, but had decreased flow rate and increased road occupancy. This suggests that vehicles proceeded with caution at lower speeds while the road density is still low.

Event 3 demonstrates an unusual traffic increase at a Saturday night. Such a peak is generally not observed during weekdays but only on some weekends. We hypothesize that it was caused by social activities that usually happen in weekends.

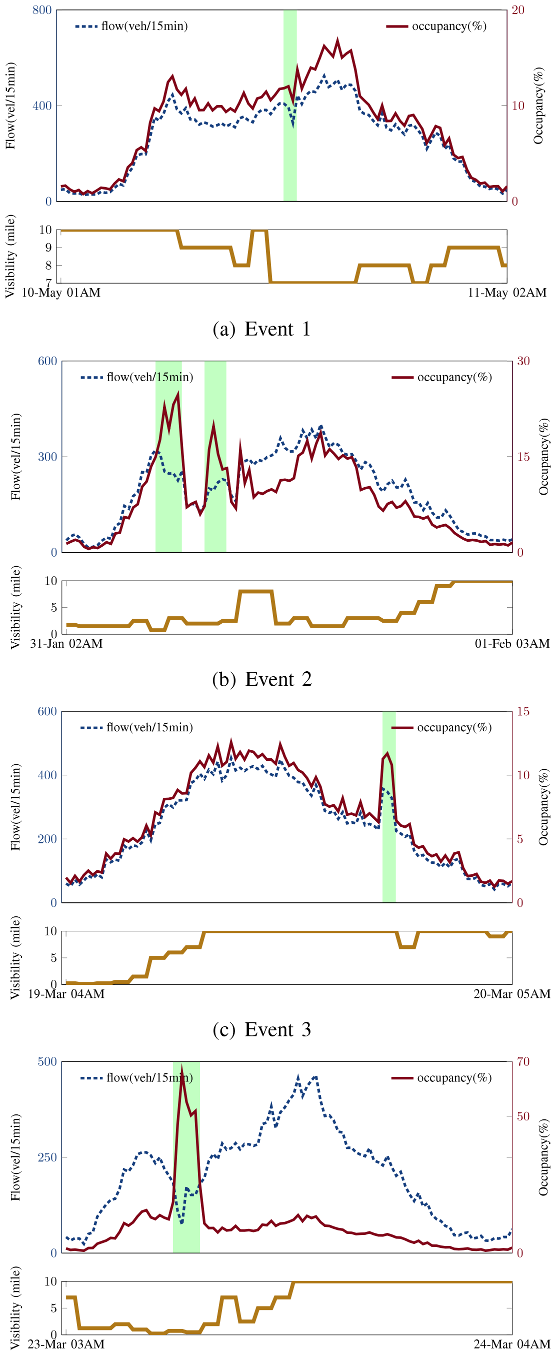
Event 4 presents a traffic jam in early morning on a weekday. There was no incident reported from the transportation agency. However, there was heavy snow and low visibility during that time period, and taking into account that it was during morning rush hour on a Wednesday, we infer the severe weather caused that traffic jam.

IV. Coupled Bayesian RPCA

In this section, we introduce the Bayesian RPCA, and describe our extension of this method to multiple coupled time

[1] Data source: MNDOT, 511MN.org

[2] Date source:weathersource.com



(a) Event 1

(b) Event 2

(c) Event 3

(d) Event 4

Fig. 3. Examples of different event types and their impact on sensor readings.

series. In our extension, the coupling of multiple timeseries generated by dynamical systems is achieved through sharing the same sparsity structure in the Bayesian RPCA. Bayesian RPCA was formally presented in [11] and used in a shopping mall human video surveillance system.

Given a measurement matrix Y, we would like to separate it into a superposition of three components $Y = L + S + E$, where L is a low rank matrix for normal background, S is a sparse matrix for rare events, and E is a noise matrix with small magnitude elements for noise. The Bayesian RPCA has strong connections to factor analysis, and has deep roots in linear Gaussian systems [4], [17].

The low rank component is expressed through SVD as $L = D\Lambda W^T$, where $\Lambda \in \mathbb{R}^{r\times r}$ is a diagonal matrix with main diagonal entries being the singular values of L, $D \in \mathbb{R}^{n\times r}$ and $W \in \mathbb{R}^{m\times r}$ are constructed from the left and right singular vectors of L. Since the rank r is unknown, non-parametric Bayesian approach treats it as a latent variable, and learns it from the data. In [11], the singular values matrix is further decomposed into $Z\Lambda$, such that $L = D(Z\Lambda)W^T$, where D, W are drawn from multivariate Gaussian distributions [20], and $Z, \Lambda \in \mathbb{R}^{k\times k}$ are drawn from multivariate Beta-Bernoulli and Gaussian-Gamma random processes. This method is similar to dictionary learning, where k is the dictionary size and D is the dictionary [11]. A Bernoulli distribution with hyperparameters having a Beta prior distribution is used to learn the binary values of the diagonal elements of Z, while a multivariate normal distribution with hyperparameters having a Gamma prior distribution is used to learn the real values of the diagonal elements in Λ. This Bayesian non-parametric approach provides a flexible way to estimate the rank of the low-rank component matrix without a preset parameter.

In learning $L = D(Z\Lambda)W^T$, requiring that the columns of D and/or W be orthonormal, complicates the computation of the complete marginal posterior distributions for D and W that are needed to construct a Gibbs sampling based Bayesian learning algorithm. Hoff [13] describes a Gibbs sampling algorithm to uniformly sample orthonormal matrices that arise from the Bingham–von Mises–Fisher posterior distribution of the left and right eigenvectors in the SVD factorization of a data matrix. In this paper, we follow an approach similar to those in [2], [11], [20], and we adopt a variational mean field approximation to the posterior distribution of D and W by assuming that each posterior distribution factors into a product of multivariate Gaussian distributions, with one Gaussian for each column of D and W. Hence, we do not orthonormalize D and W.

The sparse component S is decomposed as $S = B \circ X$, where $\circ$ denotes the element-wise multiplication of two matrices, $s_{ij} = b_{ij}x_{ij}$, where B is a binary matrix and X is a real matrix. The binary elements of B are learned from a Bernoulli process with hyperparameters having a Beta prior distribution. The columns of X are learned from a multivariate Gaussian distribution with hyperparameters having a Gamma prior distribution. In [11], Markov spatio-temporal dependencies in the sparse component are considered, since a moving object in one video frame is highly likely to show up in the next frame at a nearby location. Therefore, a Markov dependency along the columns (corresponding to video frames) is enforced in the sparse component.

In our traffic problem, an event at one time in a given day is unlikely to happen again the next day at the same time and location. However, an event will persist for some time after it happens. So when a traffic event happens, it is likely that it will persist to the next time point. Figure 4 illustrates spareness of events for the sensor 196 and the persistence of events when they happened.

When measuring many variables of a dynamical system,

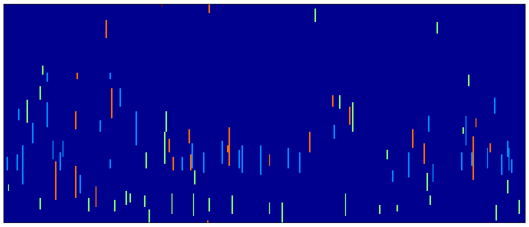


Fig. 4. Color map of matrix S showing its sparsity pattern. S is calculated from one year flow rate of Sensor 196. Each column stands for a day's data.

it is natural to consider use multiple of them to learn the system, instead of using only one variable. Besides, those variables may share some internal structure, given that they are generated from a single system/process. We extend the Bayesian RPCA method by coupling the sparsity structure of multiple variables that exhibit spatio-temporal dependencies. For example, given two data matrices Y_1 and Y_2 observing different physical quantities/variables at the same time and location, we have

$$\begin{aligned} Y_1 &= D_1(Z_1\Lambda_1)W_1^T + B \circ X_1 + E_1 \\ Y_2 &= D_2(Z_2\Lambda_2)W_2^T + B \circ X_2 + E_2, \end{aligned}$$

where the binary matrix B, capturing sparsity, is shared by both data matrices. Such sparsity structure sharing can be further extended to measurements from multiple sensors in a neighborhood, which may be affected by the same event in time. Figure 5 shows the graphical model of our extension to Bayesian RPCA, with coupled variables sharing the same sparsity structure.

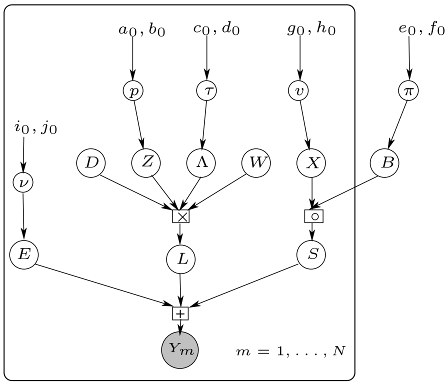


Fig. 5. Graphical model of Bayesian RPCA method with shared sparsity structure between multiple measured variables. There is one "plate" for each measured variable.

In traffic applications, loop detector sensors measure/collect multiple variables. The flow rate is a measurement of number of vehicles passing through the sensor during a unit time (e.g. 30 seconds). Usually, small flow rate means free flow on the road, while larger flow rate means higher density on the road. However, when there is a traffic jam and all vehicles move slowly or even stop, the flow rate is low, but the traffic is not free flowing. An extra variable, the road occupancy, is helpful in distinguishing between these two event types. Also, adjacent sensors may record the same events; hence they are very likely to share the same structure of sparsity of B.

V. Events Detection Experiments

A. *Experimental data*

We obtained a dataset of vehicle flow and road occupancy measured by 38 inductive loop detectors on highway I-494 from the Minnesota Traffic Observatory (MTO) for the whole year of 2011. The 38 loop detector sensors measure traffic flow and occupancy on the southern and eastern bound I-494. The dataset has time resolution of 30 seconds, which accounts for 2880 data points per day for each measurement. Inductive loop detectors fail sometimes, and the missing measurements are flagged as -1 in the dataset ($\approx 8.9\%$ values are missing).

In addition to the readings from the loop detectors on the Minnesota I-494 highway, we also gather incident reports, and weather information. We gather weather data from weathersource.com. We gather incident reports for the segment of road under study for 2011 from the RSS feeds published by the Minnesota Department of Transportation at 511MN.org. Each incident is described by start time, duration, location, camera ID, event type, and blocking. For example, on March 23, 2011 at 11:41AM in cross-section of I-494 EB and Flying Cloud Drive, there was a crash, which lasted for 45 minutes, and blocked at least one lane.

We preprocess the sensor readings by aggregating them into 15-minute resolution, which is the typical time interval used in vehicle volume counts [1]. Considering the strong daily patterns presented by the sensor data, we construct a matrix for each sensor/variable, whose columns contain the daily aggregated data. Thus, an $n \times m$ matrix collects one sensor's data for m days with n points on each day. We also remove an entire day, if the sensor was flagged as failed for the whole day. However, for short duration failures, failure, where some sensor readings during the day are flagged as -1, we keep that day's column.

To evaluate our method of detecting events, we need to identity/annotate the sensor measurements with information from ground-truth traffic events. To this end, we use the gathered incident reports with some changes, as described below. The gathered collection of incidents has some limitations. First, the collection contains only incident events, such as roadwork, crash, hazard, which often reduce the traffic flow. As discussed earlier, other events (e.g. due to social activities or severe weather) may also affect traffic flow yet they are not included in that collection. Second, the collection may contain incidents that do not impact the sensor measurements and thus can not be detected from those sensor measurements.

To mitigate these limitations, we proceed as follows. We plot the whole year's worth of flow, occupancy, their annual average for each 15 minutes in a day, reported incidents, and visibility, aligned by time. Then, we have asked three people to individually review those plots and annotate time segments that they consider fall into the following three types of traffic events: (1) events that reduce flow rate or speed, (2) free flow with unusual high traffic volume, and (3) sensor failure. Annotations where the humans originally disagree, the reviewers were asked to reach a consensus annotation. During this annotation process, the reviewers had no access to the output (detected events) of our proposed approach.

In what follows, we consider the measurements from loop detectors with IDs 196 and 197 on the eastern-bound I-494, which are within 0.5 miles of each other. The ground-truth consists of 232 events total for the two sensors: 148 type-1 events, 83 type-2 events, and 1 type-3 event.

B. *Coupling multiple variables*

We compare three methods for detecting events from measurements from a single sensor: our proposed method using sparsity coupling for the traffic flow and road occupancy, the traditional PCA on traffic flow as a baseline, and the Bayesian RPCA with traffic flow. We use the same hyperparameters and priors for the proposed method and the original Bayesian RPCA. For each method, we count the number of detected events, detected but unlabeled events,undetected but labeled events.

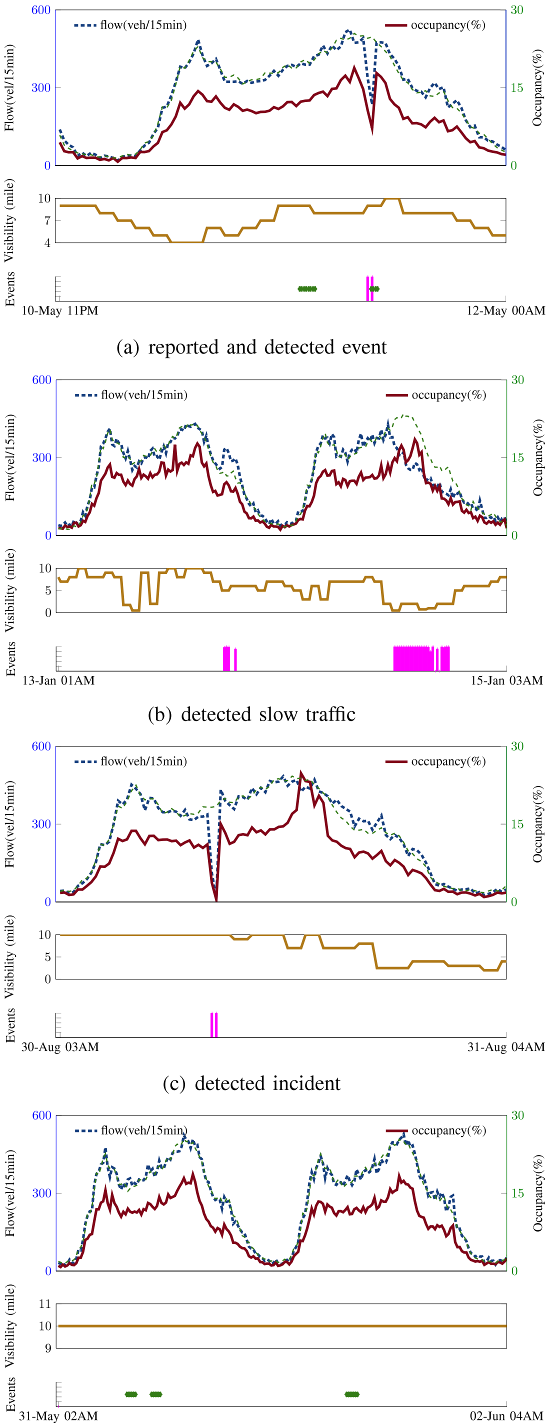
For the PCA method, we use a simple threshold on the noise (residual not captured by the principal components). We used 25 principal components (out of 96 possible components) capturing over 95% of the energy of the data (Euclidean norm). The noise threshold was set to 30 vehicles/15 mins. We decided on this threshold value so that PCA finds approximately the same number of false-positive events as the other methods. With this setting, the number of false-positives for PCA (flow/occupancy), Bayesian RPCA (flow/occupancy), and our proposed coupled Bayesian RPCA were 214/247, 281/290, and 257 respectively.

TABLE I
DETECTION ACCURACY COMPARISON OF THREE METHODS, USING DATA FROM THE SAME SENSOR.

Method	Variable	Overall	Type 1	Type 2
PCA	flow	46.1%	49.3%	39.8%
PCA	occupancy	54.3%	59.5%	45.8%
BRPCA	flow	80.2%	73.6%	91.6%
BRPCA	occupancy	81.5%	86.5%	73.5%
Coupled BRPCA	flow& occupancy	83.2%	87.2%	77.1%

Table I shows the accuracy (ratio of the number of detected events over ground-truth events) of the three methods. Our proposed coupled Bayesian RPCA has the best accuracy (83.2%) among the three, especially in finding events that have negative impacts on traffic conditions (events of type-1). For type-2 events, Bayesian RPCA using only traffic flow has better accuracy than coupling flow and occupancy in our method. Note however that type-2 events (free flow with unusual high traffic volume) have significantly less negative impact than type-1 events,

Figure 6(a) shows an incident (green dots in the *Events* axes) reported by 511MN.org, and is also annotated by the human reviewers. The incident report indicates a car crash near sensor 196 with lane blocking lasting 20 minutes. We observe reduced vehicle flow and road occupancy. The event is correctly detected by our method.



(a) reported and detected event

(b) detected slow traffic

(c) detected incident

(d) reported incidents with no impact

Fig. 6. Comparison of detected events and ground-truth events. In each *Flow* axes, solid lines are measured values; green dot line is the algorithm discovered normal value for flow. In each *Visibility* axes, the visibility from weather stations are illustrated to assist result interpretation. In each *Event* axes, green dots are reported events; red bars indicate the algorithm detected events with probability.

Figure 6(b) illustrate a case of reduced traffic speed due to low visibility. During the time period indicated by the red bars, there was heavy snow in the I-494 area with visibility lower than 2 miles. The flow rate is significantly lower than average, while the road occupancy is not reduced accordingly, suggesting that drivers proceeded with caution at low speed during that time. No incidents were reported by 511.org near that time/location, but the event is annotated by the human reviewers. The event is correctly detected by our method.

Figure 6(c) suggests a lane blocking event. There is no reported incident at the time/location, but the event is annotated by the human reviewers. The event is correctly detected by our method.

Figure 6(d) illustrates a case where there is a reported incident, which does not impact the traffic flow or occupancy. No event in the nearby time/location is annotated by the human reviewers, nor detected by our method.

C. Coupling measurements from neighboring sensors

In light of the strong correlation between measurements from neighboring sensors we discussed in Section III, we couple single variable measurements (either traffic flow or road occupancy) from adjacent sensors. Table II shows that each coupling road occupancy from neighboring sensors gives the best accuracy overall and for type-1 events. Coupling traffic flow gives the best accuracy for type-2 events.

TABLE II
DETECTION ACCURACY COMPARISON OF COUPLING DIFFERENT VARIABLES AMONG TWO SENSORS USING OUR COUPLED BRPCA METHOD.

Spatial Coupling	Overall	Type-1	Type-2
flow	82.5%	74.7%	100%
occupancy	93.7%	94.9%	90.9%

The (coupled) Bayesian RPCA can be implemented for online processing purpose. Its running time also promises real-time processing. Above experiments were running on a 64-bit machine with 8GB memory, Matlab 2011b environment. Average running time using one variable for an entire year is about 2 minutes, with 1000 burn-in and 1000 iterations in Gibbs sampling. Average running time for two coupled variables takes 4 minutes. However, in real use, the matrix size can be dramatically reduced by using only recent a few months data, or only use the same weekday's data for a specific weekday.

Another issue for real-time event detection is how to construct a matrix with future observation in a column. We can fill in average values for the 'future' observation on a day. The average values could be annual or recent a few months average, which serve as the surrogate of normal background. Newly observed values will update those fill-in ones as time forwards.

VI. CONCLUSION

We consider the problem of detecting traffic events from continuously collected measurements from road traffic condition sensors. We focus on traffic slow down, unexpected high traffic volume, and traffic jams events. Detecting such traffic events can help drivers make early decisions to choose travel routes, saving time and energy.

We propose an extension to Bayesian Robust PCA for detecting events in road traffic datastreams. Our extension couples multiple streams of sensor measurements so that they share the sparsity pattern which is inherent in the robust PCA approach. The sparsity sharing can be used both in the temporal dimension (by using different variables measured at the same location) and in the spatial dimension (by using the same variable measured at nearby locations). We experimentally demonstrate using a real dataset that our proposed method significantly improves upon the event detection accuracy of the traditional PCA and Bayesian Robust PCA methods. Furthermore, the Bayesian nature of the method facilitates a probabilistic interpretation of the detected events, while removing the tension in deciding the number of principal components to be used. Moreover, the proposed method can process traffic datastreams incrementally and thus can detect events in an online and real-time fashion.

This work broadens the RPCA's applications, which are usually seen in video surveillance [3], [11] and online recommendation systems [6], [7].

Though we focus on event detection, in the future we plan to use the low-rank component of the traffic datastream computed by the proposed method, to derive initial and boundary conditions for macroscopic road traffic models. For example, the low-rank component can be used to estimate "normal" incoming/outgoing traffic flow at the entry/exit ramps (by removing the effects of outliers/incidents), which will allow to improve the accuracy of short-term forecasting of road traffic conditions.

APPENDIX A
DERIVATION OF GIBBS SAMPLING FOR BRPCA

For brevity, we indicate vectors with lower-case bold letters, matrices with uppercase bold letters, and the ith column and the jth row of a matrix $\mathbf{A}$ with $\mathbf{A}_i$ and $\mathbf{a}_j$ respectively. Furthermore, let $\otimes$ indicate the matrix Kronecker product, $\circ$ the Hadamard matrix product, and $\mathbf{vec}(\mathbf{A}) = [\mathbf{A}_1^T \mathbf{A}_2^T \mathbf{A}_3^T \ldots]^T$ denote the vector that results from stacking the columns of matrix $\mathbf{A}$. Let $\mathbf{diag}(\mathbf{a})$ denote the diagonal matrix with vector $\mathbf{a}$ as its main diagonal.

The column stacking operator distributes over matrix addition. The Kronecker product of two vectors can be written as the regular matrix-vector product

$$\mathbf{vec}(\mathbf{a}\mathbf{b}^T) = (\mathbf{b} \otimes \mathbf{a}) = (\mathbf{b} \otimes \mathbf{I}_{\dim(\mathbf{b})})\mathbf{a}, \tag{1}$$

where $\dim(\mathbf{b})$ is the dimension of $\mathbf{b}$.

Consider matrix $\mathbf{Y}$ of observed random variables generated by a set of hidden matrix random variables, such that

$$\mathbf{Y} = \underbrace{\mathbf{D}\ \mathbf{diag}(\mathbf{z})\ \mathbf{diag}(\boldsymbol{\lambda})\ \mathbf{W}}_{\mathbf{L}} + \mathbf{B} \circ \mathbf{X} + \mathbf{E}, \tag{2}$$

where the dimensions and type of each matrix random variable is given in Table III. We want to infer the posterior distribution of the hidden random variables given the observations $\mathbf{Y}$. We assume that the elements of the added noise matrix $\mathbf{E} = (e_{ij})$ are i.i.d random variables with normal distribution $e_{ij} \sim \mathcal{N}(0, \gamma^{-1})$.

It can be shown that

$$\mathbf{vec}(\mathbf{Y}) = \mathbf{vec}(\mathbf{L}) + \mathbf{vec}(\mathbf{B} \circ \mathbf{X}) + \mathbf{vec}(\mathbf{E}), \tag{3}$$

$\mathbf{L}$ is linear in the columns of $\mathbf{D}$

$$\mathbf{L} = \sum_{j=1}^{K} z_j \lambda_j \mathbf{D}_j \mathbf{w}_j, \text{and that} \tag{4}$$

$$\mathbf{vec}(\mathbf{L}) = \sum_{j=1}^{K} z_j \lambda_j \left(\mathbf{w}_j^T \otimes \mathbf{I}_N\right) \mathbf{D}_j. \tag{5}$$

We first state the Bayes rule for linear Gaussian systems as the Theorem A.1 below (please see section 4.4.3 in [17] for a proof).

TABLE III
DESCRIPTION OF VARIABLES.

Variable	Dim	Description
$\mathbf{Y}$	$N \times M$	noisy observation
$\mathbf{D}$	$N \times K$	left singular vectors, $\mathcal{N}(\mathbf{D}_j \ ; \ \mathbf{0}, N^{-1}\mathbf{I}_N)$
$\mathbf{diag}(\mathbf{z})$	$K \times K$	binary diagonal matrix, Bernoulli$(z_j \ ; p_j)$
$\mathbf{diag}(\boldsymbol{\lambda})$	$K \times K$	diagonal matrix, $\mathcal{N}(\lambda_j \ ; \ 0, \tau^{-1})$
$\mathbf{W}$	$K \times M$	right singular vectors, $\mathcal{N}(\mathbf{w}_j^T \ ; \mathbf{0}, M^{-1}\mathbf{I}_M)$
$\mathbf{B}$	$N \times M$	binary elements, Bernoulli$(b_{ij} \ ; \pi_{ij})$
$\mathbf{X}$	$N \times M$	magnitude of sparse matrix, $\mathcal{N}(\mathbf{X}_j \ ; \ \mathbf{0}, \nu \mathbf{I}_M)$
$\mathbf{E}$	$N \times M$	noise matrix, $\mathcal{N}(e_{ij} \ ; \ 0, \gamma^{-1})$
p_j	scalar	Beta$(p_j \ ; \alpha_o, \beta_o)$
τ	scalar	$\Gamma(\tau \ ; \ a_o, b_o)$
π_{ij}	scalar	Beta$(\pi_{ij} \ ; \ a_1, b_1)$
ν	scalar	$\Gamma(\nu \ ; \ c_o, d_o)$
γ	scalar	$\Gamma(\gamma \ ; \ e_o, f_o)$

Theorem A.1 ([17]): Let $\mathbf{y}$ be a noisy observation of a hidden random variable $\mathbf{x}$. Suppose $\mathbf{x}$ generates $\mathbf{y}$ through a linear system $\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{b} + \boldsymbol{\epsilon}$, where $\boldsymbol{\epsilon}$ is Gaussian noise $\mathcal{N}(\boldsymbol{\epsilon} \ ; \ \mathbf{0}, \boldsymbol{\Sigma}_y)$. Let the prior distribution of $\mathbf{x}$ be

$$p(\mathbf{x}) = \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_x, \boldsymbol{\Sigma}_x),$$

and the likelihood of $\mathbf{y}$ given $\mathbf{x}$ be

$$p(\mathbf{y}|\mathbf{x}) = \mathcal{N}(\mathbf{y} \ ; \ \mathbf{A}\mathbf{x} + \mathbf{b}, \boldsymbol{\Sigma}_y).$$

The posterior distribution of $\mathbf{x}$ given the observations $\mathbf{y}$ is given by

$$\begin{aligned} p(\mathbf{x} \mid \mathbf{y}) &= \mathcal{N}(\mathbf{x} \ ; \ \boldsymbol{\mu}_{x|y}, \boldsymbol{\Sigma}_{x|y}) \\ \boldsymbol{\Sigma}_{x|y}^{-1} &= \boldsymbol{\Sigma}_x^{-1} + \mathbf{A}^T \boldsymbol{\Sigma}_y^{-1} \mathbf{A} \\ \boldsymbol{\mu}_{x|y} &= \boldsymbol{\Sigma}_{x|y} \left(\mathbf{A}^T \boldsymbol{\Sigma}_y^{-1} (\mathbf{y} - \mathbf{b}) + \boldsymbol{\Sigma}_x^{-1} \boldsymbol{\mu}_x\right) \end{aligned}$$

■In Gibbs sampling, we fix an order that we sample the various random variables. To perform the sampling, we need to compute the full conditional probability distribution of the sampled random variables given all the other random variables and the observations $\mathbf{Y}$.

Sampling D.

We first sample $\mathbf{D} = [\mathbf{D}_1, \ldots, \mathbf{D}_K]$, where $\mathbf{D}_k$ is the kth column of $\mathbf{D}$. We need to compute the full conditional distribution of $\mathbf{D}_k$ given all the other random variables. In one iteration of Gibbs sampling, the observation for $\mathbf{D}_k$ is found by subtracting the last step sampled spare component and all values contributed by all the other columns $\mathbf{D}_j$ from the observation matrix $\mathbf{Y}$. We denote the observation associated with $\mathbf{D}_k$ in this manner as

$$\mathbf{Y}^{-k} = \mathbf{Y} - \mathbf{B} \circ \mathbf{X} - \sum_{j=1, j \neq k}^{K} z_j \lambda_j \mathbf{D}_j \mathbf{w}_j. \tag{6}$$

Using Eqs. (3-5) it follows that

$$\mathbf{vec}(\mathbf{Y}^{-k}) = z_k \lambda_k \left(\mathbf{w}_k^T \otimes \mathbf{I}_N\right) \mathbf{D}_k + \mathbf{vec}(\mathbf{E}), \tag{7}$$

which implies the linear system

$$\mathbf{vec}(\mathbf{Y}^{-k}) = \mathbf{A}_{-k} \mathbf{D}_k + \mathbf{vec}(\mathbf{E}), \tag{8}$$

where $\mathbf{A}_{-k}$ is the $MN \times N$ matrix $z_k \lambda_k \left(\mathbf{w}_k^T \otimes \mathbf{I}_N\right)$. Hence, we can invoke Theorem A.1 to compute the posterior full conditional distribution of $\mathbf{D}_k$. In particular, given prior distribution for $\mathbf{D}_k$

$$p(\mathbf{D}_k) = \mathcal{N}(\mathbf{D}_k \ ; \ \mathbf{0}, N^{-1}\mathbf{I}_N),$$

and likelihood

$$p(\mathbf{Y}^{-k} | \mathbf{D}_k) = \mathcal{N}(\mathbf{vec}(\mathbf{Y}^{-k}) \ ; \ \mathbf{A}_{-k}\mathbf{D}_k, \gamma^{-1}\mathbf{I}_{MN}),$$

by applying Theorem A.1, the posterior from which we sample $\mathbf{D}_k$ is

$$p(\mathbf{D}_k \mid \mathbf{Y}^{-k}) = \mathcal{N}(\mathbf{D}_k \ ; \ \boldsymbol{\mu}_{\mathbf{D}_k|\mathbf{Y}^{-k}}, \boldsymbol{\Sigma}_{\mathbf{D}_k|\mathbf{Y}^{-k}}), \text{ where}$$

$$\begin{aligned} \boldsymbol{\Sigma}_{\mathbf{D}_k|\mathbf{Y}^{-k}}^{-1} &= N\mathbf{I}_N + \gamma \mathbf{A}_{-k}^T \mathbf{A}_{-k} \\ &= N\mathbf{I}_N + \gamma \sum_{j=1}^{M} \lambda_k^2 z_k^2 w_{kj}^2 \\ \boldsymbol{\mu}_{\mathbf{D}_K|\mathbf{Y}^{-k}} &= \boldsymbol{\Sigma}_{\mathbf{D}_k|\mathbf{Y}^{-k}} \left(\mathbf{A}_{-k}^T \gamma \mathbf{A}_{-k}\right) \\ &= \gamma \boldsymbol{\Sigma}_{\mathbf{D}_k|\mathbf{Y}^{-k}} \sum_{j=1}^{M} \lambda_j z_j w_{kj} \mathbf{Y}_j^{-k} \end{aligned} \tag{9}$$

Sampling z_k.

We treat z_k as hidden variable for the purpose of Gibbs sampling, with associated observation also $\mathbf{Y}^{-k}$ given by (6). Note that

$$\mathbf{vec}(\mathbf{Y}^{-k}) = \left(\lambda_k \left(\mathbf{w}_k^T \otimes \mathbf{I}_N\right) \mathbf{D}_k\right) z_k + \mathbf{vec}(\mathbf{E}). \tag{10}$$

This implies the linear system

$$\mathbf{vec}(\mathbf{Y}^{-k}) = \mathbf{a}_{-k} z_k + \mathbf{vec}(\mathbf{E}) \tag{11}$$

where

$$\mathbf{a}_{-k} = \lambda_k \left(\mathbf{w}_k^T \otimes \mathbf{I}_N\right) \mathbf{D}_k. \tag{12}$$

We assume a Bernoulli prior distribution $p(z_k) =$ Bernoulli(p_k), with the hyperparameter p_k having a Beta distribution $p_k \sim \text{Beta}(\alpha_0, \beta_0)$. The likelihood of the associated observation $\mathbf{Y}^{-k}$ given z_k is

$$p(\mathbf{Y}^{-k} \mid z_k) \sim \mathcal{N}(\mathbf{vec}(\mathbf{Y}^{-k}) \ ; \ \mathbf{a}_{-k} z_k, \gamma^{-1}\mathbf{I}_{MN}) \tag{13}$$

We sample z_k from the Bernoulli posterior distribution (as [23])

$$z_k \sim \text{Bernoulli}\left(\frac{q_1}{q_0 + q_1}\right), \text{ where} \tag{14}$$

$$\begin{aligned} q_1 &= p(z_k = 1 \mid \mathbf{Y}^{-k}) \\ &\propto p(\mathbf{Y}^{-k} \mid z_k = 1) p(z_k = 1) \\ &\propto p_k \exp\left(-\frac{\gamma}{2}\|\mathbf{vec}(\mathbf{Y}^{-k}) - \mathbf{a}_{-k}\|^2\right), \end{aligned} \quad (15)$$

$$\begin{aligned} q_0 &= p(z_k = 0 \mid \mathbf{Y}^{-k}) \\ &\propto (1 - p_k)\mathcal{N}(\mathbf{Y}^{-k} \ ; \ \mathbf{0}, \gamma^{-1}\mathbf{I}) \propto 1 - p_k, \end{aligned} \quad (16)$$

Note that the two likelihood distributions q_1 and q_0 are both Gaussian with the same covariance but different means, and therefore $q_1/(q_0 + q_1)$ is the odds ratio for z_k.

Sampling p_k.
We have a Bernoulli-Beta conjugate prior pair for the z_k and its hyperparameter p_k, and therefore the posterior of p_k is given by

$$p_k \sim \text{Beta}(\alpha_0 + z_k, \beta_0 + 1 - z_k). \quad (17)$$

Sampling λ_k.
We treat λ_k as hidden variable for the purpose of Gibbs sampling, with the associated observation also $\mathbf{Y}^{-k}$ and given by (6). Note that

$$\mathbf{vec}(\mathbf{Y}^{-k}) = \left(z_k \left(\mathbf{w}_k^T \otimes \mathbf{I}_N\right) \mathbf{D}_k\right) \lambda_k + \mathbf{vec}(\mathbf{E}). \quad (18)$$

This implies the linear Gaussian system

$$\mathbf{vec}(\mathbf{Y}^{-k}) = \mathbf{a}_{-k}\lambda_k + \mathbf{vec}(\mathbf{E}), \text{ where} \quad (19)$$

$$\mathbf{a}_{-k} = z_k \left(\mathbf{w}_k^T \otimes \mathbf{I}_N\right) \mathbf{D}_k. \quad (20)$$

We assume a Gaussian prior distribution for $\mathcal{N}(\lambda_k \ ; \ 0, \tau^{-1})$. The likelihood of the associated observation is

$$p(\mathbf{Y}^{-k} \mid \lambda_k) \sim \mathcal{N}(\mathbf{vec}(\mathbf{Y}^{-k}) \ ; \ \mathbf{a}_{-k}\lambda_k, \gamma^{-1}\mathbf{I}_{MN}). \quad (21)$$

The posterior full conditional distribution $p(\lambda_k \mid \mathbf{Y}^{-k})$ that we sample λ_k from follows by applying Theorem A.1 to the linear system $\mathbf{vec}(\mathbf{Y}^{-k}) = \mathbf{a}_{-k}\lambda_k + \mathbf{vec}(\mathbf{E})$.

Sampling W.
Regarding the Gibbs sampling of the rows of $\mathbf{W}$, which are assumed to have a Gaussian prior $\mathcal{N}(\mathbf{w}_k \ ; \ \mathbf{0}, M^{-1}\mathbf{I}_M)$, we utilize

$$\mathbf{Y}^T = \mathbf{W}^T \ \mathbf{diag}(\mathbf{z}) \ \mathbf{diag}(\boldsymbol{\lambda}) \ \mathbf{D}^T + (\mathbf{B} \circ \mathbf{X})^T + \mathbf{E}^T, \quad (22)$$

and proceed as for sampling the columns of $\mathbf{D}$ above.

Sampling B and X.
Let

$$\mathbf{S} = \mathbf{Y} - \mathbf{D} \ \mathbf{diag}(\mathbf{z}) \ \mathbf{diag}(\boldsymbol{\lambda}) \ \mathbf{W}. \quad (23)$$

For each matrix, we sample one element at a time, in turn.

Sampling b_{ij}.
We treat b_{ij} as hidden variable and associate with it the observation s_{ij}. Note that

$$s_{ij} = x_{ij}b_{ij} + e_{ij}. \quad (24)$$

We assume prior distribution $p(b_{ij}) \sim \text{Bernoulli}(\pi_{ij})$, with the hyperparameter π_{ij} having distribution $\pi_{ij} = \text{Beta}(\alpha_1, \beta_1)$. The likelihood of the associated observation is

$$p(s_{ij} \mid b_{ij}) \sim \mathcal{N}(s_{ij} \ ; \ x_{ij}b_{ij}, \gamma^{-1}). \quad (25)$$

The posterior distribution is

$$b_{ij} \sim \text{Bernoulli}\left(\frac{q_1}{q_0 + q_1}\right), \text{ where} \quad (26)$$

$$\begin{aligned} q_1 &= p(b_{ij} = 1 \mid s_{ij}) \propto p(s_{ij} \mid b_{ij} = 1) p(b_{ij} = 1) \\ &\propto \pi_{ij} \exp\left(-\frac{\gamma}{2}(x_{ij} - s_{ij})^2\right) \end{aligned} \quad (27)$$

$$q_0 = p(b_{ij} = 0 | s_{ij}) \propto (1 - \pi_{ij})\mathcal{N}(0, \gamma^{-1}) \propto 1 - \pi_{ij}, \quad (28)$$

Sampling π_{ij}.
Since b_{ij} has Bernoulli distribution with hyperparameter π_{ij} whose prior is a Beta distribution, the posterior distribution of π_{ij} is

$$\pi_{ij} \sim \text{Beta}(\alpha_1 + b_{ij}, \beta_1 + 1 - b_{ij}). \quad (29)$$

In (29), if there are Markov dependencies among a group $\mathcal{G}$ of b_{ij}'s, then all of them in the group will share the same π_{ij} value, and in Eq. (29) we replace the b_{ij} with their sum,

$$\pi_{ij} \sim \text{Beta}(\alpha_1 + \sum_{ij \in \mathcal{G}} b_{ij}, \beta_1 + \sum_{ij \in \mathcal{G}} (1 - b_{ij})). \quad (30)$$

Sampling x_{ij}.
We treat the variable x_{ij} as a hidden variable with Gaussian prior distribution $p(x_{ij}) = \mathcal{N}(x_{ij} \ ; \ 0, \nu^{-1})$, with hyperparameter $\nu \sim \text{Gamma}(c_0, d_0)$. We associate with it the observation s_{ij}, whose likelihood is

$$p(s_{ij} \mid x_{ij}) \sim \mathcal{N}(s_{ij} \ ; \ b_{ij}x_{ij}, \gamma^{-1}). \quad (31)$$

The posterior distribution of x_{ij} is

$$p(x_{ij} \ | s_{ij}) \propto \mathcal{N}(x_{ij} \ ; \ \mu_{x|s}, \sigma_{x|s}), \text{ where} \quad (32)$$

$$\sigma_{x|s}^{-1} = \nu + \gamma b_{ij}^2 \quad (33)$$

$$\mu_{x|s} = \gamma \sigma_{x|s} b_{ij} s_{ij} \quad (34)$$

Sampling ν, τ, and γ.
Often, we encounter a Gaussian random variable with known mean and unknown covariance

$$\mathbf{X}_j \sim \mathcal{N}(\mathbf{0}, \nu^{-1}\mathbf{I}_N), \quad (35)$$

where $p(\nu) = \Gamma(\nu \ ; \ c_o, d_o)$, for some positive constants c_o, d_o. To be able to perform Gibbs sampling of the random variable ν, we need to compute its posterior.

The posterior of the precision ρ (with $\Gamma(\rho; \alpha_o, \beta_o)$) of a normal distribution with known mean $\mathcal{N}(x; \mu, \rho^{-1})$, given n observations x_i from the normal distribution is a Γ distribution with parameters

$$\alpha_n = \alpha_o + n/2 \text{ and } \beta_n = \beta_o + (1/2)\sum_{i=1}^{n}(x_i - \mu)^2. \quad (36)$$

See sections 9.5 Theorem 2 and 9.9 Theorem 2 in "Optimal Statistical Decisions" by Morris DeGroot [10]. (For multivariate normal and Wishart $(a_0, \mathbf{R}_o)$ prior on its precision, the posterior of the precision is Wishart with $a_o + n$ degrees of freedom and precision $\mathbf{R}_o + \sum_{i=1}^{n}(\mathbf{X}_i - \boldsymbol{\mu})(\mathbf{X}_i - \boldsymbol{\mu})^T$)

Applying this fact to ν we get that its posterior given $\mathbf{X}$ is

$$p(\nu \mid \mathbf{X}) \propto \Gamma\left(c_o + NM/2, d_o + (1/2)\sum_{j=1}^{M} \mathbf{X}_j^T \mathbf{X}_j\right). \quad (37)$$

Similarly, since $\lambda_j \sim \mathcal{N}(0, \tau^{-1})$ and $\tau \sim \Gamma(a_0, b_o)$, it follows the the posterior of τ given $\boldsymbol{\lambda}$ is

$$p(\tau \mid \boldsymbol{\lambda}) \propto \Gamma\left(a_o + K/2, b_o + (1/2)\sum_{j=1}^{K} \lambda_j^2\right). \quad (38)$$

Finally, since $\mathcal{N}\left(\mathbf{E}\ ;\ \mathbf{0}, \gamma^{-1}\mathbf{1}_{N\times M}\right)$ and $\gamma \sim \Gamma(e_o, f_o)$, it follows that the posterior of γ given the observations $\mathbf{Y}$ and all the other random variables $\mathbf{z}$, $\boldsymbol{\lambda}$, $\mathbf{W}$, $\mathbf{B}$, and $\mathbf{X}$ (i.e. $\mathbf{E}$) is

$$\begin{aligned} &\propto \Gamma(e_o + NM/2, f_o + (1/2)\sum_{i=1}^{N}\sum_{j=1}^{M} e_{ij}) \\ &= \Gamma(e_o + NM/2, f_o + (1/2)\|\mathbf{E}\|_F^2), \quad (39) \end{aligned}$$

where $\mathbf{E}$ is given by Eq. (2).

Acknowledgment

We would like to thank IBM for partially supporting this research effort. We would also like to thank the Minnesota Traffic Observatory for providing us with of raw loop detector dataset, and thank Ms. Yulei Wang and Ms. Yao Li for their help on annotating the dataset with traffic events. We are very grateful to the anonymous reviewers for their careful reading and cogent commentary.

References

[1] Guide for interpreting short duration traffic count reports. Washington State Department of Transportation, www.wsdot.wa.gov, 2010.

[2] S.D. Babacan, M. Luessi, R. Molina, and A.K. Katsaggelos. Sparse Bayesian Methods for Low-Rank Matrix Estimation. *Signal Processing, IEEE Transactions on*, Vol.60, no.8, pp.3964-3977, 2012.

[3] S. Becker, E. J. Candès, and M. Grant. Templates for convex cone problems with applications to sparse signal recovery. *Mathematical Programming Computation*, 3(3):165–218, 2011.

[4] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[5] E. Candès, X. Li, Y. Ma, and J. Wright. Robust principal component analysis? *Journal of the ACM*, 58(2), 2011.

[6] E. J. Candès and Y. Plan. Matrix completion with noise. *Proceedings of the IEEE*, 98(6):925–936, 2010.

[7] E. J. Candès and B. Recht. Exact matrix completion via convex optimization. *Foundations of Computational Mathematics*, 9(6):717–772, 2009.

[8] E. J. Candès and T. Tao. The power of convex relaxation: Near-optimal matrix completion. *IEEE Transactions on Information Theory*, 56(5):2053–2080, 2010.

[9] J.F. Collins, C.M. Hopkins, and J.A. Martin. Automatic incident detection – TRRL algorithms HIOCC and PATERG. *TRRL Supplementary Report*, No.526, Crowthorne, Berkshire, UK, 1979.

[10] M. H. DeGroot. *Optimal Statistical Decisions*. John Wiley & Sons, 2004.

[11] X. Ding, L. He, and L. Carin. Bayesian robust principal component analysis. *Image Processing, IEEE Transactions on*, 20:3419–3430, 2011.

[12] V. Guralnik, and J. Srivastava. Event detection from time series data. *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, 33–42, 1999.

[13] P. Hoff. Simulation of the matrix Bingham-von Mises-Fisher distribution, with applications to multivariate and relational data. *Journal of Computational and Graphical Statistics*, Vol.18, Issue 2, pp.438-456, 2009.

[14] A. Ihler, J. Hutchins, and P. Smyth. Learning to detect events with markov-modulated poisson processes. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 1, 2007.

[15] F. Jiang, Y. Wu, and A. Katsaggelos. Abnormal event detection from surveillance video by dynamic hierarchical clustering. *IEEE International Conference on Image Processing*, 5:V–145–V–148, 2007.

[16] B. Morris and M. Trivedi. Real-time video based highway traffic measurement and performance monitoring. *Proceedings of the IEEE Intelligent Transportation System Conference*, pages 59–64, 2007.

[17] K. P. Murphy. *Machine Learning A Probabilistic Perspective*. MIT Press, 2012.

[18] S. Olariu and M. Weigle, editors. *Vehicular Networks From Theory to Practice*. CRC Press, 2009.

[19] H.J. Payne, E.D. Helfenbein, and H.C. Knobel. Development and testing of incident detection algorithms, Volume 2: research methodology and detailed results. *Report No. FHWA-RD-76-20*, FHWA, Washington D.C. April 1976.

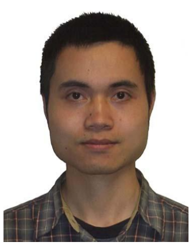
[20] R. Salakhutdinov, and A. Mnih. Bayesian Probabilistic Matrix Factorization. *Proceedings of the 25th International Conference on Machine Learning*, 880–887, Helsinki, Finland, 2008.

[21] M. E. Tipping and C. M. Bishop. Mixtures of probabilistic principal component analysis. *Neural Computation*, 11:443–482, 1999.

[22] R. Weil, J. Wootton, and A. Garcia-Ortiz. Traffic incident detection: Sensors and algorithms. *Mathematical and Computer Modeling*, 27, 257–291, 1998.

[23] M. Zhou, H. Chen, J. Paisley, L. Ren, L. Li, Z. Xing, D. Dunson, G. Sapiro, and L. Carin. Nonparametric bayesian dictionary learning for analysis of noisy and incomplete images. *IEEE Trans. Image Processing*, 21:130–144, 2012.

Konstantinos Kalpakis Konstantinos Kalpakis received the Ph.D. degree in computer science in 1994. He is currently an Associate Professor at the Department of Computer Science and Electrical Engineering Department at the University of Maryland, Baltimore County. His research interests include wireless sensor networks, query processing in distributed systems, and mining timeseries datasets.



Shiming Yang Shiming Yang received his M.S. degree in Applied Mathematics, and Ph.D. degree in Computer Science from the University of Maryland, Baltimore County. His research interests include Bayesian learning and inference for datastreams, large scale data mining and machine learning.

Alain Biem Alain Biem received a Ph.D. degree in Computer Science from the University of Paris 6, France, in 1997 with Summa Cum Laude honors. He is currently a Senior Research Scientist and project lead at the IBM T. J. Watson Research Center where he is involved in various research and development projects on machine learning, time series analysis, datastream analytics, and Big Data computing.