A Real-Time Passenger Flow Estimation and Prediction Method for Urban Bus Transit Systems

Jun Zhang, Dayong Shen, Lai Tu, Fan Zhang, Chengzhong Xu, Yi Wang, Chen Tian, Xiangyang Li, *Fellow, IEEE*, Benxiong Huang, and Zhengxi Li

***Abstract*— Bus service is the most important function of public transportation. Besides the major goal of carrying passengers around, providing a comfortable travel experience for passengers is also a key business consideration. To provide a comfortable travel experience, effective bus scheduling is essential. Traditional approaches are based on fixed timetables. The wide adoptions of smart card fare collection systems and GPS tracing systems in public transportation provide new opportunities for using the data-driven approaches to fit the demand of passengers. In this paper, we associate these two independent data sets to derive the passengers' origin and destination. As the data are real time, we build a system to forecast the passenger flow in real time. To the best of our knowledge, this is the first paper, which implements a system utilizing smart card data and GPS data to forecast the passenger flow in real time.**

***Index Terms*— Real-time, estimation, prediction, urban bus transit systems.**

Manuscript received December 16, 2015; revised May 7, 2016, September 30, 2016, and December 27, 2016; accepted March 10, 2017. Date of publication April 12, 2017; date of current version October 30, 2017. This work was supported in part by the China 973 Program under Grant 2015CB352400; in part by the Research Program of Shenzhen under Grant JSGG20150512145714248, Grant KQCX2015040111035011, and Grant CYZZ2015040311101266; in part by the National Natural Science Foundation of China under Grant 61602194, Grant 61402198, and Grant 61321491; in part by the Collaborative Innovation Center of Novel Software Technology and Industrialization; in part by the Jiangsu Innovation and Entrepreneurship (Shuangchuang) Program; in part by the China National Funds for Distinguished Young Scientists under Grant 61625205; and in part by the Key Research Program of Frontier Sciences, CAS, under Grant QYZDYSSW-JSC002, Grant NSF CMMI 1436786, and Grant NSF CNS 1526638. The Associate Editor for this paper was Y. Gao. *(Corresponding author: Lai Tu.)*

J. Zhang is with the School of Electronic Information and Communications, Huazhong University of Science and Technology, Wuhan 430074, China, and also with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China.

D. Shen is with the Research Center for Computational Experiments and Parallel Systems, National University of Defense Technology, Changsha 410073, China.

L. Tu, Y. Wang, and B. Huang are with the School of Electronic Information and Communications, Huazhong University of Science and Technology, Wuhan 430074, China (e-mail: tulai.net@gmail.com).

F. Zhang is Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China.

C. Xu is with Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China, and also with the Department of Computer Science, Illinois Institute of Technology, Chicago, IL 60616 USA.

C. Tian is with the State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210093, China.

X. Li is with the University of Science and Technology of China, Hefei 230026, China, and also with the Illinois Institute of Technology, Chicago, IL 60616 USA.

Z. Li is with the Department of Automation, North China University of Technology, Beijing, China.

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/TITS.2017.2686877

I. Introduction

BUSES are the most widely used public transportation in many cities today.

To improve the quality of bus service, a real-time system that can monitor and predict the *Passenger Flow* of the running buses is helpful. Here, *Passenger Flow* denotes the number of on-board passengers of a bus, which varies over time and space. The passenger flow can partially reflect the collective human mobility along a route and the quality of bus service in term of comfort. From a scheduling perspective, it tells you how many people travel or want to travel on a route. This information can guide the operators to allocate and schedule the bus route and timetable dynamically in fine granularity.

Current practice in Bus Transit System (BTS) operators demonstrates that manual data-collection efforts are costly and usually applicable only in small scale. The use of automated data-collection systems grow rapidly and show great potential. Automatic Fare Collection (AFC) devices that can record payments of riders using smart card, and a GPS embedded On Board Unit (OBU) that can track the bus are widely deployed. With the mature of big data systems, we have the opportunity to estimate and predict the passenger flow of every bus in urban wide BTS.

To depict the problem more clear, we can consider a concrete example as shown in Figure 1. Several buses operate in a line of route where we assume that no passing occurs among them along their whole trips. Passengers get on and off at each station, which changes the passenger flows of the buses over time and location. The solid lines and circles illustrate the segments and stations that the buses already travelled before current time, and the dash lines represent the remainder of the trips they will travel. The problem is that given the real time data of AFC transaction records and the OBU traces of the buses, how to estimate the number of riders on each bus and how to predict the number in the remainder of the trip in the near future.

While the problem looks like a straight forward counting job, the solution is not easy due to the limitation of available data. The challenges may lie in the following status quos:

- Current on bus devices do not offer a facility that can automatically and precisely count the number of the passengers getting on and off the bus, which may also be impractical to make it widely by human field investigations.

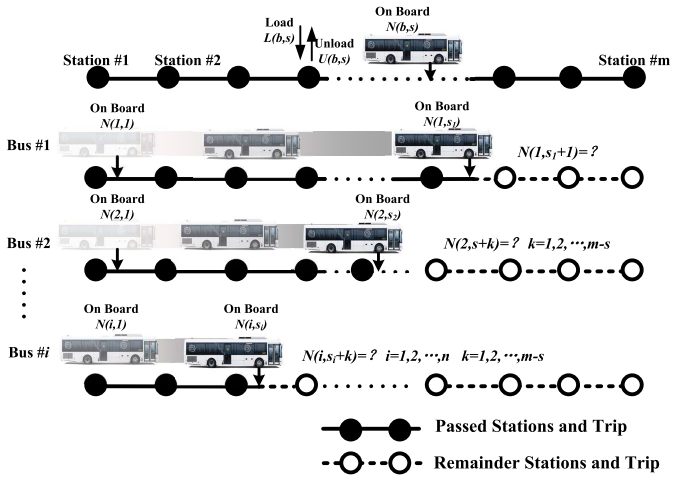


Fig. 1. An illustration of the real time passenger flow estimation and prediction problem in BTS.

- Automatically collected data such as AFC records and GPS can be useful but not adequate. Some facts cannot be observed directly, such as a passenger's getting off or someone paying by coins. So we need to make proper estimation rather than just counting.
- Finally, due to the uncertainty of people's mobility, it is also challenging to predict the passenger flow of future.

In summary, real-time passenger flow estimation and prediction in BTS are important but to our best knowledge, there is no existing approach for this problem. In this paper, we develop a system to estimate and predict the passenger flow in term of the number of riders in real-time by analyzing the AFC records and the bus GPS data. To make this possible, we make the following contributions:

- Firstly, we estimate the number of the riders getting on at each station. We derive the boarding position of a passenger by querying the GPS trace dataset with the taping time as a key. By analyzing the intervals between every two consecutive taping recodes, we derived an approximate estimation of the number of these passengers.
- Secondly, we estimate the passengers' alighting stations from the boarding records of their transit trips and of their return trips if they show regular commuter patterns, a.k.a. the *Trip Chain* analysis. For the passengers paying by coins and those whose next trips cannot be identified, we use the distribution of the alighting stations of all smart card users to approximate the probability of a coin user alighting at a station.
- Thirdly, with the real-time estimation of the number of the passengers on a bus, we further predict the number of the passengers that will be on the bus at its remainder stations.

We apply our method in several routes of buses in the city of Shenzhen and compare them with some baseline algorithms. The results show that our solution is compatible with different routes and outperforms other baseline algorithms. The rest of the paper is organized as follow: After a brief introduction of the related work in Section II, we give an overview of the problem and our solution in Section III. In Section IV, we focus on the real time estimation method and followed by Section V, which revolves around the prediction algorithm. Finally we introduce the setting for applying our solution in the city of Shenzhen and evaluate the performance in Section VI and end up the paper with conclusion in Section VII.

II. Related Work

The wide adoption of IoT provides new opportunities for using the data driven approaches in Intelligent Transportation System (ITS) [1]. As an IoT device in the public transportation system [2], the smart card can be identified by a unique serial number. Every time a smart card is taped, details of the transaction are recorded. The OBUs, usually with GPS tracing devices, can record the physical position of the vehicle at different time. Fusing the two types of data can help us to estimate the crowding in the bus [3].

Short-term traffic forecasting [4], [5] and short-term passenger demand forecasting [6] are successful applications of short-term transportation forecasting in the literature. The short-term transportation forecasting approaches can be generally divided into two categories: parametric and non-parametric techniques [4], [7], [8].

In the traditional parametric techniques, historical average [9], smoothing techniques [10], and autoregressive integrated moving average (ARIMA) [11] have been applied to forecast transportation demand. Particularly, ARIMA has become one of the common parametric forecasting approaches since the 1970s. The ARIMA model has been widely applied in forecasting short-term traffic data such as traffic flow, travel time, speed, and occupancy [12], [13]. In addition, with the characteristics of seasonality and trends in traffic data, some researchers have applied seasonal ARIMA to predict traffic flow [14], [15] and international air passenger flow [16], [17]. As stated in Brooks [18], ARIMA performs well and robustly in modeling linear and stationary time series. However, the applications of ARIMA or seasonal ARIMA models are limited because they assume linear relationships among timelagged variables so that they may not capture the structure of non-linear relationships [19].

For the non-parametric techniques, several methods have been used to forecast the transportation demand such as neural networks [4], [20], non-parametric regression [7], [21], Kalman filtering models [22], and Gaussian maximum likelihood [23]. Among these non-parametric techniques, neural networks have been frequently adopted as the modeling approach because they possess the characteristics of adaptability, nonlinearity and arbitrary function mapping capability [19]. Essentially, neural networks can deal with complex non-linear problems without a priori knowledge regarding the relationships between input and output variables [19]. Recently, several previous studies have developed neural network based models for traffic and transportation forecasting which include multilayer perceptron neural networks [9], Kalman filter based multilayer perceptron [24], time-delay neural networks [25], radial basis function neural networks, dynamic neural networks, state-space neural networks, and the support vector machine for regression, etc.

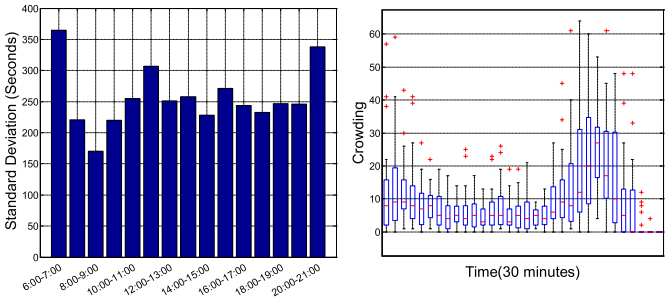


Fig. 2. The standard deviation of arrival time and the passenger flow fluctuations.

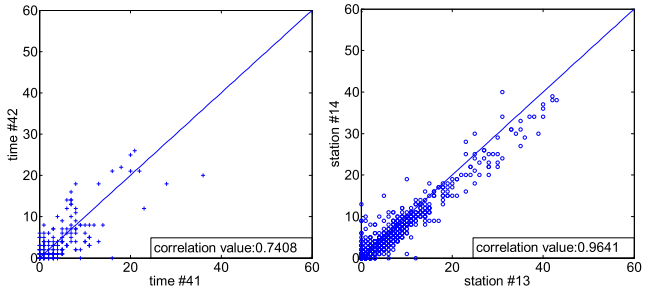


Fig. 3. Temporal and Spatial correlations in the number of riders on bus.

III. Overview

In this section, we first illustrate the motivation of a real time passenger flow estimation and prediction system for BTS. Then based on the ability of the deployed automatical data collection devices in current BTS, we formulate the problem as a model to estimate and predict a time series values of matrices and finally present a overview of the work flow of the system.

A. Motivation

The motivation of a real time passenger flow estimation and prediction system lies in the uncertainty existing in urban wide mobility and traffic flow. The left plot in Figure 2 shows the different periods' standard deviation of a line's arrival time in 15 days. We can see in different periods, the arrival time varies greatly. The max standard deviation can reach 365 seconds. Such fluctuation can be observed over time and stations in the collected data in current BTS. The right plot in Figure 2 shows the box plot of passenger flow of the buses at one stations over time. We divide the operation hours of a bus line into periods of 30 minutes and investigate the passenger flow in the periods in different days. We found that in most periods, the passenger flow of the same periods in different days can be quite different.

The correlation between adjacent buses is considered as the representation of *Temporal and spatial smoothness* of series. We can imagine the passenger flow at a certain station increasing or decreasing gradually. The correlations are found in our preliminary results of the number of riders calculated from historical data. We show two cases at a certain time and a station in Figure 3. The horizontal axis represents the passenger flow at time t or at station n and the vertical axis represents the passenger flow at time $t+1$ or at station $n+1$. We can see that the values in the adjacent time slots and stations have correlations.

With such a real time bus passenger flow estimation and prediction, a lot of potential applications both for bus operators and passengers can be developed upon it. For one example, we embedded the passenger flow information in a Bus Tracker App. Like traffic monitoring functions in map application, such additional information can improve the App users' experience by eliminating their curiosities about coming buses' crowding and also help them to decide which bus to get on. All of these applications can be done based on this system.

B. Problem Formulation

Based on the motivation and preliminary correlation analysis, the estimation and prediction are important and feasible in running BTS. To ease the representation, we formulate the problem as estimation and prediction of values of time series as follows:

$N(i, j)$ denotes the number of passengers on Bus $\sharp i$ at Station $\sharp j$ after the bus already loaded and unloaded at the station. $L(i, j)$ denotes the number of passengers boarding the bus, which further composes from two parts, i.e., $L(i, j) = L_s(i, j) + L_c(i, j)$. L_s and L_c stand for the numbers of riders paying by Smart cards and Coins respectively. In current BTS with AFC, L_s can be observed while L_c can only be estimated. $U(i, j)$ denotes the number of passengers alighting the bus. As the passenger's alighting behavior is not observable from the AFC data, we decompose $U(i, j)$ as $U(i, j) = U_h(i, j) + U_p(i, j)$. Here U_h denotes the number of passengers whose destinations are inferred from their historical trip chain pattern and U_p denotes the number of passengers whose destinations are estimated based on a probability model. We will discuss the estimation method in Section IV.

$$\tilde{\mathbb{N}}(t) = \begin{pmatrix} \tilde{N}_{11} & \tilde{N}_{12} & \dots & \tilde{N}_{1,k} & \hat{N}_{1,k+1} & \dots \\ \tilde{N}_{21} & \tilde{N}_{22} & \dots & \hat{N}_{2,k} & \hat{N}_{2,k+1} & \dots \\ \vdots & \vdots & \ddots & \ddots & \vdots & \vdots \\ \tilde{N}_{i,1} & \dots & \tilde{N}_{i,j} & \hat{N}_{i,j+1} & \vdots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \vdots & \vdots \end{pmatrix}_{B \times S} \quad (1)$$

Based on above notation, it is obvious that $N(i, j) = N(i, j-1) + L(i, j) - U(i, j)$. Suppose there are B bus services from the first bus service to the last one on a day, and the number of stations is S. The $N(i, j)$ will form a $B \times S$ matrix which we denote $\mathbb{N}$. At a certain time t, if the Bus $\sharp i$ has passed Station $\sharp j$ and has not reached Station $\sharp j + 1$. We estimate the values $\tilde{N}(i, k)$ for all $k \leq j$ and predict the values $\hat{N}(i, k)$ for all $k > j$. Therefore we get a time series of partially estimated and partially predicted matrices $\tilde{\mathbb{N}}(t)$ as Eq. 1, where our goal is to make estimation and prediction errors of $\tilde{\mathbb{N}}(t)$ to the true $\mathbb{N}$ as small as possible.

C. The Architecture and Datasets

The architecture of the estimation and prediction system is shown as Figure 4. Three datasets are involved for estimation and prediction. The table fields and descriptions are elaborated in Table I. The GPS dataset contains the GPS coordinates of

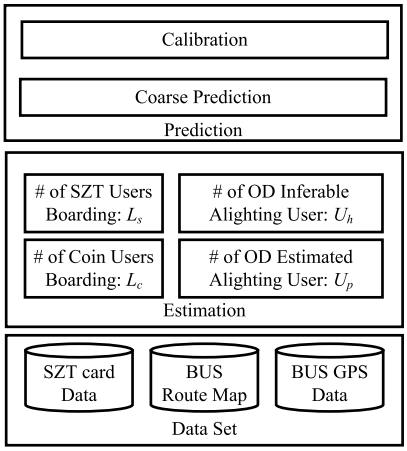


Fig. 4. The data processing work flow.

every bus every 20-40 seconds. The smart card dataset records every smartcard's users' boarding event. The bus route map is static data and used for mapping the passenger's boarding event to the station. Collecting the three datasets to localize the boarding events will be discussed in Section IV-A

By analyzing the boarding events, the data process work flow is then divided into two tasks as follows:

1) Estimation: Firstly, we estimate the numbers of passengers on buses by estimating the passengers ODs. Not only the ODs of smart card users, but also the coin users are considered. The estimation of the passengers' origins is based on mining the temporal and spatial features of the smart card tapping events, which will be discussed in Section IV-B. The estimation of the passengers' destinations is based on trip chain analysis and a probability model and will be presented in Section IV-C.

2) Prediction: After estimation, we then build a model to predict the passenger flow. This model mainly contains two steps: Coarse prediction and Calibration. The coarse prediction is based on analyzing the historical data to see which passenger flow pattern in the historical estimations is most similar to current one. Then we use the result of coarse prediction as the state transit function of an Extended Kalman Filter (EKF) and apply the EKF to the current passenger flow to predict future value. Section V-A and V-B will discuss the two steps in details respectively.

IV. Estimation

In this section, we describe our design of estimating the passenger flow of the running buses. To begin with, Section IV-A first introduces the methods that locate each AFC record's corresponding boarding station. Then in Section IV-B, we present an algorithm for calculating the number of the passengers boarding the bus at a station, which also includes how to estimate the number of those paying by coins. In Section IV-C we further propose a probabilistic model to estimate the number of the alighting passengers at each passed station.

Therefore, with both numbers of the boarding and alighting passengers, we finally obtain an estimated value of the number of the on board passengers.

TABLE I
GPS Dataset and Smart Card Dataset

GPS dataset		Smart card dataset	
Content	Remarks	Content	Remarks
OBU ID	On Board Unit ID	Serial number	It is unique for different records
Vehicle ID	Vehicle registration ID.	Card ID	The number of SZT smart card
Line ID	The line number of the bus	FCD ID	Fare Collection Device ID
Position state	Located or un-located	Transaction type	Metro: Get on/off, Bus: Get on
Longitude	The longitude of the vehicle	Time	The time of tapping card
Latitude	The latitude of the vehicle	Name	Metro: station name, Bus: line name
Time	The time of obtaining the location	Vehicle ID	Vehicle registration ID

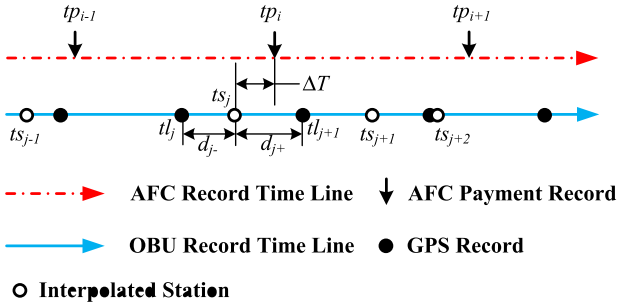


Fig. 5. Determine the vehicle's line.

A. Time Synchronization and Boarding Event Localization

As is mentioned in the overview of the model in Section III, the only data with respect to the passengers that can be observed are their records of payment by smart cards when they get on. However because there is no location field in the AFC records, one labelling step that tags a get-on station to every smart card payment record is prerequisite. Otherwise we may mis-located a passenger's boarding event to a wrong station.

To label a smart card payment record with a station, we match the time stamps in AFC records and OBU records. We then can locate the stations of AFC records by matching them with the OBU records. However there may exist time difference between the AFC devices and OBU devices as they work independently. Moreover, the GPS locations are sampled every 20-40 seconds in OBU. There do have chances that there is no GPS record at the location of a station, which means that there may be no time stamp in OBU records that can match some boarding events. So in order to match the time stamps, we need to interpolate the time stamps in OBU records and synchronize the time stamps in AFC records and OBU records.

Figure 5 illustrate the interpolation and time synchronization process for boarding event localization. We first interpolate the time at which the bus arrives at a station in the OBU records timeline. The interpolation is simply calculated linearly by the driving distances between the station and its nearest GPS records, as expressed in Eq. 2.

$$ts_j = tl_j + \frac{(tl_{j+1} - tl_j)d_{j-}}{d_{j-} + d_{j+}} \tag{2}$$

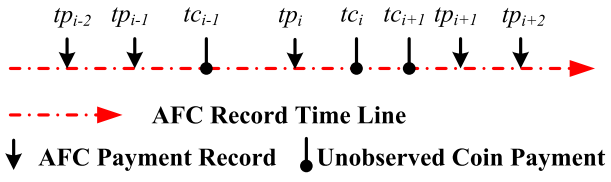


Fig. 6. Illustration of boarding events in the timeline.

The synchronization is then processed as finding a time offset ΔT to minimize the sum of the differences between each AFC payment time and the time of the bus arriving at its *corresponding station* for all the AFC records along a whole bus trip. Here the *corresponding station* of an AFC payment is the station whose arriving time is closest to the payment time after calibrating ΔT in the timeline. The synchronization process can be expressed as Eq. 3.

$$\Delta T = \arg\min_{\Delta T} \sum_i \min_j |tp_i - \Delta T - ts_j| \tag{3}$$

Because we ignore some facts such as buses' speed variety and stopping time at a station, We admit that the time offset ΔT derived from Eq. 3 may be not precisely accurate and have error. However with Eq. 3, the *corresponding station* of each AFC payment tp_i can also be figured out as the station j where $j = \arg\min_j |tp_i - \Delta T - ts_j|$. As we can assume that the buses' stopping time at a station is much less than their travelling time between adjacent stations, the calculated *corresponding stations* should be correct in most cases.

B. Estimation of $L(i, j)$

Based on above time synchronization and boarding event localization process, we can then label every AFC payment with its corresponding station. As a result, we can easily count the passengers who use smart card and get on Bus i at Station j, i.e., $L_s(i, j)$. However, if we further hope to estimate the total number of the boarding passengers $L(i, j)$, including that of those using coins, i.e. $L_c(i, j)$, current collected data failed to give direct observations. So we went for field investigation and found that the time gaps between any two consecutive smart card payment events can be a clue to infer the hidden $L_c(i, j)$.

Figure 6 magnifies the boarding events in the timeline. Through field investigation, we found that passengers boarding the bus usually form a queue and get on one by one. They pay their ride fees either by smart card or by coins one after another. So the payment time can be an approximate arithmetic sequence. If the time gap between two consecutive smart card payment events is larger, it is more probable to have one or more coin payment events between them. Then the question can be that "Given an observed t as the time gap, what is the probability distribution of X, where X is the number of the coin users boarding during t?"

To answer the question, we manually collect data by recording 20 video clips at 20 stations where there are totally 40 smart card users and 20 coin users getting on buses. We extract the time of every payment event and construct a histogram of the time intervals between two events. We found that the payment events almost fit poisson process very well,

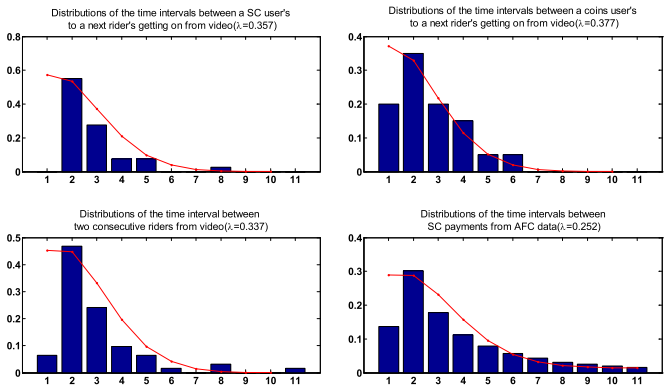


Fig. 7. Time distribution of each getting on event.

as shown in Figures 7. Therefore, using the field investigation data as observed samples, we obtain the intensity parameter $\lambda = 0.34$ for the poisson process, which means that averagely a passenger takes 3 seconds to get on the bus. Finally we use this λ in estimating $L_c(i, j)$ as Eq. 4,

$$L_c(i, j) = \sum_{k=1}^{L_s(i,j)-1} \arg\max_n P\left(n; \lambda(tp_{k+1} - tp_k)\right) \tag{4}$$

where $P(n; \lambda\tau) = \frac{e^{-\lambda\tau}(\lambda\tau)^n}{n!}$ is a poisson distribution with associated parameter $\lambda\tau$, representing the probability of the number of events in time interval $(t, t + \tau]$.

In case that we don't have other source to observe the coin users in current infrastructure, such estimation can be a practical solution in current system. And we are witnessing that the smart card is being the trend in urban transportation system. We believe with larger portion of users using smart cards, more boarding events are observable. The estimation accuracy can be further improved.

Finally, with $L_s(i, j)$ counted and $L_c(i, j)$ estimated, we can get the estimation of the total boarding passengers as $L(i, j) = L_s(i, j) + L_c(i, j)$.

C. Estimation of $U(i, j)$

With the number of getting on passengers known, the next task is to estimate the number of passengers getting off at each station. Then we are able to output the number of passengers on the bus. However, as in Shenzhen neither smart card users nor coins users need extra operations before they get off, there is no direct way of observing or counting the alighting behavior in our dataset. So we estimate the number of alighting passengers based on trip chain analysis from empirical data. Specifically, we differentiate the passengers into 3 types: 1) those who use smart card and show strong regularity in historical records of round trip ODs or transit rides; 2) those who use smart card other than the first type and those who use coins. 3) those who use smart card and take transit ride after alighting current bus. The numbers of each type of the passengers who alight from Bus $\sharp i$ at Station j are $U_h(i, j)$, $U_p(i, j)$ and $U_t(i, j)$ respectively.

1) Estimation Based on Historical Regularity: For the first kind of passengers, we estimate each passenger's getting off

station based on the regularity in his/her historical ODs or transit pattern. More specifically, commute people may usually have fixed ODs everyday and their trips' destinations can be inferred from the origin of corespondent return trips. If one or multiple transits involve in one's trip, the destination of each segment of trip can be inferred from the origin of the next segment.

Therefore, we first analyze the historical data and extract *trip tuples* of $< R_{ID}, O_i, T_i >$ where O_i and T_i represent the origin and the time of the rider R_{ID} paying his ith trip. If the trip is paid by smart card, R_{ID} is identified by the smart card ID. Otherwise, the trips are paid by coins, which are not identifiable and excluded from the regular trips discussion. Given an identifiable tuple $< R_{ID}, O_i >$, if O_{i+1} has a larger probability than p_{th} to be one certain station s, we name the trip of R_{ID} originating O_i to be a regular trip. After we identify all the regular trip tuples from the historical record, we can then make estimation of the destination of new trip of R_{ID} from O_i, D_i to be s, as expressed in Eq. 5.

$$D_i \approx \arg\max_{O_{i+1}} \{P | P = \mathbb{P}\left(O_{i+1} | R_{ID}, O_i\right), P \geq P_{th}\} \quad (5)$$

where $x \approx y$ stands for x and y are two adjacent stations such as two stations in outbound and return directions respectively of the same bus line at the same location, or transit stations at same locations shared by two lines.

Practically, we require the sample size to be larger than 10 to compute the empirical probability of $P = \mathbb{P}(O_{i+1} | R_{ID}, O_i)$ and set $P_{th} = 80\%$. If no P meets the requirement of P_{th} or the sample size in historical data, the trip will be named as a random trip. Note that the *trip tuples* will incrementally update in the historical data. As the time that the system run goes longer, it is more probable to find out these regular trips in the dataset. Therefore, for this type of passengers, we can estimate their getting off station when we observed their smart card payment at the getting on stations. By counting the numbers of this type of passengers who get off at a Station j from Bus $\sharp i$, we can obtain the estimation of $U_h(i, j)$ for every passed station.

2) Dispatch Based on Common Distribution Assumption: In cases that there are not enough samples in the historical dataset to help us to derive the D_i of a correspondent trip $< R_{ID}, O_i, T_i >$, we assume that D_i has the same distribution as that of regular trips. The assumption can also be interpreted as the distribution of the destination of a trip is independent to whether the trip is a regular trip, as shown in Eq. 6.

$$\mathbb{P}(D_i | R_{ID}, O_i) \perp \mathbb{P}(< R_{ID}, O_i, T_i > \text{is a regular trip}) \quad (6)$$

Based on the assumption, we calculated the empirical distribution of D on condition of O from the observable ODs of regular trips in historical data, denoted as $\mathbb{P}(D|O)$. Then we dispatch the non-regular trips including estimated unidentifiable trips paid by coins from all passed stations to different destinations based on $\mathbb{P}(D|O)$, as expressed in Eq. 7.

$$U_p(i, j) = \sum_{k=1}^{j-1} L_p(i, k) \mathbb{P}(D = j | O = k) \quad (7)$$

where $L_p(i, k)$ is the number of non-regular trips, including $L_c(i, k)$, originating from Station k on Bus $\sharp i$.

With U_h and U_p calculated, the number of the passengers getting off at a station can be preliminary estimated as the sum of U_h and U_p.

3) Estimation Amendment Based on Transit Payment: Besides the real time estimation for regular trips and non-regular trips, we applied further amendment operation if we found confliction between the real time estimation and later transit payment record. For instance, suppose we first estimate that a rider R_{ID}'s regular trip $< R_{ID}, O_i, T_i >$ should head to $\hat{D}_i$ at T_d based on R_{ID}'s historical travel regularity. However another payment record shows that R_{ID} has taken a transit ride at another Station O_{i+1} near D_i at time T_{i+1} where $D_i \neq \hat{D}_i$, which means that R_{ID} gets off at D_i rather than $\hat{D}_i$. In such case, we need to amend the estimation based on such observed fact. Likewise, for a non-regular trip, if observed transit fact conflicts with what we dispatch based on empirical probability, we also adopt the amendment. All the amendment can not be real time as transits take time. But as transits usually take not too long time, the delay should be acceptable.

With the above 3 steps, we output $U(i, j)$ for all passed stations and can finally compute the bus passenger flow estimations $\tilde{N}(i, j)$ for every bus at every station.

V. Prediction

In this section, we present the proposed model to predict the short-term passenger flow. We build a 2-Step Real Time Prediction ($2RTP$) model based on Extended Kalman Filter Model. Basically the model contains two steps: Coarse prediction based on historical data, which is introduced in Section V-A and Calibration based on Extended Kalman Filter prediction, which is discussed in Section V-B.

A. Coarse Prediction Based on Historical Data

To predict the passenger flow, we firstly search the historical data to find the passenger flow pattern that is most similar to current estimation. The similarity is defined a Equation 8, where S is the similarity of matrix $\tilde{N}$ and matrix N. $\tilde{N}$ is the real-time estimation and N is the passenger flow of one day in historical data. Operator $<, >$ denotes the inner product, which in our case is the sum of the products of corresponding elements of the two matrices.

$$S = \frac{< \tilde{N}, N >}{\sqrt{< \tilde{N}, \tilde{N} >} * \sqrt{< N, N >}} \quad (8)$$

We assume that if current passenger flow pattern is similar to that of some day in the history, the following passenger flow may change similarly as the pattern on that day. More particularly, with coarse prediction, if we input the sequence of the passenger flow estimation $\{x_1, x_2, x_3, \ldots, x_n\}$, we get a sequence $\{u_1, u_2, u_3, \ldots, u_n, u_{n+1}\}$, where they are similar in period $1 \sim n$, and u_{n+1} is output as the coarse prediction value of next period.

B. Calibration Based on Extended Kalman Filter

With coarse prediction, we then use an extended Kalman Filter [26] based predictor to calibrate the coarse prediction

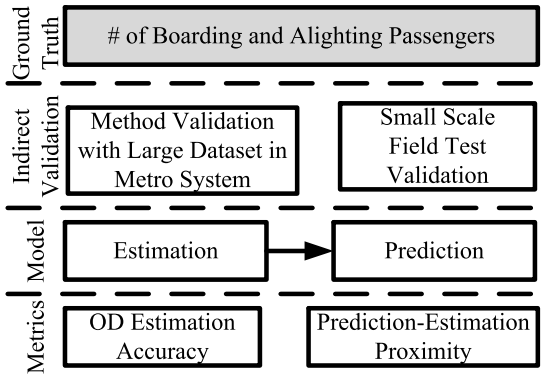


Fig. 8. The model of evaluation.

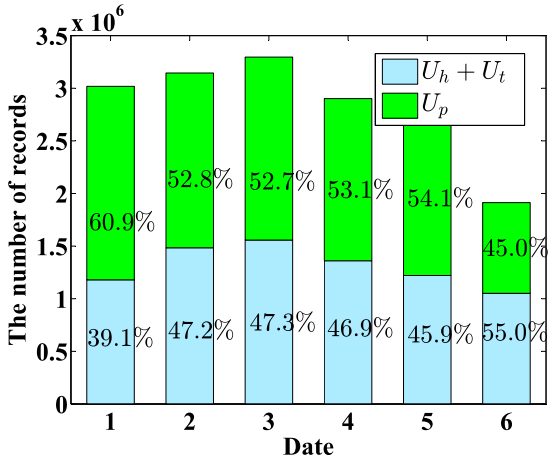


Fig. 9. The Proportion of the Trip-chain Inferrable ODs.

and output the final predictive value. The prediction is running for every station separately. The transition function f of the EKF is a piecewise function that fitting the historical sequence $\{u_1, \cdots, u_{n+1}\}$ as Equation 9. Both the predicted state x_k and the observation state z_k are the passenger flow. Therefore, the observation function h is expressed as Equation 10. By constantly calculating x_k iteratively, we compute the calibrated predictive value.

$$f(x_{k-1}, u_{k-1}) = x_{k-1} + \frac{u_k - u_{k-1}}{u_{k-1} - u_{k-2}}(x_{k-1} - x_{k-2}) \quad (9)$$

$$h(x_k) = x_k \quad (10)$$

VI. Evaluation

In this section, we evaluate the performance of the system. We first present the method and experimental data for evaluation in Section VI-A and then evaluate the performance of estimation and prediction in Section VI-B and VI-C.

A. The Method and Experiment for Evaluation

The overall performance of the system depends on the accuracy of estimation and prediction. The passenger flow is calculated based on estimation of the numbers of boarding and alighting passengers at each station. Therefore we use the accuracy of OD estimations of each trip as the metric to evaluate the performance of passenger flow estimation. On the other hand, we assess the proximity of the predicted value to the estimated value in the future to evaluate the prediction model itself. We use these two metrics to evaluate the proposed model. If the system can estimate the OD of each trip accurately and can predict future estimation accurately, we can then consider that the system work well. However it is hard to get the ground truth of the OD in large scale to evaluate the accuracy. So we use two methods in different scales to indirectly evaluate the methods proposed in the paper, as shown in Figure 8.

Firstly we apply the estimation model in the metro system, where the AFC system work similarly to that in the bus system, except that passengers will tap the smart card again when they exit the destination station. Consequently we have the ground truth of the OD of every trip. We evaluate the accuracy of OD estimation in the metro system. The evaluation result can indirectly validate the estimation model in estimating passengers flow of smart card users in large scale. Secondly we did small scale field experiments to evaluate the OD estimation of the trips whose OD can be not inferred, such as the coin users.

The prediction module uses current estimation to predict the future value. In condition that the estimation works well and the prediction result is close to the estimation in the future, the prediction model is then acceptable. We compare our prediction method with serval baseline methods with same estimation input. We also use the same future estimation value as criteria.

The evaluation is performed on the study on bus Line $\sharp B691$ in ShenZhen. $\sharp B691$ contains 18 stations one-way and passes through many residential areas and business areas.

B. Evaluation of the Estimation

The number of boarding passengers are generally measurable by counting the smart card tapping records and analyzing the intervals between two tapping records. The number of alighting passengers are estimated based on trip chain inference and a probability model. So we first evaluate the proportions of the two types estimations. Then we use two methods in different scales to evaluate the accuracy of trip destination estimation.

1) The Proportion of the Trip-Chain Inferrable ODs: Firstly, we present the proportions of the two types OD estimations. We use the AFC data in 6 days as shown in Figure 9. The blue boxes represent the numbers of trips whose destinations are inferred by trip chain model. The green boxes represent the numbers of trips whose destinations are estimated based on the probability model. We find that about 39.1% to 55.0% of destinations can be inferred from trip chain model. The rest are estimated based on the probability model, which are usually occasional trips of smart card users or trips paid by coins.

2) The Accuracy of Destination Estimation: Secondly, we analyze the accuracy of the destination estimation.

a) Large Scale Metro Data Validation: As bus and metro passengers have similar characteristics of travel and the destination of a metro trip is observable, we use large scale metro

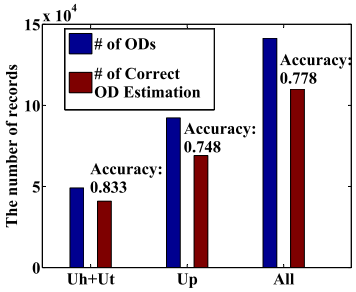


Fig. 10. The Accuracy of the D Estimation in Metro System.

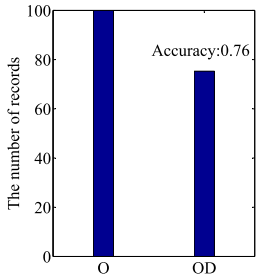


Fig. 11. The Accuracy of the D Estimation in the Field Experiment.

data to validate our estimation model. Figure 10 gives the result of 1.56 million trip samples in the metro system. We find that the accuracy of the destination estimation using trip chain model is about 83.3% and the accuracy of the estimation using the probability assignment model is about 74.8%. The overall accuracy is about 77.8%.

b) Small Scale Field Experiments: We then evaluate the accuracy in bus system from small scale field survey. We collect 100 trips of about 20 participants. These OD trips are recorded. Result in Figure 11 shows that the proposed estimation method in the experiments can get an accuracy of 76%.

C. Evaluation of the Prediction

To evaluate the prediction, we firstly investigate the error distribution of the Extended Kalman Filter. Secondly, we use the estimation values both as input and evaluation reference to compare $2RTP$ with several baseline prediction methods.

1) Error Distribution Analysis: Extended Kalman Filter can be applied to nonlinear system where observation noises are assumed to be gaussian white noise. So firstly to validate the availability of EKF, we study the distribution of the observation noise. Observation noise is the error between coarse prediction and true estimation value. The distribution of the error is shown in Figure 12, where we can see that it approximately obeys Gaussian distribution. To tests the randomness of error sequence, we study the autocorrelation of the error sequence and test them using Q-Test. Figure 13 presents the autocorrelation of the error sequence. We can see that the autocorrelation values are relatively very small except the value at zero. The Q-Test result shows that at the confidence level of 85%, it can accept the hypotheses that the error sequence is white noise. Based on the above results, we validate the availability EKF in the prediction.

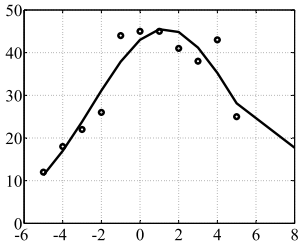


Fig. 12. Distribution of the Observation Noise.

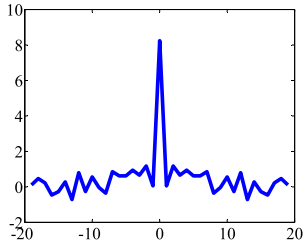


Fig. 13. Autocorrelation of the Error Sequence.

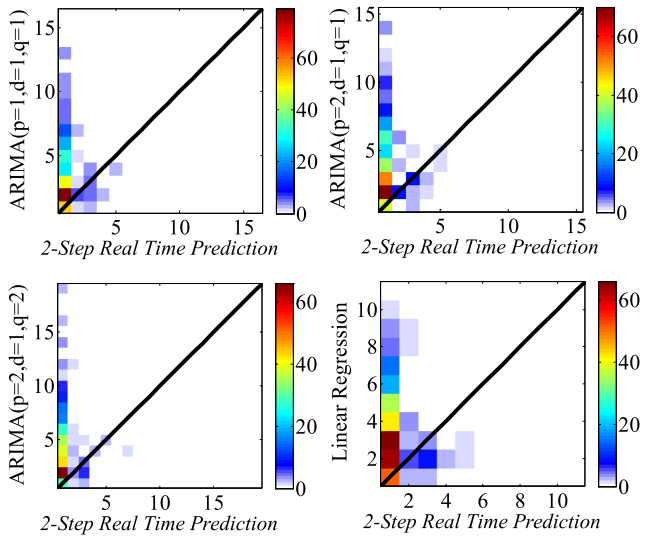


Fig. 14. The Prediction Error Comparison between $2RTP$ and Baseline Models.

2) Prediction Results Analysis: We compare the prediction accuracy of the proposed method with two baseline methods:

- ARIMA: The ARIMA model has been widely applied in forecasting short-term traffic data such as traffic flow. In $ARIMA(p, d, q)$, parameters p, d, and q are the order of the autoregressive model, the degree of difference, and the order of the moving-average model. We choose three common parameter combinations in the comparison.
- Linear Regression: We use different periods of historical data to train the model. Each period has a linear regression model. After getting the parameters, we can predict the passenger flow.

We compare the prediction error in 18 stations in two directions in 10 different periods, which produce $18 \times 2 \times 10 = 360$ sample points for each methods. Each plot of Figure 14 shows the comparison between the proposed $2RTP$ model and one baseline method. The color temperature of the block (x, y)

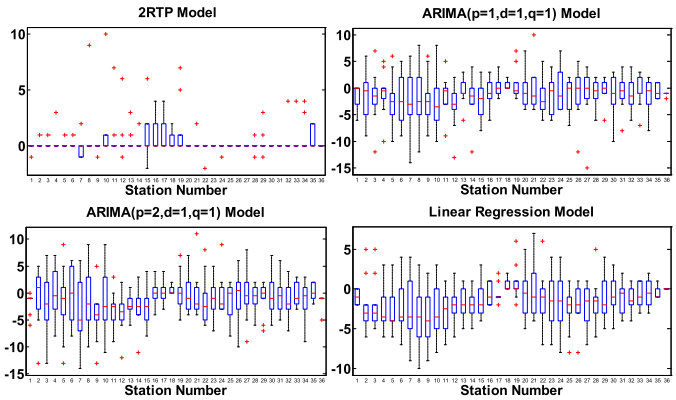


Fig. 15. The Prediction Errors in Different Stations.

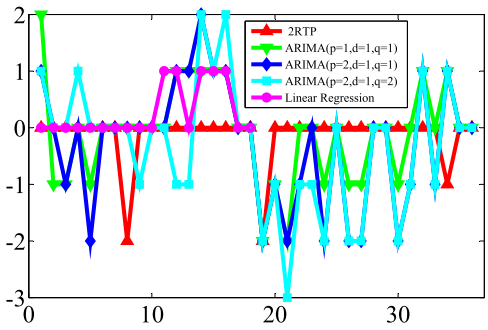


Fig. 16. The Crowding Rate Prediction Errors in Different Stations.

TABLE II
THE RMSE OF DIFFERENT MODELS

Model	RMSE
$2RTP$	1.2845
ARIMA(1,1,1)	3.9402
ARIMA(2,1,1)	4.148
ARIMA(2,1,2)	4.9256
Linear Regression	3.1323

in each plot represents the number of samples points whose error is x with $2RTP$ model and y with the baseline method. We can obviously see that a larger portion of sample points locate above the diagonal $y = x$, which infers that the $2RTP$ model has less prediction errors than the baseline methods. The box-and-whisker plots in Figure 15 further show the prediction errors distribution in the 36 stations. Again we can see that the $2RTP$ model has less mean prediction errors in most stations than the baseline methods. Moreover We also find that except some fliers, the prediction errors with $2RTP$ model in many stations are zero, which means the $2RTP$ can predict the passenger flow accurately in many instances.

Thirdly, we use root mean square error (RMSE) to quantify the error, as shown in Table. II. We can also see that the $2RTP$ model outperform other baseline method in the the prediction accuracy.

Practically, rather than the precise passenger flow, the passengers may care more about the crowding on the bus. Therefore we rate the crowding based on the number of passengers on the bus using K-Means. The crowding rate reference is shown in Table. III. Then we evaluate the accuracy of predictive crowding rate in different stations, which is illustrated in Figure 16. Results also show that $2RTP$ model can predict the crowding rate more accurately than the other models.

TABLE III
THE RESULT OF k-MEANS

Crowding Rate	Number of Passengers	Description
1	0-3	Empty
2	3-6	Medium
3	6-14	Full
4	14-26	Crowded
5	More than 26	Very Crowded

VII. CONCLUSION

In this paper, we present a system to analyze and predict the passenger flow in real-time. The data input of the system are the GPS trace and smart card payment records. We build a model to fuse these two datasets to estimate the passenger flow by deriving the origin and destination of passengers. We then build a 2 Step Real-Time Prediction model that uses both historical data and recent value to predict the future passenger flow. Compared with existing prediction models that only use historical data or recent value, the proposed $2RTP$ prediction outperforms them in prediction accuracy in most time and stations.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their valuable comments.

REFERENCES

[1] F.-Y. Wang, "Parallel control and management for intelligent transportation systems: Concepts, architectures, and applications," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 3, pp. 630–638, Sep. 2010.
[2] M. Yu, D. Zhang, Y. Cheng, and M. Wang, "An RFID electronic tag based automatic vehicle identification system for traffic iot applications," in *Proc. Chin. Control Decision Conf. (CCDC)*, May 2011, pp. 4192–4197.
[3] X. Cheng *et al.*, "Electrified vehicles and the smart grid: The ITS perspective," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 4, pp. 1388–1404, Aug. 2014.
[4] E. I. Vlahogianni, J. C. Golias, and M. G. Karlaftis, "Short-term traffic forecasting: Overview of objectives and methods," *Transp. Rev.*, vol. 24, no. 5, pp. 533–557, Sep. 2004.
[5] 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.
[6] T.-H. Tsai, C.-K. Lee, and C.-H. Wei, "Neural network based temporal feature models for short-term railway passenger demand forecasting," *Expert Syst. Appl.*, vol. 36, no. 2, pp. 3728–3736, Mar. 2009.
[7] B. L. Smith, B. M. Williams, and R. K. Oswald, "Comparison of parametric and nonparametric models for traffic flow forecasting," *Transp. Res. C, Emerg. Technol.*, vol. 10, no. 4, pp. 303–321, Aug. 2002.
[8] N. Zhang, F. Y. Wang, F. Zhu, D. Zhao, and S. Tang, "DynaCAS: Computational experiments and decision support for ITS," *IEEE Intell. Syst.*, vol. 23, no. 6, pp. 19–23, Nov. 2008.
[9] B. L. Smith and M. J. Demetsky, "Traffic flow forecasting: Comparison of modeling approaches," *J. Transp. Eng.*, vol. 123, no. 4, pp. 261–266, Jul. 1997.
[10] B. M. Williams, P. K. Durvasula, and D. E. Brown, "Urban freeway traffic flow prediction: Application of seasonal autoregressive integrated moving average and exponential smoothing models," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 1644, no. 1, pp. 132–141, Jan. 1998.
[11] J. V. Hansen, J. B. McDonald, and R. D. Nelson, "Time series prediction with genetic-algorithm designed neural networks: An empirical comparison with modern statistical models," *Comput. Intell.*, vol. 15, no. 3, pp. 171–184, Aug. 1999.

[12] M. S. Ahmed and A. R. Cook, *Analysis of Freeway Traffic Time-Series Data by Using Box-Jenkins Techniques*, vol. 722. Washington, DC, USA: Transportation Research Board, 1979.
[13] M. M. Hamed, H. R. Al-Masaeid, and Z. M. B. Said, "Short-term prediction of traffic volume in urban arterials," *J. Transp. Eng.*, vol. 121, no. 3, pp. 249–254, May 1995.
[14] B. M. Williams and L. A. 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, Nov. 2003.
[15] M. C. Tan, S. C. Wong, J. M. Xu, Z. R. Guan, and P. Zhang, "An aggregation approach to short-term traffic flow prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 1, pp. 60–69, Mar. 2009.
[16] J. Faraway and C. Chatfield, "Time series forecasting with neural networks: A comparative study using the airline data," *J. Roy. Statist. Soc., C (Appl. Statist.)*, vol. 47, no. 2, pp. 231–250, Jan. 1998.
[17] C. Lim and M. McAleer, "Time series forecasts of international travel demand for australia," *Tourism Manage.*, vol. 23, no. 4, pp. 389–396, Aug. 2002.
[18] C. Brooks, *Introductory Econometrics for Finance*. Cambridge, U.K.: Cambridge Univ. Press, 2014.
[19] G. Zhang, B. E. Patuwo, and M. Y. Hu, "Forecasting with artificial neural networks: The state of the art," *Int. J. Forecasting*, vol. 14, no. 1, pp. 35–62, 1998.
[20] M. Dougherty, "A review of neural networks applied to transport," *Transp. Res. C, Emerg. Technol.*, vol. 3, no. 4, pp. 247–260, Aug. 1995.
[21] S. Clark, "Traffic prediction using multivariate nonparametric regression," *J. Transp. Eng.*, vol. 129, no. 2, pp. 161–168, Mar. 2003.
[22] Y. Wang, M. Papageorgiou, and A. Messmer, "Real-time freeway traffic state estimation based on extended Kalman filter: A case study," *Transp. Sci.*, vol. 41, no. 2, pp. 167–181, May 2007.
[23] Y. Tang, W. H. Lam, and P. L. Ng, "Comparison of four modeling techniques for short-term aadt forecasting in Hong Kong," *J. Transp. Eng.*, vol. 129, no. 3, pp. 271–277, May 2003.
[24] P. Vythoulkas, "Alternative approaches to short term traffic forecasting for use in driver information systems," *Transp. Traffic Theory*, vol. 12, pp. 485–506, 1993.
[25] H. Zhang, "Recursive prediction of traffic conditions with neural network models," *J. Transp. Eng.*, vol. 126, no. 6, pp. 472–481, Dec. 2000.
[26] K. Hoshina, "Extended Kalman filter," Ph.D. dissertation, School Eng., Tokyo Univ. Agriculture Techn., Tokyo, Japan, 2013.

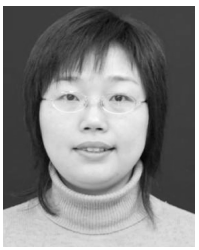
Jun Zhang received the B.S. degree from Beijing University of Posts and Telecommunications, in 2010 and the M.S. degree from the Department of Electronics and Information Engineering, Huazhong University of Science and Technology, China, in 2012, where he is currently working toward the Ph.D. degree. His research interests include bus scheduling, big data analysis, big-data-driven systems, and spatiotemporal data mining.

Dayong Shen received the bachelor's and master's degrees in system engineering from NUDT in 2011 and 2013, respectively. He is currently working toward the Ph.D. degree with the Social Transportation and Social Logistics. He has rich experience in designing and implementing parallel logistics system projects. His research interests include intelligent scheduling, artificial intelligence algorithm and parallel social systems.

Lai Tu received the B.S. degree in communication engineering and the Ph.D. degree in information and communication engineering from Huazhong University of Science and Technology, China, in 2002 and 2007, respectively. From 2007 to 2008, he was a Post-Doctoral Fellow with the Department of EIE, Huazhong University of Science and Technology. From 2009 to 2010, he was a Post-Doctoral Researcher with the Department of CSIE, Nation Cheng Kung University, Taiwan. He is currently an Associate Professor with the School of Electronic and Information and Communications, Huazhong University of Science and Technology. His research areas include urban computing, human behavior study, mobile computing, and networking.

Fan Zhang received the Ph.D. degree in communication and information system from Huazhong University of Science and Technology in 2007. He was a Post-Doctoral Fellow with The University of New Mexico and with University of Nebraska-Lincoln, from 2009 to 2011. He is currently an Associate Professor with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences. His research topics include big data processing, data privacy and network security, and wireless networks.

Chengzhong Xu received the Ph.D. degree from The University of Hong Kong in 1993. He is currently a Professor with the Department of Electrical and Computer Engineering, Wayne State University (WSU), USA. He also holds an Adjunct Appointment with the Shenzhen Institute of Advanced Technology, Chinese Academy of Science, and as the Director of the Institute of Advanced Computing and Data Engineering. He has authored over 200 papers in journals and conferences. His research interest is in parallel and distributed systems and cloud computing. He was the Best Paper Nominee of the 2013 IEEE High Performance Computer Architecture, and the Best Paper Nominee of the 2013 ACM High Performance Distributed Computing. He was a recipient of the Faculty Research Award, the Career Development Chair Award, and the President's Award for Excellence in Teaching of WSU, and the Outstanding Oversea Scholar Award of NSFC. He serves on a number of journal editorial boards, including IEEE TRANSACTIONS ON COMPUTERS, IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, IEEE TRANSACTIONS ON CLOUD COMPUTING, *Journal of Parallel and Distributed Computing*, and *China Science Information Sciences*.



Yi Wang received the B.S., M.S., and Ph.D. degrees from the Department of Electronics and Information Engineering, Huazhong University of Science and Technology, China, in 2000, 2003, and 2009, respectively. She is currently a Lecturer with the School of Electronics Information and Communications, Huazhong University of Science and Technology. Her research interest is big data for smart transportation.

Chen Tian received the B.S., M.S., and Ph.D. degrees from the Department of Electronics and Information Engineering, Huazhong University of Science and Technology, China, in 2000, 2003, and 2008, respectively. He was an Associate Professor with the School of Electronics Information and Communications, Huazhong University of Science and Technology. From 2012 to 2013, he was a Post-Doctoral Researcher with the Department of Computer Science, Yale University. He is currently an Associate Professor with the State Key Laboratory for Novel Software Technology, Nanjing University, China. His research interests include data center networks, network function virtualization, distributed systems, Internet streaming, and urban computing.

Xiangyang Li (F'15) received the bachelor's degree from the Department of Computer Science and a bachelor's degree from the Department of Business Management, Tsinghua University, China, in 1995, and the M.S. and Ph.D. degrees from the Department of Computer Science, University of Illinois at Urbana–Champaign, in 2000 and 2001, respectively. He is currently a Professor with College of Computer Science and Technology, University of Science and Technology of China, and with the Illinois Institute of Technology. He is also a Distinguished Visiting Professor with Xi'an Jiaotong University and with the University of Science and Technology of China. His research interests include wireless networking, mobile computing, security and privacy, cyber–physical systems, and algorithms. He is a recipient of China NSF Outstanding Overseas Young Researcher (B). He and his students received three best paper awards, including the ACM MobiCom 2014, the COCOON 2001, and the IEEE HICSS 2001, and the One Best Demo Award ACM MobiCom 2012. He and his students received five best paper awards, including the IEEE GlobeCom 2015, the IEEE HPCCC 2014, the ACM MobiCom 2014, the COCOON 2001, and the IEEE HICSS 2001, one best paper award runner up, one best demo award at the ACM MobiCom 2012, and was selected as the best paper candidate at the ACM MobiCom 2008 and the ACM MobiCom 2005. He is an Editor of several journals, including IEEE TRANSACTION ON MOBILE COMPUTING. He has served many international conferences in various capacities, including the ACM MobiCom, the ACM MobiHoc, and the IEEE MASS. He is an ACM Distinguished Scientist.

Benxiong Huang is currently a Doctoral Supervisor and a Professor with the Department of Electronic and Information Engineering, Huazhong University of Science and Technology, and the Vice-Director of the National Engineering Laboratory of Next generation Internet access system, and the Secretary-General of the Innovation Institute of internet of things. His research interests cover communication system, next generation mobile internet, signal processing, and social computing.

Zhengxi Li is currently a Doctoral Supervisor, a Professor, and the Vice-President of the North China University of Technology. His research interests cover intelligent traffic control and management, control theory and control engineering, and electric drive technology.