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

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

***Abstract*—Transportation systems might be heavily affected by factors such as accidents and weather. Specifically, inclement weather conditions may have a drastic impact on travel time and traffic flow. This study has two objectives: first, to investigate a correlation between weather parameters and traffic flow and, second, to improve traffic flow prediction by proposing a novel holistic architecture. It incorporates deep belief networks for traffic and weather prediction and decision-level data fusion scheme to enhance prediction accuracy using weather conditions. The experimental results, using traffic and weather data originated from the San Francisco Bay Area of California, corroborate the effectiveness of the proposed approach compared with the state of the art.**

***Index Terms*—Data fusion, deep learning, intelligent transportation systems (ITS), traffic prediction, weather information.**

I. Introduction

With more than 1 billion cars on roads today, which is expected to double to around 2.5 billion by 2050 [2], designing superefficient navigation and safer travel is becoming a major challenge 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 pour in from everywhere: weather, roads, traffic, and social network streams. An interesting example is the city of Eindhoven, The Netherlands, 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 are not only stored and processed but used for predicting and decision making in a timely and accurate way.

Manuscript received October 16, 2015; revised February 14, 2016 and April 11, 2016; accepted May 28, 2016. Date of publication June 28, 2016; date of current version December 14, 2016. The review of this paper was coordinated by the Guest Editors.

A. Koesdwiady and F. Karray are with the Centre for Pattern Analysis and Machine Intelligence, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: abkoesdw@uwaterloo.ca).

R. Soua was with the Centre for Pattern Analysis and Machine Intelligence, University of Waterloo, Waterloo, ON N2L 3G1, Canada. He is now with the Interdisciplinary Centre for Security, Reliability and Trust (SnT), University of Luxembourg, 1359 Luxembourg.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVT.2016.2585575

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 that they can better choose the ideal departure time and the best routes to avoid gridlock. However, traffic prediction is very challenging. Drivers interact with surrounding connected things (infrastructure, traffic light, cars), as well as with each other. During their trip, they are also exposed to changing weather conditions. Favorable or inclement weather will drastically influence drivers on roads. Statistics made available by the Federal Highway Administration reveal that 23% of vehicle accidents 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 pretrip 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 autocorrelation 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 influential 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 the 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 this 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 this 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 of traffic parameters (speed, flow, density) over time. This is crucial for drivers to plan their trip or for trip rescheduling

purposes. Traffic flow prediction approaches can be mainly classified in three categories: 1) parametric approach; 2) nonparametric approach; and 3) hybrid approach [5].

The main techniques used in the first class are autoregressive integrated moving average (ARIMA)-based models [6]–[9] and Kalman filtering [10]–[13]. However, the linearity of the 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 much 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.

Nonparametric regression [14], [15] is a widely used technique. In [15], an online boosting regression technique that ensures traffic prediction under abnormal traffic conditions was proposed. Otherwise, boosting is disabled. In [16], a support vector regression was used to establish the prediction model, whereas 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 usually use one hidden layer. Simulations have shown that one hidden layer would not be enough to describe the complicated relationship between the inputs and the outputs of the prediction model.

To handle the rigidity of time series and nonparametric models when complex nonlinear traffic states are present, some studies have investigated hybrid approaches [19]–[22]. By combining several techniques, these approaches are more adaptive. Although the aforementioned 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 the Global Positioning System, loop detectors, and smartphones. Few studies have tackled the problem of analyzing the tight correlation between traffic data and external factors such as weather [23]–[25]. Nevertheless, in [24] and [25], linear regression, which is a parametric model, was used 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 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 significant impact on traffic flow is widely acknowledged [31], [32].

III. Technical Background

Here, we detail the core components of our approach, which are DBNs, multitask 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

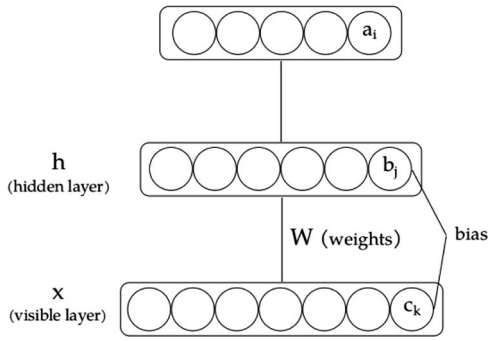

Fig. 1. RBMs.

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 Fig. 1.

RBMs are developed based on the interlayer interaction energy and the 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 $\mathbf{W}$ are 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 (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:

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

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

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

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

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

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 in [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

$$P(\mathbf{h}|\mathbf{x}) = \prod_i P(h_i|\mathbf{x})$$
$$P(\mathbf{x}|\mathbf{h}) = \prod_k P(x_k|\mathbf{h}). \tag{7}$$

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

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

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

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

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

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

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

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

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

Having done unsupervised training, the parameters are used to initialize neural networks that will be trained using a supervised back-propagation algorithm. In this paper, the neural networks are trained to learn several tasks at the same time, where each task consists of traffic flow prediction on 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. Fig. 2 depicts a multitask regression stacked on top of the stacked RBMs.

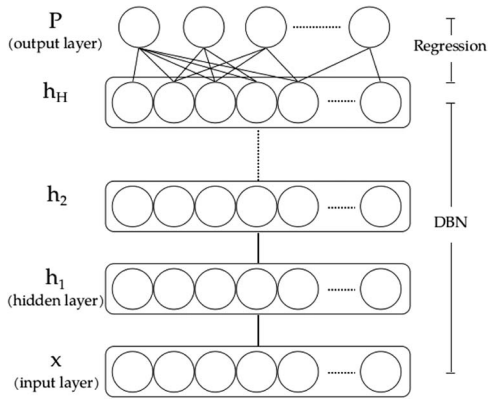

Fig. 2. DBN architecture for a multitask regression.

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 most well-known data fusion classifications was provided by Dasarathy [36]. Data fusion techniques are roughly classified into five categories, as follows.

- Data In Data Out: This is the basic data fusion method. Data fusion is conducted immediately after the data are gathered from sensors. Both inputs and outputs are raw data.
- Data In Feature Out: This category uses raw data from sources to extract features able to better describe an entity.
- Feature In Feature Out: Both inputs and outputs are features. The purpose of feature fusion is to improve the existing one or obtain new features.
- Feature In Decision Out (FEI-DEO): The inputs are features, whereas the outputs are decisions. Hence, decisions are made based on features fed to the system.
- Decision In Decision Out (DEI-DEO): This is also called 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 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 a better prediction about the traffic flow.

A. Correlation Study

We collected traffic data every 5 min for four months from 47 freeways. In addition, soft and hard weather data are collected from 16 stations scattered in the San Francisco Bay

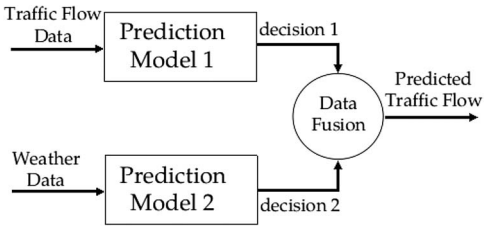

Fig. 3. Decision-level data fusion for traffic flow prediction using weather data.

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

In this work, we investigate the autocorrelation coefficients of traffic flow in freeways in the 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 is analyzed. Traffic networks are considered as nonlinear systems. However, the standard correlation coefficient, i.e., the 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 Spearman's rank correlation coefficient. This method measures the statistical dependence between two variables. The rank of the variables is then correlated using the 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 Fig. 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 up until eight steps in the past for 47 roads are considered, the size of the input will be $8 \times 47 = 376$. For the output of 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 that of 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 a least square estimate method according to the following equations:

$$Y = \Phi^{\top} W$$
$$\hat{W} = \left(\Phi^{\top}\Phi\right)^{-1}\Phi^{\top} Y \tag{12}$$

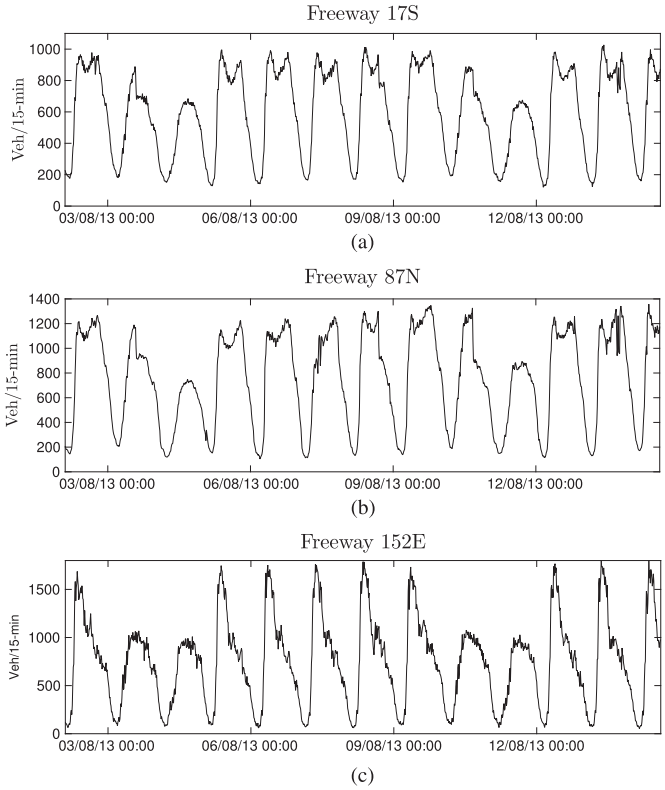

Fig. 4. Portion of traffic flow data with different traffic volumes. (a) Freeway with low traffic flow. (b) Freeway with medium traffic flow. (c) Freeway with high traffic flow.

where

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

V. Experimental Settings

Here, we describe our data sets and scenario settings.

A. Data Sets

Traffic flow and weather data sets are obtained from the Caltrans Performance Measurements Systems (PeMS) [38] and the MesoWest project, respectively [39].

The traffic flow is measured every 30 s using inductive-loop sensors deployed throughout the freeways. These data are then aggregated into 5-min duration by PeMS. Furthermore, based on the recommendation of Highway Capacity Manual 2010 [40], the traffic data are aggregated further into 15-min duration. In the collected data set, 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. Fig. 4 plots the two-week 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 shown in the figure, traffic flow witnesses peak points during rush hours and reaches its lowest point at night.

The weather data set is obtained from 16 National Weather Service 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 conditions. Originally, the weather data are sampled every 1 h, and the traffic flow data are sampled every 15 min. To align both data, the weather data are resampled every 15 min using linear interpolation of the previous and the 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 one-step intervals. For the weather data, we select only a one-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 are 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 conditions. All of these items are normalized to real numbers in the range of [0, 1], except for weather conditions, which are soft data. Weather conditions consist of 24 unique values, which are represented by

$$\text{weath\_cond} = \{\text{clear}, \text{cloudy}, \text{foggy}, \text{rain}, \text{snow}, \text{overcast} \quad \text{smoke}, \ldots\}. \tag{13}$$

To incorporate the weather conditions in the DBN-based prediction, the soft-data representation is redefined 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 predictors are identical. Specifically, predictor model 2 predicts the traffic flow using data coming from the weather stations. In this paper, the traffic flow and weather data are collected in the weekdays and weekends from August 1, 2013 to November 25, 2013. We use the data from August to October for the training and those from November for testing.

B. Scenario Settings

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

1) traffic flow prediction model: ARIMA, ANNs, and DBNs;
2) feature settings: original, detrended, 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, and 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 Section V-C.

In the second scenario, the selected prediction model is tested using different feature settings to assess the impact of these settings on the prediction performances. Mainly, we investigate the detrended and weekend/weekday versions of the traffic flow data. The goal of the detrending 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 $f_i \in \mathcal{R}^T$ denote the 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 $h = 0, 1, \ldots, 23$, and $d = 1, \ldots, 7$. From these definitions, the following seasonal variation component is defined:

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

where

$s_i(h,d) \triangleq$ seasonal variation at hour h and day d for road i;

$f_i(t) \triangleq$ traffic flow of road i at instant time t;

$g(t) \triangleq$ indexing operator for h and d for a given time t;

$|A|$ denotes the cardinality of set A.

- Use the preceding definitions and equation to calculate the detrended version $\delta f_i(t)$ of the traffic flow $f_i(t)$ of the freeway i, i.e.,

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

As shown in Fig. 4, the 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 the 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-square error (RMSE) and mean absolute error (MAE). These indexes are defined as

$$\text{MSE} = \left(\frac{1}{T}\sum_{t=1}^{T}(y_t - \hat{y}_t)^2\right)^{\frac{1}{2}}$$

$$\text{MAE} = \frac{1}{T}\sum_{t=1}^{T}|y_t - \hat{y}_t| \tag{16}$$

where y_t and $\hat{y}_t$ are the actual and predicted traffic flows at time t, respectively. We use these indexes to measure the linear score that averages the error with the same weight and to measure

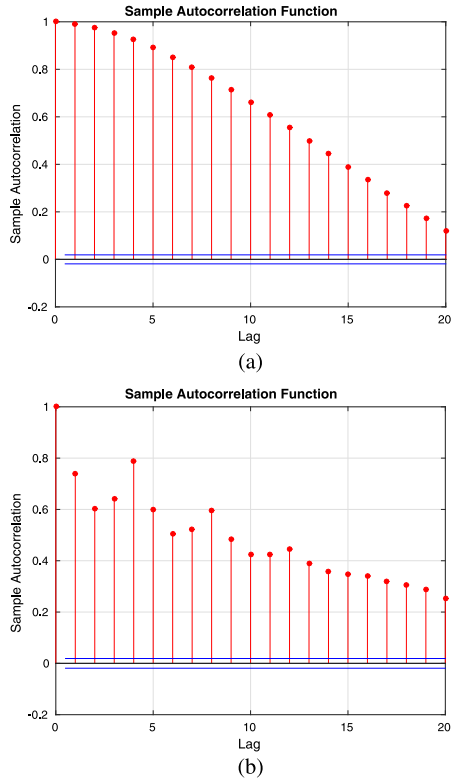

Fig. 5. Autocorrelation of the traffic flow data. (a) Original traffic flow data. (b) Detrended traffic flow data.

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., detrended and weekend/weekday versions of traffic flow data.
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) Autocorrelation: 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. Fig. 5 shows the autocorrelation results for both the original [see Fig. 5(a)] and detrended [see Fig. 5(b)] traffic flow data. For both figures, each lag corresponds to a 15-min time step. The height of the red lines represents the correlation between traffic flow at (t) and (t-time lag). The two blue lines correspond to the 95% confidence intervals of the correlation coefficients.

As can be observed in Fig. 5(a), even up to 20 lags (5 h) in the past, the original traffic flow values exhibit significant correlation. For the detrended 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 to October 2013). The experiment shows that the optimized lag number is 8, as depicted in Fig. 6. Although the training has a smaller error, the figure shows that the error of the testing process converges after seven lags; hence, eight lags are selected in the subsequent experiments.

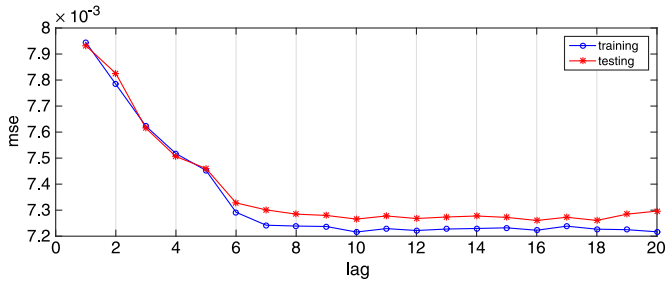

Fig. 6. Cross-validation of lag numbers.

2) 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 *Wind S.* are discarded since they have large correlation coefficients with respect to the highest cross-correlation coefficient, i.e., *Temp. Hum*, *CHC*, and *Wind S.* are considered as redundant weather variables. The final feature subset is plotted in Fig. 7. *Temp.*, *Wind G.*, and *Wt. Con* have a positive correlation with respect to traffic flow. Obviously, traffic volume increases when weather presents high temperatures (daytime) and decreases at night. Furthermore, we notice that traffic flow is inversely correlated to *Vis* (-0.12). Indeed, during low-visibility conditions such as fog and rain, the average speed is lower than its normal level, which yields lower traffic flow. The conclusion made here 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 is determined using auto/cross correlation discussed in the previous section. We set the number of hidden units and epochs of the one-hidden-layer ANNs set through cross-validation on the

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

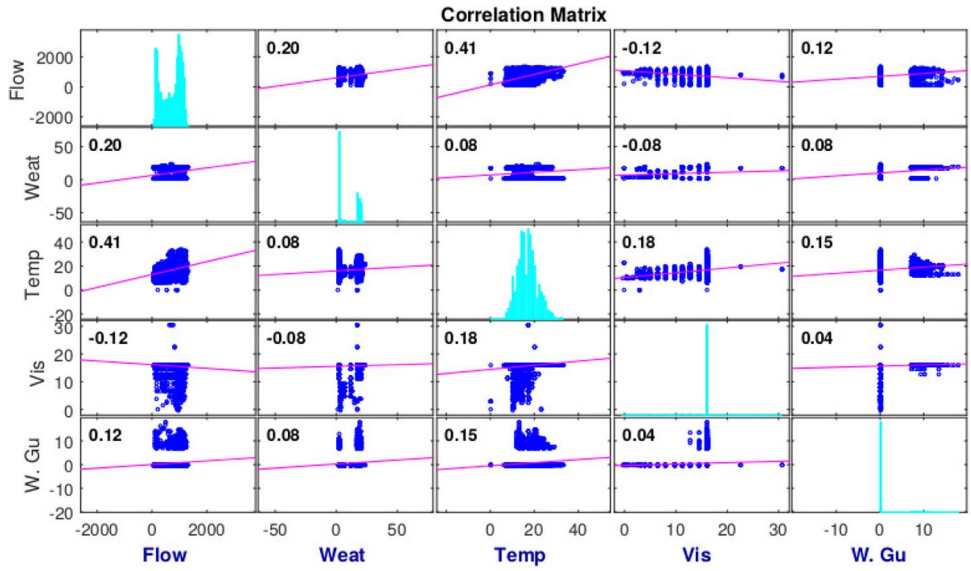

Fig. 7. Cross correlation between the traffic flow and the most pertinent weather variables.

training data. The hidden units are varied between [50, 300] in steps of 10. Moreover, the number of epochs is varied between [25, 250] in steps of 25. The best architecture for ANNs is one hidden layer with 90 hidden units, and the number of epochs is equal to 150.

In the case of DBNs, we choose the layer size from two to five layers. The number of nodes in each layer is chosen from [50, 300] in the steps of 50. The number of epochs is crucial to the learning phase. Therefore, we vary the epochs' range from 50 to 500 in the steps of 50. The best architecture for DBNs consists of three hidden layers with 250, 200, and 100 hidden units in the first, second, and third hidden layers, respectively. The best number of epochs 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 with the most relevant state-of-the-art prediction techniques, namely, ARIMA and ANN. Fig. 8 presents the error values of DBNs and ARIMA and ANN in terms of MAE and RMSE. We can see from the figure that, for a 15-min traffic flow prediction task, DBN performs best. It presents an average MAE equal to 0.07 and an average RMSE equal to 0.05. Compared with DBNs, ANNs have a 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 detrended and weekday/weekend versions: Three typical freeways, i.e., 17S, 87N, and 15E, are chosen to represent low, medium, and high traffic flows. We conduct experiments and evaluate the performance based on MAE and RMSE. The results are shown in Table II.

The reported results attest to the fact 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. On the other hand, the detrended version presents 0.06 and 0.09 as MAE and RMSE values, respectively. It is worth noting that feeding prediction model with detrended traffic data outperforms when the traffic flow is high. Overall, the original traffic data outperform the detrended version. This revelation can be explained by the fact that the detrended traffic data remove the temporal context of the

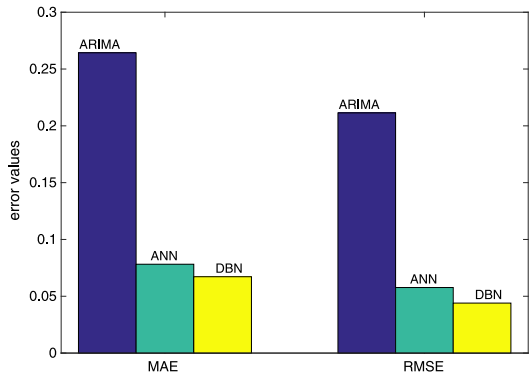

Fig. 8. Error comparison of DBN with the other approaches.

TABLE II
PERFORMANCE COMPARISON OF DBN USING ORIGINAL DATA WITH THE DETRENDED VERSION AND THE 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

features. However, with the original data, we retain the spatial and temporal relation of the features.

Moreover, the results show that our DBNs, when we consider the weekday/weekend version, present a higher average error. For example, in the 87N freeway, the DBNs have an RMSE value equal to 0.0455 considering the whole traffic, whereas the weekday/weekend version shows an RMSE equal to 0.285. This high error value is due to the few quantity of data during weekend. Second, traffic data are 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: In the following experiment, we investigate the performances of data fusion in different levels. To this end, we differentiate two levels: at the feature level, which is denoted by $\mathrm{T}+\mathrm{T}-\mathrm{DF1}$, and at the decision level. Our decision-level data fusion for traffic flow prediction using weather data follows the DEI-DEO model. Furthermore, at the decision level, we examine the performance of three variants of data fusion techniques as follows:

- least square, which is denoted by $\mathrm{T}+\mathrm{T}-\mathrm{DF2}-\mathrm{LS}$;
- ANNs, which are denoted by $\mathrm{T}+\mathrm{T}-\mathrm{DF2}-\mathrm{ANN}$;
- DBNs, which are denoted by $\mathrm{T}+\mathrm{T}-\mathrm{DF2}-\mathrm{DBN}$.

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 the $\mathrm{T}+\mathrm{T}-\mathrm{DF1}$ 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 with $\mathrm{T}+\mathrm{T}-\mathrm{DF2}-\mathrm{LS}$, $\mathrm{T}+\mathrm{T}-\mathrm{DF2}-\mathrm{ANN}$, and $\mathrm{T}+\mathrm{T}-\mathrm{DF2}-\mathrm{DBN}$, which represent the decision-level data fusion. From the 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 advantage of DBNs for decision fusion is clearer in medium traffic flow (87N freeway). Indeed, $\mathrm{T}+\mathrm{T}-\mathrm{DF2}-\mathrm{DBN}$ has MAE and RMSE equal to 0.0250 and 0.0356, respectively. Meanwhile, $\mathrm{T}+\mathrm{T}-\mathrm{DF2}-\mathrm{ANN}$ has MAE and RMSE equal to 0.0262 and 0.0366, respectively.

For illustration purposes, we also plot in Fig. 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). These results can be explained by the fact that the majority of the freeways have medium and low traffic flows. Hence, we have more data regarding medium flow, and our DBN-based architecture is able to provide more accurate road traffic prediction.

C. Complexity Analysis

In terms of time complexity, the pretraining process of the RBMs depends on the number of neurons at each hidden layer, the hidden layers at each RBM, and the batch size. In the supervised part, the duration of the training depends on the number of epochs, the hidden layer, the neurons at each hidden layer, and the 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 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 terms of space complexity, the memory required for the pretraining and supervised training processes of the DBNs is substantially larger than the implementation or the testing process. This issue can be considered as one of the limitations of deep architectures. In this paper, we try to find the best configuration of models that gives the highest accuracy and, at the same time, try to maximize the utilization of available resources.

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**

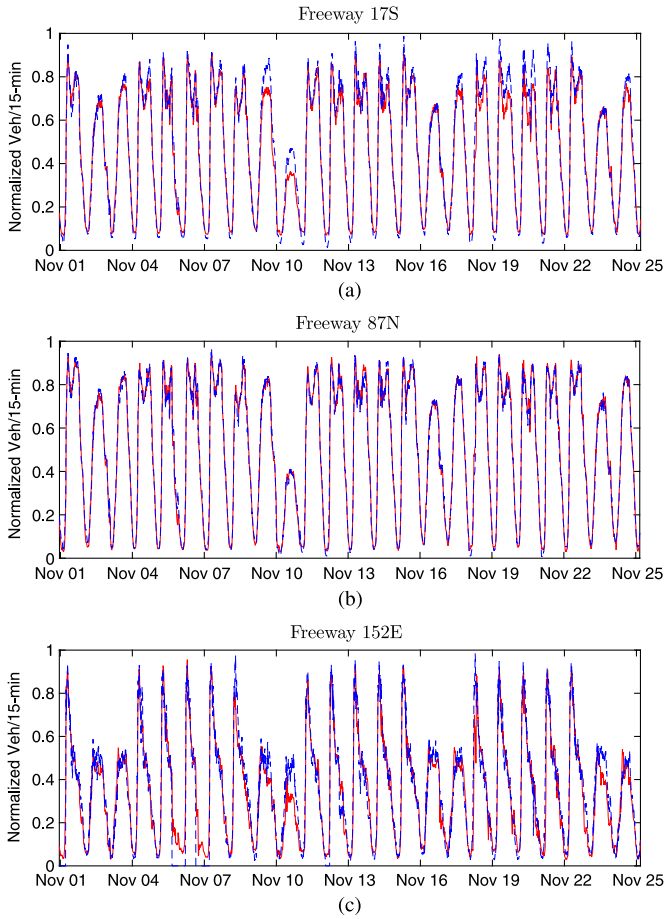

Fig. 9. Traffic flow prediction of freeway with various traffic volumes. (a) Freeway with low traffic flow. (b) Freeway with medium traffic flow. (c) Freeway with high traffic flow.

VII. CONCLUSION

Weather condition is a prevalent issue facing transportation safety and traffic management authorities. Current research on traffic prediction mainly focuses on data traffic history and neglects weather conditions. In this paper, we have 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 have 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 nontraditional and rich information from social networks to improve short- and long-term traffic predictions.

REFERENCES

[1] "UN-Habitat—United Nations human settlements programme," Unhabitat, Nairobi, Kenya, 2015, accessed Dec. 23, 2015. [Online]. Available: http://www.unhabitat.org/pmss/listItemDetails.aspx?publicationID=3387
[2] "Transport outlook 2012: Seamless transport for greener growth," Org. Econ. Co-operation Develop., Paris, France, 2012, accessed Dec. 23, 2015. [Online]. Available: http://www.internationaltransportforum.org/pub/pdf/12Outlook.pdf
[3] "IBM Traffic management solution," IBM, Armonk, NY, USA, 2012, accessed Dec. 23, 2015. [Online]. Available: http://www.ibm.com/smarterplanet/us/en/traffic_congestion/article/traffic-management-and-prediction.html
[4] "How do weather events impact roads?" Fed. Hwy. Admin., Washington, DC, USA, 2015, accessed Dec. 23, 2015. [Online]. Available: http://www.ops.fhwa.dot.gov/weather/q1_roadimpact.htm
[5] V. Lint and V. Hinsbergen, "Short-term traffic and travel time prediction models," *Artif. Intell. Appl. Critical Transp.*, vol. E-C168, pp. 22–41, 2012.
[6] M. Cools, E. Moons, and G. Wets, "Investigating the variability in daily traffic counts through use of arimax and sarimax models," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 2136, pp. 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," *J. Transp. Eng.*, vol. 129, no. 6, pp. 664–672, 2003.
[8] S. R. Chandra and H. Al-Deek, "Predictions of freeway traffic speeds and volumes using vector autoregressive models," *J. Intell. Transp. Syst.*, vol. 13, no. 2, pp. 53–72, 2009.
[9] W. Min and L. Wynter, "Real-time road traffic prediction with spatio-temporal correlations," *Transp. Res. C, Emerging Technol.*, vol. 19, no. 4, pp. 606–616, 2011.
[10] J. Guo and B. Williams, "Real-time short-term traffic speed level forecasting and uncertainty quantification using layered Kalman filters," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 2175, pp. 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," *IET Intell. Transp. Syst.*, vol. 8, no. 3, pp. 308–321, May 2014.
[12] S. Jin, D.-h. Wang, C. Xu, and D.-f. Ma, "Short-term traffic safety forecasting using Gaussian mixture model and Kalman filter," *J. Zhejiang Univ. Sci. A*, vol. 14, no. 4, pp. 231–243, 2013.
[13] Z. Jiang, C. Zhang, and Y. Xia, "Travel time prediction model for urban road network based on multi-source data," in *Proc. 9th ICTTS*, 2014, vol. 138, pp. 811–818.

[14] H. Chang, Y. Lee, B. Yoon, and S. Baek, "Dynamic near-term traffic flow prediction: System oriented approach based on past experiences," *IET Intell. Transp. Syst.*, vol. 6, no. 3, pp. 292–305, Sep. 2012.
[15] T. Wu, K. Xie, D. Xinpin, and G. Song, "A online boosting approach for traffic flow forecasting under abnormal conditions," in *Proc. 9th Int. Conf. FSKD*, May 2012, pp. 2555–2559.
[16] J. Hu, P. Gao, Y. Yao, and X. Xie, "Traffic flow forecasting with particle swarm optimization and support vector regression," in *Proc. IEEE 17th ITSC*, Oct. 2014, pp. 2267–2268.
[17] A. Khosravi, E. Mazloumi, S. Nahavandi, D. Creighton, and J. W. C. Van Lint, "A genetic algorithm-based method for improving quality of travel time prediction intervals," *Transp. Res. C, Emerging Technol.*, vol. 19, no. 6, pp. 1364–1376, 2011.
[18] K. Kumar, M. Parida, and V. K. Katiyar, "Short term traffic flow prediction for a non urban highway using artificial neural network," in *Proc. 2nd CTRG*, 2013, vol. 104, pp. 755–764.
[19] J. McCrea and S. Moutari, "A hybrid macroscopic-based model for traffic flow in road networks," *Eur. J. Oper. Res.*, vol. 207, no. 2, pp. 676–684, 2010.
[20] S. A. Zargari, S. Z. Siabil, A. H. Alavi, and A. H. Gandomi, "A computational intelligence-based approach for short-term traffic flow prediction," *Expert Syst.*, vol. 29, no. 2, pp. 124–142, 2012.
[21] M.-W. Li, W.-C. Hong, and H.-G. Kang, "Urban traffic flow forecasting using Gauss-SVR with cat mapping, cloud model and PSO hybrid algorithm," *Neurocomput.*, vol. 99, pp. 230–240, Jan. 2013.
[22] Y. Zhang, Y. Zhang, and A. Haghani, "A hybrid short-term traffic flow forecasting method based on spectral analysis and statistical volatility model," *Transp. Res. C, Emerging Technol.*, vol. 43, pp. 65–78, 2014.
[23] S. Dunne and B. Ghosh, "Weather adaptive traffic prediction using neurowavelet models," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 1, pp. 370–379, Mar. 2013.
[24] Y. Zhao, A. Sadek, and D. Fuglewicz, "Modeling the impact of inclement weather on freeway traffic speed at macroscopic and microscopic levels," *Transp. Res. Rec. J. Transp. Res. Board*, vol. 2272, pp. 173–180, 2012.
[25] L. Lin, M. Ni, Q. He, J. Gao, and A. W. Sadek, "Modeling the impacts of inclement weather on freeway traffic speed: an exploratory study utilizing social media data," presented at the Transportation Research Board Annual Meeting, Washington, DC, USA, Jan. 2015.
[26] D. Ciresan, U. Meier, and J. Schmidhuber, "Multi-column deep neural networks for image classification," in *Proc. IEEE CVPR*, 2012, pp. 3642–3649.
[27] R. Abdelmoula, "Noise robust keyword spotting using deep neural networks for embedded platforms," M.S. thesis, Dept. Electr. Comput. Eng., Univ. Waterloo, Waterloo, ON, Canada, 2016.
[28] L. Nie, M. Wang, L. Zhang, S. Yan, B. Zhang, and T.-S. Chua, "Disease inference from health-related questions via sparse deep learning," *IEEE Trans. Knowl. Data Eng.*, vol. 27, no. 8, pp. 2107–2119, Aug. 2015.
[29] Y. Lv, Y. Duan, W. Kang, Z. Li, and F.-Y. Wang, "Traffic flow prediction with big data: A deep learning approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 865–873, Apr. 2015.
[30] W. Huang, G. Song, H. Hong, and K. Xie, "Deep architecture for traffic flow prediction: Deep belief networks with multitask learning," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 5, pp. 2191–2201, Oct. 2014.
[31] E. I. Vlahogianni and M. G. Karlaftis, "Comparing traffic flow time-series under fine and adverse weather conditions using recurrence-based complexity measures," *Nonlinear Dyn.*, vol. 69, no. 4, pp. 1949–1963, 2012.
[32] I. Tsapakis, T. Cheng, and A. Bolbol, "Impact of weather conditions on macroscopic urban travel times," *J. Transp. Geogr.*, vol. 28, pp. 204–211, 2013.
[33] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, 2006.
[34] M. A. Carreira-Perpinan and G. E. Hinton, "On contrastive divergence learning," in *Proc. 10th Int. Workshop Artif. Intell. Statist.*, 2005, pp. 33–40, 2005.
[35] B. Khaleghi, A. Khamis, F. O. Karray, and S. N. Razavi, "Multisensor data fusion: A review of the state-of-the-art," *Inf. Fusion*, vol. 14, no. 1, pp. 28–44, 2013.
[36] F. Castanedo, "A review of data fusion techniques," *Sci. World J.*, vol. 2013, 2013, Art. no. 704504.
[37] D. Liu, S.-Y. Cho, D.-M. Sun, and Z.-D. Qiu, "A Spearman correlation coefficient ranking for matching-score fusion on speaker recognition," in *Proc. IEEE TENCON*, 2010, pp. 736–741.
[38] "Caltrans performance measurement system," Calif. Dept. Transp., Sacramento, CA, USA, 2015, accessed Dec. 23, 2015. [Online]. Available: http://pems.dot.ca.gov/
[39] "MesoWest status update log," Univ. Utah, Salt Lake City, UT, USA, 2015, accessed Dec. 23, 2015. [Online]. Available: http://mesowest.utah.edu/
[40] Highway Capacity Manual. Volumes 1–4, Transp. Res. Board, Washington, DC, USA, 2010.
[41] J. He, W. Shen, P. Divakaruni, L. Wynter, and R. Lawrence, "Improving traffic prediction with tweet semantics," in *Proc. 23rd Int. Joint Conf. Artif. Intell.*, 2013, pp. 1387–1393.
[42] D. L. Hall, M. McNeese, J. Llinas, and T. Mullen, "A framework for dynamic hard/soft fusion," in *Proc. 11th Int. Conf. Inf. Fusion*, Jun. 30–Jul. 3, 2008, pp. 1–8.

Arief Koesdwiady (S'16) received the B.Eng. degree in physics engineering from Institute Teknologi Bandung, Bandung, Indonesia, in 2008 and the M.Sc. degree in control system engineering from King Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia, in 2013. He is currently working toward the Ph.D. degree with the Centre for Pattern Analysis and Machine Intelligence, 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 received the M.Sc. degree in computer science and networking from Ecole Nationale des Sciences des l'Informatique (ENSI), Manouba, Tunisia, in November 2010. From December 2010 to February 2014, he pursued the Ph.D. degree with the HIPERCOM team, INRIA, France. He received the Ph.D.(Hons.) degree in computer science from Pierre and Marie Curie University, Paris, France, in 2014.

His thesis investigated data gathering in multi-channel wireless sensor networks. From April 2014 to November 2015, he was a Postdoctoral Researcher with the Pattern Analysis and Machine Intelligence Group, University of Waterloo, Waterloo, ON, Canada. He is currently a Research Associate with the Interdisciplinary Centre for Security, Reliability and Trust (SnT), University of Luxembourg.

Fakhreddine Karray (S'89–M'90–SM'00) received the Ing. Dipl. degree in electrical engineering from the University of Tunis, Tunis, Tunisia, in 1984 and the Ph.D. degree from the University of Illinois, Urbana-Champaign, in 1989.

He is currently a Professor of electrical and computer engineering with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada, and the Associate Director of the Pattern Analysis and Machine Intelligence Laboratory. He is a coauthor of a textbook on soft computing entitled *Soft Computing and Intelligent Systems Design* (Addison-Wesley, 2004). He serves as an Associate Editor for the *International Journal of Robotics and Automation*, the *International Journal of Control and Intelligent Systems*, and the *International Journal of Image Processing*.. He has served as Chair/Co-Chair of more than 12 international conferences and technical programs and was the General Co-Chair of the IEEE Conference on Logistics and Automation, China, in 2008 and the General Co-Chair of the International Conference on Systems, Circuits and Signals, Tunisia, in 2009. He is the holder of 13 U.S. patents in the areas of mechatronics and intelligent systems. His research interests are in autonomous systems and intelligent man-machine interfacing and mechatronics design, on which he has extensively authored. He is the local Waterloo Chapter Chair of the IEEE Control Systems and the IEEE Computational Intelligence Societies. He serves as an Associate Editor for the IEEE TRANSACTIONS ON MECHATRONICS, the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS-PART B, and *IEEE Computational Intelligence Magazine*.