Congestion Avoidance Algorithm using Extended Kalman Filter

Sung-Soo Kim
Digital Content Research Division
ETRI, South Korea
sungsoo@etri.re.kr

Yong-Bin Kang
Centre for Distributed Systems and Software Engineering
Monash University, Australia
yongbin.kang@infotech.monash.edu.au

Abstract

The prediction of traffic congestion is quite an important issue in vehicle navigation to smoothly control traffic flow, and improve the quality of driver's convenience. However, it is not easy to make accurate predictions since traffic change is highly nonlinear and complex dynamic process.

First, we present a new traffic prediction algorithm on the basis of the combined knowledge of both the historical and the real-time traffic information. Based on this traffic prediction result, this paper presents a novel routing technique capable of providing intelligent route services completely adequate to dynamic route guidance systems.

In our experiments, we have performed the proposed algorithms on two road networks; one of the complex urban areas and the city. Overall, the results of traffic prediction indicate that our prediction algorithms provide more accurate (nearly 90%) traffic information compared with previous traffic prediction solutions. In addition, our implementation of route determination provides the adaptive routes for traffic conditions, as well as scalable routing services for users' preferences.

1 Introduction

Vehicle navigation systems were originally designed as electronic enhancements of maps automatically indicating the current position of the vehicle and a route to the destination. Many research attempts have been conducted to design better navigation systems, by examining the variety of traffic information available. However, previous research efforts have revealed that implementation of navigation system with congestion avoidance is difficult since congested traffic exhibits very complex dynamic behavior [1, 6].

In order to provide practical navigation service, it is very important that we forecast the traffic congestion in the near future. This is because the quality of the route guidance can be strongly affected by the near future traffic events on different travel routes. The navigation systems in the market today have the technical limitation that for dynamic route guidance only current traffic and congestion information is usually considered. These systems recommend the route for the shortest estimated travel time based on the current state of traffic information. However, traffic conditions can change during a journey noticeably due to the dynamic nature of the traffic. So, traffic congestion can increase vehicle travel times in a traffic network considerably.

Moreover, traditional systems heavily depend on the historical traffic patterns to predict the traffic congestion. To predict the traffic congestion, we present a new traffic prediction algorithm that utilizes both historical traffic pattern and real-time traffic information to detect the near future traffic congestion.

Some of the novel contributions of our work include:

- *Accuracy.* A novel traffic prediction algorithm which is more accurate than traditional prediction methods which is based on historical traffic patterns.
- *Scalability.* A scalable route determination method to handle a variety of route preferences based on standard OpenLS specification by integrating with the proposed traffic prediction algorithm.
- *Adaptability.* A new navigation system architecture using above two algorithms for avoiding the traffic congestion under the dynamic traffic environments.

2 Related Work

This section briefly surveys related work in major three tasks in the navigation system. We can classify the general problem of dynamic route guidance into three categories: traffic estimation, traffic prediction, and route determination.

Traffic Estimation: Traffic estimation is an important part of an on-line road traffic management system. A key aspect of traffic estimation is the underlying the traffic models. Most of the existing traffic models can be classified as so-called *data-driven* and *macroscopic models* [2, 4, 5].

0-7695-3038-9/07 $25.00 © 2007 IEEE
DOI 10.1109/ICCIT.2007.147

In our work, we adopt the *macroscopic traffic models* [4] to model the traffic states, since the models provide full insight into the *physical meaning* of vehicular propagation and the cause of possible delays on the urban streets of interest.

Traffic Prediction: In general, the approaches for travel prediction can be divided into two classes in terms of facility type of interest: *freeways* and *urban streets* [10]. While prediction of travel time for freeways has been extensively studied in the literature, research on travel time prediction of urban street is quite limited. In our work, we propose the traffic prediction algorithm for one of the complex urban streets. The traffic prediction method also can be classified into three classes in terms of the traffic sources which are used for prediction [2]: methods which are based on *only historical traffic patterns*, methods which are based on historical patterns on the basis of *real-time traffic information*, and methods which perform dynamic traffic prediction with a *dynamic traffic state of the future*.

Route Determination: Shortest path algorithms optimize the costs of a journey in a network. The shortest paths problem in road networks has been the subject of extensive research for many years resulting in the publication of a large number of scientific papers [6, 7, 8].

In our work, we do not attempt to solve or propose yet another variation of the shortest path problems. Our research focus is to present a new route determination method based on the traffic prediction reflecting on dynamic change of real-time traffic information and historical traffic patterns. Therefore, we will introduce a new traffic prediction algorithms for complex urban streets. And then, a routing algorithm which is integrated with the proposed traffic prediction algorithm will be presented.

3 System Overview

In this section, we give an overview of the architectural features of congestion-avoidable navigation system in terms of traffic model for estimation, traffic prediction and route determination.

In order to estimate the traffic conditions, we use the method based on the existing sensors; induction traffic loops, monitoring cameras, road construction database, location data of GPS (Global Position System) equipped vehicles and buses. The loop sensors periodically transfer the following data to the central server in *traffic center*; loop data ID, time of transfer, acquisition time, location ID, number of vehicles per hour, average speed, and occupancy.

The proposed system consists of several subsystems; traffic center, point of interest (POI) server, routing server, traffic prediction server, and navigation client. A navigation client has a position unit (GPS receiver), a digital map, and communication functionality. The major functions of traffic center are client position computation using position determination equipment (PDE), traffic estimation, and manage the real-time traffic databases. The common services work

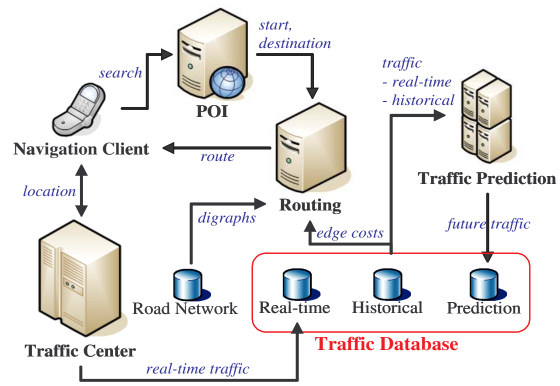

Figure 1. System architecture for the proposed navigation system.

as follows:

1. Navigation client requests the optimal route according to he/his start position and destination position.

2. Point of interest (POI) or directory server finds the location of a specific or nearest place, product or service. Then POI server or user passes starting location and destination to the routing server.

3. Routing server performs route determination using line digraphs according to the user's route preferences such as *shortest*, *fastest*, and *easiest* routes based on standard OpenLS specification [3].

4. Traffic prediction server compute the near future traffic and update prediction database periodically.

5. Traffic center periodically broadcasts the real-time traffic information to clients via communication networks.

The important three server-side tasks for our proposed navigation systems are *traffic estimation*, *traffic prediction*, and *route determination*.

1. **Traffic Estimation.** Estimate traffic state of the road network and then store traffic information to real-time traffic database every five minutes.

2. **Traffic Prediction.** Predict the near future traffic speed $\hat{v}_k$, which is evaluated by real-time speed v_k in real-time traffic database and historical traffic patterns $\tilde{v}_k$ in historical traffic database at kth time step.

Notation	Description
R_n	a given traffic network
T_r	real-time traffic database
T_h	historical traffic pattern database
T_p	traffic prediction database
G	a graph for a given traffic network
D	a line digraph for G
v_k	actual real-time speed of a link at kth time step
$\tilde{v}_k$	historical speed of a link at kth time step
$\hat{v}_k$	predicted speed of a link at kth time step
$v_k(i)$	v_k of ith link in D
$\tilde{v}_k(i)$	$\tilde{v}_k$ of ith link in D
$\hat{v}_k(i)$	$\hat{v}_k$ of ith link in D

Table 1. Notation used in this paper.

3. **Route Determination.** Perform the Dijkstra's algorithm to guarantee the optimal route for each line digraph D according to route preference. In this task, each line digraph is periodically updated according to the traffic prediction databases.

Our research focus is the improvement of traffic prediction and route determination in order to provide more effective dynamic service to user.

4 Algorithms

In this section, we present the detail of a novel congestion avoidance algorithm for providing the effective routes. To facilitate subsequent discussion, we summarize the relevant notational convention in this paper in Table 1.

4.1 Quality Control for Traffic Estimation

In most cases, we can get real-time traffic information (v_k) such as link speeds of all links in G. However, a few data is invalid value which were not measured correctly due to acquisition errors such as sensor failures. We call these invalid data as *missing edges* (or links) in G. Therefore, we should control these edges to improve the traffic data quality. In our experiments, the missing edges become nearly 2% of total traffic data in each acquisition interval (5 min.). Two kinds of missing edges can be measured according to the extent of these edges - (a) *one specific missing edge* and (b) *all missing edges can be found in a specific region.*

In order to predict $\hat{v}_k$, we measure not only $\tilde{v}_k$ simply, but through v_k and $\tilde{v}_k$. If v_k is similar to that of the of regular cases then, the result of $\hat{v}_k$ will be obtained by the difference between v_k and $\tilde{v}_k$. Otherwise, we apply a method on the basis of Kalman Filter for finding a practically useful convergence by combining updates with v_k and $\tilde{v}_k$. Particularly, it should be taken notice of that *unexpected situations* (e.g., traffic congestion, road construction, traffic accident) are found to provide these situations to drivers accurately. To find out such edges, a comparison method between v_k and $\tilde{v}_k$ was required.

Let $v_k(i)$ be v_k and $\tilde{v}_k(i)$ be $\tilde{v}_k$, where a connected edge i in G. Then, unexpected situations for all i can be detected by the following expression,

$$|v_k(i) - \tilde{v}_k(i)| > \delta, \quad (1)$$
$$\text{where } \delta = \alpha * \sqrt{\frac{\sum_{d=1}^{n} (\tilde{v}_k(i) - \tilde{v}_{k-d}(i))^2}{n}}$$

where δ is adjusting threshold which is believed as boundary of unexpected traffic situations and constant α represent average speed deviation within a given total periods of time n.

In this present work, there are two possible traffic prediction models will be defined differently, in the case of when these unexpected situations appeared or not.

4.2 Traffic Prediction based on Historical Traffic Patterns

In this section, we describe the method that predict the future traffic condition, which is applied when a deviation between v_k and $\tilde{v}_k$ can be regarded as reasonable small. We found that each link had similar speed pattern for the same day and time interval.

Based on the facts that have been observed so far, traffic patterns which were classified by the same periods of days and times have been shown quite similar aspect. Therefore, we attempt to define the prediction speed on each i at prediction time $k+d$ as a future value by using the following equation,

$$\hat{v}_{k+d}(i) = v_k(i) \pm f(i), \quad (2)$$
$$\text{where } f(i) = \beta * \sqrt{\frac{\sum_{d=0}^{m} (v_k(i) - \tilde{v}_k(i))^2}{m}}$$

where $\beta = v_k(i) - \sigma(v_k(i))$ is average speed deviation within a given total period of time m. The $\sigma(v_k(i))$ denotes a standard error of the link i obtained from the historical data. The result of this evaluation turned out to be proceed smoothly and provide desired prediction speed. The details of prediction algorithm based on historical patterns is shown in **Algorithm 1**.

4.3 Traffic Prediction using Extended Kalman Filter

We now introduce a method that makes use of a novel approach based on Kalman filter for traffic prediction. The idea is to use state space models from dynamic linear systems theory to capture the evolution of a system. Kalman

Algorithm 1 Prediction based on historical patterns

for each time k=0 to m **do**
  for each link i=0 to n **do**
    $v_k(i)$ = *getRealTimeSpeed*(k, i);
    $\tilde{v}_k(i)$ = *getHistoricalPattern*(k, i);
    β = *getDeviation*$(v_k, \tilde{v}_k)$;
    $f(i)$ = *computeChange*$(\beta, v_k, \tilde{v}_k)$;
    if *IsIncreasing*$(v_k, \tilde{v}_k)$ **then**
      $\hat{v}_k(i) = v_k(i) + f(i)$;
    else
      $\hat{v}_k(i) = v_k(i) - f(i)$;
    end if
  end for
end for

filtering is a powerful method because it can be used not only for estimation but also for prediction. The mathematical model used in the derivation of the Kalman filter is a reasonable representation for many problems of practical interest, including control problems as well as estimation problems. The Kalman filter model is also used for the analysis of measurement and estimation problems.

In order to predict more accurate speed of near future, the *Extended Kalman Filter* (EKF) is adapted to the proposed algorithm. Input to the EKF are the current day and time and historical traffic patterns T_h. Parameters of the EKF are explained in Table 2. The EKF process is composed of two groups of equations: *predictor* (time upate) and *corrector* (measurement update). The *predictor* updates the previous $(k-1)$th state and its uncertainty to the predicted values at the current kth time step. The $n \times n$ matrix A relates the state at the previous time step $k-1$ to the state at the current step k, in the absence of either a driving function of process noise. The $n \times l$ matrix B relates the optional control input $u \in R^l$ to the state x.

$$\begin{aligned} \hat{x}_k^- &= A\hat{x}_{k-1}^- + Bu_{k-1} \\ P_k^- &= AP_{k-1}A^T + Q, \\ \hat{z}_k &= \vec{h}(\hat{x}_k^-, d_k, t_k). \end{aligned}$$

Corrector equations correct the predicted state value $\hat{x}_k^-$ based on the residual of actual measurement z_k and measurement estimate $\hat{z}_k$. The Jacobian matrix linearizes nonlinear measurement function:

$$\begin{aligned} K_k &= P_k^- H_k^T (H_k P_k^- H_k^T + R)^{-1}, \\ \tilde{z}_k &= z_k - \hat{z}_k, \\ \hat{x}_k &= \hat{x}_k^- + K_k \cdot \tilde{z}_k, \\ P_k &= (I - K_k H_k) P_k^-. \end{aligned}$$

The details of prediction algorithm using extended Kalman Filter is shown in **Algorithm 2**.

Param.	Description
d_k	a day of the week at kth time step
t_k	a time of the day at kth time step
z_k	*measurement*; link speed at kth time step
$\hat{z}_k$	*measurement prediction*; predicted measurement at kth time step
$\tilde{z}_k$	*residual*; $\tilde{z}_k = z_k - \hat{z}_k$
x_{k-1}	real value of a link speed
$\hat{x}_{k-1}$	*state*; filter's estimate of the state value at $(k-1)$th time step $\hat{x}_{k-1} = E(x_{k-1} \| Z_{k-1}, D_{k-1}, t_{k-1})$
$\hat{x}_k^-$	*state prediction*; predicted state estimate at kth time step given measurements up to $(k-1)$th step $\hat{x}_k^- = E(x_k \| Z_{k-1}, D_{k-1}, t_{k-1})$
A	a $n \times n$ matrix which represents spatial relationship among links.
Q	*process noise* a 2×2 covaiance matrix
R	*measurement noise* a 2×2 covaiance matrix
P_{k-1}	*state uncertainty*; 2×2 uncertainty covariance matrix at $(k-1)$the time step, $P_{k-1} = E[(x_{k-1} - \hat{x}_{k-1})(x_{k-1} - \hat{x}_{k-1})^T]$,
P_k^-	*state uncertainty prediction*; predicted state uncertainty at kth time step given measurement up to $(k-1)$th time step, $P_k^- = E[(x_k - \hat{x}_k^-)(x_k - \hat{x}_k^-)^T]$
$\vec{h}$	*measurement function*; returns the speed
$\hat{z}_k$	*measurement estimate*; the current speed estimate given the day and time, $\hat{z}_k = \vec{h}(\hat{x}_k^-, d_k, t_k)$,
H_k	*Jacobian* Jacobian matrix of $\vec{h}$
K_k	*Kalman Gain*; 2×2 matrix.

Table 2. Parameters used in the Extended Kalman Filter.

5 Performance Evaluation

In this section, we present details of our experimental results and evaluate its performance. To evaluate the proposed approach, we forecast the average speed of cars on a link including crossroads in the area of Kang-Nam in Seoul, Korea for up to three hour ahead. The target map included 598 intersections and 821 links, and 21,538 cars were initially arranged on the road network.

The three prediction methods according to the traffic data sources for prediction were evaluated for comparative analysis as follows :

1. **P**(PATTERN) - by the only historical traffic pattern data
2. **PR**(PATTERN+REAL-TIME) - by the historical traffic pattern and real-time data

Algorithm 2 Prediction using Kalman Filter

initializeFilter();
for each time k=0 to m **do**
  $\tilde{h}_k$ = *getHistoricalPattern*(d_k, t_k);
  for each link i=0 to n **do**
    // **predictor**
    $\hat{x}_k^-$ = *predictSpeed*($\hat{x}_{k-1}^-$, $\tilde{h}_k$);
    P_k^- = *predictErrorCovariance*(P_{k-1});
    // **corrector**
    K_k = *computeKalmanGain*(P_k^-, H_k^T, H_k, R);
    $\hat{x}_k$ = *updateEstimate*($\hat{x}_k^-$);
    P_k = *updateErrorCovariance*(P_k^-);
  end for
end for

3. **PRK**(PATTERN+REAL-TIME+KALMAN) - by the historical traffic pattern, real-time data with Kalman filtering

Three different measures of effectiveness are used in this research for evaluating the performance of the traffic prediction: average deviation error, root mean squared error (RMSE), and min/max error. To analyze the accuracy of these cases, the average deviation errors on all the edges are given by the following expression,

$$\varepsilon(k) = \frac{\sum_{i=1}^{n} |\hat{v}_k(i) - v_k(i)|}{n} \qquad (3)$$

which denotes a average value of the difference between the actual and the prediction speed at time k.

We measured a traffic network in every 5 minutes intervals during hours (08:00 $\sim$ 09:00) to obtain a real-time speeds and predicted in every 5 minutes and the length of prediction time were 3 peak hours (08:05 $\sim$ 11:05). The experimental results are obtained through investigating for 4 weeks at the 95% confidence interval. Figure 2 shows the corresponding results of our experiments, which illustrates the rates of edges (0 $\sim$ 25%) in the average deviation of errors between the actual speed v_k and the prediction speed $\hat{v}_k$ (1km $\sim$ 20km). As you can see in this graph, the rates for **PRK** were always at higher level(about 12%, 23%, 17%) than the others, PR (about 8%, 23%, 12%) and **P** (about 4%, 18%, 16%) between 1km to 3km, during the lowest scope of errors, while nearly the same for the rest of scope of errors (4km $\sim$ 20km). These indicate that the rate of edges for **PRK** gives us minimal deviation error between v_k and $\hat{v}_k$ and can provide higher trustworthy prediction model than the two others; **P** and **PR**. We also compared the performance of accuracy among three methods by investigating the RMSE between v_k and $\hat{v}_k$.

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{v}_k(i) - v_k(i))^2} \qquad (4)$$

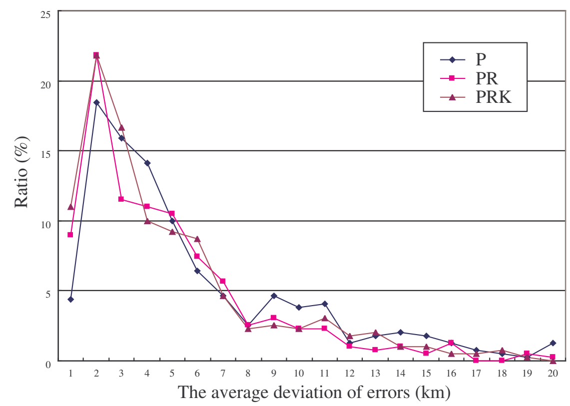

Figure 2. Comparison results of the three prediction methods.

The result shows that the accuracy of the **PRK** improves rather than others. As shown in Fig. 3, the RMSEs of the **PRK** are between 5 and 8 km. In order to evaluate the **PRK**

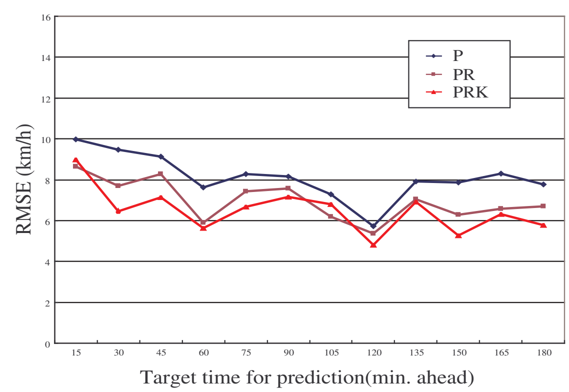

Figure 3. Experimental results correspond to times.

in detail, we measured the maximum and minimum errors for traffic prediction time step. Fig. 4 shows the maximum error for prediction. As we can see in this graph, although the **PRK** method has an maximum error at 15 minute time step for prediction, it will adjusts the traffic situations reducing the prediction error from 30 minute time step.

6 Conclusions and Future Work

We have presented novel traffic prediction algorithms in vehicle navigation systems. We also proposed a new route plan to guide a fastest route from a given destination, which based on future traffic prediction. The data structure of a

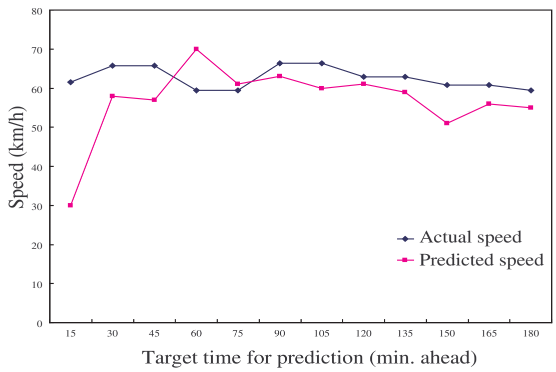

Figure 4. Maximum prediction error for the PRK method.

liner dual graph is the our primal structure which represents all pair of consecutive edges and allow individual weighting. We could minimize travel time cost and solve the turn problem that may be occurred at the intersection area with the graph.

To predict the future traffic condition, we first measured real-time traffic condition for reflecting on dynamic changing of current traffic flows. The traffic prediction model on the basis of real-time and cumulative traffic patterns was proposed to provide more realistic and accurate traffic information to drivers. If the measured real-time velocity was permitted within satisfiable threshold, traffic prediction was determined by adopting the cumulative traffic patterns. Otherwise, Extended Kalman filter method was adopted in case of the traffic congestion or unexpected circumstances.

We finally combined predicted traffic velocity and road topology for generating reliable weight cost function, and Dijkstra's algorithm was applied to travel a fastest route. The results of our works are supposed to provide driver with truly valuable traffic information of near future. Furthermore, this work may also be extended to the fields in the services of LBS solution and telematics.

Acknowledgements

This work was supported by the IT R&D program of MIC/IITA. [07MC1610, Development of Function Extensible Real-Time Renderer]

References

[1] B. S. Kerner, C. Demir, R. G. Herrtwich, S. L Klenov, H. Rehborn, M. Aleksić, A. Haug, DaimlerChrysler AG, Traffic State Detection with Floating Car Data in Road Networks, In *Proceedings of the 8th International IEEE Conference on Intelligent Transportation Systems*, pp. 700-705, 2005.

[2] B. S. Kerner, H. Rehborn, M. Aleksić, A. Haug, DaimlerChrysler AG, Traffic Prediction Systems in Vehicles, In *Proceedings of the 8th International IEEE Conference on Intelligent Transportation Systems*, pp. 251-256, 2005.

[3] OpenLS Initiative, OpenLS Route Determination Service Specification, *OpenGIS Interoperability Program Report* (OGC 02-090), Open GIS Consortium Inc., 11 Nov. 2002.

[4] Hitoshi Kanoh,Takeshi Furukawa, Souichi Tsukahara, Kenta Hara, Hirotaka Nishi, and Hisahi Kurokawa, Short-Term Traffic Prediction Using Fuzzy C-Means and Celluar Automata in a Wide-Area Road Network, In *Proceedings of the 8th International IEEE Conference on Intelligent Transportation Systems*, pp. 984-988, 2005.

[5] Lyudmila Mihaylova, René Boel, A Particle Filter for Freeway Traffic Estimation, In *Proceedings of the 43rd International IEEE Conference on Decision and Control*, pp. 2106-2111, 2004.

[6] Michael Mahut, An Heuristic Algorithm for Simulation-based Dynamic Traffic Assignment, In *Proceedings of the 8th International IEEE Conference on Intelligent Transportation Systems*, pp. 239-244, 2005.

[7] Stefano Pallottino, Maria Grazia Scutella, Shortest Path Algorithms in Transportation Models: Classical and Innovative Aspects, Technical Report, Univ. of Pisa, 1998.

[8] Stephan Winter, Modeling Costs of Turns in Route Planning, *GeoInformatica*, Vol. 6, No. 4, pp. 345-360, 2002.

[9] Hao Liu, Henk J. van Zuylen, Hans van Lint, Yusen Chen, and Ke Zhang, Prediction of Urban Travel Times with Intersection Delays, In *Proceedings of the 8th International IEEE Conference on Intelligent Transportation Systems*, pp. 1062-1067, 2005.

[10] Taehyung Kim, Hyoungsoo Kim and David J. Lovell, Traffic Flow Forecasting: Overcoming Memoryless Property in Nearest Neighbor Non-Parametric Regression, In *Proceedings of the 8th International IEEE Conference on Intelligent Transportation Systems*, pp. 965-969, 2005.