COMPRESSION, COMPRESSED SENSING AND COMPRESSED PREDICTION OF URBAN TRAFFIC DATA

Technical field

The present invention relates to methods and systems for sensing and prediction of traffic conditions, primarily, but not exclusively, road traffic conditions.

Background

Large cities may face transportation and mobility issues, with traffic jams being quite common, especially during rush hours. By appropriately guiding drivers, traffic jams may potentially be avoided or at least limited. To this end, traffic guidance systems require accurate information about current and future traffic states.

Traffic applications typically rely on the global positioning system (GPS) to provide location based services to end users. GPS devices include in-car global positioning system (GPS) devices and internet enabled GPS devices.

In-car GPS may be used to provide (usually at extra cost) real time traffic updates and/or may use historical traffic data to provide the most efficient route. Due to its price and the subscription cost of the traffic data, such devices have a limited share in the market of traffic applications.

Internet-enabled GPS devices are frequently based on mobile software applications running on platforms such as iOS or Android, with traffic information (if any) being provided by third parties. These mobile applications can typically be categorized as either "turn by turning" or traffic applications. Turn-by-turning mobile applications provide guidance to a desired destination without any information about traffic conditions. On the other hand, traffic mobile applications typically provide "traffic on map" information, with various levels of quality and network coverage. These colored maps depict a recent traffic state, which is based on historical data. For example, Google collects the vehicle speeds of its users (if location services are enabled on the users' phones) and infers current traffic state from these samples. However, these

systems suffer from scalability issues (because they require a high number of samples) and privacy issues (because individual users may be identifiable).

More importantly, in-car and internet GPS devices provide only information about the current state of the road network. A driver relying on a route guidance algorithm implemented by such a device may encounter congestion during the course of a trip, since the state information provided by the GPS device may rapidly become out of date. Accordingly, it would be desirable to provide algorithms which can perform prediction of road traffic conditions for multiple horizons, and which can deal with large diverse road networks.

It has previously been known that data driven algorithms such as Support Vector Regression (SVR) can perform traffic prediction with high accuracy. In addition, a confidence in the predicted state can be inferred through Bayesian SVR algorithms. However, such algorithms may not be scalable. Further, they may not be able to make predictions in real-time.

It would be desirable to provide a method of predicting traffic conditions which overcomes the above disadvantages, or which at least provides a useful alternative.

Summary

In one aspect, the present invention relates to a real-time compression method for predicting vehicular traffic conditions within a geographical area, the geographical area including a network of vehicular roads having subsets of roads, each subset of roads including a group of at least a segment of some of the roads which are selected as being suitable to represent the network of roads, the selection being based on weightage assigned to each road of the network; the assignment being based on correlation of historical vehicular traffic data of the network to respective roads of the network, the method comprising:

(i) providing spatial relations data indicative of a spatial relationship between the selected segments of some of the roads and the rest of the roads of the network;

(ii) obtaining current traffic conditions data by real-time detection of vehicular traffic conditions at the selected segments of the roads; and

(iii) predicting future traffic conditions for the network of roads within the geographical area from the current traffic conditions data at the selected segments of the roads and the spatial relations data.

Advantageously, the present invention provides scalable architectures for real-time traffic estimation and prediction systems. By the use of the spatial relations data, it is possible to use measurements of only a subset of the road segments in order to infer conditions for the entire network. The method may be used to infer current traffic conditions for the network and/or to predict future traffic conditions for the network.

In certain embodiments, the spatial relations data is stored in a plurality of mobile devices belonging to respective users, and the current traffic conditions data is received by each mobile device. Accordingly, if the acquisition of traffic data is being handled by a server, estimation and/or prediction of the traffic conditions may be delegated to the mobile devices, freeing up resources on the server.

The method may further comprise displaying the predicted future traffic conditions on each mobile device.

In certain embodiments, the weightages are obtained by singular value decomposition of a matrix formed from the historical vehicular traffic data. The selected segments may be selected by sampling using the weightages as probabilities.

In certain embodiments, the prediction is performed using support vector regression or Bayesian support vector regression.

The method may comprise determining current traffic conditions for the network from the current traffic conditions data at the selected segments of the roads and the spatial relations data.

In another aspect, the invention relates to a computer-readable storage medium having stored thereon computer-executable instructions which, when executed by at least one computing device, cause the at least one computing device to perform the method according to any of the preceding paragraphs.

In a further aspect of the present invention, there is provided a system for predicting vehicular traffic conditions within a geographical area, the geographical area including a network of vehicular roads having subsets of roads, each subset of roads including a group of at least a segment of some of the roads which are selected as being suitable to represent the network of roads, the selection being based on weightage assigned to each road of the network; the assignment being based on correlation of historical vehicular traffic data of the network to respective roads of the network, the system comprising:

at least one processor; and

at least one non-volatile storage device, the at least one non-volatile storage device having stored thereon spatial relations data indicative of a spatial relationship between the selected segments of some of the roads and the rest of the roads of the network;

the at least one processor being configured to:

(i) obtain current traffic conditions data from real-time detection of vehicular traffic conditions at the selected segments of the roads; and

(ii) predict future traffic conditions for the network of roads within the geographical area from the current traffic conditions data at the selected segments of the roads and the spatial relations data.

In certain embodiments, a first processor is configured to perform operation (i), and a second processor is configured to perform operation (ii). The first processor may be located at a server computer system, and the second processor may be located at a mobile device in communication with the server computer system.

The at least one non-volatile storage device may comprise a plurality of storage devices of a corresponding plurality of mobile devices belonging to respective users, and each of the plurality of storage devices may have stored thereon the spatial relations data and the current traffic conditions data.

The at least one processor may be configured to predict the future traffic conditions using support vector regression or Bayesian support vector regression.

In certain embodiments, the at least one processor is configured to determine current traffic conditions for the network from the current traffic conditions data at the selected segments of the roads and the spatial relations data.

In a further aspect of the present invention, there is provided a column-based method for compressing vehicular traffic data within a geographical area, the geographical area including a network of vehicular roads, the column-based method comprising:

(i) correlating historical vehicular traffic data of the network to respective vehicular roads of the network;

(ii) assigning a weightage to each vehicular road of the network based on the correlation;

(iii) selecting at least a segment of some of the roads based on the assigned weightage which are suitable to represent the network of roads; and

(iv) determining spatial relations data indicative of a relationship between the selected segment of the roads and the rest of the roads of the network;

wherein current traffic conditions of the network are computable using current traffic conditions data indicative of current traffic conditions at the selected segments of the roads and the spatial relations data.

In certain embodiments, assigning the weightage may include performing singular value decomposition of the historical vehicular data.

In any of the above embodiments, the spatial relations data may represent a relationship matrix having X number of columns and Y number of rows; wherein

X = total number of road segments in the network, and

Y = total number of road segments in the selected road segments.

In a yet further aspect, the present invention provides a system for compressing vehicular traffic data within a geographical area, the geographical area including a network of vehicular roads, the system comprising: at least one processor; and a non-volatile storage device having stored thereon computer-readable program instructions which, when executed, cause the at least one processor to perform the method according to any one of the three immediately preceding paragraphs.

Brief Description of the Drawings

Embodiments of the invention will now be described, by way of non-limiting example only, with reference to the accompanying drawings in which:

Fig. 1 is a schematic diagram of a traffic data sensing and prediction system;

Fig. 2 is a flow chart of a compression method implemented in the system of Fig. 1;

Fig. 3 is a flow chart of a sensing method according to embodiments;

Fig. 4 is a flow chart of a prediction method according to embodiments;

Fig. 5 shows comparative plots of percent root mean distortion (PRD) error *vs.* compression ratio for (a) a method implemented according to the process of Fig. 2 and (b) a method implemented according to the process of Fig. 3;

Fig. 6 shows comparative plots of percent root mean distortion (PRD) error *vs.* compression ratio for a method implemented according to the process of Fig. 4, for (a) a 5 minute prediction horizon and (b) a 30 minute prediction horizon; and

Fig. 7 shows comparative plots of mean average percentage error (MAPE) *vs.* compression ratio for a method implemented according to the process of Fig. 4, for (a) a 5 minute prediction horizon and (b) a 30 minute prediction horizon.

Detailed Description of Embodiments

Referring to Fig. 1, there is shown a traffic data compression, sensing and prediction system 100. The system 100 comprises a traffic data server 110 which acquires traffic data indicative of one or more measured traffic parameters relating to vehicle traffic along respective road segments (also referred to herein as links) in a road network. The traffic parameters may include speed and volume, for example. The traffic parameters may be derived from signals measured by inductive-loop sensors or other traffic sensors, in known fashion.

The server 110 is in communication with a plurality of user devices, for example laptop or desktop computers 130 and mobile computing devices 140 (e.g., smartphones and tablet computing devices), via a wide area network 102 such as the Internet. In the case of mobile computing devices 140, these may connect to the network 102 via a cellular data network and Internet gateway. Server 110 transmits traffic information to application software executing on the user devices 130, 140, relating to the current and

predicted traffic parameters in particular geographical areas (i.e., along the road segments). The traffic information is overplotted on a road map which is displayed by the user devices, so that a user can assess, based on the current and/or predicted traffic parameters, the best route to travel along the road network. Alternatively, the application software may be configured to determine one or more suggested routes based on the current and/or predicted traffic parameters, and to display the one or more suggested routes on the road map.

The server 110 receives a continuous stream of traffic data representing speed and/or volume along each road segment in the network, at a succession of times, at a given sampling period (e.g., 5 minutes or 15 minutes).

In general terms, the server 110 implements a process or set of processes which enables prediction of a future state of the road network substantially in real time. The road network is compressed by representing it in terms of a subset of suitable roads or road segments, and generating spatial relations data indicative of a relationship between the subset and the rest of the network. Accordingly, current or predicted values of the traffic parameters for the subset can be extrapolated, using the spatial relations data, to obtain the values for the whole network. The sensing and prediction applications which use the compression require less computational resources than previously known approaches. In addition, the use of compression enables at least some of the computations to be delegated to users' devices 140 (e.g., mobile phones), which are increasingly getting more computational resources.

Referring to Figs. 2 to 4, in preferred embodiments, the processes implemented by server 110 (in conjunction with user mobile devices 140, if any computations are to be delegated) include the following:

- a compression process 200;
- a sensing process 300;
- a training process 400; and
- a prediction process 450.

The compression process 200 is carried out entirely by server 110 in an offline process, in order to generate a relationship matrix which can be transmitted to mobile

devices 140 (and periodically updated if necessary), in order for the mobile devices 140 to perform the sensing process 300 and/or prediction process 450. The training process 400 is preferably carried out in offline mode by server 110. In alternative embodiments, the compression process may be carried out by server 110 while the sensing process 300, training process 400 and/or prediction process 450 may be carried out by one or more additional servers (not shown) which are in communication with server 110, for example via the Internet 102, and/or via a local area network. In a further variant, all of the computations of processes 200, 300, 400 and 450 may be performed by server 110, with only the final results (current and predicted network state) being transmitted to mobile devices 140.

Compression 200

Compressing the network is a preprocessing step which enables real-time estimation and prediction of large scale traffic network. During this process it is possible to: (1) identify the most representative links in the network which will be most important for the estimation and prediction phases; and (2) learn relationship functions which will be used to disseminate traffic information from the subset of the road segments to the entire network in both estimation and prediction processes.

Referring to Fig. 2, compression process 200 takes as its input historical data representing traffic information for the entire network. The data stream includes speed for each road or road segment in the network, generated from inductive loop detectors, for example. Privacy of traffic information as well as high reliability and spatial coverage are major benefits of using inductive-loop sensors as the source of the real-time traffic stream. The input data may be represented (step 210) as a matrix **A**, the columns of which represent road segments, and the rows of which represent time intervals.

The matrix **A** may be approximated by a column-based (CX) matrix approximation $\widehat{\mathbf{A}}$:

$$\widehat{\mathbf{A}} = \mathbf{CX},$$

where $\mathbf{A} \in \mathbb{R}^{m \times n}$, $\mathbf{C} \in \mathbb{R}^{m \times c}$ is a matrix consisting of c columns of the matrix $\mathbf{A}$, and $\mathbf{X} \in \mathbb{R}^{c \times n}$ is a matrix that expresses every column of $\mathbf{A}$ in terms of the basis provided by

the columns of $\mathbf{C}$. $\mathbf{A}$ has m rows corresponding to m time intervals and n columns corresponding to n road segments. The c columns of matrix $\mathbf{C}$ ($c < n$) correspond to the subset of c road segments which are most suitable for representing the road network.

Next, at steps 212-216, the process 200 performs column selection, in order to construct the matrix $\mathbf{C}$ by finding the most suitable columns. Testing all possible choices of c columns is typically not practical. Therefore, in embodiments of the present invention, column selection is carried out by SVD sampling. The present inventors have found that the SVD sampling method yields the best reconstruction accuracy. SVD sampling assigns higher selection probability to the road segments with larger traffic speed variations (if speed is used as the traffic parameter of interest) or traffic volume variations (if volume is the traffic parameter of interest).

At step 212, the process 200 performs singular value decomposition (SVD) of the historical data (matrix $\mathbf{A}$) of the entire network:

$$\mathbf{A} = \mathbf{XSY}^{\mathrm{T}}$$

where $\mathbf{X}$ and $\mathbf{Y}$ are unitary matrices whose columns are left and right singular eigenvectors respectively of $\mathbf{A}$. The matrix $\mathbf{S}$ contains the singular values of $\mathbf{A}$.

At step 214, the Euclidean norm of each of the top k right singular vectors of matrix $\mathbf{A}$ (i.e., the columns of $\mathbf{Y}$) is calculated, to assign a score E_{a_i} to each column. The top k right singular vectors are the k singular vectors having the k largest singular values. For example, k can be chosen to be equal to c. The score E_{a_i} can be converted into a weightage or probability P_{a_i} which can be used to sample (step 216) the columns:

$$\mathrm{P}_{a_i} = \frac{1}{k} E_{a_i} = \frac{1}{k} \sum_{j=1}^{k} v_{ij}^2 \quad \forall\ i = 1, \ldots n, \tag{1}$$

where v_{ij} is the i-th coordinate of the j-th right singular vector.

Once the c columns have been sampled, matrix $\mathbf{C} \in \mathbb{R}^{m \times c}$ having them as its columns can be constructed (step 218). Next, at step 220, the relationship matrix $\mathbf{X} \in \mathbb{R}^{c \times n}$,

which will allow representation of the columns of matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$ in terms of columns of the matrix $\mathbf{C}$, is computed. The matrix $\mathbf{X}$ can be regarded as an extrapolation matrix that maps the subnetwork associated with $\mathbf{C}$ to the entire network represented by $\mathbf{A}$. For given matrices $\mathbf{C}$ and $\mathbf{A}$, we compute the relationship matrix $\mathbf{X}$ as $\mathbf{X} = \mathbf{C}^{+}\mathbf{A}$, where $\mathbf{C}^{+}$ is the Moore-Penrose pseudo-inverse of matrix $\mathbf{C}$.

In the following, subscripts h, p and f are used to denote historical, present and future values.

Suppose that the matrix $\mathbf{C}_h$ contains the observed traffic states of the c specific locations in the network, such that $\{c_1, \ldots, c_c\} \subseteq \{a_1, \ldots, a_n\}$. Then, the data matrix $\mathbf{A}_h$ can be approximated as $\widehat{\mathbf{A}}_h = \mathbf{C}_h \mathbf{X}_h$, where the matrix $\mathbf{X}_h$ contains the relationships between the traffic conditions at different locations in the network. Hence, instead of storing the large matrix $\mathbf{A}_h$, the two smaller matrices $\mathbf{C}_h$ and $\mathbf{X}_h$ can be stored by server 110. The compression ratio (CR) of this low-dimensional approximation is given by:

$$CR_h = \frac{mn}{mc + cn}. \qquad (2)$$

The output of compression process 200 is the relationship matrix $\mathbf{X}_h$, which can be stored by server 110 for the purpose of inferring, from the c specific locations, traffic parameters in the remainder of the network, and/or predicting future traffic parameters on the basis of predicted values at the c specific locations, as will now be described.

Real time compressed sensing 300

Sensing process 300 aims to estimate the traffic state of the whole traffic network by observing only a subset of the network, in real time. This process is referred to herein as real time compressed sensing for large traffic networks.

Referring to Fig. 3, sensing process 300 preferably has, at step 310, a previously computed relationship matrix $\mathbf{X}_h$ as an input, which can be used to infer the full-network matrix $\mathbf{A}$ for any given $\mathbf{C}$. In alternative embodiments, step 310 may comprise determining an $\mathbf{X}_h$ using historical data $\mathbf{A}_h$ and a column matrix $\mathbf{C}_h$, by a compression process 200, for example. If a pre-computed relationship matrix is used, the process

300 only requires traffic data from a small number of sensors during online operation (compressed sensing).

In addition to the relationship matrix $\mathbf{X}_h$, the sensing process 300 also obtains data indicative of the subset of roads or road segments for which current traffic data are to be measured or otherwise obtained. Next, at step 312, the current data for the subset of roads or road segments is retrieved, and a row vector $\mathbf{c}_p^i \in \mathbb{R}^{1\times c}$ is constructed from the retrieved values. Under the assumption that the traffic conditions are stationary, so that a fixed matrix $\mathbf{X}$ allows us to accurately reconstruct the original data matrix $\mathbf{A}$ from $\mathbf{C}$, the present network state $\widehat{\boldsymbol{\alpha}}_p^i$ can be estimated at step 314 according to $\widehat{\boldsymbol{\alpha}}_p^i = \mathbf{c}_p^i \mathbf{X}_h \quad \forall i = k, \ldots, m$, where $\widehat{\boldsymbol{\alpha}}_p^i \in \mathbb{R}^{1\times n}$ is a row vector which represents the current state of the entire network for test data ($i = k, \ldots, m$). Row vector $\mathbf{c}_p^i \in \mathbb{R}^{1\times c}$ contains the information about current traffic conditions at c specific locations in the network. Matrix $\mathbf{X}_h$ is the relationship matrix, learned from a training data set. The compression ratio for compressed sensing can be defined as n/c.

Large traffic networks contain a diverse set of road segments. In certain embodiments, homogeneous subnetworks may improve the overall performance of compressed sensing. The method may comprise dividing the traffic network into s mutually exclusive subnetworks such that $\boldsymbol{\alpha}_p = [\alpha_1 \ldots \alpha_s]$ where $\boldsymbol{\alpha}_i \in \mathbb{R}^{1\times n_i} \quad \forall i = 1, \ldots, s$. Then, compressed sensing can be performed for each subnetwork separately. The results of the clustered subnetworks may be merged to infer the traffic state of the entire network. A number of choices of temporal and/or spatial clustering is possible. In some embodiments, simple clustering based on different road categories may be applied.

The overall performance of the compressed sensing method 300 may be sensitive to the "compressibility" of the network and "non-stationarity" in the traffic data. For compression, the traffic data can be represented as a product of two low-rank matrices, i.e., the subnetwork data matrix and the most appropriate relationship matrix. As the compression is lossy, it is expected that the reconstructed matrix $\widehat{\mathbf{A}}_h$ will be different from the original historical data matrix $\mathbf{A}_h$. The non-stationarity is due to the fact that the matrix $\mathbf{X}_h$ is inferred from training (historical) data instead of the current data. The matrix $\mathbf{A}_p = \left[\boldsymbol{\alpha}_p^k \ldots \boldsymbol{\alpha}_p^m\right]^T$ is not available, and the goal is to infer that matrix by

extrapolating the matrix $\mathbf{C} = [\mathbf{c}_p^k \dots \mathbf{c}_p^m]^T$ according to the CX decomposition discussed above. Obviously, the matrix $\mathbf{X}_p$ cannot be extracted from the current data $\mathbf{A}_p$, since the matrix $\mathbf{A}_p$ is not available. Instead, $\mathbf{X}_h$ can be determined from the training data set $\mathbf{A}_h$. Since traffic is not perfectly stationary, this approximation will induce an additional reconstruction error. This is referred to herein as the error due to non-stationarity of traffic spatial relationships. To quantify this error, let us call $\mathbf{B} = \mathbf{C}_p\mathbf{X}_p$ the reconstruction of the data matrix $\mathbf{A}_p$, assuming the latter is available to compute the CX decomposition. The reconstruction $\hat{\mathbf{A}}_p = [\hat{\boldsymbol{\alpha}}_p^k \dots \hat{\boldsymbol{\alpha}}_p^m]^T$ in the scenario of compressed sensing is less accurate, since we need to replace $\mathbf{X}_p$ (determined from the test data matrix $\mathbf{A}_p$) by $\mathbf{X}_h$ (determined from training data matrix $\mathbf{A}_h$). The mean squared error (MSE) incurred for compressed sensing can be written as:

$$\frac{1}{rn}\|\mathbf{A}_p - \hat{\mathbf{A}}_p\|_F^2 = \frac{1}{rn}\|(\mathbf{A}_p - \mathbf{C}_p\mathbf{X}_p) - (\mathbf{C}_p\mathbf{X}_h - \mathbf{C}_p\mathbf{X}_p)\|_F^2, \quad (3)$$

$$= \frac{1}{rn}\|(\mathbf{A}_p - \mathbf{B}) - (\hat{\mathbf{A}}_p - \mathbf{B})\|_F^2, \quad (4)$$

$$= \frac{1}{rn}\Big(\sum_{i=k}^{m}\sum_{j=1}^{n}(a_{ij} - b_{ij})^2 + \sum_{i=k}^{m}\sum_{j=1}^{n}(\hat{a}_{ij} - b_{ij})^2 - 2\sum_{i=k}^{m}\sum_{j=1}^{n}(a_{ij} - b_{ij})(\hat{a}_{ij} - b_{ij})\Big), \quad (5)$$

where r represents the number of time instances in test data matrix $\mathbf{A}_p$. The first component of the error corresponds to the compressibility of the network and the second component is due to the non-stationarity of spatial patterns within the network (see (5)). The third component of the error refers to the correlations between aforementioned error components (see (5)). To make this interpretation more explicit, we rewrite (5) as:

$$\mathrm{MSE}_{\mathrm{est}} = \mathrm{MSE}_{\mathrm{com}} + \mathrm{MSE}_{\mathrm{ns}} - 2\xi_{\mathrm{est}}, \quad (6)$$

where ξ_{est} is the correlation coefficient between compressibility and non-stationarity. The behavior of these errors for different compression ratios is discussed below in the Experimental results section.

Notably, low-dimensional Principal Component Analysis (PCA) models cannot be used for compressed sensing, since PCA requires data from all sensors for both offline and online operations.

Real time compressed prediction 450

Analogously to compressed sensing process 300, real time prediction of large and diverse traffic networks can be performed by system 100. Broadly speaking, future traffic states are predicted for a subset of roads, for example using data driven algorithms such as support vector regression (SVR) or Bayesian Support Vector Regression (BSVR) to perform traffic multi-horizon prediction. Preferably, each prediction horizon is treated as an independent function estimation problem in order to provide modularity. A combination of offline training and online prediction, using temporal features, can be used. Mapping functions between input features and predicted values can be learned using a training (historical) data set.

Referring to Fig. 4, in certain embodiments, compressed prediction comprises an offline training process 400 and an online prediction process 450, which implements support vector regression (SVR). SVR was chosen for prediction, as it a commonly used algorithm to perform road traffic prediction.

Training process 400 is conducted over a subset of roads or road segments, for example the subset used to generate the relationship matrix $\mathbf{X}_h$ as outlined above. At step 402, the road segments of the subset are selected, and at step 404, historical data for the traffic parameter for the selected road segments are obtained. These historical data, for example, the current and past speed values observed on a particular road segment, can be used to train the SVR. Training data are generated from the historical data at step 406. For each time instance t in the historical data, training data can be generated as follows: define a vector of input features $\mathbf{x} = [v(t)\ v(t - dt)\ \ldots\ v(t - k.dt)]^T$ and a corresponding target value $y = v(t + k.dt)$. The value $v(t)$ is the speed at time t, $v(t - dt)$ the speed at the previous sampling interval $t - dt$, and so on. The $\mathbf{x}$ vectors and their corresponding y values are then used to train the SVR, at step 408, in a manner known in the art. The result of training process 400 is a set of SVR parameters 420 which can be used as input to prediction process 450. The training process 400 may be carried

out for multiple prediction horizons (e.g., three prediction horizons: 5 minutes into the future, 15 minutes into the future, and 30 minutes into the future) to determine respective sets of SVR parameters 420 for respective prediction horizons.

Once the SVR parameters 420 have been obtained, they can be stored by server 110 and used for prediction process 450, which may be executed by server 110. In other embodiments, server 110 transmits the SVR parameters and relationship matrix $\mathbf{X}_h$ to user devices 140 so that the user devices 140 can execute prediction process 450.

The prediction process 450 iterates (step 458) over the selected road segments (step 454) used in the construction of relationship matrix $\mathbf{X}_h$ in order to generate (step 456) a predicted value of the traffic parameter for each selected segment, for each prediction horizon. Accordingly, process 450 generates prediction data 460 indicative of a row vector $\hat{\mathbf{c}}_f^i \in \mathbb{R}^{1\times c}$ containing the predicted values of the traffic variable at the c selected locations at the i^{th} time instance. Then, at step 462, prediction data indicative of predicted values for the entire network are generated, extrapolating $\hat{\mathbf{c}}_f^i$ using the previously computed relationship matrix $\mathbf{X}_h$: $\widehat{\boldsymbol{\alpha}}_f^i = \hat{\mathbf{c}}_f^i \mathbf{X}_h, \forall i = k, \ldots, m$, where $\widehat{\boldsymbol{\alpha}}_f^i \in \mathbb{R}^{1\times n}$ contains the predictions for all locations at the i^{th} time instance.

Errors in the predicted values can be estimated according to the following procedure. We can write $\mathbf{c}_f^i = \hat{\mathbf{c}}_f^i + \Delta\mathbf{c}^i$ where $\Delta\mathbf{c}^i$ represents the prediction error for the subnetwork at time i. Furthermore, let $\mathbf{D} = \mathbf{C}_f\mathbf{X}_h$ be the estimated network profile, during the entire observational period, without any prediction error in $\mathbf{C}_f$ ($\mathbf{C}_f = \left[\mathbf{c}_f^k \ldots \mathbf{c}_f^m\right]^T$). Then, the MSE between predicted $\widehat{\mathbf{A}}_f$ ($\widehat{\mathbf{A}}_f = \left[\widehat{\boldsymbol{\alpha}}_f^k \ldots \widehat{\boldsymbol{\alpha}}_f^m\right]^T$) and true future values $\mathbf{A}_f$ ($\mathbf{A}_f = \left[\boldsymbol{\alpha}_f^k \ldots \boldsymbol{\alpha}_f^m\right]^T$) can be written as:

$$\frac{1}{rn}\|\mathbf{A}_f - \hat{\mathbf{A}}_f\|_F^2 = \frac{1}{rn}\|(\mathbf{A}_f - \mathbf{C}_f\mathbf{X}_h) - (\hat{\mathbf{C}}_f\mathbf{X}_h - \mathbf{C}_f\mathbf{X}_h)\|_F^2, \tag{7}$$

$$= \frac{1}{rn}\|(\mathbf{A}_f - \mathbf{D}) - (\hat{\mathbf{A}}_f - \mathbf{D})\|_F^2, \tag{8}$$

$$= \frac{1}{rn}\Big(\sum_{i=k}^{m}\sum_{j=1}^{n}(a_{ij} - d_{ij})^2 + \sum_{i=k}^{m}\sum_{j=1}^{n}(\hat{a}_{ij} - d_{ij})^2 - 2\sum_{i=k}^{m}\sum_{j=1}^{n}(a_{ij} - d_{ij})(\hat{a}_{ij} - d_{ij})\Big), \tag{9}$$

where the first component of the error corresponds to the non-stationarity of spatial patterns within the network and the second component is due to inaccurate predictions. The third component of the error shows the correlations between these two error components. We refer to (9) as MSE for compressed prediction. We rewrite (9) in more explicit form:

$$\mathrm{MSE}_{\mathrm{total}} = \mathrm{MSE}_{\mathrm{est}} + \mathrm{MSE}_{\mathrm{pred}} - 2\xi_{\mathrm{pred}}, \qquad (10)$$

where ξ_{pred} refers to correlation between non-stationarity and predictability. By substituting (6) in (10), we obtain:

$$\mathrm{MSE}_{\mathrm{total}} = \mathrm{MSE}_{\mathrm{com}} + \mathrm{MSE}_{\mathrm{ns}} + \mathrm{MSE}_{\mathrm{pred}} - 2\xi_{\mathrm{est}} - 2\xi_{\mathrm{pred}}. \quad (11)$$

Hence, the total error of compressed prediction can be decomposed into four error components: (i) error due to compression; (ii) error due to changes in spatial relationships (non-stationarity); (iii) error due to inaccurate predictions; (iv) correlations among the previous error components.

Compressed prediction provides significant reduction in computational complexity by explicitly predicting the traffic variables for only a small subset of road segments in the network. Compressed prediction involves two computations: (i) prediction of the traffic conditions at representative locations in the network and (ii) extrapolation of the predicted values to the entire network. In the former, the computational complexity depends on the underlying prediction algorithm, and is proportional to the number of locations c in the subnetwork. The second step (extrapolation) requires a single matrix-vector multiplication with complexity $O(cn)$. In practice, the predictions at each link in the subnetwork are computationally complex. By contrast, the extrapolation can be executed much faster. Therefore, by performing prediction only for a small subnetwork, the computational complexity can be drastically reduced.

De-centralized system architecture

In certain embodiments, traffic information is disseminated from server 110 to user's mobile devices 140. The end users are provided with the traffic information from a small subset of links (compressed states: current/predicted), as discussed above in relation to processes 300, 400, 450. The user's device 140 can then compute the traffic state (estimation/prediction) for the whole network with the help of predefined relationship functions stored in the device.

Embodiments of the method described above can form the core component for many traffic related and location based applications. Moreover, the method has a modular architecture. Hence, it can be easily implemented in distributed configurations. In particular, real-time compressed sensing 300 and real-time compressed prediction 450 can be implemented on client devices (mobile phones) 140. Relationship matrix $\mathbf{X}_h$ can be stored at users' mobile devices 140 for this purpose. By downloading accurately predicted speeds of a small subset of the links in the network, a user's cell phone 140 will be able to generate the current and future traffic states for the whole network. Calculated values can be graphically shown on a map on a display of the user's device 140.

User devices 140 may each have stored thereon mobile application program instructions and data which, when executed by a processor of the user device 140, cause it to perform real-time compressed sensing and real-time compressed prediction. An installation package of the mobile application may include the latest version of the extrapolation matrix $\mathbf{X}_h$. This matrix may be periodically updated by the server 110. Current and predicted speed values (or values of other traffic parameters) for selected links in the network can be provided on the user's demand. Then the user mobile phone 140 will be able to perform network-wide real time estimation / prediction, based on the data for the selected links and the extrapolation matrix.

Estimating reliability of prediction

The predicted traffic states are prone to uncertainty. Prediction accuracy may vary for different test conditions. Actual prediction error can only be calculated once field data

becomes available. For instance, consider a 15 minute prediction horizon. In this case, we can only calculate prediction error after 15 minutes. Consequently, applications which use prediction data remain vulnerable to variations in prediction error. Algorithms that can anticipate such variations in prediction accuracy will be highly desirable. These algorithms will allow ITS applications such as route guidance to take into account the uncertainty associated with predicted network state. In certain embodiments, therefore, the present invention may perform traffic prediction by employing Bayesian Support Vector Regression (BSVR). This technique combines the advantages of SVR and Bayesian inference. BSVR estimates the variance (termed as error bars) associated with each predicted state. The error bars provided by BSVR help us to identify predicted states with high uncertainty. It has previously been shown that BSVR can anticipate large prediction errors with reliable sensitivity and low false alarm rate. The error bars from BSVR can be used to inform end users about reliability of predicted traffic state for each time instance.

Experimental results of compression, compressed sensing and compressed prediction of urban traffic

To evaluate the methods described above, we consider a large traffic network in Singapore. The traffic network contains diverse types of roads, belonging to different categories (CAT A, B, ..., E). The data set contains the average speed at each link of a large transportation network for a period of three months (August - October 2011) . Traffic data was provided by the Land Transportation Authority (LTA). The sampling interval of provided data set is 5 minutes. We selected 17967 links which had less than 5% of missing values. Imputation was performed by applying the Low Dimensional CP Weighted OPTimization (LDCP-WOPT) imputation method as it is able to deal with the large data set (see M.T. Asif *et al.*, *Proc. 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3527-3531).

The data set was represented as a matrix $\mathbf{A}$ as explained above. The data were divided into a training set for selecting the subset of links, training the predictive model, and learning the relationship between the subset of links and the rest of the network. The August and September data were used as the training set and the remaining data used as a testing set. SVR was applied as described above (processes 400 and 450) to the

subset of links (compressed prediction) and to the whole network (uncompressed prediction).

We use percent root mean distortion (PRD) error to evaluate the reconstruction error of low-dimensional models. The percent root mean distortion (PRD) quantifies the reconstruction error:

$$\mathrm{PRD}(\%) = \frac{\| \mathbf{A} - \mathbf{CX} \|_F}{\| \mathbf{A} \|_F}.$$

An alternative measure of the reconstruction error is the mean absolute percentage error (MAPE) defined as:

$$\mathrm{MAPE} = \frac{1}{mn} \sum_i \sum_j \frac{|a_{i,j} - \hat{a}_{i,j}|}{a_{i,j}},$$

We use the mean squared error (MSE) to assess the impact of several error components on the overall error. MSE is defined as:

$$\mathrm{MSE} = \frac{1}{mn} \|\mathbf{A} - \mathbf{CX}\|_F^2.$$

The CX method for the compression of traffic data was evaluated. The above-described SVD sampling strategy was applied to find an appropriate set of columns. Sampling was repeated five times and the average reconstruction accuracy reported. Our benchmark is PCA, as it is considered as the optimal linear transformation.

Fig. 5(a) shows the compression performance 510 of the presently described embodiments contrasted with PCA (curve 512). As expected, PCA slightly outperforms the proposed method in terms of reconstruction error. However, as mentioned earlier, the low-dimensional model obtained by PCA cannot be used in compressed estimation and prediction, since it requires data from all the links to be used. In contrast, the low-dimensional models generated by the CX-method are perfectly suitable for both

applications, this being the main advantage of the presently described approach compared to PCA.

An important question about the presently described CX method is whether the sampling scheme leads to subnetworks that are stable over time. To assess the stability of the subnetwork generated by the SVD sampling method, it was applied to each of the three months (August, September, October 2011) of traffic data separately. For each month, the road segments were sorted in descending order according to the probability assigned by the SVD sampling method. Hence, the most representative roads were at the top of these lists. Next the first *k* links of each list were selected, with *k* corresponding to 10%, 25%, and 50% of the links in the network. If the subnetwork is stable across time, the three shortlists of top-*k* links should have many links in common. The results of this analysis are summarized in Table 1, where the percentage of common links is provided. As can be seen from this Table, most links in the subnetwork are consistently selected for all three months, suggesting that the SVD sampling method results in a subnetwork that is stable over time.

Table 1: Overlap (%) among the k links with the highest selection probability (calculated by the SVD sampling method) in the three months of data.

	Aug.	Sep.	Oct.
Aug.	100	93.88	92.26
Sep.	93.88	100	92.82
Oct.	92.26	92.82	100

(a) k = 10% of all roads

	Aug.	Sep.	Oct.
Aug.	100	93.61	91.67
Sep.	93.61	100	92.14
Oct.	91.67	92.14	100

(b) k = 25% of all roads

	Aug.	Sep.	Oct.
Aug.	100	94.65	93.12
Sep.	94.65	100	93.49
Oct.	93.12	93.49	100

(c) k = 50% of all roads

We also investigated the case of compressed sensing. The aim was to reconstruct the average speed at each link in the entire network by collecting data from a small subset of roads. The relationship matrix $\mathbf{X}_h$ was determined from the training set (data from Aug-Sep 2011), and the reconstruction error was assessed on the test set (data from October, 2011).

Fig. 5(b) shows the reconstruction accuracy of the presently described method for three different approaches: In the first approach, shown as curve 522, we selected the subset of road segments according to the SVD sampling scheme described above. In the second approach, shown as curve 524, we clustered the network according to the

category of the road. For each cluster, we selected the subset of the road segments using the SVD sampling scheme. Then, we performed compressed sensing for each cluster separately. In the third approach, shown as curve 520, we used the same set of roads as defined in the second approach to perform network estimation. Unlike in the second approach, we did not perform any clustering.

Intuitively, the difference between the second 524 and the third approach 520 shows the gain obtained by network clustering. Fig. 5(b) indicates that applying the compressed sensing method to different road categories leads to better estimation performance for the entire network. As expected, the reconstruction accuracy of all three approaches increases with the size of subnetwork.

We also investigated the error of compressed sensing in more detail. In the following, we consider the subset of links as defined in the SVD scheme without clustering (third approach). The overall compressed sensing (estimation) error is caused by information loss due to compression of traffic data and changes in traffic behavior between the training and testing periods. Table 2 shows the MSE of the individual error components, the correlation between the two error components and the total MSE, for different compression ratios. As can be seen from Table 2 the non-stationarity of the traffic data is the main contributor to the estimation error.

As expected, the error associated with the compressibility of traffic data increases with the compression ratio. Furthermore, Table 2 also shows that there is no correlation between compressibility and non-stationarity of traffic data.

Table 2: Prediction performance of the proposed and traditional methods.

	2	4	6	8	10
MSE (Comp)	0.01	2.18	7.27	11.95	15.93
MSE (Ns)	19.77	29.07	31.20	29.23	27.35
ξ(Corr)	0.00	0.00	0.00	0.00	0.00
Total MSE	19.78	31.25	38.47	41.18	43.28

In the compressed prediction process described earlier, the future state of a small subset of roads is used to predict a future traffic condition for the whole network. After

training (by process 400) the SVR on the training data, compressed prediction (process 450) was applied to the test data. This was compared to the traditional (baseline) approach in which the speed for every road segment is explicitly predicted. Fig. 6 and 7 depicts the prediction accuracy of the present and traditional methods for different prediction horizons and various compression ratios. In Fig. 6(a), for a 5 minute prediction horizon, the PRD error of compressed prediction is shown as curve 610 while the PRD error of traditional prediction is shown as horizontal line 612. Similarly, in Fig. 6(b), for a 30 minute prediction horizon, the PRD error of compressed prediction is shown as curve 620 while the PRD error of traditional prediction is shown as horizontal line 622. In Fig. 7(a), for a 5 minute prediction horizon, the MAPE error of compressed prediction is shown as curve 710 while the MAPE error of traditional prediction is shown as horizontal line 712. Similarly, in Fig. 7(b), for a 30 minute prediction horizon, the MAPE error of compressed prediction is shown as curve 720 while the MAPE error of traditional prediction is shown as horizontal line 722.

As expected, the compressed method has slightly larger PRD and MAPE error than the traditional approach. This additional error decreases with decreasing compression ratio. Also, the additional error decreases for large prediction horizons (see Fig. 6(b) and 7(b)). Naturally, it is difficult to predict for larger horizons even with the traditional approach (see line 622 in Fig. 6(b) and line 722 in Fig. 7(b)). Hence, in such cases the error due to prediction tends to become the dominant component.

We decomposed the MSE of compressed prediction into estimation (EST) and prediction (SVR) components. Table 3 shows the contribution of these two error components as well as the correlation between them for a 5 minute prediction horizon. As can be seen from Table 3, the estimation error increases with the compression ratio. This increase in estimation error is mainly due to non-stationarity of the error component (see Table 2). Table 3 shows that the prediction error tends to be dominant for smaller compression ratio, i.e., when a significant portion of the network is explicitly predicted. From Table 3, we can also see that there is some correlation between the two error components.

The presently described embodiments provide a compressed prediction process which provides substantial reduction in computational complexity, by explicitly predicting the variables for only a small representative set, followed by (fast and efficient)

extrapolation to the entire network. This reduction in computation time is obtained at the expense of a small increase in the prediction error (see Fig 6 and 7). The computation times for the compressed and traditional methods are shown in Table 4. For the purpose of benchmarking, we tested the compressed and uncompressed prediction algorithms on a 2.67 GHz MacPro server on a single core with 32GB of random-access memory (RAM). We assumed that training phases are performed offline for both methods. Prediction time for the compressed method involves the time to predict the traffic variable for a subset of links and the time required to perform network wide extrapolation. As Table 4 shows, the latter can be neglected. Consequently, the required computation time for compressed prediction is proportional to the number of the road segments in C. In other words, the reduction in computational complexity is to a good approximation proportional to the compression ratio of the compressed method (see Table 4).

Table 3: The MSE error of compressed prediction for different compression ratios.

	2	4	6	8	10
MSE (EST)	19.78	31.25	38.47	41.18	43.28
MSE (SVR)	33.73	28.23	24.15	20.95	18.61
ξ(Corr)	5.80	6.01	5.52	4.83	4.42
Total MSE	41.91	47.46	51.58	52.47	53.05

Table 4: Computation time (in seconds) of the compressed method for 5 minute prediction horizon. The traditional approach requires 91.63 sec to perform prediction for the whole network.

Compression Ratio	2	4	6	8	10
SVR	45.82	22.91	15.27	11.45	9.16
Matrix multiplication	0.33	0.24	0.15	0.12	0.11
Total	46.15	23.15	15.42	11.57	9.27
Complexity Savings	49.6%	74.7%	83.2%	87.4%	89.9%

Embodiments of the proposed method can be embedded in existing heterogeneous traffic databases such as Inrix and Google. The presently described method of large scale traffic prediction can be further used in traffic control centers. Accurate prediction of future traffic conditions on macro/city-scale brings great benefits for existing

intelligent transportation systems (ITS) applications such as Electronic Road Pricing (ERP) and Variable Message Signs (VMS). For example, if congestion is going to appear in a particular road/area, then traffic controllers will suggest (via VMS) or force (via ERP) drivers to avoid that road/area.

Embodiments of the invention may have one or more of the following advantages (as summarised in Table 5 below):

- Real time estimation and prediction (up to 1 hour into the future) for large and diverse traffic networks.
- Current and future traffic information for the entire network can be calculated at a user's mobile device. This allows the user to be timely informed about the potential congestion along his route.
- Unlike internet-enabled and social GPS methods, embodiments of the present method are independent of the number of probe sensors and their locations in the network. The present method can use any type of input traffic data representing values of a traffic parameter for respective road segments, but in certain embodiments can rely on real-time stream data, generated from inductive-loop detectors.
- High spatial coverage of the traffic loops allows network wide prediction. A real-time data stream can be provided for all the roads in the network.
- When loop detectors are the primary source of the real-time data stream, privacy is not an issue.
- The implementation and system architecture may be relatively simple. The real-time and predicted traffic information for the subset of roads can be sent to a user's mobile device, which can determine network traffic information for the rest of the network by extrapolation using relationship functions from the past (historical) data.
- Scalable and real-time prediction for the whole network. Some computations can be assigned to users' mobile devices to further reduce the overhead of the communication network.

Table 5: comparison of traffic information technologies

	In-car GPS device	Internet-enabled GPS	Social GPS maps	Proposed method
Representative	Garmin, Tom-Tom	G-maps^, Inrix^	Waze^	
Price	High	Low	Low	Low
Real time data	Yes*	Yes	Yes	Yes
Predicted data	No	No	No	Yes
Penetration rate	Low	High	Medium	Low
Coverage	High	Medium	Low	High
Privacy restriction	Medium	High	Low	Low

^ - mobile apps *-additional cost

5

Claims

1. A real-time compression method for predicting vehicular traffic conditions within a geographical area, the geographical area including a network of vehicular roads having subsets of roads, each subset of roads including a group of at least a segment of some of the roads which are selected as being suitable to represent the network of roads, the selection being based on weightage assigned to each road of the network; the assignment being based on correlation of historical vehicular traffic data of the network to respective roads of the network, the method comprising:

(i) providing spatial relations data indicative of a spatial relationship between the selected segments of some of the roads and the rest of the roads of the network;

(ii) obtaining current traffic conditions data by real-time detection of vehicular traffic conditions at the selected segments of the roads; and

(iii) predicting future traffic conditions for the network of roads within the geographical area from the current traffic conditions data at the selected segments of the roads and the spatial relations data.

2. A method according to claim 1, wherein the spatial relations data is stored in a plurality of mobile devices belonging to respective users, and the current traffic conditions data is received by each mobile device.

3. A method according to claim 1 or 2, further comprising displaying the predicted future traffic conditions on each mobile device.

4. A method according to any one of the preceding claims, wherein the weightages are obtained by singular value decomposition of a matrix formed from the historical vehicular traffic data.

5. A method according to claim 4, wherein the selected segments are selected by sampling using the weightages as probabilities.

6. A method according to any one of the preceding claims, wherein the prediction is performed using support vector regression or Bayesian support vector regression.

7. A method according to any one of the preceding claims, comprising determining current traffic conditions for the network from the current traffic conditions data at the selected segments of the roads and the spatial relations data.

8. A method according to any one of the preceding claims, wherein the spatial relations data represents a relationship matrix having X number of columns and Y number of rows; wherein

X = total number of road segments in the network, and

Y = total number of road segments in the selected road segments.

9. A computer-readable storage medium having stored thereon computer-executable instructions which, when executed by at least one computing device, cause the at least one computing device to perform the method according to any of claims 1 to 8.

10. A system for predicting vehicular traffic conditions within a geographical area, the geographical area including a network of vehicular roads having subsets of roads, each subset of roads including a group of at least a segment of some of the roads which are selected as being suitable to represent the network of roads, the selection being based on weightage assigned to each road of the network; the assignment being based on correlation of historical vehicular traffic data of the network to respective roads of the network, the system comprising:

at least one processor; and

at least one non-volatile storage device, the at least one non-volatile storage device having stored thereon spatial relations data indicative of a spatial relationship between the selected segments of some of the roads and the rest of the roads of the network;

the at least one processor being configured to:

(i) obtain current traffic conditions data from real-time detection of vehicular traffic conditions at the selected segments of the roads; and

(ii) predict future traffic conditions for the network of roads within the geographical area from the current traffic conditions data at the selected segments of the roads and the spatial relations data.

11. A system according to claim 10, wherein a first processor is configured to perform operation (i), and a second processor is configured to perform operation (ii).

12. A system according to claim 11, wherein the first processor is located at a server computer system, and the second processor is located at a mobile device in communication with the server computer system.

13. A system according to any one of claims 10 to 12, wherein the at least one non-volatile storage device comprises a plurality of storage devices of a corresponding plurality of mobile devices belonging to respective users, and wherein each of the plurality of storage devices have stored thereon the spatial relations data and the current traffic conditions data.

14. A system according to any one of claims 10 to 13, wherein the at least one processor is configured to predict the future traffic conditions using support vector regression or Bayesian support vector regression.

15. A system according to any one of claims 10 to 14, wherein the at least one processor is configured to determine current traffic conditions for the network from the current traffic conditions data at the selected segments of the roads and the spatial relations data.

16. A system according to any one of claims 10 to 15, wherein the spatial relations data represents a relationship matrix having X number of columns and Y number of rows; wherein

X = total number of road segments in the network, and

Y = total number of road segments in the selected road segments.

17. A column-based method for compressing vehicular traffic data within a geographical area, the geographical area including a network of vehicular roads, the column-based method comprising:

(i) correlating historical vehicular traffic data of the network to respective vehicular roads of the network;

(ii) assigning a weightage to each vehicular road of the network based on the correlation;

(iii) selecting at least a segment of some of the roads based on the assigned weightage which are suitable to represent the network of roads; and

(iv) determining spatial relations data indicative of a relationship between the selected segment of the roads and the rest of the roads of the network;

wherein current traffic conditions of the network are computable using current traffic conditions data indicative of current traffic conditions at the selected segments of the roads and the spatial relations data.

18. A column-based method according to claim 17, wherein assigning the weightage includes performing singular value decomposition of the historical vehicular data.

19. A column-based method according to claim 17 or 18, wherein the spatial relations data represents a relationship matrix having X number of columns and Y number of rows; wherein

X = total number of road segments in the network, and

Y = total number of road segments in the selected road segments.

20. A system for compressing vehicular traffic data within a geographical area, the geographical area including a network of vehicular roads, the system comprising: at least one processor; and a non-volatile storage device having stored thereon computer-readable program instructions which, when executed, cause the at least one processor to perform the method according to any one of claims 17 to 19.