This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2016.2585575, IEEE Transactions on Vehicular Technology

Improving Traffic Flow Prediction With Weather Information in Connected Cars: A Deep Learning Approach

Arief Koesdwiady, *Student Member, IEEE*, Ridha Soua, and Fakhri Karray, *Senior Member, IEEE*

***Abstract*—Development of Intelligent Transportation Systems (ITS) is fast becoming a crucial component in the design of future smart cities given major requirement for enacting super-efficient navigation. Transportation systems might be heavily affected by factors such as accidents and weather. Specifically, inclement weather conditions may have drastic impact on travel time and traffic flow. The current study has two objectives: first, to investigate a correlation between weather parameters and traffic flow, and to improve traffic flow prediction by proposing a novel holistic architecture. It incorporates: 1) deep belief networks for traffic and weather prediction and, 2) decision-level data fusion scheme to enhance prediction accuracy using weather conditions. The experimental results, using traffic and weather data originated from San Francisco Bay Area of California, corroborate the effectiveness of the proposed approach compared to the state of the art.**

***Index Terms*—Intelligent transportation systems, traffic prediction, weather information, deep learning, data fusion.**

I. Introduction

Presently, our cities suffer from ever increasing population growth, pollution leading to pressure on existing transportation infrastructure. The situation will be even more critical in future. Recent statistics indicate that 60% of the world population will be living in cities by 2050 [1]. With more than a billion cars on the roads today that is expected to double to around 2.5 billion by 2050 [2], designing super-efficient navigation and safer travel journey are becoming a major challenges for transportation authorities. The development of Intelligent Transportation Systems (ITS) is a cornerstone in the design and implementation of smart cities. Indeed, building more roads will not radically solve the problem of large traffic congestion, fuel consumption, longer travel delays and safety.

Connected cars evolve in a data-rich environment where they consistently generate and receive a variety of data that pours in from everywhere: weather, roads, traffic and social networks streams. The abstraction level of data ranges from raw low-level sensor data to higher-level information transmitted to connected cars and people for high quality predictive decision making. An interesting example is the city of Eindhoven where authorities have collaborated with IBM to design a traffic management solution that merges data coming from in-vehicles sensors with traffic gathered from roads [3]. Data is not only stored and processed but used for predicting and making decision in a timely and accurate way.

Arief Koesdwiady, Ridha Soua, and Fakhri Karray are with the Centre for Pattern Analysis and Machine Intelligence, University of Waterloo, ON, Canada (corresponding author e-mail: abkoesdw@uwaterloo.ca)

Traffic flow prediction is one of the major functions of ITS that should be achieved. It is particularly useful for planning proactive operations to alleviate road congestion. Moreover, timely and accurate prediction offers guidance to drivers so they can better choose the ideal departure time and best routes to avoid gridlock. However, traffic prediction is very challenging. Drivers interact with surrounding connected things (infrastructure, traffic light, cars) and with each other. During their trip, they are exposed also to changing weather conditions. Favorable or inclement weather will influence drastically drivers in roads. Statistics made by the Federal Highway Administration (FHWA) reveal that 23% of road crashes are due to adverse weather conditions [4]. Hence, it is crucial to investigate the impact of weather on traffic prediction. Quantifying such an impact will help transportation operators and road users in terms of better pre-trip planning and best navigation strategies that may match weather conditions.

Motivated by the close relationship between traffic on roads and weather conditions, this paper proposes a deep learning novel approach that predicts traffic flow by fusing the big data provided by traffic history and weather data. First, we study the road traffic auto-correlation to determine the required number of previous steps of traffic flow needed to achieve high prediction accuracy. Then, we investigate the cross-correlation between traffic measurements and weather conditions at various granularity. The purpose is to select the most influencing weather variables. Moreover, we propose a comprehensive prediction architecture that incorporates Deep Belief Networks (DBNs) and data fusion to derive more accurate road traffic prediction. The decisions obtained by DBNs about traffic will be fused together at decision-level to provide better decisions regarding traffic flow prediction accuracy. Our approach will be compared to different levels and techniques of data fusion.

The rest of the paper is organized as follows. Section II describes the current literature on traffic prediction. Then, we detail our traffic prediction approach based on deep learning and data fusion that incorporate traffic history and weather information. Section VI describes our experimental results and findings. We conclude the paper in Section VII.

II. State of the art

An efficient ITS should be capable of providing road users with continuous pertinent information regarding the evolution

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2016.2585575, IEEE Transactions on Vehicular Technology

of traffic parameters (speed, flow, density) over time. This is crucial for drivers to plan their trip or for trip rescheduling purpose. Traffic flow prediction approaches can be mainly classified in three categories:1) parametric 2) non parametric and 3) hybrid approaches [5].

Main techniques used in the first class are autoregressive integrated moving average based (ARIMA) models [6], [7] [8] [9], Kalman filtering [10]–[13]. However, the linearity of time series approach presents an inconvenience for traffic prediction. Traffic flow has stochastic and nonlinear nature. In addition, these techniques predict traffic on each road separately, for example ARIMA. Since transportation networks are complex and very correlated, it is crucial to predict traffic flow from a network perspective. Thus, time series based approaches are more prone to large errors in traffic forecasting.

Non-parametric regression [14], [15] is a widely used technique. In [15], authors propose an online boosting regression technique that ensures traffic prediction under abnormal traffic conditions. Otherwise, boosting is disabled. In [16], a Support Vector Regression (SVR) was used to establish the prediction model while Particle Swarm optimization was used to optimize the model's parameters. A panoply of Artificial Neural Networks (ANNs) were proposed to predict traffic flow [17], [18]. However, ANNs training algorithms suffer from the problem of local minima. Moreover, ANNs use usually one hidden layer. Simulations have shown that one hidden layer would not be enough to describe the complicated relationship between inputs and outputs of the prediction model.

To handle the rigidity of time series and non-parametric models when complex nonlinear traffic states are present, some works have investigated hybrid approaches [19]–[22]. By combining several techniques, these approaches are more adaptive. Although the above mentioned hybrid models are flexible, they do not fully take profit from spatial information collected from the whole road network. Moreover, these studies rely crucially on only information collected by sensors such as GPS, loop detectors and smart-phones. Few works tackled the problem of analyzing the tight correlation between traffic data and external factors such as weather [23]–[25]. Nevertheless, authors of [24], [25] used linear regression, which is a parametric model, to forecast traffic.

Deep learning has attracted researchers from various domains [26]–[28].This was motivated by the fact that other techniques require a prior knowledge of specific domains for feature extraction and selection. Deep learning could learn features with a less prior knowledge. The successful use of deep learning has motivated researchers in the area of traffic flow prediction [29], [30]. However, these recent studies did not take into account weather factors whose their significant impact on traffic flow is widely acknowledged [31], [32].

III. Technical Background

In this section, we detail the core components of our approach that are Deep Belief Networks (DBNs), Multi-Task Learning (MTL) and decision-level data fusion.

A. MTL in DBNs

DBNs are formed by stacking Restricted Bolztmann Machines (RBMs), which are then trained greedily using unsupervised algorithms [33]. An RBM is an undirected graphical model that has no connections within visible ($\mathbf{x}$) or hidden ($\mathbf{h}$) units, as illustrated in Figure 1.

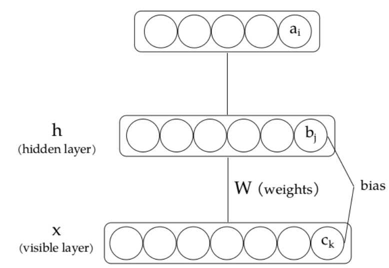


Fig. 1. Restricted Boltzmann Machines

RBMs are developed based on the inter-layer interaction energy and bias energy of each layer. Mathematically, the energy function of an RBM is defined as:

$$E(\mathbf{x},\mathbf{h}) = -\mathbf{b}^\top\mathbf{x} - \mathbf{c}^\top\mathbf{h} - \mathbf{h}^\top\mathbf{W}\mathbf{x} \tag{1}$$

where $\mathbf{b}$ and $\mathbf{c}$ are bias vectors associated with the hidden and visible layers respectively, and $\mathbf{W}$ is the weights between the visible and hidden layers. We denote the set of parameters $\mathbf{b},\mathbf{c},$ and $\mathbf{W}$ by θ. An RBM defines a distribution, which involves the hidden units, over the visible units. The distribution of observing a particular visible and hidden units configuration based on the energy function is given by:

$$P(\mathbf{x},\mathbf{h}) = \frac{e^{-E(\mathbf{x},\mathbf{h})}}{Z} \tag{2}$$

where Z is the partition function. Using Equation 2, the distribution of observing a set of visible units is defined as:

$$P(\mathbf{x}) = \sum_{\mathbf{H}} P(\mathbf{x},\mathbf{h}) = \exp\frac{-F(\mathbf{x})}{Z} \tag{3}$$

where $F(X)$ is known as the free energy term, and is defined as follows:

$$F(\mathbf{x}) = \exp\left(\mathbf{c}^\top\mathbf{x} + \sum_{j=1}^{H}\log(1+\exp(b_j + \mathbf{W}_{(j,:)}\mathbf{x}))\right)/Z \tag{4}$$

To train an RBM, we need to minimize the average negative log-likehood (NLL) function of all training points ($t = 1, ..., T$). The NNL function is set as follows:

$$NLL = \frac{1}{T}\sum_t l(f(\mathbf{x}^{(t)})) = \frac{1}{T}\sum_t -\log P(\mathbf{x}^{(t)}) \tag{5}$$

Then, the NNL function is optimized with respect to θ using stochastic gradient descent algorithm; the result of the optimization is defined as:

$$\begin{aligned}\frac{\partial -\log P(\mathbf{x}^{(t)})}{\partial\theta} &= \mathbb{E}_{\mathbf{h}}\left[\frac{\partial E(\mathbf{x}^{(t)},\mathbf{h})}{\partial\theta}\middle|\mathbf{x}^{(t)}\right] \\ &\quad - \mathbb{E}_{\mathbf{x},\mathbf{h}}\left[\frac{\partial E(\mathbf{x},\mathbf{h})}{\partial\theta}\right]\end{aligned} \tag{6}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2016.2585575, IEEE Transactions on Vehicular Technology

The second term in the right-hand side of the equation is hard to compute since we have to make an exponential sum over both $\mathbf{x}$ and $\mathbf{h}$. To address this problem, we use the contrastive divergence algorithm proposed by [34].

To perform the Gibbs sampling, the conditional distributions $P(\mathbf{h}|\mathbf{x})$ and $P(\mathbf{x}|\mathbf{h})$ have to be computed according to Equation 7.

$$\begin{aligned} P(\mathbf{h}|\mathbf{x}) &= \prod_i P(h_i|\mathbf{x}) \\ P(\mathbf{x}|\mathbf{h}) &= \prod_k P(x_k|\mathbf{h}) \end{aligned} \tag{7}$$

If $\mathbf{x}_k$ and $\mathbf{h}_j$ are binary units, the following applies

$$\begin{aligned} P(h_j = 1|\mathbf{x}) &= \frac{1}{1+\exp(-(b_j + \mathbf{W}_{(j,:)}\mathbf{x}))} \\ &= \mathrm{sigm}(b_j + \mathbf{W}_{(j,:)}\mathbf{x}) \\ P(x_k = 1|\mathbf{h}) &= \frac{1}{1+\exp(-(c_k + \mathbf{h}^\top \mathbf{W}_{(:,k)}))} \\ &= \mathrm{sigm}(c_k + \mathbf{h}^\top \mathbf{W}_{(:,k)}) \end{aligned} \tag{8}$$

After the negative sample $\mathbf{x}(k) = \tilde{\mathbf{x}}$ is estimated, the point estimate of the expectation in Equation 6 is computed as follows:

$$\begin{aligned} \mathbb{E}_{\mathbf{h}}\left[\frac{\partial E(\mathbf{x}^{(t)},\mathbf{h})}{\partial\theta}\middle|\mathbf{x}^{(t)}\right] &\approx \frac{\partial E(\mathbf{x}^{(t)},\tilde{\mathbf{h}})}{\partial\theta} \\ \mathbb{E}_{\mathbf{x},\mathbf{h}}\left[\frac{\partial E(\mathbf{x},\mathbf{h})}{\partial\theta}\right] &\approx \frac{\partial E(\tilde{\mathbf{x}},\tilde{\mathbf{h}})}{\partial\theta} \end{aligned} \tag{9}$$

Even with only one step of sampling, i.e., $k = 1$, this algorithm has been proven to be successful for unsupervised training [34]. Using Equation 9, the parameter θ of the RBMs are updated according to the following equation:

$$\theta^{(t+1)} = \theta^{(t)} - \alpha\left(\nabla_\theta(-\log P(\mathbf{x}^{(t)}))\right) \tag{10}$$

For each parameter $\mathbf{W}$, $\mathbf{b}$, and $\mathbf{c}$, the update equations are given below:

$$\begin{aligned} \mathbf{W}^{(t+1)} &= \mathbf{W}^{(t)} - \alpha\left(\mathbf{h}(\mathbf{x}^{(t)})\mathbf{x}^{(t)^\top} - \mathbf{h}(\tilde{\mathbf{x}})\tilde{\mathbf{x}}^\top\right) \\ \mathbf{b}^{(t+1)} &= \mathbf{b}^{(t)} - \alpha\left(\mathbf{h}(\mathbf{x}^{(t)}) - \mathbf{h}(\tilde{\mathbf{x}})\right) \\ \mathbf{c}^{(t+1)} &= \mathbf{c}^{(t)} - \alpha\left(\mathbf{x}^{(t)} - \tilde{\mathbf{x}}\right) \end{aligned} \tag{11}$$

where $\mathbf{h}(\mathbf{x}) \triangleq \mathrm{sigm}(\mathbf{b} + \mathbf{W}\mathbf{x})$.

Having done with the unsupervised training, the parameters are used to initialize neural networks that will be trained using a supervised back-propagation algorithm. In this work, the neural networks are trained to learn several tasks at the same time, where each task consists of traffic flow prediction at each road. This type of learning is known as MTL [30]. Intuitively, MTL is suitable to be applied in a transportation network where several freeways and stations are connected to each other. Figure 2 depicts a multi-task regression stacked on top of the stacked RBMs.

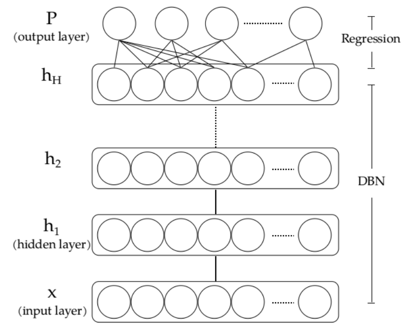


Fig. 2. DBN architecture for multi-task regression

B. Data Fusion

Data fusion deals with the synergistic combination of data and information from one or multiple sources to provide improved information with higher quality and reliability [35]. One of the the most well-known data fusion classification was provided by Dasarathy [36]. Data fusion techniques are roughly classified into five categories.

- data in-data out (DAI-DAO): is the basic data fusion method. Data fusion is conducted immediately after the data are gathered from sensors. Both inputs and ouputs are raw data.
- data in-feature out (DAI-FEO): this category uses raw data from sources to extract features able to better describe an entity.
- feature in-feature out (FEI-FEO): Both inputs and outputs are features. The purpose of feature fusion is to improve existing one or obtain new features.
- feature in-decision out (FEI-DEO): the inputs are features while the outputs are decisions. Hence, decisions are made based on features fed to the system.
- Decision In-Decision Out (DEI-DEO): is called also decision fusion. Decisions are fused to obtain enhanced or new decisions.

Our architecture for traffic flow prediction will follow the DEI-DEO model.The traffic flow predictions provided by DBNs using past values of the traffic flow and the current weather data will be fused to provide more accurate future traffic flow prediction. This particular scheme avoids compounding prediction errors that may ensue had weather data been predicted rather than been used as an real information.

IV. Deep Learning and Data Fusion based Traffic Flow Prediction Approach

Understanding the correlation between traffic indicators and weather conditions is crucial. This is the first step in our study. Then, with the extracted weather data as a complement, we can get better prediction about the traffic flow.

A. Correlation Study

We collect traffic data every five minutes for 4 months from 47 freeways. In addition, soft and hard weather data are collected from 16 stations scattered in the Bay Area.

Computing correlation involving huge number of paired time-series (weather and traffic), ensures pertinent feature selection and best prediction accuracy [23].

In the present work, we investigate the auto-correlation coefficients of traffic flow in freeways in San Francisco, Bay Area. Moreover, the cross-correlation between traffic flow and various weather parameters time-series (humidity, wind speed, temperature,etc) in the same area are analyzed. Traffic networks are considered as nonlinear systems. However, the standard correlation coefficient, i.e., Pearson product-moment correlation coefficient, represents the linear relation between variables. Thus, it is not appropriate for our study. We will use the well-known and simple nonlinear correlation coefficient known as the Spearman's rank correlation coefficient. This method is suitable when the data follow a curve instead of a straight line and are insensitive to noise. This method measures the statistical dependence between two variables. The rank of the variables are then correlated using Pearson correlation coefficient [37]. Then the selected features will be fed to our DBNs.

B. Decision-Level Data Fusion For Traffic Flow Prediction using DBNs

To the best of our knowledge, we are the first to include weather conditions in a DBN based framework. In our proposed architecture, the traffic flow and weather data are predicted separately using DBNs. The result of each prediction is then merged using data fusion techniques. This type of fusion scheme follows the DEI-DEO scheme discussed in section III-B. The proposed architecture is illustrated in Figure 3. In the prediction model 1 part, the input of the model is historical traffic flow data. The historical traffic flow data of each road are concatenated with other traffic flow data in other roads. As an example, if the historical traffic flow data until 8 steps in the past for 47 roads are considered, the size of the input will be $8 \times 47 = 376$. For the output of the prediction model 1, the future values of traffic flow of 47 roads are selected. In the case of prediction model 2, the input of the model is the current weather variables selected using cross-correlation. The output is identical with the prediction model 1, i.e., the future values of traffic flow of 47 roads.

In the data fusion part of the architecture, several methods of data fusion are tested: weighted average, ANNs, and DBNs. In the former technique, the output of each prediction model is weighted to produce better traffic flow predictions. These weights are calculated using least-square estimate method

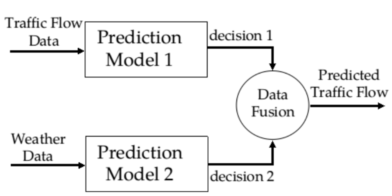


Fig. 3. Decision-Level data fusion for traffic flow prediction using weather data.

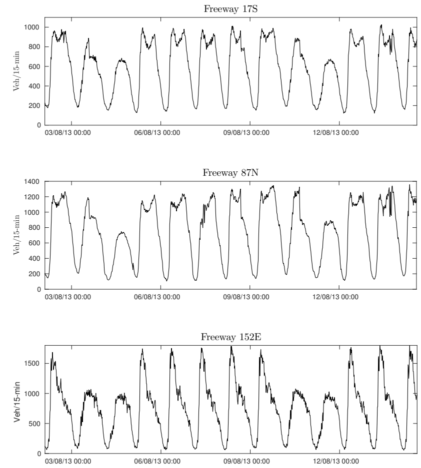


Fig. 4. A portion of traffic flow data with different traffic volume. Freeway with low (a), medium (b), and high (c) traffic flow.

according to the following equations:

$$\begin{aligned} Y &= \Phi^\top W \\ \hat{W} &= (\Phi^\top \Phi)^{-1} \Phi^\top Y \end{aligned} \tag{12}$$

where

$Y \triangleq$ output of the data fusion
$\Phi \triangleq$ the vector of regressors, which are functions of input vector X
$W \triangleq$ the weights of the data fusion inputs

V. Experimental Settings

In this section, we describe our data sets and scenario settings. The former allows us to accommodate the correlation analysis as well as for training and testing the DBNs. The latter allows us to show the merit of our decision-Level data fusion.

A. Datasets

We need two data sets: one contains traffic flow measurements and the other contains weather data. Traffic flow and weather datasets in this area are obtained from the Caltrans Performance Measurements Systems (PeMS) [38] and MesoWest project respectively [39].

The traffic flow is measured every 30 secondsusing inductive-loop sensors deployed throughout the freeways. These data then are aggregated into 5-minute duration by PeMS. Furthermore, based on the recommendation of Highway Capacity Manual 2010 [40], the traffic data are aggregated further into 15-minute duration. In the collected dataset, there are 47 roads, where the traffic flow of each road is calculated based on the average of all the loop detectors in that particular road.

Figure 4 plots the 2-weeks traffic flow patterns in selected freeways with low, medium, and high traffic flow. These categories are manually clustered based on the mean of the traffic flow in the 47 freeways. As can be seen in the figure, traffic flow witnesses peak points during rush hours, and reaches its lowest point during night.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2016.2585575, IEEE Transactions on Vehicular Technology

The weather dataset is obtained from 16 National Weather Service (NWS) network stations scattered throughout the district.Each station provides data about the temperature, humidity, visibility, wind speed and gust, dew point, cloud layer height and general weather condition. Originally, the weather data are sampled every one hour and the traffic flow data are sampled every 15 minutes. To align both data, the weather data are resampled every 15 minute using linear interpolation of the previous and next values of the temperature data at each point.

The inputs for the prediction models are the traffic flow of all roads at the previous k-step intervals and weather data from 16 stations at the previous 1-step intervals. For the weather data, we select only 1-step interval since the traffic flow prediction will be mainly based on the past traffic flow data. Moreover, the data coming from the weather station is a complement. The traffic flow data are represented as the number of vehicles per 15-min at a particular freeway. These data are normalized to real numbers in the range of $[0, 1]$. The weather data consist of temperature, humidity, visibility, wind gust and speed, dew point, cloud layer height, and weather condition. All of these items are normalized to real numbers in the range of $[0, 1]$, except for the weather condition which is soft-data. Weather condition consists of 24 unique values, which are represented by:

$$\begin{aligned} \text{weath\_cond} = \{&clear, cloudy, foggy, rain, \\ &snow, overcast, smoke, ...\} \end{aligned} \tag{13}$$

In order to incorporate the weather conditions in the DBN-based prediction, the soft-data representation is re-defined as an integer number in the range of $[1, 10]$, for instance $clear = 1,\ \ cloudy = 2$.

The output of the predictor is the k-step ahead of traffic flow prediction values. In our proposed architecture, the training outputs of both the predictors are identical. Specifically, the predictor model 2 predicts the traffic flow using data coming from the weather stations. In this work, the traffic flow and weather data are collected in the weekdays and weekends from August 1^{st}, 2013 to November 25^{th}, 2013. We use the data of August to October for the training and November for the testing.

B. Scenario Settings

We apply the traffic flow and weather datasets to different scenarios. These scenarios are defined based on:

1) Traffic flow prediction model: ARIMA, ANNs, and DBNs.
2) Feature settings: original, de-trended, and weekend/weekday traffic flow data.
3) Data fusion scheme: FEI-DEO and DEI-DEO data fusion levels.

For the first scenario, we assess the most appropriate model for traffic flow prediction by feeding ARIMA, ANNs, DBNs with the original traffic flow data. Based on the experiment results, we select the prediction model with the lowest performance indexes. These indexes are defined in V-C.

In the second scenario, the selected prediction model is tested using different feature settings to asses the impact of these settings on the prediction performances. Mainly, we investigate the de-trended and the weekend/weekday versions of the traffic flow data. The goal of the de-trending is to remove the fluctuation caused by the variation of hours and days of each week. This can be done by estimating the seasonal variation component and subtracting this component from the original traffic flow data [41], as follows:

- let $f_i \in \mathcal{R}^T$ denotes time-series data of the traffic flow at freeway i for T total number of time stamps.
- define the hour of day and day of week pair (h, d), where $h = 0, 1, ..., 23$ and $d = 1, ..., 7$. From these definitions, the following seasonal variation component is defined as the following

$$s_i(h, d) = \frac{\sum_{\{t|g(t)=(h,d)\}} f_i(t)}{|\{t|g(t) = (h, d)\}|} \tag{14}$$

where

$$\begin{aligned} s_i(h,d) &\triangleq \text{seasonal variation at hour } h \text{ and day } d \\ &\quad\ \text{for road i.} \\ f_i(t) &\triangleq \text{traffic flow of road } i \text{ at instant time } t. \\ g(t) &\triangleq \text{an indexing operator for } h \text{ and } d \\ &\quad\ \text{for a given time } t. \\ |A| &\ \text{denotes the cardinality of set A.} \end{aligned}$$

- use the above definitions and equation to calculate the de-trended version $\delta f_i(t)$ of the traffic flow $f_i(t)$ of the freeway i:

$$\delta f_i(t) = f_i(t) - s_i(h, d), \ \ \forall g(t) = (h, d) \tag{15}$$

As can be seen in Figure 4, traffic flow shows different patterns during the weekday and weekend. Therefore, we conduct two experiments with two groups of data: weekend and weekday traffic flow data. These groups of data are trained separately using two different settings of the selected model. The outputs are then concatenated and compared to the ground truth using the performance indexes.

In the last scenario, we investigate the traffic flow and weather data fusion at different levels. Essentially, we incorporate the relevant weather data as additional features to the traffic flow prediction model. The incorporation level differs on the data fusion schemes. In the FEI-DEO scheme, the data fusion is performed at the feature level. This scheme is compared with our novel architecture based on the DEI-DEO scheme. In this latter, the data fusion of the traffic flow and weather data is performed at the decision level.

C. Performance Indexes

We use two performance indexes: Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE). These indexes are

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2016.2585575, IEEE Transactions on Vehicular Technology

defined as:

$$\begin{aligned} \text{MSE} &= \left(\frac{1}{T} \sum_{t=1}^{T} (y_t - \hat{y}_t)^2 \right)^{\frac{1}{2}} \\ \text{MAE} &= \frac{1}{T} \sum_{t=1}^{T} |y_t - \hat{y}_t| \end{aligned} \tag{16}$$

where y_t and $\hat{y}_t$ are the actual and predicted traffic flow at time t respectively. We use these indexes to measure the linear score that averages the error with the same weight, and to measure the residuals by assigning larger weights to larger errors. These measures are represented by MAE and RMSE respectively.

VI. Analysis and Results

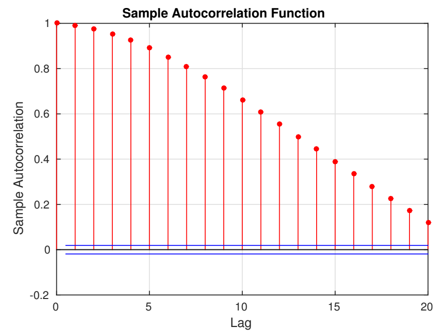
To make the proposed architecture tractable, we consider the following steps:

1) We begin by determining the required number of previous steps of traffic flow and influencing weather variables before a traffic flow prediction is conducted.
2) We compare the performances of DBNs with various prediction models, i.e., ARIMA and ANNs.
3) We further analyze the feature settings, i.e., d-trended and weekend/weekday version of traffic flow data, to study their impact on the prediction performances.
4) We integrate the weather data in the prediction using data fusion techniques. We compare the performances of the two data fusion levels: FEI-DEO and DEI-DEO

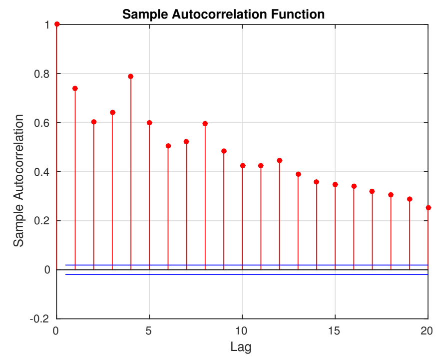
A. Correlation Analysis

1) Auto-Correlation: The purpose of the analysis is to test if the previous k-step intervals of the traffic flow have any correlation with the k-step ahead prediction. Figure 5 shows the auto-correlation results for both original (Figure 5(a)) and de-trended (Figure 5(b)) traffic flow data. For both the figures, each lag corresponds to 15 minutes time step. The height of the red lines represents the correlation between traffic flow at (t) and $(t - \text{time lag})$. The two blue lines correspond to the 95% confidence intervals of the correlation coefficients.

As can be observed in Figure 5 (a), even up to 20 lags (five hours) in the past, the original traffic flow values exhibit significant correlation. For the de-trended version, traffic flow values have significant correlation up to more than 20 lags. However, we limit the lag number to 20 in cross-validation processes due to computational constraint. In our cross-validation process, we fixed the predictor's parameters, and we train the predictor using different lag numbers. The outputs of the predictors are observed to determine the optimized lag number. The cross-validations are conducted only on the training set (August-October 2013). The experiment shows that the optimized lag number is 8, as depicted in Figure (6). Although the training has smaller error, the figure shows that the error of the testing process converge after 7 lag; hence, 8 lag is selected in the subsequent experiments.



(a) Original traffic flow data



(b) Detrended traffic flow data

Fig. 5. Auto correlation of the traffic flow data.

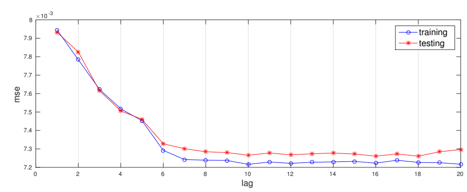


Fig. 6. Cross-validation of lag numbers

2) Cross-Correlation: To properly select the most weather factors that shape traffic flow behavior, it is necessary to investigate the weather and traffic correlation. Cross-correlation results are depicted in Table I. The labels *Tr. Flow, Wt. Con, Temp, Hum, Vis, Wind. G, Dew. P, CHCH, and Wind S.* refer to *Traffic flow, Weather Condition (cloudy, sunny, snowy, etc), Temperature, Humidity, Visibility, Wind Gust, Dew Point, Cloud Height Coverage and Wind Speed* respectively.

In table I, the cross-correlation coefficient between each pair of weather variables and traffic flow is represented by a real number. To select the pertinent feature subsets we follow the definition given in [42]. *Dew P,* has small correlation coefficient with respect to traffic flow. Hence, this variable is discarded. In addition, Hum, CHC, and $WindS.$ are discarded since they have large correlation coefficients with respect to the highest cross-correlation coefficient, i.e., $Temp.$ Hum, CHC, and $WindS.$ are considered as redundant weather variables.

The final feature subset is plotted in Figure 7. $Temp$, $WindG.$ and $Wt.Con$ have positive correlation with respect to traffic flow. Obviously, traffic volume increases when weather presents high temperatures (daytime) and decrease during

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2016.2585575, IEEE Transactions on Vehicular Technology

TABLE I
TRAFFIC FLOW AND WEATHER VARIABLES CROSS-CORRELATION COEFFICIENTS

	Tr. Flow	Wt. Con	Temp	Hum	Vis	Wind G.	Dew P.	CHC	Wind S.
Tr. Flow	1.00	0.20	0.41	-0.32	-0.12	0.12	0.09	0.31	0.34
Wt. Con	0.20	1.00	0.08	0.21	-0.08	0.08	0.28	0.84	0.23
Temp	0.41	0.08	1.00	-0.52	0.18	0.15	0.46	0.18	0.56
Hum	-0.32	0.21	-0.52	1.00	-0.22	-0.14	0.43	0.00	-0.34
Vis	-0.12	-0.08	0.18	-0.22	1.00	0.04	-0.02	-0.08	0.10
Wind G.	0.12	0.08	0.15	-0.14	0.04	1.00	0.04	0.12	0.38
Dew P.	0.09	0.28	0.46	0.43	-0.02	0.04	1.00	0.19	0.25
CHC	0.31	0.84	0.18	0.00	-0.08	0.12	0.19	1.00	0.31
Wind S	0.34	0.23	0.56	-0.34	0.10	0.38	0.25	0.31	1.00

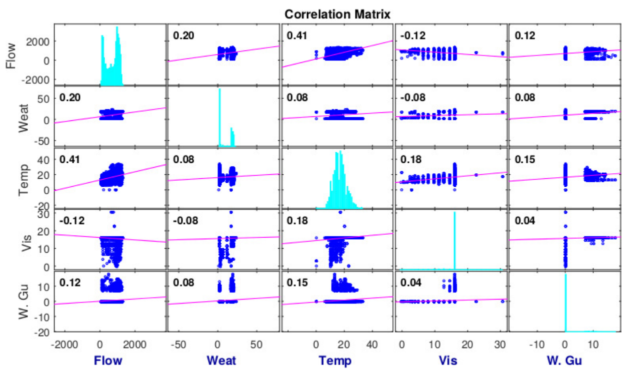


Fig. 7. Cross-correlation between traffic flow and most pertinent weather variables

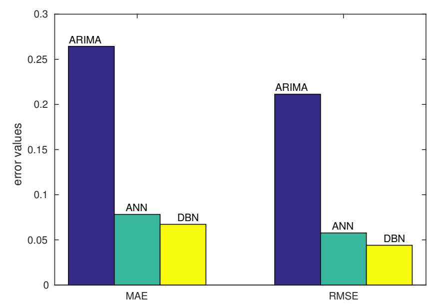


Fig. 8. Errors comparison of DBN with other approaches

nighttime. Furthermore, we can notice that traffic flow is inversely correlated to Vis (-0.12). Indeed, during low visibility condition such as fog, rain, etc, average speed is lower than its normal level which yields to lower traffic flow.

The conclusion made in this section vindicates the decision to use only pertinent weather data as part of the inputs in the traffic flow prediction architecture.

B. Traffic Flow Prediction and Data Fusion Analysis

1) Predictor Architecture Settings: Our prediction architecture includes several parameters that should be defined and tuned. For ARIMA, the number of regression variables are determined using auto-cross correlation discussed in the previous section. We set the number of hidden units and epoch of the 1-hidden layer ANNs set through cross-validation on the training data. The hidden units are varied between $[50, 300]$ in the step of 10. Moreover, the number of epoch is varied between $[25, 250]$ in the step of 25. The best architecture for ANNs is 1-hidden layer with 90 hidden units, and number of epoch is equal to 150.

In the case of DBNs, we choose the layer size from 2 to 5 layers. The number of nodes in each layer is chosen from $[50, 300]$ in the step of 50. The number of epochs is crucial to the learning phase. Therefore, we vary the epochs range from 50 to 500 in the step of 50. The best architecture for DBNs consists of 3 hidden layers, 250, 200, 100 hidden units in the first, second, and third hidden layers respectively. The best number of epoch of the DBNs training is found to be 100 epochs.

2) Prediction Results: In the following, we focus on the traffic flow prediction and data fusion performances.

DBN versus ARIMA and ANN:

We compare DBN to the most relevant state of the art prediction techniques namely ARIMA and ANN. Figure 8 presents the error values of DBNs as well as ARIMA and ANN in terms of MAE and RMSE. We can see from the figure that for 15-min traffic flow prediction task, DBN exhibits the best performances. It presents an average MAE equal to 0.07 and an average RMSE equal to 0.05. Compared to DBNs, ANNs has lower performance (0.08 for MAE and 0.06 for RMSE) but higher performances than ARIMA. The results confirm the merit of DBN to predict the actual traffic flow. DBNs are selected to be the prediction model in our proposed architecture.

Effect of de-trended and weekday/weekend version:

We explore the impact of the de-trending on our DBNs. Three typical freeways, i.e., 17S, 87N and 15E, are chosen to represent low, medium, and high traffic flow. We conduct experiments and evaluate the performance based on MAE and RMSE. The results are shown in Table II.

The reported results attest that using original traffic flow ensures better prediction performances when the traffic flow is low and medium. Indeed, for medium traffic, DBNs fed with original data have 0.034 and 0.045 as MAE and RMSE values respectively. Whereas, de-trended version presents 0.06 and 0.09 as MAE and RMSE values respectively. This results can be explained by the fact that the majority of the freeways has

TABLE II
PERFORMANCE COMPARISON OF DBN USING ORIGINAL DATA WITH THE DE-TRENDED VERSION AND WEEKEND-WEEKDAY PARTITION

		Original	De-trended	Weekend/ Weekday
Low Traffic (17S)	MAE	**0.0410**	0.0635	0.1904
	RMSE	**0.0599**	0.0848	0.2441
Med Traffic (87N)	MAE	**0.0347**	0.0602	0.2166
	RMSE	**0.0455**	0.0916	0.2851
High Traffic (152E)	MAE	0.0850	**0.0541**	0.1980
	RMSE	0.1161	**0.0894**	0.2781
Average	MAE	**0.0487**	0.0677	0.2004
	RMSE	**0.0701**	0.1040	0.2659

medium traffic flow. It is worth noting that feeding prediction model with de-trended traffic data outperforms when the traffic flow is high. Overall, the original traffic data outperforms de-trended version. This revelation can be explained by the fact that de-trended traffic data removes the temporal context of the features. However, with original data we retain the spatial and temporal relation of the features.

Moreover, the results show that our DBNs, when we consider weekday/weekend version, present higher average error. For example, in the 87N freeway, the DBNs have an RMSE value equal to 0.0455 considering the whole traffic while weekeday/weekend version shows an RMSE equal to 0.285. This high error value is due to the few quantity of data during weekend. Secondly, traffic data is time-series data. Thus, analyzing weekday and weekend data separately hides the time correlation inside the data between weekday and weekend.

Traffic and weather data fusion analysis:

Existing studies ignores weather conditions when using DBNs to predict traffic flow in transportation networks. In the following experiment, we investigate the performances of data fusion in different levels. To this end, we differentiate two levels: at feature level, which is denoted by $T+W-DF1$, and at decision level. Our decision-level data fusion for traffic flow prediction using weather data follows the DEI-DEO model. Furthermore, at decision level, we examine the performance of three variants of data fusion techniques:

- Least Square, which is denoted by $T+W-DF2-LS$.
- ANNs, which is denoted by $T+W-DF2-ANN$.
- DBNs, which is denoted by $T+W-DF2-DBN$.

The results of average values of MAE and RMSE are summarized in Table III. The first and second columns show the average error values of the traffic flow prediction based on only traffic and weather data respectively. The third column depicts the results of the data fusion at feature level. In general, fusing the data at feature level exhibits better prediction performances than using traffic or weather data only as input for the predictor.

In $T+W-DF1$ model, the DBNs' parameters are tuned simultaneously for traffic and weather data. Thus, the degree of freedom to tune the parameters is limited. Therefore, this level of data fusion presents inferior performances compared to $T+W-DF2-LS$, $T+W-DF2-ANN$ and $T+W-DF2-DBN$ which represent the decision level data fusion. From results depicted in Table III, it is clear that using DBNs for decision fusion significantly outperforms LS and ANNs for different traffic flow levels (Low, medium and high). The merit of DBNs for decision fusion is more clear in medium traffic flow (87N freeway). Indeed, $T+W-DF2-DBN$ has MAE and RMSE equal to 0.0250 and 0.0356 respectively. Meanwhile, $T+W-DF2-ANN$ has MAE and RMSE equal to 0.0262 and 0.0366 respectively.

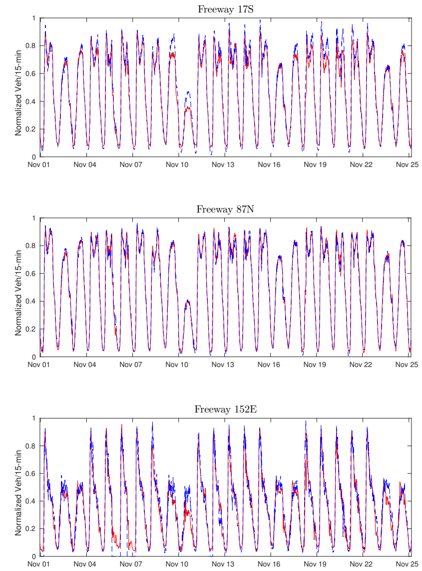


Fig. 9. Traffic flow prediction of freeway with various traffic volume. Freeway with low (a), medium (b), and high (c) traffic flow.

For illustration purpose, we also plot in Figure 9 the profile of predicted traffic flow values using decision-level data fusion based on DBNs versus the ground truth for the three representative freeways (17S, 87N and 15E). From this figure, we can see that our proposed architecture enhanced with weather data better tracks the traffic flow pattern in medium traffic flow (87N freeway). This results can be explained by the fact that the majority of the freeways has medium and low traffic flow. Hence, we have more data regarding medium flow and our DBNs based architecture is able to provide more accurate road traffic prediction. Low accuracy for high traffic is due to the scarcity of data regarding freeways with high traffic.

C. Complexity Analysis

In term of time complexity, the pre-training process of the RBMS depends on the number of neurons at each hidden layer, hidden layers, at each RBM and the batch size. In the supervise part, the duration of the training depends on the number of epoch, hidden layer, neurons at each hidden layer, and batch size. This two-stage process requires intensive computation time when the complexity of the model is increased. However, this process does not affect the implementation aspect of the DBNs. Once the final configuration of the DBNs is obtained, the testing process does not require intensive computation. For city-size traffic flow prediction applications, it can be

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2016.2585575, IEEE Transactions on Vehicular Technology

TABLE III
PERFORMANCE COMPARISON OF DBN

		Traffic Only	Weather Only	T+W-DF1	T+W-DF2-LS	T+W-DF2-ANN	T+W-DF2-DBN
Low Traffic (17S)	MAE	0.0410	0.2188	0.0480	0.0455	0.0424	**0.0352**
	RMSE	0.0599	0.2722	0.0575	0.605	0.0580	**0.0481**
Med Traffic (87N)	MAE	0.0347	0.2282	0.0499	0.0342	0.0262	**0.0250**
	RMSE	0.0455	0.2921	0.0583	0.0444	0.0366	**0.0356**
High Traffic (152E)	MAE	0.0850	0.2209	0.0472	0.0853	0.0714	**0.0654**
	RMSE	0.1161	0.2754	0.1038	0.1150	0.1012	**0.0954**
Average	MAE	0.0487	0.2192	0.0416	0.0491	0.0433	**0.0405**
	RMSE	0.0701	0.2722	0.06557	0.0700	0.0638	**0.0603**

assumed that a considerable amount of offline resources for the training process are available. Indeed, traffic authorities can have access to high-performance tools and libraries to power innovative machine learning applications in the cloud, data centers and workstations. Furthermore, the implementation of the proposed model requires significantly fewer resources.

In term of space complexity, the memory required for the pre-training and supervised training processes of the DBNs is substantially larger than the implementation or testing process. Other methods such as ANNs and ARIMA require fewer time and space in the training process, since the number of parameters is fewer than that of deep architectures. This issue can be considered as one of the limitation of deep architectures. In this work, we try to find the best configuration of models that gives highest accuracy, and, in the same time, try to maximize the utilization of available resources.

VII. CONCLUSION

One of the fundamental premises of smart cities is to improve quality of life by developing smart mobility. The ultimate objective is to ensure super-efficient navigation and safer travel journey. However, weather condition is a prevalent issue facing transportation safety and traffic management authorities. Wind gust, visibility, and high temperature affect traffic flow. Current research on traffic prediction mainly focus on data traffic history and neglect weather conditions. In this paper, we provided insights into the impact of weather conditions on traffic flow through correlation analysis. Moreover, we have studied how these critical weather data will affect the future traffic flow prediction. Accordingly, we proposed a comprehensive prediction architecture that incorporates DBNs and data fusion to derive more accurate traffic flow prediction in San Francisco, Bay Area using traffic flow history and weather data. We have differentiated several scenarios to highlight the merit of using data fusion at decision level in our proposed architecture. Experiment results show that our data-driven urban traffic system prediction outperforms the state of the art techniques. This higher traffic prediction accuracy ensures better operation and management traffic strategies.

Future work will involve the collection of non-traditional and rich information from social networks to improve short- and long-term traffic prediction.

REFERENCES

[1] Unhabitat.org. UN-Habitat - United Nations Human Settlements Programme. http://www.unhabitat.org/pmss/listItemDetails.aspx?publicationID=3387, 2015. [Online; accessed 23-December-2015].

[2] Organisation for Economic Co-operation and Development. Transport Outlook 2012: Seamless Transport for Greener Growth. http://www.internationaltransportforum.org/pub/pdf/12Outlook.pdf, 2012. [Online; accessed 23-December-2015].

[3] IBM. IBM traffic management solution. http://www.ibm.com/smarterplanet/us/en/traffic_congestion/article/traffic-management-and-prediction.html, 2012. [Online; accessed 23-December-2015].

[4] Federal Highway Administration. How Do Weather Events Impact Roads? http://www.ops.fhwa.dot.gov/weather/q1_roadimpact.htm, 2015. [Online; accessed 23-December-2015].

[5] Van Lint and Van Hinsbergen. Short-term traffic and travel time prediction models. *Artificial Intelligence Applications to Critical Transportation*, pages 22–41, 2012.

[6] Mario Cools, Elke Moons, and Geert Wets. Investigating the variability in daily traffic counts through use of arimax and sarimax models. *Transportation Research Record: Journal of the Transportation Research Board*, 2136:57–66, 2009.

[7] B Williams and L Hoel. Modeling and forecasting vehicular traffic flow as a seasonal arima process: Theoretical basis and empirical results. *Journal of Transportation Engineering*.

[8] Srinivasa Ravi Chandra and Haitham Al-Deek. Predictions of freeway traffic speeds and volumes using vector autoregressive models. *Journal of Intelligent Transportation Systems*, 13(2):53–72, 2009.

[9] Wanli Min and Laura Wynter. Real-time road traffic prediction with spatio-temporal correlations. *Transportation Research Part C: Emerging Technologies*, 19(4):606 – 616, 2011.

[10] Jianhua Guo and Billy Williams. Real-time short-term traffic speed level forecasting and uncertainty quantification using layered kalman filters. *Transportation Research Record: Journal of the Transportation Research Board*, 2175:28–37, 2010.

[11] N. Barimani, A.R. Kian, and B. Moshiri. Real time adaptive non-linear estimator/predictor design for traffic systems with inadequate detectors. *Intelligent Transport Systems, IET*, 8(3):308–321, May 2014.

[12] Sheng Jin, Dian-hai Wang, Cheng Xu, and Dong-fang Ma. Short-term traffic safety forecasting using gaussian mixture model and kalman filter. *Journal of Zhejiang University SCIENCE A*, 14(4):231–243, 2013.

[13] Zhou Jiang, Cunbao Zhang, and Yinxia Xia. Travel time prediction model for urban road network based on multi-source data. *Procedia - Social and Behavioral Sciences*, 138(0):811 – 818, 2014. The 9th International Conference on Traffic and Transportation Studies (ICTTS 2014).

[14] H. Chang, Y. Lee, B. Yoon, and S. Baek. Dynamic near-term traffic flow prediction: systemoriented approach based on past experiences. *Intelligent Transport Systems, IET*, 6(3):292–305, September 2012.

[15] Tianshu Wu, Kunqing Xie, Dong Xinpin, and Guojie Song. A online boosting approach for traffic flow forecasting under abnormal conditions. In *Fuzzy Systems and Knowledge Discovery (FSKD), 2012 9th International Conference on*, pages 2555–2559, May 2012.

[16] Jianming Hu, Pan Gao, Yunfei Yao, and Xudong Xie. Traffic flow forecasting with particle swarm optimization and support vector regression. In *Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on*, pages 2267–2268, Oct 2014.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVT.2016.2585575, IEEE Transactions on Vehicular Technology

[17] Abbas Khosravi, Ehsan Mazloumi, Saeid Nahavandi, Doug Creighton, and JWC Van Lint. A genetic algorithm-based method for improving quality of travel time prediction intervals. *Transportation Research Part C: Emerging Technologies*, 19(6):1364–1376, 2011.

[18] Kranti Kumar, M. Parida, and V.K. Katiyar. Short term traffic flow prediction for a non urban highway using artificial neural network. *Procedia - Social and Behavioral Sciences*, 104(0):755 – 764, 2013. 2nd Conference of Transportation Research Group of India (2nd CTRG).

[19] Jennifer McCrea and Salissou Moutari. A hybrid macroscopic-based model for traffic flow in road networks. *European Journal of Operational Research*, 207(2):676 – 684, 2010.

[20] Shahriar Afandizadeh Zargari, Salar Zabihi Siabil, Amir Hossein Alavi, and Amir Hossein Gandomi. A computational intelligence-based approach for short-term traffic flow prediction. *Expert Systems*, 29(2):124–142, 2012.

[21] Ming-Wei Li, Wei-Chiang Hong, and Hai-Gui Kang. Urban traffic flow forecasting using gausssvr with cat mapping, cloud model and {PSO} hybrid algorithm. *Neurocomputing*, 99:230 – 240, 2013.

[22] Yanru Zhang, Yunlong Zhang, and Ali Haghani. A hybrid short-term traffic flow forecasting method based on spectral analysis and statistical volatility model. *Transportation Research Part C: Emerging Technologies*, 43, Part 1:65 – 78, 2014. Special Issue on Short-term Traffic Flow Forecasting.

[23] S. Dunne and B. Ghosh. Weather adaptive traffic prediction using neurowavelet models. *Intelligent Transportation Systems, IEEE Transactions on*, 14(1):370–379, March 2013.

[24] Yunjie Zhao, Adel Sadek, and Daniel Fuglewicz. Modeling the impact of inclement weather on freeway traffic speed at macroscopic and microscopic levels. *Transportation Research Record: Journal of the Transportation Research Board*, 2272:173–180, 2012.

[25] Lei Lin, Ming Ni, Qing He, Jing Gao, Adel W Sadek, and Transportation Informatics Tier I Director. Modeling the impacts of inclement weather on freeway traffic speed: An exploratory study utilizing social media data. *to appear in Transportation Research Record: Journal of Transportation Research Board*, 2015.

[26] Dan Ciresan, Ueli Meier, and Jürgen Schmidhuber. Multi-column deep neural networks for image classification. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3642–3649. IEEE, 2012.

[27] Ramzi Abdelmoula. Noise robust keyword spotting using deep neural networks for embedded platforms. Master's thesis, University of Waterloo, 2016.

[28] Liqiang Nie, Meng Wang, Luming Zhang, Shuicheng Yan, Bo Zhang, and Tat-Seng Chua. Disease inference from health-related questions via sparse deep learning. *Knowledge and Data Engineering, IEEE Transactions on*, 27(8):2107–2119, 2015.

[29] Yisheng Lv, Yanjie Duan, Wenwen Kang, Zhengxi Li, and Fei-Yue Wang. Traffic flow prediction with big data: A deep learning approach. *Intelligent Transportation Systems, IEEE Transactions on*, 16(2):865–873, April 2015.

[30] Wenhao Huang, Guojie Song, Haikun Hong, and Kunqing Xie. Deep architecture for traffic flow prediction: Deep belief networks with multitask learning. *Intelligent Transportation Systems, IEEE Transactions on*, 15(5):2191–2201, Oct 2014.

[31] EleniI. Vlahogianni and MatthewG. Karlaftis. Comparing traffic flow time-series under fine and adverse weather conditions using recurrence-based complexity measures. *Nonlinear Dynamics*, 69(4):1949–1963, 2012.

[32] Ioannis Tsapakis, Tao Cheng, and Adel Bolbol. Impact of weather conditions on macroscopic urban travel times. *Journal of Transport Geography*, 28:204 – 211, 2013.

[33] Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh. A fast learning algorithm for deep belief nets. *Neural computation*, 18(7):1527–1554, 2006.

[34] Miguel A Carreira-Perpinan and Geoffrey E Hinton. On contrastive divergence learning. In *Proceedings of the tenth international workshop on artificial intelligence and statistics*, pages 33–40. Citeseer, 2005.

[35] Bahador Khaleghi, Alaa Khamis, Fakhreddine O. Karray, and Saiedeh N. Razavi. Multisensor data fusion: A review of the state-of-the-art. *Information Fusion*, 14(1):28 – 44, 2013.

[36] Federico Castanedo. A review of data fusion techniques. *The Scientific World Journal*, 2013:19 pages, 2013.

[37] Di Liu, Siu-Yeung Cho, Dong-mei Sun, and Zheng-Ding Qiu. A spearman correlation coefficient ranking for matching-score fusion on speaker recognition. In *TENCON 2010-2010 IEEE Region 10 Conference*, pages 736–741. IEEE, 2010.

[38] California Departement of Transportation. Caltrans Performance Measurement System. http://pems.dot.ca.gov/, 2015. [Online; accessed 23-December-2015].

[39] The University of Utah. MesoWest Status Update Log. http://mesowest.utah.edu/, 2015. [Online; accessed 23-December-2015].

[40] Highway Capacity Manual. Volumes 1-4.(2010). *Transporation Research Board*, 2010.

[41] Jingrui He, Wei Shen, Phani Divakaruni, Laura Wynter, and Rick Lawrence. Improving traffic prediction with tweet semantics. In *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence*, pages 1387–1393. AAAI Press, 2013.

[42] D.L. Hall, M. McNeese, J. Llinas, and T. Mullen. A framework for dynamic hard/soft fusion. In *Information Fusion, 2008 11th International Conference on*, pages 1 – 8, 30 2008-july 3 2008.

Arief Koesdwiady received the B.Eng degree in physics engineering from Institute Teknologi Bandung, Indonesia, and the M.Sc. degree in control system engineering from King Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia, in 2008 and 2013, respectively. He is currently pursuing the Ph.D. degree from the Centre for Pattern Analysis and Machine Intelligence at the University of Waterloo, Waterloo, ON, Canada. His current research interests include intelligent transportation systems, machine learning, deep learning, big data, and data fusion.

Ridha Soua Ridha Soua is currently a Research Associate at SnT, University of Luxembourg. He obtained a MSC diploma in Computer Science and Networking from Ecole Nationale des Sciences des l'Informatique (ENSI) in Tunisa in November 2010. From December 2010 to February 2014, he was PhD student at INRIA, HIPERCOM team. He obtained his PhD in computer sciences with honors from Pierre & Marie Curie University. His thesis investigated data gathering in multichannel wireless sensor networks. He was also a Postdoctoral Researcher in the Pattern Analysis & Machine Intelligence Group at University of Waterloo (Canada) from April 2014 to November 2015.

Fakhreddine Karray received the Dip-Ing. degree in electrical engineering from ENIT, Tunis, Tunisia, and the Ph.D. degree from the University of Illinois, Urbana Champaign, Champaign, IL, USA. He is the University Research Chair Professor in Electrical and Computer Engineering and Co-Director of the Center for Pattern Analysis and Machine Intelligence Center at the University of Waterloo, Waterloo, ON, Canada. His current research interests include intelligent systems, soft computing, sensor fusion, and context aware machines with applications to intelligent transportation systems, cognitive robotics, and natural man-machine interaction. He has co-authored over 400 technical articles, a textbook on soft computing and intelligent systems, six edited textbooks, and 20 textbook chapters. He holds 15 U.S. patents. Dr. Karray has Chaired/Co-Chaired several international conferences in his area of expertise and has served as a Keynote/Plenary Speaker on numerous occasions. He has also served as an Associate Editor/Guest Editor for a number of journals, including Information Fusion, the IEEE Trans. on CYBERNETICS, the IEEE Trans. on Neural Networks and Learning, the IEEE Trans. on Mechatronics, and the IEEE Computational Intelligence Magazine. He is the Chair of the IEEE Computational Intelligence Society (CIS) Chapter in Kitchener-Waterloo,