Online Traffic Speed Forecasting Considering Multiple Periodicities and Complex Patterns

Alexander Chen
National Taiwan University of Science and Technology
Taipei, Taiwan
alex@arkross.com

Hsing-Kuo Pao
National Taiwan University of Science and Technology
Taipei, Taiwan
pao@mail.ntust.edu.tw

Yuh-Jye Lee
National Chiao Tung University, Hsinchu, Taiwan
yuhjye@math.nctu.edu.tw

ABSTRACT

Intelligent Transportation Systems (ITS) has been developed to aid drivers and other road-users to make a better travel decision. In recent years, many research efforts have been devoted in this field. Being one kind of time-series data, we can analyze the traffic data following the general aspects of studying time-series, which contains the analysis of periodicity of many kinds. This work highlights the study on the (long-term) multiple periodicities that could be found in traffic data while also considers more specific aspects such as unexpected short-term patterns, spatial relationship and feature correlations. Thanks to the periodicity of traffic data, most experienced drivers can tell how the traffic state will be on the road with given specific time and location. We aim to propose an approach with many of the above aspects to reach a quality traffic speed forecasting. We choose Gaussian process regression as the base model to realize the approach. Given the forecasting that considers all the above aspects, we enjoy the speed forecasting performance with MAE equal to one to two mph at its peak performance for a challenging speed forecasting 30-minute ahead of the current time.

Keywords

Gaussian process, Intelligent Transportation Systems, periodicity, traffic forecasting.

1. INTRODUCTION

1.1 Motivation

Accurate and real-time traffic state forecasting has been a critical problem in *Intelligent Transportation Systems*. In general, traffic data contains complex patterns which could be hard to fully understand and to predict. The traffic data, as one kind of time-series data may contain periodic patterns that repeat continuously through time. Knowing the rush hours in 8AM morning and 5PM evening everyday gives a common sense periodic pattern for road-users like us to avoid traffic congestion for efficient commuting between homes and offices. To consider a longer period of repetitive patterns, we may see late rush hours on Monday morning and early rush hours on Friday evening which happen in a weekly basis. In general, a robust prediction model should be able to understand multiple periodic patterns that are hidden in the data for traffic forecasting. In this work, one of the main goals is to propose a multi-periodicity model that can integrate the hints from multiple periodicities for traffic speed prediction.

Aside from the various periodic patterns, traffic data is often highly dynamic, and it is not uncommon that we see a sudden change in traffic state at any moment. These changes might be temporary, but the effect is usually significant. A little change in one vehicle speed may affect the overall traffic speed in the interconnected segments on the road. We call the patterns the short-term patterns and the periodic patterns that we discussed in the previous paragraph the long-term patterns. Other than the short-term patterns, to further enhance the traffic forecasting, we can consider the spatial relationship between data collecting sensors if we know the traffic information from more than a single location. Also, we can rely on collecting more features, such as *occupancy* and *flow* that could be related to traffic speed to help the traffic speed forecasting. In this work, we aim to consider all the above complex patterns as well as the multiple periodic patterns for a robust traffic state forecasting.

We adopt Gaussian process [11] as the key method to build our forecasting model. Given the methodology, we have the mechanism to integrate the information from multiple periodicities, short-term patterns, spatial relationship, and feature correlations in a single forecasting model. Following the maximum likelihood principle, we look for the best parameter set for prediction given massively historical data for training. We use the conjugate gradient method to achieve the optimal likelihood. To deal with the massive traffic data, we also propose a mechanism that can consider both the short-term and long-term data simultaneously and the mechanism can decide when to acquire additional data and to re-train the model to keep the most recent state in real-time forecasting. The overall methodology is designed for traffic state forecasting; however, the general framework can be used for other time series data of many kinds as well.

We evaluate the proposed model on a public San Diego ITS sensor data from RDE dataset collections [12]. It is a public data resource that is continuously maintained and we can expect massive data collection for as accurate as possible prediction. To speak of traffic speed prediction, due to the high dynamicity in the traffic data, sometimes even using the

Machine learning for large scale transportation systems 2016 San Francisco, California, USA

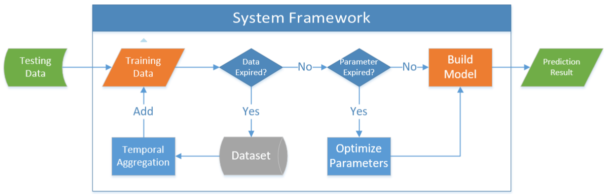


Figure 1: Model Framework

current traffic state value as the prediction value can yield a result with relatively low error. This is more likely to be true when we only predict the traffic in a short period, say, next 5 minutes traffic state. According to one of the methodologies used by INRIX[1] to predict travel time [9], it is meaningful to predict the traffic 30 minutes ahead of time. By having the next 30-minute traffic speed forecasting result, commuters may have enough time for them to re-consider their original traveling plan and make the decision about changing routes or staying in the original path to reach the final destination. Based on the experiment results, we show that the proposed method can reach one or two mph MAE on the San Diego ITS dataset.

1.2 Related Work

Various methods and approaches have been founded in recent years to continuously improve the reliability and accuracy of road traffic state forecasting. One of the common methods used for time series, especially traffic data is Autoregressive Integrated Moving Average (ARIMA) model [8, 2] along with its variations, seasonal ARIMA model [6]. Kalman filtering [1, 18] has been proposed from a data-driven approach aspect for traffic forecasting. Gaussian Process Regression (GPR) [19, 3, 17, 4] has also been researched to model traffic data despite its cubic time complexity. Aside from GPR, other kernel-based approach such as support vector machines [16] has also been developed to forecast traffic conditions. In recent years, deep learning (including neural networks) approach has also been proposed [10, 14, 21, 20] to deal with traffic data when massive data is available. Other traffic speed forecasting methods are also developed [7, 5, 15] to specifically tackle the traffic speed related issues.

1.3 Research Framework for Massive Data Inputs

Overall, the proposed method contains a framework that can deal with online prediction with massive data inputs while maintaining an acceptable level of accuracy. After an initial preprocessing on the raw data, we operate temporal aggregation before the model training. By doing it, we hope that the periodic patterns become more obvious without losing too much details. Aggregation can also save some space and reduce time complexity by reducing the number of data which we want to include in the model. On the side, we optimize the parameters periodically to keep the model close to the most recent situations. The details are shown in Figure 1.

The rest of the article is organized as follows. Before we go on to introduce the proposed method, we discuss the main focused problem in Section 2. Afterwards, in Section 3, we elaborate the details of the proposed framework. There are various models that we discuss for the complex patterns in the traffic data: the multi-periodicity model, and the models considering short-term inputs, spatial relationship and feature correlations. The experiment results are shown in Section 4 and in Section 5, we summarize our presentation.

[1] INRIX (http://inrix.com/) is a company which deals with traffic data using Big Data and Internet of Things (IoT) technology.

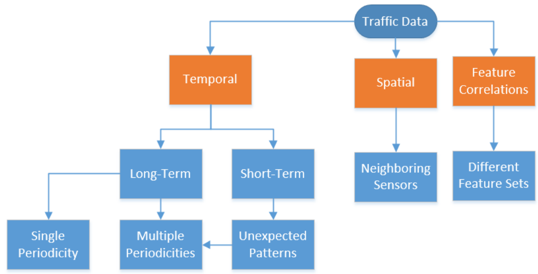


Figure 2: Different aspects of road traffic data

2. TRAFFIC SPEED FORECASTING IN INTELLIGENT TRANSPORTATION

In traffic state forecasting, it is one of the most common approaches to use the past data to predict what could happen in the coming moments. However, the volatility in the traffic data has been a big issue to tackle for such a data-driven approach. The road traffic data, as one kind of time series data contains several different aspects which we can consider them individually or plurally in its data pattern explanation. To be able to effectively predict what could happen in the future or near future, we usually consider the *temporal* relationship, the *spatial* relationship and feature correlations in the data, as shown in Figure 2. In the temporal relationship, we can consider the short-term and the long-term bahaviors in the data and for the long-term behavior, the multiple periodicities is one of the highlighted points we would like to address in this study. We take turn to discuss each of the aspects to illustrate what the proposed method is for this work.

2.1 Periodicity

Periodicity, or a periodic pattern in traffic data, is highly influenced by environmental factors such as seasons, daylight, and social behaviors such as working hours. A series of data like traffic data may contain mixed periodic patterns due to multiple factors, further discussed below.

Daily Patterns

During the day, people have a good vision of the road, and therefore they are able to drive in different styles. Moreover, we can easily observe what could influence the traffic patterns for people to go on and go off from work on weekdays. During the rush hours, there will almost always be a congestion in some sections of a road. An everyday driver knows the patterns and can try a detour to avoid the congestion in advance to save time. This kind of patterns usually repeats itself almost everyday. When we consider a daily model, we treat each single day independently and the data from different days but at the same time are aggregated together in the daily statistics computation.

In Figure 3, we see that there is a drop in traffic speed

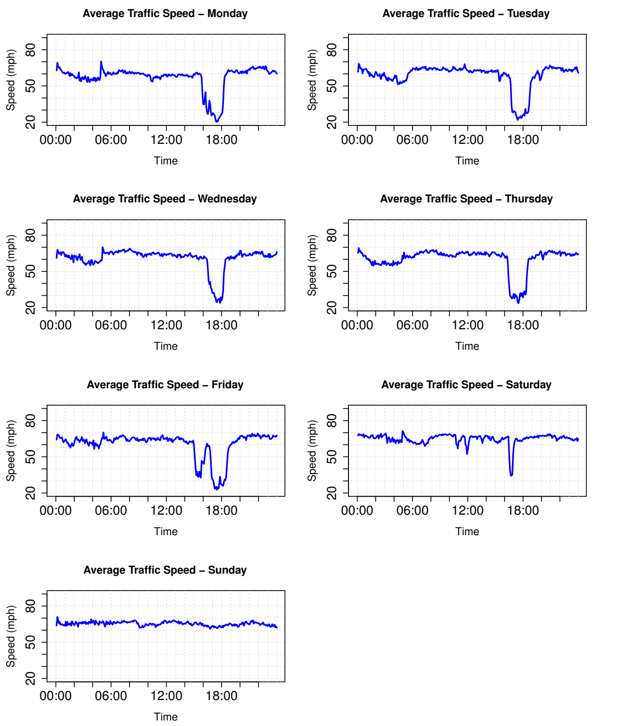


Figure 3: The daily traffic speed patterns on seven days, a result collected from Nov. 17, 2010 to Nov. 23, 2010.

at around 4 PM almost everyday (especially on weekdays). The details might be slightly different in different days (and in different locations), but the overall patterns are similar. Technically, we can operate a temporal aggregation to group the data of the same time in a day together across different days (e.g., several 8AM mornings' data) to discover the daily patterns without showing too much details.

Weekly Patterns

Other than the daily patterns, we often observe weekly patterns when a longer time of data are collected. Most business is run in a weekly basis, therefore, we see its effect on road traffic also in weekly periodic patterns. Taxi drivers know how to increase the possibility of picking up potential passengers at a certain time on a weekday; also, we often see heavy inbound traffic on Monday morning and outbound traffic on Friday afternoon or evening when most people work in town and live out of town. That is to say, we should be able to distinguish between the patterns of Monday morning and Friday morning when we look into more details of the data. By saying this, a good model that considers weekly patterns should treat the data from Monday morning and Friday morning differently. That is, the data from the same time but in different weeks may be aggregated together for analysis; however, the data from the same time but in different days may be collected separately as they may real heterogenous patterns caused by different factors.

Let us refer to Figure 4 where we consider a whole week simultaneously for analysis. By doing it, we can easily observe different patterns between weekdays and weekend and between different weekdays sometimes. We notice the severe

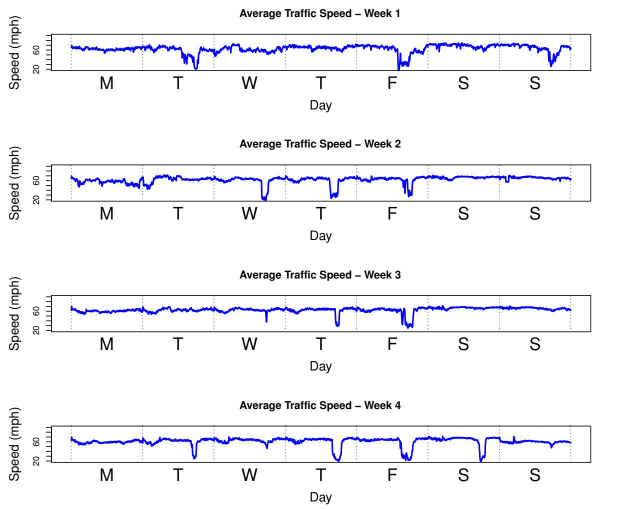


Figure 4: The weekly traffic speed patterns of four consecutive weeks, a result collected from Oct. 18, 2010 to Nov. 14, 2010.

congestions at Friday 4 PM at those four weeks; however, we may or may not see the similar patterns at the same time in some other days in the same or different weeks. In general, we may see similar patterns for the data that are observed at the same time in a week but from different weeks.

To effectively predict traffic states in some future moments, in this work, we would like to consider both the daily and weekly patterns, the two main periodic patterns for traffic data analysis which are considered *long-term* patterns. Also, we take into account the unexpected patterns or called *short-term* patterns. To have both, with some additional help from the estimation of spatial correlations and feature correlations, we can build a more powerful model than before.

2.2 Unexpected Short-Term Patterns

Based on what we observe in the previous section (Section 2.1), although daily and weekly patterns are likely to happen, there are still some exceptions in some days. Usually these exceptions could be caused by some environmental factors such as sudden rainfall or snowfall, or by some human factors such as accidents. Weather factors can cause unusual drop in traffic speed since the drivers need to slow down to avoid slipping over the rainwater or snow. This kind of unusual pattern may last for a few minutes to a few hours. On the other hand, if the unusual pattern is caused by accidents or sudden braking, the pattern going from normal to congestion will fluctuate very quickly but only persist for a few moments. In Figure 5, we illustrate the unexpected speed pattern, the current pattern (blue) that is deviated from the mean value (dashed line) of some other days, taken from a specific sensor (sensor No. 2 in Figure 8). Going by this thought, we consider including short-term data in the model training. Since we want to see the recent trends in more detail, we collect data with higher sampling rate than the usual (compared to the long-term data collection).

2.3 Spatial Considerations

Other than the temporal relationships in data, it is inter-

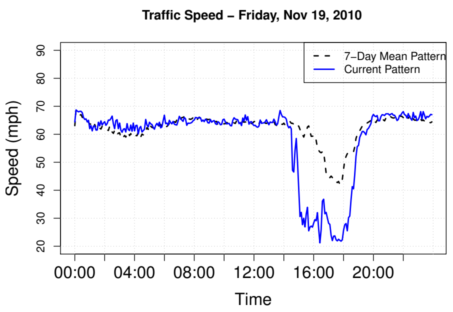


Figure 5: One day traffic speed pattern with a congestion in the afternoon (blue), which is deviated from the mean pattern (dashed) that is computed based on a period of seven days.

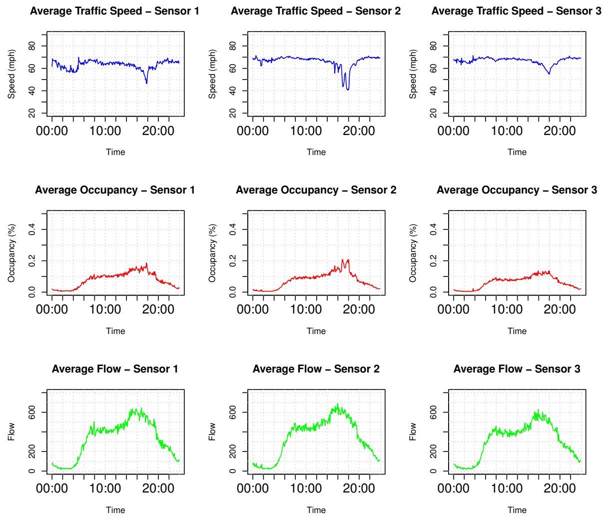


Figure 6: The speed, occupancy, and flow information of three nearby sensors

esting to study the spatial relationship in data. Intuitively, when we want to forecast traffic speed at a location, we can use the nearest sensors' data from the focused location to complete the task. The closer a sensor is to a location, the more relevant it becomes in general. Additionally, a traffic state in a section of the road might be similar to the nearby sections of the same road. For example, if there is a traffic jam at sensor s, it is likely that there will be similar traffic jam happened at sensor $s-1$ and $s+1$ at the same or similar time. Figure 6 shows the similar traffic states from three nearby sensors on I5N highway at November 10, 2010.

2.4 Feature Correlations

To predict traffic speed in some near future moments, we may rely on some other features to enhance the prediction accuracy. In reality, vehicle speed could be affected by many factors. First, drivers may have different preferences in driving their vehicles. Some drivers like to drive slowly, while some others like to drive aggressively. Second, the road occupancy may also affect the traffic speed. On an empty freeway, drivers are free to choose their own driving speed. On

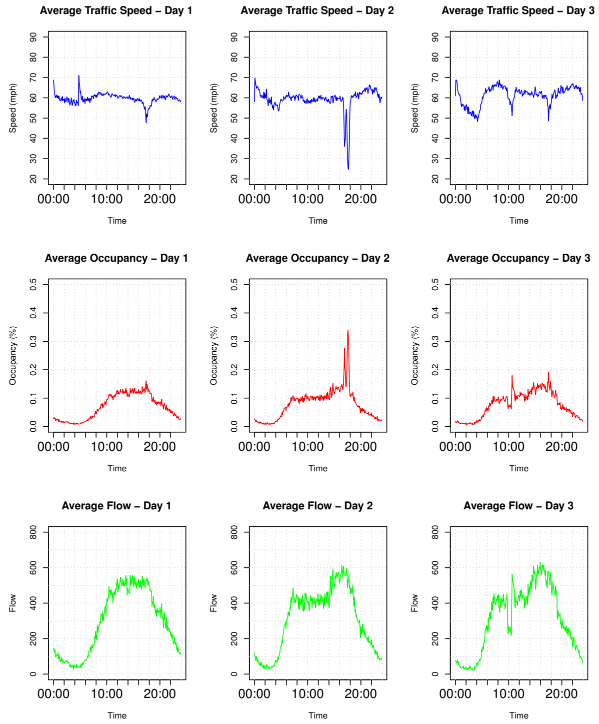


Figure 7: The speed, occupancy, and flow information on different days

the other hand, on an occupied freeway, drivers are forced to reduce their speed to some degree.

In Figure 7, we can see the correlation between speed, occupancy, and flow. When the occupancy is fluctuating, the speed is also significantly dropping (see the second day). We can also see that when the occupancy is sigificantly rising, the flow is also rising, although it rises more slowly. However, at the third day, we can see that the flow also hits the same amount around the same time with the second day. This time, the speed does not drop as much as what we observe on the second day. The occupancy does not rise as significantly either. Based on the above observation, we can say that when occupancy rises, the speed is very likely to decrease, especially when the occupancy reaches its limit on a road. On the other hand, we can have a high traffic flow without having a high occupancy. This means that the distribution of the vehicles is almost uniform and there is no road section with too many vehicles crammed together which cause traffic jam.

3. PROPOSED METHOD

We introduce Gaussian process, the base model that shall be used to deal with various patterns in traffic data via several model variants.

3.1 Gaussian Process Regression

Gaussian process is a non-parametric model as the model can adjust by itself when the input data and the number of the input data change. Gaussian Process Regression (GPR) is a probabilistic model that can solve regression problem under the Gaussian process framework. A Gaussian process [11] is fully specified by its mean function $m(\mathbf{x})$ and covariance function $k(\mathbf{x}, \mathbf{x}')$, where $\mathbf{x}, \mathbf{x}'$ denote the data feature vectors. Given a historical dataset $\mathcal{D} \equiv \{\mathbf{d}_i = (\mathbf{x}_i, y_i) : i = 1, 2, \ldots\}$, where $\mathbf{x}_i$ is a data feature vector and y_i is its corresponding value, the objective of GPR is to find a function f which can describe the relationship between $\mathbf{x}_i$ and y_i in high accuracy.

Formally, a Gaussian process can be formulated as:

$$f(\mathbf{x}) \sim GP(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')) \,, \tag{1}$$

with the undecided mean function $m(\mathbf{x})$ and covariance function $k(\mathbf{x}, \mathbf{x}')$. To consider measurement errors from the sensor data, we need to allow some small difference between the true value and the observed data in the regression problem. We choose a Gaussian noise model, which is written as:

$$y = f(\mathbf{x}) + \epsilon, \quad \epsilon \sim \mathcal{N}(0, \sigma_n^2) \,, \tag{2}$$

where σ^2 denotes the noise variance.

One of the key components in Gaussian process is its covariance (kernel) function k, which defines the pairwise relation between two feature vectors $\mathbf{x}$ and $\mathbf{x}'$. In this work, we deal with various kinds of data features including *occupancy*, *flow*, *time*, and *location*. To describe the relationship between two data points, we take one of the most common approaches which is the *squared exponential* function, written as follows:

$$\begin{aligned} k(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp\left(-\frac{1}{2}(\mathbf{x} - \mathbf{x}')^T \mathbf{\Lambda}^{-1} (\mathbf{x} - \mathbf{x}')\right) \\ + \sigma_n^2 \delta(\mathbf{x}, \mathbf{x}') \,, \end{aligned} \tag{3}$$

in which $\delta(\mathbf{x}, \mathbf{x}')$ is the Kronecker delta function, and $\mathbf{\Lambda} = \mathrm{diag}(\lambda_1^2, \lambda_2^2, \ldots)$ is a (diagonal) matrix to control the individual length-scale of each input variable given to the model. Since the data features are heterogenous, we use a variant of squared exponential function which also includes Automatic Relevance Determination (ARD) (to be discussed later) to decide the parameters in $\mathbf{\Lambda}$.

In traffic speed forecasting, given the current traffic conditions $\mathbf{x}_*$ for prediction, the current traffic speed y_* is predicted (probabilistically) by GPR as:

$$\begin{aligned} &y_* \mid \mathbf{x}_*, \mathbf{X}, Y \sim GP(k(\mathbf{x}_*, \mathbf{X}) \left[k(\mathbf{X}, \mathbf{X}) + \sigma_n^2 I\right]^{-1} Y, \\ &k(\mathbf{x}_*, \mathbf{x}_*) - k(\mathbf{x}_*, \mathbf{X}) \left[k(\mathbf{X}, \mathbf{X}) + \sigma_n^2 I\right]^{-1} k(\mathbf{X}, \mathbf{x}_*)) \,, \end{aligned} \tag{4}$$

where $\mathbf{X}$ and Y contain all the historical data including the feature vectors $\mathbf{X}$ and their corresponding values Y. The first part in Eq. 4 is the mean function which gives the traffic speed prediction and the second part provides the prediction confidence.

Conjugate Gradient Method

Conjugate Gradient method [13] is an iterative method for solving sparse linear systems. In GPR, both the mean function and covariance function have their respective adjustable parameters, which we call the hyperparameters θ. Choosing the best hyperparameters is one of the challenges introduced in GPR. However, it is very difficult to guess what hyperparameters that may offer the best result. We assume that by fitting the model into the training data with a good choice of the hyperparameters, we can have the best prediction result following the *maximum likelihood* principle. The Gaussian process log likelihood function is defined as:

$$\begin{aligned} \log P(f(\mathbf{x}) \mid \theta, \mathbf{x}) = -\frac{1}{2} f(\mathbf{x})^T k(\theta, \mathbf{x}, \mathbf{x}')^{-1} f(\mathbf{x}) \\ -\frac{1}{2} \log\det(k(\theta, \mathbf{x}, \mathbf{x}')) - \frac{|x|}{2} \log 2\pi \,. \end{aligned} \tag{5}$$

Maximizing this marginal likelihood towards θ fits the model into the data, and therefore completes the specification of the Gaussian process $f(\mathbf{x})$. We use *Conjugate Gradient* method in order to achieve the optimum.

3.2 Data Selection and Aggregation

The traffic data consists of data of a few features that are collected in different sensors and at different moments. The spatial and temporal relationships built in the data implies how we should take advantage of the relationships in the prediction task. That is also one of the reasons that we adopt Gaussian process as the key component for traffic state prediction as the process can naturally describe the relationships by choosing an appropriate kernel for pairwise data.

One data item in the traffic data can be written as

$$\mathbf{x} = (\mathbb{x}, s, t\} \,,$$

with the core features $\mathbb{x}$ and the information about the data collecting sensor $s \in S$ for sensor set S and data collecting time $t \in \{1, 2, \ldots\}$. The raw data are collected in a streaming format for every sensor. We usually operate some sampling (or smoothing) techniques on the original data for better efficiency and noise removal. Specifically, we aggregate every m data

$$\mathbf{x}_t, \mathbf{x}_{t+1}, \ldots, \mathbf{x}_{t+m-1} \,,$$

into one where $\mathbf{x}_{t+k} = (\mathbb{x}_{t+k}, s, t+k)$ which are all collected in a single sensor s for the moments from t to $t+m-1$. The aggregation is done via a simple averaging computation and the aggregated result is recorded as $\mathbf{x}_i = \mathbf{x}_i^t = (\mathbb{x}_i^t, s, t)$, a data feature set for a data item $\mathbf{d}_i$ in $\mathcal{D} \equiv \{\mathbf{d}_i = (\mathbf{x}_i, y_i)\}$, where $\mathbb{x}_i^t$ is obtained from averaging the items in the set $\mathbb{x}_t, \mathbb{x}_{t+1}, \ldots, \mathbb{x}_{t+m-1}$. We repeat the aggregation for n steps until we have n records of data for every sensor. That is the training base for the prediction on a single sensor. We can surely consider the training given several sensors at the same time.

3.3 Model Building

When we observe a time series data like traffic data, we might see more than just one periodic pattern. Some say traffic jam always happens right before evening. Some say it happens only at Thursday and Friday evenings. In fact, both assumptions might be true to some degree. Rather than just considering only one, we try to consider the possibility of having multiple periodic patterns in our approach. More than the consideration on multi-periodicities, we also take into account the short-term patterns as well as the spatial and feature correlations in the final traffic state prediction.

Regarding to the study on multi-periodicities, we basically consider two kinds of approaches when we have enough belief to say that the analyzed data may consist of more than one periodicity. The first approach is to build multiple models, with each model has its own periodicity. The output from these models will then be combined to form the final result. We call this model *Composite-Periodicity* model or

CP model, which is made from multiple single-periodicity model. The second approach is to include all periodic patterns at once in one model. We call this *Multi-Periodicity* model or MP model, further discussed below.

We design a framework which allows traffic forecasting in a streaming fashion using GPR (Figure 1). In our method, we use one Gaussian process each time we do one prediction. Given the Gaussian process computation barrier, we have to carefully select our training data and the time for model retraining. In this framework, we update the model once a while. When a new data comes, we check first whether our current model has expired. If the model expires, it will be updated with a new training data. If the model has yet to expire, it will be reused for forecasting. By reusing the model this way, we can save some time from having to rebuild the model too frequently.

Single-Periodicity Model

First, we define a period p in which it is assumed to have a periodic or repetitive pattern on given data. For example, $p =$ *one day* means that any pattern that happens on one day will be (almost) repeated in another one day, up to noises or measurement errors. We work on a modified time t_p as a feature for learning given a periodicity p.

For data time index $T = \{t_1, t_2, \ldots\}$ and a given periodicity p, we can define an equivalence relation $\sim_p$ on T where two indices t_j and t_k are equivalent, written as $t_j \sim_p t_k$ if $t_j \equiv t_k \bmod p$. Given the relation, the *equivalence classes* or *quotient space* of T by $\sim_p$, denoted by $T/\sim_p$ include the indices with different remainders if divided by p, written as $T/\sim_p \equiv \{0, 1, 2, \ldots, p-1\}$. In reality, if we can choose the minimum unit in the time domain as one minute, then $p = 1440$ for one-day periodicity. In this study, we chose the minimum unit to be five minutes and $p = 1440/5 = 288$ for one-day periodicity.

The index drawn from the quotient space will be added into the feature set as an additional time index feature according to a given periodicity p. Given a pre-decided periodicity p, the feature set $\mathbf{x} = (\not{x}, s, t_p)$ for the single-periodicity model has the time information t_p drawn from the quotient space $T/\sim_p$. The result from GPR based on the single-periodicity model is given by:

$$y = f(\mathbf{x} = \{\not{x}, s, t_p\}) \sim GP(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')) \,,$$

for $t_p \in T/\sim_p$. We can also work on a slightly different feature set as:

$$\mathbf{x} = (\not{x}, s, t, t_p) \,,$$

where t is the original time stamp (without thrown away) and t_p is the time index drawn from the quotient space such as t could be the absolute time and t_p can be the time on a day for a daily-periodicity model for traffic forecasting. The difference between the two choices is that the former combines data of the same time on a day (but maybe with different days) together in statistics computation while the latter treat the data from the same time on a day but different days differently with a positive distance on time in the kernel computation.

Multi-Periodicity Model

We can discuss a complex case where we have multiple periodicities observed in a given data set. For given multiple periodicities $p_1, p_2, \ldots, p_\ell, \ldots$, we can define the quotient spaces $T/\sim_{p_1}, T/\sim_{p_2}, \ldots, T/\sim_{p_\ell}, \ldots$ and use them to describe the data by using a new feature set written as:

$$\mathbf{x} = (\not{x}, s, t_{p_1}, t_{p_2}, \ldots, t_{p_\ell}, \ldots\} \,,$$

where t_{p_ℓ} is drawn from the quotient space of periodicity p_ℓ and $\not{x}$ and s denote the core features and the sensor information respectively as before.

For traffic data, we can study the prediction problem under multiple periodicities such as daily, weekly or even yearly. To address those periodicities, we can have the data features written as:

$$\mathbf{x} = (\not{x}, s, t_{\text{daily}}, t_{\text{weekly}}, t_{\text{yearly}}) \,.$$

In this case, for instance, two features t_{daily}'s from two data should share the same value if two have the same time on a day (such as 4PM) even they are from different days. Likewise, the feature t_{weekly} should have the value ranged from all moments in a week, such as "8AM on Monday" or "4PM on Friday", etc.

Composite-Periodicity Model

We can consider an alternative approach to combine the information from multiple periodicities, in fact, a straightforward approach which computes the weighted average of the results from several single periodicity models, described as:

$$y = \sum_{p \in P} w_p f_p(\mathbf{x} = \{\not{x}, s, t_p\}) \,,$$

where $\sum_p w_p = 1$ and f_p is the regressor that is for the single periodicity t_p.

Adding Short-Term Data

For each of the above models, either with single periodicity or multiple periodicities, we can add a few most recent data to the training set for modeling. The augmented data set becomes:

$$\mathcal{D} \cup \mathcal{D}_{\text{short}} \equiv \{\mathbf{d}_i = (\mathbf{x}_i, y_i)\} \cup \{(\not{x}_{t-c}, s, t-c; y_{t-c}), \\ (\not{x}_{t-c+1}, s, t-c+1; y_{t-c+1}), \ldots, (\not{x}_{t-1}, s, t-1; y_{t-1})\} \,,$$

for a c period of time to be considered as the short-term data. As mentioned before, the data in $\mathcal{D}$ that are for periodicity analysis are called the long-term data to be distinguished from the short-term ones.

Mean Function

Gaussian process is fully specified by its mean and covariance functions. In this study, the mean function $m(\mathbf{x}) = m(\not{x}, s, t)$ for GPR is obtained by computing the average of all the historical data with the same time feature t (and sensor feature s) to be the value for the mean function. Note that t can be t_p with a periodicity p for a periodicity model.

Parameter Estimation

Following the data selection procedure in Subsection 3.2, we update the hyperparameters θ of the GPR kernel based on different choices of the training set. The hyperparameters $\theta \in \Theta$ are saved for each time index t_p given a periodicity p and will be reused for some adjustable period of time before they get expired and need to be updated. When a data for prediction has a time index t_p, we check whether in the Θ there is a set of hyperparameters with exact match of the time index. If any hyperparameters θ have the same time index t_p, the hyperparameters will be used. Otherwise, the last used hyperparameters right before this moment will be selected instead. For multi-periodicity models, we consider the shortest periodicity period to decide the hyperparameters.

4. EXPERIMENTS AND RESULTS

In this section, we present the experiment results for traffic state forecasting based on the proposed method. First, we introduce the dataset for traffic forecasting evaluation in Subsection 4.1 and afterwards, we explain the evaluation method. Finally, we demonstrate the experiment results in Subsection 4.2.

4.1 Dataset and Evaluation

We chose San Diego ITS sensor data from RDE dataset collections [12] for evaluation. We selected five sensors' data from I5N freeway to be experimented with, as shown in Figure 8. The data in the set were collected in the year 2010 with five-minute sampling rate. We also used the same sensor set's data to test our forecasting result. As explained before, we aim to predict the next 30-minutes traffic speed starting from November 10, 2010 until November 23, 2010. To prepare the long-term data, we do hourly aggregation to the data. The processed dataset consists of features listed as follows:

- The Core features (denoted by $\mathbb{x}$)
  - Flow: Number of vehicles count
  - Occupancy: How long the sensor is occupied by vehicles (in percentage)
- Other Features
  - Sensor information: Sensor's coordinates (in latitude and longitude) or simply sensor index, denoted by s.
  - Time information: Including the time on a day t_{daily} and the time in a week t_{weekly}, for daily and weekly periodicities respectively.

After all, the traffic speed is the target value we aim to predict. In most experiments, we find out that the contribution of adding the flow information is limited (also check Figure 6) and decide to exclude the feature in most of the experiments.

In all experiments, we evalute the proposed method by calculating the Mean Absolute Error (MAE) and Mean Absolute Percentage Error (MAPE).

4.2 Results

We take turn to investigate the performance for various models which include single-periodicity (SP) model, multi-periodicity (MP) model, the models with short-term data, and the models with other considerations such as adding spatial information and adding other related features for traffic forecasting.

Single-Periodicity Model.

Similar moments of the day may have similar traffic conditions even they are from different days; also, similar days of the week may share similar traffic patterns even they are from different weeks. Considering the daily and weekly periodicities in the SP model may show better performance

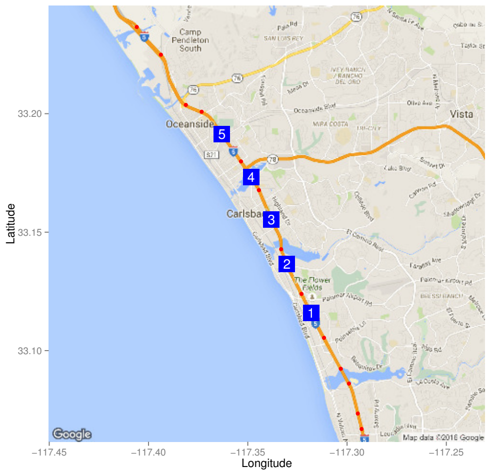


Figure 8: The sensors located in I5N where we use the data collected in them to evaluate the proposed method.

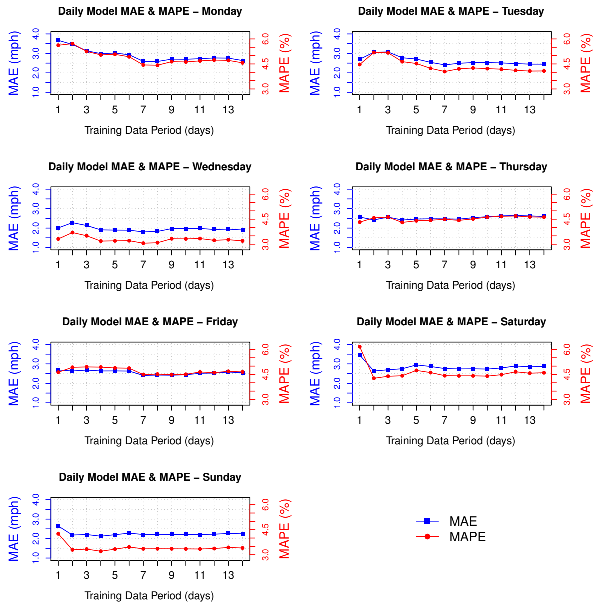


Figure 9: The prediction error of SP model with the daily periodicity for seven different days of the week.

than the model without such considerations. In this series of experiments, we use previous several days and weeks for training to see how the single-periodicity models perform with the daily and weekly periodicities.

To speak of the SP model with *daily* periodicity, we discuss the model given different numbers of days for training. Given previous one up to 14 days for training, the performance on the SP model with the additional feature t_{daily} is in Figure 9. As we can observe, the more days for training, the better the model performance is. More specifically, we need about two days for training to reach converged performance during the weekend and need about four to seven days to reach converged performance for the weekdays.

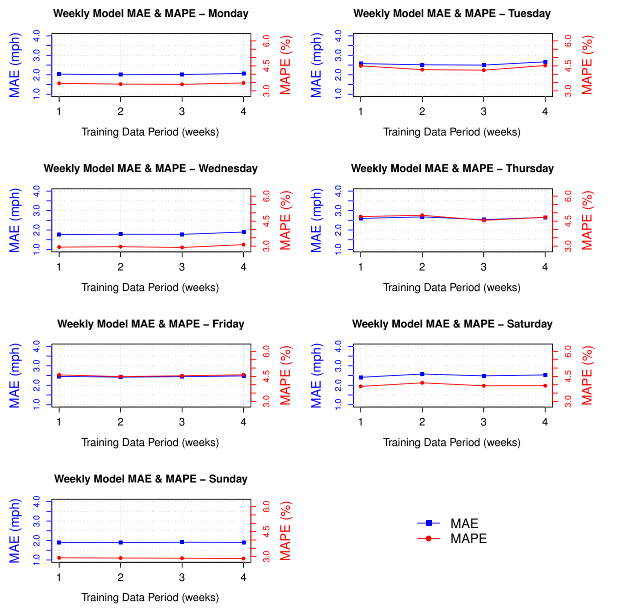
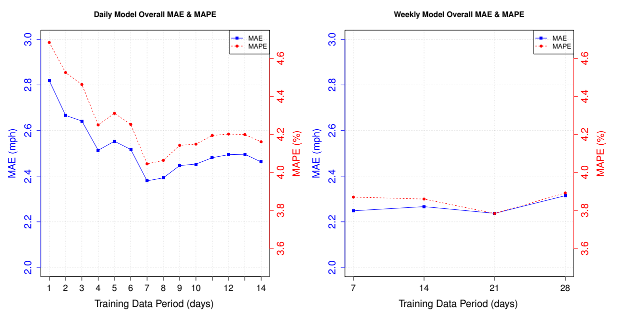


Figure 10: The prediction error of SP model with the weekly periodicity for seven different days of the week.

To speak of the SP model with *weekly* periodicity, we work on the model give different numbers of weeks for training. We experiment the model with previous one to four weeks of training and the results are shown in Figure 10. The additional feature t_{weekly} to be added to the feature set, i.e., the time of the week is a time index which starts from Monday 0:00 and incrementally increasing up to before Sunday 23:59.



(a) (b)

Figure 11: Overall prediction error with different data periods: (a) daily model, (b) weekly model.

We compare the two SP models' overall forecasting performance in Figure 11, with the daily and weekly periodicities. We can see that the SP weekly model prediction is superior compared to the SP daily model, even with the same amount of training data. This implies that considering weekly periodicity is more helpful than considering daily periodicity. Note that we have a clear error drop when we collect the data of seven days for training. It is due to that having a seven-day data, we collect another day of data of similar patterns to the current traffic conditions.

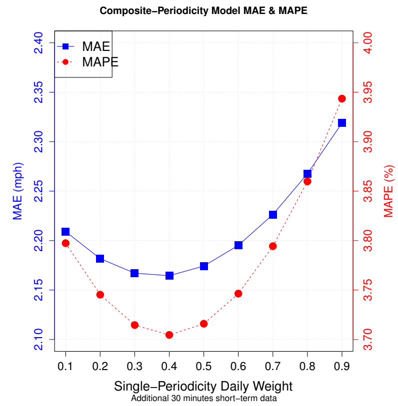


Figure 12: The MAE & MAPE on the CP Model with the daily and weekly periodicities.

Multi-Periodicity Model.

Table 1: The performance (MAE and MAPE) on various SP, MP and CP models

Model	MAE (mph)	MAPE (%)
SP (daily)	2.38	4.04
SP (weekly)	2.25	3.86
MP (daily-weekly)	**2.11**	**3.61**
CP (daily and weekly)	2.16	3.70

The MP model considers multiple periodic patterns at once. In this study, to make use of both daily and weekly repetitive patterns, we include two time indices in the model: the time on a day t_{daily} and the time in a week t_{weekly}. We experiment using the data for previous seven days and also for two neighboring upstream and downstream sensors in the model training. The overall (average) performance is shown in Table 1. Based on the results, we can conclude that the SP model with weekly periodicity works better than the SP model with daily periodicity; also, the MP model with daily and weekly periodicities works the best among all of them, even if compared to the CP model with daily and weekly periodicities (to be discussed next).

Composite-Periodicity Model.

We can simply combine the two single-periodicity models by weighted average to enhance the prediction performance. In Figure 12 we see the results from the two SP models with daily and weekly periodicities combined together with different weights. First, we can see that by combining multiple SP models using weighted average, we can achieve a better result. To choose the best weight, we test different weights on the SP model with daily periodicity from 0.1 to 0.9 and find out that the best weight is close to 0.4 (and the weight for the SP weekly model is 0.6). However, the model's performance is not as good as the MP model's which considers both daily and weekly periodicities.

Models with Short-Term Data.

In this experiment, we add some short-term data into various models to see the result. The short-term data are directly included and be combined with the long-term data for

Table 2: The MAE and MAPE of the MP (daily-weekly) model with short-term data.

Data Period	MAE (mph)	MAPE (%)	Time (s)
None	2.056584	3.52	0.033
10 minutes	1.458071	2.54	0.036
20 minutes	1.458075	2.54	0.036
30 minutes	1.410530	2.47	0.037
1 hour	1.410602	2.47	0.037
2 hours	1.410663	2.47	0.037

Table 3: All experiment results for various SP, MP and CP models.

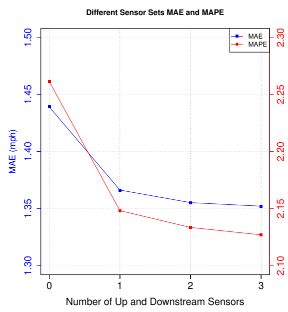
Model		**MAE** (mph)	**MAPE** (%)
SP	daily (long-term)	2.38	4.04
	daily + short-term	1.77	3.05
	weekly (long-term)	2.25	3.86
	weekly + short-term	1.70	3.01
MP	long-term	2.11	3.61
	long + short-term	**1.41**	**2.47**
CP	daily + weekly (long-term)	2.16	3.70
	long + short-term	1.65	2.88

model training. We try adding up to two hours of short-term data and see the performance.

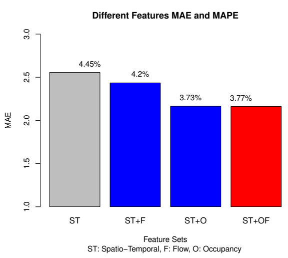
From Table 2, we can see that including short-term data has a big significance in the forecasting performance. However, adding more than 30 minutes of data no longer increases the accuracy. We can assume that the short-term data for more than 30 minutes ago is not very relevant anymore to the current traffic conditions. To summarize all the results, we refer to Table 3. The MP model again works the best among all. Especially, the MAE for MP (daily-weekly) is decreased from 2.11 to 1.41 if adding the short-term data into the training.

Spatial Considerations.

The main part of this work is to focus on the study of multiple periodicities for traffic data. Still, we would like to explore the performance if we collect the information from more than one sensor. In this series of experiments, we want to observe the effect of using multiple neighboring sensors



(a)



(b)

Figure 13: (a) The forecasting errors with: (a) different sensor sets and (b) different feature sets.

Table 4: Linear correlation (Pearson's product-moment correlation coefficient)

	Prev. Speed	**Occupancy**	**Flow**
Current Speed	0.975	-0.559	-0.157

compared to using only one sensor. In the local view scenario, we only use the past information from a sensor s to forecast the traffic speed at the same sensor s for a future moment. For the global view, we forecast traffic speed at sensor s using the information from several neighboring sensors from s such as $\{\cdots, s-2, s-1, s+1, s+2, \cdots\}$ as well.

In previous experiments, we basically set the features set to be: $(\cancel{x}, s, \mathbf{t}_{\text{periodicity}})$ for the time index vector given by $\mathbf{t}_{\text{periodicity}}$ for multiple periodicities. Now the same features are collected from each of the sensors if the sensor is considered a neighbor of the sensor s. For evaluation, we set the training period to be seven days and use the daily SP model as the main method for comparison. The Sensor No. 4 (in Figure 8) shall be used to test how effective the proposed model is. We can observe the experiment result in Figure 13(a). Using one up to two upstream and downstream sensors can significantly improve the prediction result.

Feature Correlations.

The performance of traffic speed prediction can be further enhanced by knowing some other relevant features' information. In this subsection, we investigate the correlation between occupancy and speed, and also between flow and speed to decide the best feature set for speed forecasting. We set the model to be the MP (daily-weekly) model and use seven days for model training.

What we want to know is whether having occupancy and flow also aids us in the forecasting. In Table 4, we can see that the occupancy data will somehow be useful in the forecasting, and the flow data may not be that useful due to the weak correlation between the flow and speed data. Figure 13(b) and Figure 14 show the experiment results with different feature sets. From the experiment results we can say that including occupancy information does help the speed forecasting. However, adding one more feature to the set, the flow information may not improve the forecasting result. This might be caused by the weak correlation between flow and speed. Note that we have a very strong correlation between the previous speed and the current speed. It suggests that it is appropriate to use only the historical speed data as well as the spatial and temporal indices for speed forecasting. That is, we can expect good results from a process model, such as a Gaussian process without the occupancy and flow information for traffic speed forecasting.

5. CONCLUSIONS

In this work, we proposed a general framework for traffic speed forecasting. In the framework, several models have been demonstrated to show the contribution on traffic forecasting. We focus on the study of multiple periodicities that could be seen in the traffic data, a streaming data with temporal relationships. Based on our study, the multi-periodicity model works better than the single-periodicity and the composite-periodicity models. We have tested the idea with a daily-weekly multi-periodicity model. The data

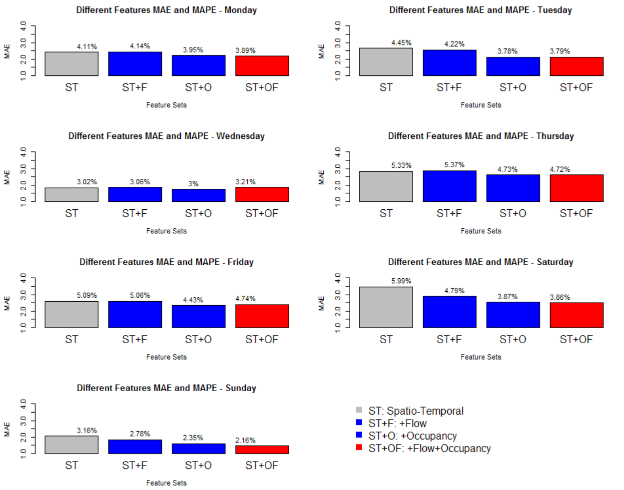


Figure 14: Prediction error with different feature sets and different days in a week.

with multiple periodicities is under a long-term consideration, which also needs to add a short-term data to have the best performance for speed forecasting as the traffic speed is often highly dynamic. Other than the temporal relationships, we consider the spatial relationship, also, the feature correlations in the data for speed forecasting to further improve the result. In the end, the proposed model outputs a peak performance for its MAE equal to one to two mph in the error, which is good enough for real applications.

6. ACKNOWLEDGEMENTS

Research partially supported by the Ministry of Science and Technology Grant MOST 104-2221-E-011-165 and also supported in part by the Ministry of Science and Technology, National Taiwan University, and Intel Corporation under Grants MOST 105-2633-E-002-001 and NTU-ICRP-105R104045.

7. REFERENCES

[1] P. J. Bickel, C. Chen, J. Kwon, J. Rice, E. van Zwet, and P. Varaiya. Measuring traffic. *Statist. Sci.*, 22(4):581–597, 11 2007.

[2] C. Chen, J. Hu, Q. Meng, and Y. Zhang. Short-time traffic flow prediction with ARIMA-GARCH model. In *Intelligent Vehicles Symposium (IV), 2011 IEEE*, pages 607–612, June 2011.

[3] J. Chen, K. H. Low, and C. K.-Y. Tan. Gaussian process-based decentralized data fusion and active sensing for mobility-on-demand system. In *Proc. Robotics: Science and Systems Conference*, 2013.

[4] X.-Y. Chen, H.-K. Pao, and Y.-J. Lee. Efficient traffic speed forecasting based on massive heterogenous historical data. In *2014 IEEE International Conference on Big Data (Big Data), Workshop on Large Data Analytics in Transportation Engineering*, pages 10–17, 2014.

[5] J. Dauwels, A. Aslam, M. T. Asif, X. Zhao, N. Mitrovic, A. Cichocki, and P. Jaillet. Predicting traffic speed in urban transportation subnetworks for multiple horizons. In *Control Automation Robotics Vision (ICARCV), 2014 13th International Conference on*, pages 547–552, Dec 2014.

[6] B. Ghosh, B. Basu, and M. OâĂŹMahony. Bayesian time-series model for short-term traffic flow forecasting. *Journal of Transportation Engineering*, 133(3):180–189, 2007.

[7] J. Guo and B. M. Williams. Real-time short-term traffic speed level forecasting and uncertainty quantification using layered kalman filters. *Transportation Research Record*, 2175:28–37, 2010.

[8] M. M. Hamed, H. R. Al-Masaeid, and Z. M. B. Said. Short-term prediction of traffic volume in urban arterials. *Journal of Transportation Engineering*, 121(3):249–254, 1995.

[9] INRIX. private conversations, 2015.

[10] Y. Lv, Y. Duan, W. Kang, Z. Li, and F. Y. Wang. Traffic flow prediction with big data: A deep learning approach. *IEEE Transactions on Intelligent Transportation Systems*, 16(2):865–873, April 2015.

[11] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2006.

[12] Research Data Exchange. `https://www.its-rde.net/home`.

[13] J. R. Shewchuk. An introduction to the conjugate gradient method without the agonizing pain. Technical report, Carnegie Mellon University, Pittsburgh, PA, USA, 1994.

[14] E. I. Vlahogianni, M. G. Karlaftis, and J. C. Golias. Optimized and meta-optimized neural networks for short-term traffic flow prediction: A genetic approach. *Transportation Research Part C: Emerging Technologies*, 13(3):211–234, 2005.

[15] J. Wang and Q. Shi. Short-term traffic speed forecasting hybrid model based on chaos-wavelet analysis-support vector machine theory. *Transportation Research Part C: Emerging Technologies*, 27:219–232, 2013.

[16] C. H. Wu, J. M. Ho, and D. T. Lee. Travel-time prediction with support vector regression. *IEEE Transactions on Intelligent Transportation Systems*, 5(4):276–281, 2004.

[17] H. Xiao and C. Eckert. Lazy Gaussian process committee for real-time online regression. *Proceedings of the 27th AAAI Conference on Artificial Intelligence (AAAI)*, pages 969–976, 2013.

[18] Y. Xie, Y. Zhang, and Z. Ye. Short-term traffic volume forecasting using Kalman filter with discrete wavelet decomposition. *Computer-Aided Civil and Infrastructure Engineering*, 22(5):326–334, 2007.

[19] Y. Xu, Q. Kong, and Y. Liu. Short-term traffic volume prediction using classification and regression trees. In *Intelligent Vehicles Symposium (IV), 2013 IEEE*, pages 493–498, 2013.

[20] R. Yasdi. Prediction of road traffic using a neural network approach. *Neural Computing & Applications*, 8(2):135–142, 1999.

[21] H. Yin, S. C. Wong, J. Xu, and C. K. Wong. Urban traffic flow prediction using a fuzzy-neural approach. *Transportation Research Part C: Emerging Technologies*, 10(2):85–98, 2002.