Transportation Research Part C 81 (2017) 153–171

Contents lists available at ScienceDirect

Transportation Research Part C

journal homepage: www.elsevier.com/locate/trc

Measuring fine-grained metro interchange time via smartphones

Weixi Gu [a,1], Kai Zhang [b,1], Zimu Zhou [c], Ming Jin [d], Yuxun Zhou [d], Xi Liu [a], Costas J. Spanos [d], Zuo-Jun (Max) Shen [f], Wei-Hua Lin [e], Lin Zhang [a,*]

[a] Tsinghua University, Tsinghua-Berkeley Shenzhen Institute, China
[b] Graduate School at Shenzhen, Tsinghua University, China
[c] ETH Zurich, Computer Engineering and Networks Laboratory, Switzerland
[d] University of California at Berkeley, Department of Electrical Engineering and Computer Sciences, United States
[e] The University of Arizona, Department of Systems and Industrial Engineering, United States
[f] University of California, Department of Industrial Engineering and Operations Research, United States

ARTICLE INFO

Article history:
Received 12 October 2016
Received in revised form 23 May 2017
Accepted 25 May 2017
Available online 7 June 2017

Keywords:
Underground public transport
Location-based service
Smartphone
Crowdsourcing

ABSTRACT

High variability interchange times often significantly affect the reliability of metro travels. Fine-grained measurements of interchange times during metro transfers can provide valuable insights on the crowdedness of stations, usage of station facilities and efficiency of metro lines. Measuring interchange times in metro systems is challenging since agent-operated systems like automatic fare collection systems only provide coarse-grained trip information and popular localization services like GPS are often inaccessible underground. In this paper, we propose a smartphone-based interchange time measuring method from the passengers' perspective. It leverages low-power sensors embedded in modern smartphones to record ambient contextual features, and utilizes a two-tier classifier to infer interchange states during a metro trip, and further distinguishes 10 fine-grained cases during interchanges. Experimental results within 6 months across over 14 subway lines in 3 major cities demonstrate that our approach yields an overall interchange state inference F1-measurement of 91.0% and an average time error of less than 2 min at an inference interval of 20 s, and an average accuracy of 89.3% to distinguish the 10 fine-grained interchange cases. We also conducted a series of case studies using measurements collected from crowdsourced users during 3 months, which reveals findings previously unattainable without fine-grained interchange time measurements, such as portions of waiting time during interchange, interchange directions, usage of facilities (stairs/escalators/lifts), and the root causes of long interchange times.

1. Introduction

Interchange times are well-known to exhibit high variability (Zhang and Yao, 2015) and are a source of major uncertainty for the quality of services in public transportation (Aguiléra et al., 2014). Unexpected crowds at transfer stations and

* Corresponding author.
E-mail addresses: guweixigavin@gmail.com (W. Gu), zhangkai@sz.tsinghua.edu.cn (K. Zhang), zimu.zhou@tik.ee.ethz.ch (Z. Zhou), jinming@berkeley.edu (M. Jin), yxzhou@berkeley.edu (Y. Zhou), xi.liu@sz.tsinghua.edu.cn (X. Liu), spanos@berkeley.edu (C.J. Spanos), maxshen@berkeley.edu (Z.-J. Shen), whlin@email.arizona.edu (W.-H. Lin), linzhang@tsinghua.edu.cn (L. Zhang).
[1] These authors contributed equally to this work.

http://dx.doi.org/10.1016/j.trc.2017.05.014
0968-090X/

excessive delays of prior transportation can easily lead to miss-connection to the next vehicle and long perceived waiting time at the stations (Fan et al., 2016). Fine-grained interchange time measurements can provide transport operators and authorities with detailed factors that may affect the traffic reliability, including congestion levels in the passages, crowdedness on platforms and usage dynamics in transit facilities such as stairs, escalators and elevators, among others, and better understand the capacities of stations (Xu et al., 2014). Such information can assist in optimizing transport timetables that minimize travel times (Sels et al., 2016; Wu et al., 2015) and improving the transit experience from a passenger's perspective (Parbo et al., 2015).

Measuring fine-grained interchange times at transfer stations is challenging, especially in metro systems. Agency-operated automatic data collection systems, *e.g.* automatic fare collection (AFC), only provide large-scale overall trip times suitable for metro network planning and evaluation (Pelletier et al., 2011). While research has been conducted to estimate several interchange times (walking, waiting and transfer) via temporal modeling using AFC data (Zhang and Yao, 2015), the prior assumptions on passenger flows and distributions might not always be valid in practice. Alternatively, extensive research has been performed to obtain direct public transit measurements from passenger smartphones through crowdsourcing (Misra et al., 2014). GPS information from passengers and vehicles enables passenger tracking in road traffic (Bierlaire et al., 2013; Carrel et al., 2015), but it is often inaccessible in underground metro systems. Some researchers leverage cellular networks operated underground to quantify quality of service and passenger flows in metros (Aguiléra et al., 2014), but cell tower signaling alone is insufficient to support the measurement of fine-grained interchange times. The rich embedded sensors on smartphones have also attracted increasing interest in extracting trip information by sensor fusion. Various inertial sensors (Feng and Timmermans, 2013; Hemminki et al., 2013; Sankaran et al., 2014) are employed to identify transportation modes such as being stationary, walking, and riding a bus, tram, or metro and so forth. These low-cost sensors are also utilized to robustly track stops and runs of metros (Yu et al., 2014; Lee and Han, 2014; Higuchi et al., 2015; Stockx et al., 2014). However, none of the previous work has been able to provide information for fine-grained interchange times from the passengers' perspective.

The rapid development of sensors in smartphones has triggers many advanced research works (Gu, 2017; Zhou et al., 2013; Shangguan et al., 2014; Yang et al., 2014; Gu et al., 2014a). In this paper, we propose a smartphone-based approach to the measurement of fine-grained interchange time. Unlike previous works (Yu et al., 2014; Lee and Han, 2014; Higuchi et al., 2015; Thiagarajan et al., 2010; Stockx et al., 2014) that track the *Stop* and *Running* of a single metro, we focus on detecting the *Interchange* of passengers during a metro trip. Although some works (Ohashi et al., 2014; Gu et al., 2016a) have explored automatically splitting trips into walking, staying and in-vehicle states, they fail to provide detailed time information that passengers spend in passages, on platforms, or using various facilities (*e.g.* stairs, escalators, elevators) during metro interchanges. By invoking carefully selected smartphone sensors to monitor ambient magnetic fields, acceleration and cell tower signal strengths, we provide a first-of-its-kind interchange time measurement service that automatically distinguishes and measures the durations of 10 common passenger states (going up/down by an elevator/escalator/stairs, walking up/down on an escalator, walking through a passage, waiting on platforms or at the entrance of elevator/escalator) at a metro transfer station. We extract representative features from raw sensory data, which are fed into a two-tier cost-sensitive Naive Bayes classifier (Elkan, 2001) and a Conditional Random Field (CRF) (Lafferty et al., 2001).

For validation, we implemented our approach as an Android application, and conducted extensive field evaluations with 32 volunteers in 6 months, covering 14 metro lines in 3 major cities in China (Beijing, Shanghai and Shenzhen). Experimental results demonstrate an overall interchange state identification F1-measurement (Sokolova and Lapalme, 2009) of 91.0%, an average error in time of less than 2 min, an average accuracy of 89.3% in distinguishing the 10 common passenger states during interchanges. We also conducted a series of case studies using measurements from crowdsourced users during 3 months and analyzed detailed interchange times at interchange stations in the 3 major cities in China from both macro and micro points of view.

- From a macro point of view, we find that 12 out of the 34 transfer stations in our evaluation have long interchange times (over 360 s). Interchange times at most interchange stations have high variability, which increases during peak hours and decreases in off-peak hours. The portion of interchange times also varies among the multiple routes for the same origin-destination pair.
- From a micro point of view, we show that passengers can spend 50% of their interchange times waiting on the platform, which can be optimized with metro re-scheduling. A decomposition of interchange states into 10 cases reveals the passengers' behaviors, *e.g.* the transferring directions and the usage of stairs/escalators/lifts during peak and off-peak hours. Further analysis on waiting and walking times shows that the root cause of long interchange times during peak hours is mainly blockages at the entrances of stairs, escalators or lifts.

In the rest of the paper, we first discuss the literature review in Section 2, and then outline our methodology in Section 3. Sections 4 and 5 detail the technical solutions to interchange state inference and fine-grained interchange case inference. We evaluate our methods in Section 6 and present a series of case studies in Section 7. Finally we conclude the paper in Section 8.

2. Literature review

2.1. Passenger's tracking

There has been active research using automatic data collection systems (*e.g.* fare collection) (Briand et al., 2017), geographic information systems (GIS) and smartphones for passenger activities tracking in public transportation (Gu et al., 2016b; Vlahogianni and Barmpounakis, 2017).

A majority of these studies aim to identify transportation modes during a trip. Feng and Timmermans (2013) utilized GPS systems and accelerometers to differentiate walking, cycling, running, and riding a motorcycle, bus, car, metro or tram with a Bayesian belief network model. Hemminki et al. (2013) leveraged accelerometers only to identify more transportation modes including being stationary, walking, and riding a bus, tram, metro or train. Other works focus more on the purpose and quality of trips. Zhao et al. (2013) proposed a theoretical framework to model and estimate excess journey times using smartcard data. Aguiléra et al. (2014) employed cell network data to estimate travel times, metro occupancy levels and origin-destination flows. Carrel et al. (2015) harnessed fine-grained location information and automatic vehicle location, fare collection and passenger count data to track passengers in detail via a location matching scheme.

Despite these research efforts, it incurs unique challenges to track passengers and their travel experiences underground, where popular GIS such as GPS becomes inaccessible. Therefore metro timetables and smartphone sensors are commonly employed for metro state inference and tracking underground. Yu et al. (2014) employed acceleration and cell tower signals to estimate the stop and go states of metros. However, the acceleration patterns might be significantly reduced in modern metros. Lee and Han (2014) exploited ambient magnetic fields in metros to differentiate their motion states. StationSense (Higuchi et al., 2015) further improved the estimation robustness by fusion acceleration and magnetic measurements. However, they cannot distinguish stops between stations from those at stations, which can trigger false arrival reminders and erroneous interchange time calculations. Thiagarajan et al. (2010) fed acceleration data and a fixed timetable into an HMM model to locate metros, which is robust to in-between stops. SubwayPS (Stockx et al., 2014) added gyroscopes to further track metros between stations. TransitLabel (Elhamshary et al., 2016) utilizes smartphone sensors to automatically add semantic information *e.g.* entrance gates, platforms, waiting lines to the complicated floor plans of metro stations.

In this work, we aim to accurately identify the interchange state of passengers during a metro trip using smartphone sensors. Our method is built upon previous research on smartphone-based metro tracking, yet advances the state-of-the-art by further classifying the interchange state into 10 fine-grained interchange cases, which facilitate to perform detailed interchange time analysis and mine passenger behaviors during metro transfers.

2.2. Indoor localization

Indoor localization has been extensively studied via different methods so far (Youssef, 2015). The most ubiquitous smartphone-based solutions are either the Pedestrian Dead Reckoning (PDR) (Beauregard and Haas, 2006) or Wi-Fi fingerprinting (Xiao et al., 2011) techniques. The principle of PDR is to measure the travelled distance and walking directions using the inertial sensors of smartphones, and to dead-reckon the user from a reference point afterwards (Zou et al., 2017a; Chen et al., 2015; Qiu et al., 2016). It often assumes a map of the building and works best with narrow corridors. While the common presence of sensors in smartphones supports PDR to be an attractive technique for smartphone localization, its notable limitation lies in increasing errors accumulated over time without timely adjustments in between, especially under the complex and large-scale underground scenarios. There will be a significant performance degradation if the trajectories of users are relatively unconstrained by corridors. The core of Wi-Fi fingerprinting techniques is to compare the intensity of the received signal and the method of "fingerprinting" (Zou et al., 2017b; Xiao et al., 2016; Lu et al., 2016; Zou et al., 2016), which requires calibration to construct a prior wireless map for the building. However, the map calibration is time consuming, and Wi-Fi access point deployment is usually sparse underground. Any changes in the environment such as adding or removing furniture or structures underground may disturb the "fingerprint" that corresponds to each location, degrading the localization performance.

Compared with the indoor position systems, we focus on the semantic state detection rather than the absolute physical coordinates. The selected features, *i.e.*, magnetic intensity, acceleration and RSSI, are used to track a passenger's states during the metro trip. For example, we choose relatively simple features from the acceleration traces because we focus on the semantic states, *e.g.* walk and stand, rather than the absolute physical coordinates. Therefore, the acceleration traces are only used to differentiate whether the passenger is walking or not. Consequently, issues such as drift, inaccurate values of the gravity, the body and navigation coordination are not major concerns in our case. As a mobile solution which tracks the semantic states of passengers, and avoids dealing with noise in indoor localization problems (*i.e.*, errors caused by drift, inaccurate values of gravity, the body and navigation coordination), we believe that simple inference models are beneficial and hold potential for implementation on commodity smartphones. They will also help to save smartphones energy and CPU resources.

3. Methodology

A metro trip from passengers' perspective usually consists of two states.

- *Interchange:* A passenger transfers from one metro to another, including both waiting at an interchange station and walking or taking lifts/elevators to another platform.
- *In-Metro:* A passenger is on a metro, which can either be *Running* or *Stop* at a station or in the tunnel.

We propose a two-tier inference model to detect the state of passengers and further record the fine-grained interchange state during transfer. The first tier model infers whether the passenger is in *Interchange* or *In-Metro* state leveraging features extracted from multiple smartphone sensors including magnetic sensor, accelerometer and GSM. It then feeds the features into a cost-sensitive Naive Bayes model for passenger state inference. Passenger state is inferred every 20 s, which roughly corresponds to the shortest possible state in the metro trip. Table 1 summarizes the sensory features used and their usage to infer passenger states during a metro trip.

Once the passenger is detected to be in the *Interchange* state, the second tier inference model further identifies the fine-grained *Interchange* cases (including taking an elevator or escalator up/down, walking upstairs/downstairs, waiting on a platform or entrance of an elevator/escalator, walking through a passage, walking up/down on the escalator) via a set of sensory measurements from an accelerometer and barometer. The sensory data and their corresponding usages for the fine-grained *Interchange* case identification are described in Table 2.

We elaborate on the feature selection and the two-tier inference models sequentially in the following sections.

4. Passenger state inference

4.1. Feature selection

Passenger state inference extracts robust features from four sensory measurements including magnetic intensity, acceleration, GSM RSSI and time statistics.

4.1.1. Magnetic intensity features

Features extracted from magnetic intensities are utilized to infer the start of the metro *Running* for the *In-Metro* state.

Principle: The contemporary metro operation systems are generally driven by the electric engine. These engines, which are distributed among each cart, rely on the interaction between winding currents and magnetic fields to generate force. During operation, significant fluctuations of the current generate strong magnetic fields, which in turn create a considerable torque to push the metro forward (Lee and Han, 2014).

Thus the magnetic intensity during metro operations is normally greater compared to that in other phases.

Fig. 1 shows an illustrative trace of magnetic intensity of a metro trip with a sampling rate of 50*Hz*. The magnetic intensity is calculated as $M = \sqrt{M_x^2 + M_y^2 + M_z^2}$, where M_x, M_y, and M_z are of the magnetic intensities on the *X*-, *Y*- and *Z*-axis respectively, and μT is the unit. The trace is filtered by a moving average window of size 500 (*i.e.*, 10 s). The time series marked blue denotes the variation of magnetic intensities, and the colored bar at the bottom represents the ground truth of two states during the entire trip. As is shown, the magnetic field increases abruptly when a passenger is in the train, and stays at a relatively low and stable level during interchanges. Thus, the variation of magnetic intensities can differentiate the two states of *Interchange* and *In-Metro*.

Magnetic features F_{mag}: Given a magnetic intensity sequence $\mathbb{M} = \{M(t_1), \ldots, M(t_k), \ldots, M(t_N)\}$, we first filer the raw data using a moving average filter of size *W*. Denote the mean $\overline{\mathbb{M}} = \frac{1}{N}\Sigma_{i=1}^{N} M(t_i)$. If $\overline{\mathbb{M}}$ exceeds a preset threshold δ_{mag} with a larger confidence probability **P** (*i.e.* $\geqslant 50\%$), it implies a high level of magnetic intensity, and possibly a state of *In-Metro*. Otherwise, it might be a state of *Interchange*. We extract the magnetic intensity features every *inference interval* (20 s), *i.e.*, $N = 1000$ under a sampling rate of 50 Hz. We empirically set the window size $W = 500$ and the threshold $\delta_{mag} = 70$ μT to optimize local experiment performance. The confidence probability **P** is learnt by a statistic model on the training dataset, which can be well applied in different underground environments.

4.1.2. Acceleration features

Features extracted from acceleration are used to infer both *In-Metro* and *Interchange* states.

Principle: Acceleration has been employed to detect the running and stopping of metros (Thiagarajan et al., 2010; Stockx et al., 2014; Yu et al., 2014), and we extend its usage to recognize passengers walking (*e.g.*, through a passage or stairs) or waiting (*e.g.*, on the platform or at the entrance of an elevator/escalator) during interchanges. As an illustrative example, we asked 20 volunteers to record accelerations during their metro trips at a sampling rate of 50*Hz*. We then calculate the *differences* of accelerations $A = \{a_1, \ldots, a_i, \ldots, a_n\}$, where $a_i = \sqrt{a_{xi}^2 + a_{yi}^2 + a_{zi}^2}$, a_{xi}, a_{yi} and a_{zi} represent the acceleration difference of two successive acceleration amplitudes at each axis. The difference operation on the acceleration trace assists to eliminate

Table 1
Summary of sensory data, extracted features, and usage for passenger state inference in the first tier.

Sensory data	Features	Usage
Magnetic field	Mean	Infer *Interchange* and *In-Metro* (on a running train)
Acceleration	Energy and period of acceleration difference	Infer *Interchange* (walking and waiting) and *In-Metro* (on a stop and running train)
GSM RSSI	Energy, standard deviation	Infer *Interchange* and *In-Metro* (on a stop/running train)

Table 2
Summary of sensory data, extracted features, and usage for fine-grained interchange case inference in the second tier.

Sensory data	Features	Usage
Acceleration	Period of acceleration difference	Infer *Walk* and *Non-walk*
Pressure	Height gradient	Infer 10 cases of *Interchange* duration

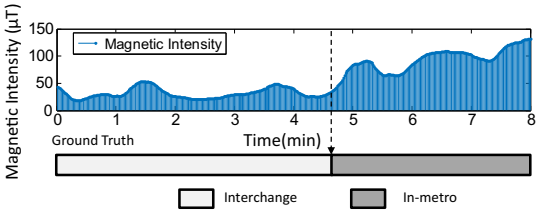

Fig. 1. Magnetic field trace during a metro trip.

the noise interference and highlight its characteristics in temporal dynamics. Fig. 2 plots the acceleration differences for 20 s after adopting a smoothing window of 50 samples for four states:

- Case 1: passenger walking during an interchange.
- Case 2: passenger waiting during an interchange.
- Case 3: passenger on a stopped train.
- Case 4: passenger on a running train.

We define *energy* as $En = \sum_{i=1}^{n} a_i^2$ and *period* as the bin corresponding to the strongest peak by performing a Fast Fourier Transformation (FFT) on $\{a_i\}_{i=1}^{n}$. As shown in Fig. 2, the repetitive pattern of walking exhibits a shortest period within the range of [0.8 s, 1.2 s], which corresponding to the walking frequency of human being spanning from 0.5 Hz to 2 Hz (Pachi and Ji, 2005). Such a *period* is hardly observed in acceleration traces for non-walking states. We also observe that the *energy* of the acceleration is relatively low when passenger waits for a metro on the platform, while slightly larger (10–80) when the passenger is on a halted metro. This notable difference in the *energy*, even if the passenger is stationary, comes from the vibration of the metro's running engine, which cannot be smoothed with a window of 50 samples. The energies for a walking passenger and a running metro are often dramatically larger, due to the larger acceleration amplitudes. Inspired by the above observations, we choose *energy* and *period* to distinguish *Interchange* (Case 1, Case 2) and *In-Metro* (Case 3, Case 4).

Acceleration feature F_{acc}: The workflow of acceleration feature extraction is detailed as follows. After collecting a 1000-sample acceleration sequence within an *Inference interval*, we first compute its difference, and then smooth it with a micro rolling average window of 50 samples. Then we calculate the period and judge whether it lies within the closed interval [0.5 s, 2 s]. If so, it is labeled as *walking*. Otherwise, we calculate its energy. If the energy is lower than a predefined threshold of $\epsilon = 5$, it is labeled as *waiting*. Otherwise, it is labeled to *In-Metro*. Both *walking* and *waiting* belong to the *Interchange* state.

4.1.3. GSM RSSI features

Features extracted from the GSM received signal strength index (RSSI) are used to infer both *In-Metro* and *Interchange* states.

Principle: GSM is the main communication medium for a metro. As GSM signals can be attenuated during a metro trip, the variance of the GSM RSSI may indicate different states during a metro trip. Generally, a metro station is installed with several child cell-sites to guarantee communication quality underground, while few cell-sites are installed in the metro tunnels, leading to a weaker signal strength.

Fig. 3 shows a trace of the GSM RSSI during a metro trip. We divide the *In-Metro* state into *Stop* and *Running* subcases as in the previous subsection. The blue bar plots the raw RSSI, and the red line illustrates the smoothed RSSI. As is shown, the RSSI drops gradually as the metro departs from the station, and stays at a low level in the tunnel. When the train approaches the next station, the RSSI climbs up and arrives at a peak when the metro halts at a station. In addition, during transfers at an

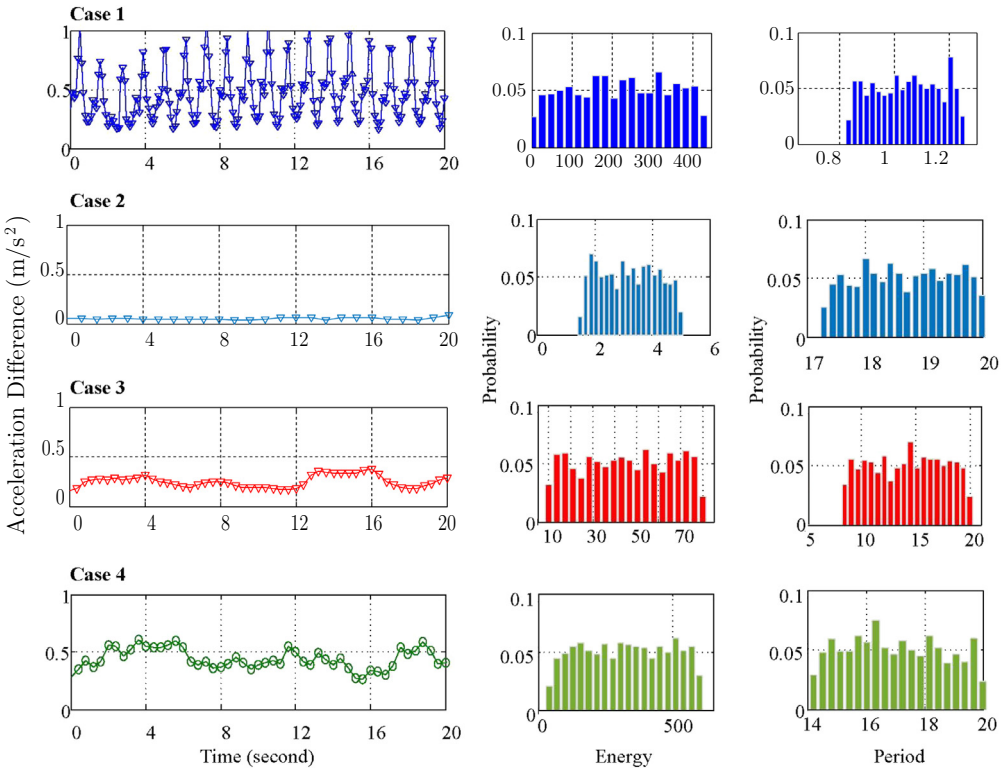

Fig. 2. Acceleration traces of four states and the probability density distributions of *energy* and *period*.

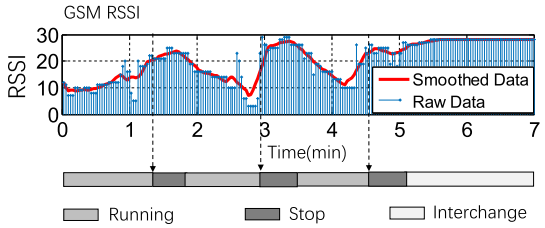

Fig. 3. RSSI trends in a metro trip.

interchange station, the RSSI remains stable at a high value. Thus the RSSI profiles of *Running*, *Stop* and *Interchange* exhibit a unique pattern.

RSSI feature F_{rss}**:** Given an RSSI sequence $R = \{r_1, r_2, \ldots r_n\}$ smoothed by a rolling window lasting for 10s, we adopt two efficient criteria to capture the distinctive pattern of RSSI. The first is energy $En = \Sigma_{i=1}^{n} r_i^2$, where r_i denotes the strength of the ith sample in the sequence.

The second is the standard derivation *std*. Compared with *In-Metro* (*i.e.*, *Stop* and *Running*), the RSSI of *Interchange* usually exhibits small *std*.

Fig. 4 illustrates the distribution of the two criteria based on 23 metro trip records from 20 volunteers. As is shown, the three RSSI profile classes are divided into three clusters by the two criteria, and hence can be classified with a decision tree model.

Fig. 5 shows the decision tree model for GSM RSSI profile classification based on our evaluation. Note the subcases of *Running* and *Stop* are labeled as *In-Metro* in the decision tree model. Specifically, the paths from root to leaf represent classification rules, and the splitting features and thresholds are determined depending on the information gain calculated by entropy.

4.2. State inference model

We modify the standard Naive Bayes classifier with cost-sensitive learning for *Interchange* and *In-Metro* state inference. We adopt this model for passenger state inference for the following reasons:

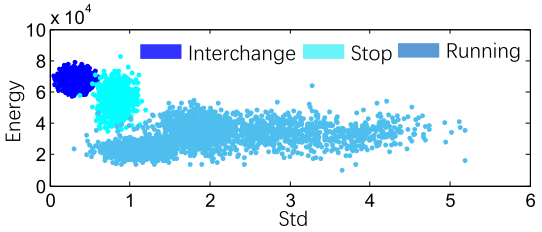

Fig. 4. RSSI feature distributions.

- A Bayes classifier is suitable for conditional and independent contextual features extracted from smartphone sensors. Since the three dimensional features (*i.e.* magnetic field, acceleration difference, and RSSI) are not correlated during the metro trip, we can assume they are independent and apply Naive Bayes model to make a classification.
- The cost-sensitive algorithm can deal with imbalanced metro trip data. As the duration of an interchange is only a small portion of a whole metro trip, there will be much fewer measurements collected during *Interchange* than *In-Metro*. Such imbalances in the measurement can lead to bias when training the standard Naive Bayes model. The cost-sensitive algorithm imposes a higher cost on the minority class to avoid such a bias, thus ensuring the inference accuracy with an imbalanced training dataset.
- The cost-sensitive Naive Bayes model provides fast, scalable and efficient model building and scoring, and often outperforms the more sophisticated algorithms for binary classification.

4.2.1. Problem formulation

Given a set of variable features, $\vec{X} = \{X_1, \ldots, X_i, \ldots, X_N\}$, where $X_i \in \{F_{mag}, F_{acc}, F_{rss}\}$ is the ith attribute of feature, we construct a posterior probability $P(Y_t|\vec{X})$ for a state $Y_t \in \{In\text{-}Metro, Interchange\}$ at each *inference interval t*. Note that the state might change during an *inference interval*. We label the state as the one with the longest duration. For instance, if a passenger is in-metro during the first 5 s , and gets off to transfer for the last 15 s, his/her state is labeled as *Interchange*.

4.2.2. State inference

Traditional Naive Bayes model: According to the Bayes Theorem and conditional independence assumptions of features X_j given Y_t, the posterior probability $P(Y_t = y_i|\vec{X})$ can be represented as:

$$p(Y_t = y_i|\vec{X}) = \frac{P(Y_t = y_i)P(\vec{X}|Y_t = y_i)}{\sum_k P(Y_t = y_k)P(\vec{X}|Y_t = y_k)} = \frac{P(Y_t = y_i)\prod_j P(X_j|Y_t = y_i)}{\sum_k P(Y_t = y_k)\prod_j P(X_j|Y_t = y_k)} \tag{1}$$

where y_i denotes the ith possible value of Y_t. X_j denotes the jth attribute of $\vec{X}$, and the summation in the denominator is over all legal values of the random variable Y_t. After learning the distributions $P(Y_t)$ and $P(X_j|Y_t)$ from the training dataset, the traditional Naive Bayes model infers the state Y_t with the most probable value based on the following rule:

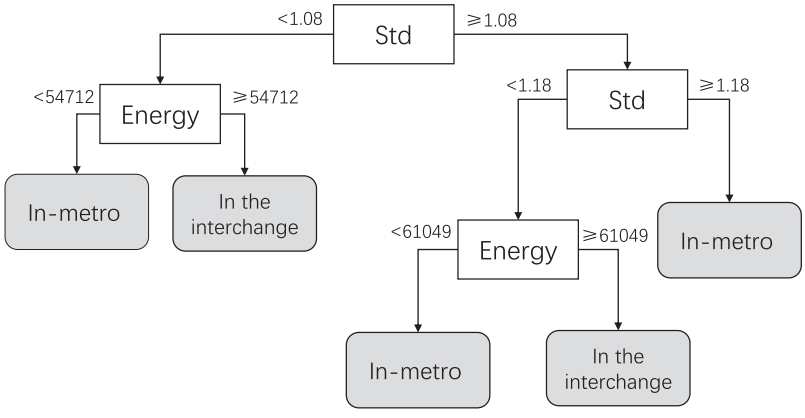

Fig. 5. Decision tree for GSM RSSI profile classification.

$$Y_t \leftarrow \underset{y_i}{argmax} \frac{P(Y_t = y_i)\prod_j P(X_j|Y_t = y_i)}{\sum_k P(Y_t = y_k)\prod_j P(X_j|Y_t = y_k)} \tag{2}$$

Cost-Sensitive Naive Bayes model: Instead of selecting the class with maximal probability, the cost-sensitive Naive Bayes Model is applied to minimize the risk function, which is constructed in Eq. (3):

$$R(Y_t = y_j|\vec{X}) = \sum P(Y_t = y_i|\vec{X}) \times C(y_i, y_j) \tag{3}$$

where $P(Y_t = y_i|\vec{X})$ can be obtained from Eq. (1). The cost $C(y_i, y_j)$ denotes the cost of the sample belonging to state y_i which is misclassified to state y_j, whose value is calculated as Eq. (4). p_i is the percentage of training samples with state y_i in the total training samples, and p_j is that of state y_j in the training samples.

$$C(y_i, y_j) = \begin{cases} (\frac{p_i}{p_j})^{\beta}, p_i > p_j; \\ (\frac{p_j}{p_i})^{\alpha}, p_i < p_j; \\ 1, p_i = p_j; \\ 0, i = j; \end{cases} \tag{4}$$

The state Y_t is finally predicted as the class with minimal value of risk function as follows:

$$Y_t \leftarrow \underset{y_i}{argmin} R(Y = y_i|\vec{X}). \tag{5}$$

In the cost-sensitive Naive Bayes model, we purposely increase the cost of a sample belonging to a minority state which is misclassified as a sample belonging to a majority state so as to boost the corresponding value of the risk function. This simple yet efficient algorithm can handle with the problem of an imbalance in the training dataset, which is well applied on the metro scenarios.

5. Fine-grained interchange state inference

After determining that a passenger is in the *Interchange* state, a fine-grained interchange case prediction module is triggered to further categorize the passenger interchange process. According to the transferring modes, we summarize the fine-grained interchange states into the following ten cases, which cover most modes that passengers transfer from one metro line to another.

1. UEL: passenger going **u**p by an **el**evator.
2. DEL: passenger going **d**own by an **el**evator.
3. UES: passenger going **u**p by an **es**calator.
4. DES: passenger going **d**own by an **es**calator.
5. WUS: passenger **w**alking **u**p**s**tairs.
6. WDS: passenger **w**alking **d**own**s**tairs.
7. WAI: passenger **wai**ting on a platform or at the entrance of an elevator/escalator.
8. WTP: passenger **w**alking **t**hrough a **p**assage.
9. WDE: passenger **w**alking **d**own on an **e**scalator.
10. WUE: passenger **w**alking **u**p on an **e**scalator.

As discussed in Section 4.1.2, we can leverage the repetitive patterns in acceleration to differentiate walking-related interchange states (WUS, WDS, WTP, WDE, WUE) and non-walking states (UEL, DEL, UES, DES, WAI). However, acceleration alone is insufficient to distinguish the above ten cases. Therefore, we combine the accelerometer and barometer embedded in smartphones for a fine-grained interchange state inference. The insight is that barometer is able to measure changes in pressure which correspond to changes in height. Fig. 6 shows an illustrative measurement of the height variance during 5 s for the 10 cases. As is shown, the slope of height variance varies when passengers ascend or descend using different tools for interchange.

Fig. 7 further plots the height gradient distributions of the above 10 cases from 8080 measurements collected at 13 main interchange stations in Beijing, Shanghai and Shenzhen. As is shown, the absolute height gradient of taking an elevator up/-down (0.06–0.15) and walking up/down via an escalator (0.06–0.19) are larger than those of others. This is because the elevator moves perpendicularly, rising faster than escalators. When passengers are walking on the escalator, the total height change speed is the sum of those for walking on staircases and a moving escalator. The height gradient of walking in a passage and waiting on a platform, is stable from −0.02 to 0.02. However, the height gradient of an escalator going up/down and upstairs/downstairs are similar, ranging from 0.02 to 0.06 in absolute terms, which cannot be easily distinguished by measuring the height but is distinguishable with the repetitive patterns in acceleration. Based on the above observations, we

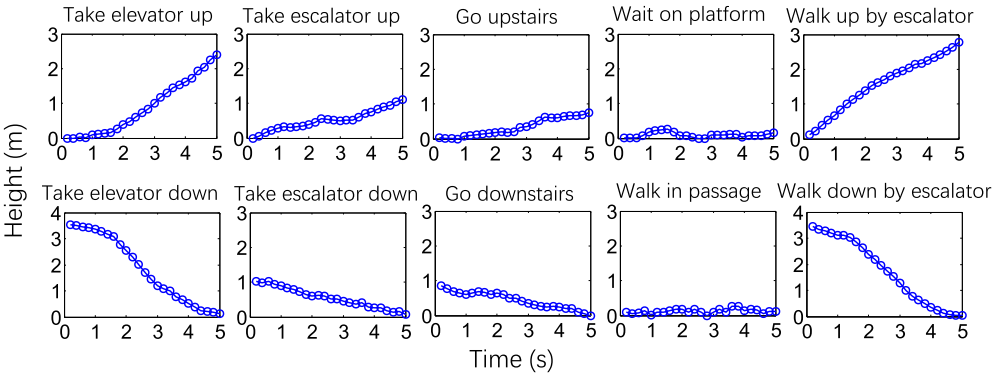

Fig. 6. Height variance for the 10 interchange cases.

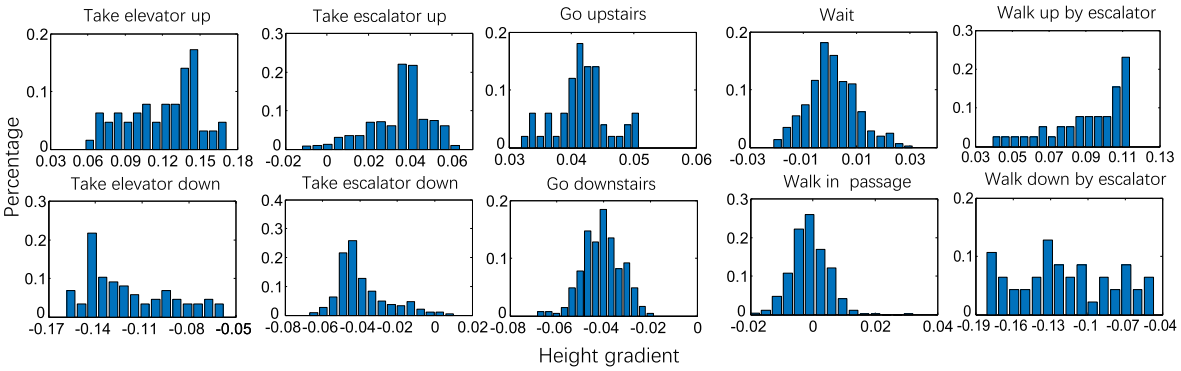

Fig. 7. Distribution of height gradient for the 10 interchange cases.

combine the *gradient* of height variance with the *period* of acceleration to recognize the above 10 cases during interchange, and summarize their profiles in Table 3. The thresholds are empirically set based on our experimental measurements.

5.1. Interchange case prediction model

We adopt a classic temporal graphical model, conditional random field (CRF) (Lafferty et al., 2001), to infer the interchange cases. Different to the previous temporal statistic models (Zhou and Spanos, 2016; Zhou and Spanos, 2016), CRF is able to depict the temporal relations between the observable features as well as the connections in the hidden stages (*i.e.*, the 10 fine-grained cases during interchange), which has been widely used in numerous applications (Gu et al., 2014b; Gu et al., 2016c). This calculates a single log-linear distribution over label sequences $\vec{Y}$ based on particular observation features $\vec{X}$ by maximizing the conditional probability $p(\vec{Y}\,|\,\vec{X})$. Compared to HMMs, CRFs allow features to depend on several states to account for long-term effects. Moreover, it also outperforms Maximum Entropy Markov models (McCallum et al., 2000) by resolving the label bias issue (Lafferty et al., 2001; Zhou et al., 2016). These advantages enable the CRF to characterize the relations between the observable contextual features and the hidden states in more depth, making it more suitable for fine-grained interchange case inference.

Denote the hidden variable sequence as $\vec{Y} = \{Y_1, Y_2, \ldots, Y_t\}$, where Y_t belongs to the 10 cases representing the predicted state of *Interchange case recognition interval t*. Similar to SubSection 4.2, the case is labeled as the one with the longest period if multiple cases occur in an *Interchange case recognition interval*.

The observation sequence is represented by $\vec{X} = \{X_1, X_2, \ldots, X_t\}$, where each X_t is the corresponding period and height gradient features extracted at the *Inference interval t*. For a label sequence $\vec{Y}$ with $N+1$ variables, its conditional probability $p(\vec{Y}\,|\,\vec{X})$ on the entire observation sequence $\vec{X}$ is given by the normalized form of potential functions,

$$p(\vec{Y}\,|\,\vec{X}) = \frac{1}{Z_{\vec{\lambda}}(\vec{X})} exp\left(\sum_{t=1}^{N}\sum_{i=1}^{M}\lambda_i f_i(Y_{t-1}, Y_t, \vec{X}, t)\right), \tag{6}$$

Table 3
The features of 10 interchange case for classification.

Category	Period feature	Height gradient feature	Description case
Walking	[0.5–2 s]	<−0.06	WDE: walk down on the escalator
		−0.06 to −0.02	WDS: walk downstairs
		−0.02 to 0.02	WTP: walk through a passage
		0.02–0.06	WUS: walk upstairs
		>0.06	WUE: walk up on the escalator
Non-walking	<0.5 s and >2 s	<−0.06	DEL: take elevator down
		−0.06 to −0.02	DES: take escalator down
		−0.02 to 0.02	WAI: wait on platform or entrance of elevator/escalator
		0.02–0.06	UES: take escalator up
		>0.06	UEL: take elevator up

where $Z_{\vec{\lambda}}(\vec{X})$ is the normalization factor, $f(Y_{t-1}, Y_t, \vec{X}, t)$ is a real-valued feature function to express the empirical distribution of the training data, and λ is the weight of each feature function (Lafferty et al., 2001). Specifically, the M feature functions consist of two parts,

$$\sum_{i=1}^{M}\lambda_i f_i(Y_{t-1}, Y_t, \vec{X}) = \sum_{k=1}^{m} v_k fs_k(Y_t, \vec{X}, t) + \sum_{j=1}^{n}\mu_j ft_j(Y_{t-1}, Y_t, \vec{X}, t) \tag{7}$$

The first part is the state feature function $fs(Y_t, \vec{X}, t)$, which describes the correlations between the entire observation feature sequence $\vec{X}$ and the label at the current phase t. The second part is the transition state function $ft(Y_{t-1}, Y_t, \vec{X}, t)$, which establishes the transition relations of the observation sequence $\vec{X}$ with the previous and current hidden states at the intervals $t-1$ and t. This function can well characterize the temporal associations of the hidden cases. The weights v and μ of these feature functions are calculated by maximizing the conditional log-likelihood of the labeled sequences. We adopt the L-BFGS (Liu and Nocedal, 1989) for efficient training, and apply the Viterbi Algorithm (Lou, 1995) to infer the hidden states based on the potential functions.

In summary, the work flow of fine-grained interchange case inference is as follows. After an *Interchange* state is identified, we first divide its duration into 4 parts, whose duration approximately equals to the shortest possible case period in the interchange. Second, we calculate the period and height gradient features, and feed them into a CRF model. Finally the output label of the CRF model is treated as the current case.

6. Validation

This section presents the evaluation methodologies and validates the proposed method to accurately infer interchange states and measure interchange times.

6.1. Experimental setup

6.1.1. Prototype implementation

We implement our two-tier interchange state inference module as an Android application, and test it on the mainstream Android operation systems (*i.e.* OS 4.0, 5.0 and 6.0) in several popular commercial mobile phones (*e.g.* Samsung Galaxy S6, Nexus 3, HTC Desire A6, HUAWEI 4C). Since our application is independent of platforms, we envision it to be easily extended to other mobile OS like iOS.

6.1.2. Data collection

We install our system on the smartphones of participants, and each of them launches it at the beginning of each metro trip. Then the application invokes the corresponding sensors to collect measurements. Participants manually label the states of the sensory measurements every *State inference interval* and *Interchange case recognition interval*, which are sufficient to make a good inference in our case. If more than one state/case occurs in the *State inference interval*/*Interchange case recognition interval*, we label the interval with the majority one.

We use these labels as the ground truth in our evaluations. In total, 3840 sets of metro trips have been conducted by 32 volunteers in 6 months, covering 14 metro lines in Beijing, Shanghai and Shenzhen, across 37 interchange stations, where each metro trip lasts for 40 min on average. During the measurements, the volunteers are free to stand or sit in the metro vehicle while hold their smartphones, and put them in pockets or in bags as shown in Fig. 8. They can use their phones as per normal while in a metro trip.

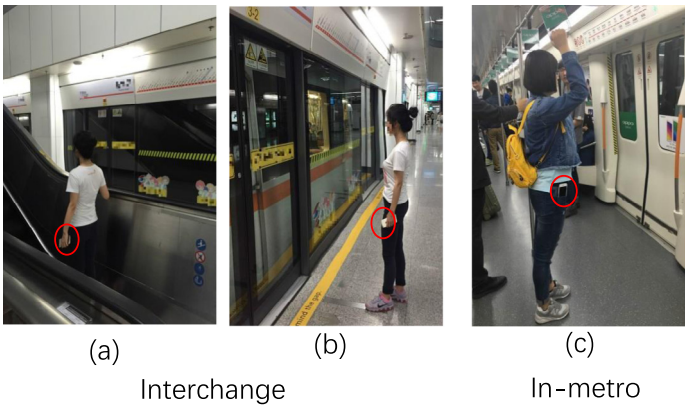

Fig. 8. Illustration of experiment cases.

6.2. State inference accuracy

In this subsection, we details the performance of *Interchange* and *In-Metro* state inference.

Inference accuracy of states. Fig. 9 shows the results of 10-fold cross validation on *Interchange* and *In-Metro* in our experimental dataset using a traditional Naive Bayes model and our cost-sensitive Naive Bayes model. As is shown, the average F1-measurement of the cost-sensitive Naive Bayes Model on *Interchange* ($F1 = 91.0\%$) is higher than that of the traditional Naive Bayes Model ($F1 = 82.9\%$). The F1-measurements of *In-Metro* state remain high ($F1 \approx 92.0\%$) on both models. The results validate that the cost-sensitive Naive Bayes model addresses the imbalanced training dataset problem and achieves a high inference accuracy for *Interchange* even with a smaller amount of training data. We find that most inference errors occur during state transitions, because contextual features may not change immediately when the passenger's state changes.

Effectiveness of feature selection. To show the effectiveness of the selected features, we use the training dataset to train our cost-sensitive Naive Bayes classifier, and evaluate its inference performance in the testing dataset by adding features. The results are shown in Table 4. As can be seen, the precision and recall (Sokolova and Lapalme, 2009) both improve with more features, and finally reach 89.6% and 92.4% of *Interchange*, and 93.6% and 91.3% of *In-Metro*, showing that the selected features are helpful for inference. By integrating the multi-dimensional features, statistics models in our work enable to copy with the sensory noise brought by the environment or personal activities (*i.e.* playing games).

Accumulated inference error time of two states. Fig. 10 shows the timing errors of two states during each metro trip. 97% of the timing errors for *Interchange* and *In-Metro* are about 2 min and 3.5 min, respectively. The accumulates time error of the overall metro trip, therefore, is under 6 min given an interference interval of 20 s. Although the inference performance of the *Interchange* state ($F1 = 92.4\%$) is not as good as that of *In-Metro* state ($F1 = 90.9\%$), its timing error is much shorter, due to its limited proportion of a metro trip.

Baselines comparison. We compare the inference results of *Interchange* with the approaches based on the analysis of the automatic fare collection (AFC) data. This method has been widely applied on many transportation related works (Sun and Xu, 2012; Zhang and Yao, 2015), where the *interchange time* (IT) can be calculated in Eq. (8) (Zhang and Yao, 2015).

$$IT = T_d - T_o - T_{act} - T_{egt} - T_{InMetro} \tag{8}$$

T_d indicates the destination time of exiting. T_o dedicates the origin time of entering. T_{act} states the access time from the origin entrance to the platform. T_{egt} points to the egress time from the final platform to the destination exit. $T_{InMetro}$ is the duration of in the train. Specifically, T_d and T_o are obtained by the AFC date. $T_{InMetro}$ is achieved by the subway timetable. T_{act} and T_{egt} are modeled by comparing from the origin entrance/ destination exit time with the train's timetable. Fig. 11

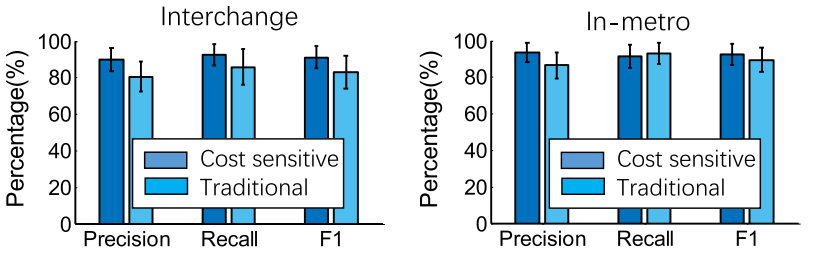

Fig. 9. Traditional Naive Bayes model vs. cost-sensitive Naive Bayes model.

Table 4
Effectiveness of features on *Interchange* detection.

Model	In-Metro		Interchange	
Feature	Precision (%)	Recall (%)	Precision (%)	Recall (%)
F_{mag}	34.1	32.5	29.9	31.1
$F_{mag} + F_{acc}$	77.4	68.9	74.7	78.5
$F_{mag} + F_{acc} + F_{rss}$	93.6	91.3	89.6	92.4

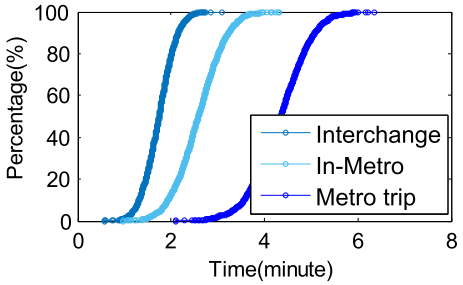

Fig. 10. Accumulated inference error time.

compares the inference accuracy of our method and AFC based method. As we can see, the advantages of our methods are more prominent with the increasing number of interchange stations in a metro trip. This is mainly because the parameters of a temporal model in AFC-based analysis are often invalid in practice. Personal behaviors (*i.e.* step frequency, walking speed) is hard to model with some specific models.

6.3. Accuracy of fine-grained interchange case inference

We extract 16,650 accurately predicted measurements of *Interchange* states from the first layer as the dataset to evaluate the fine-grained interchange case inference module. We further divide the measurements of each *Interchange* state into 4 segments, whose periods are the same as the *interchange case recognition duration*. In total, 66,600 samples of 10 interchange cases are used for evaluation. Participants manually labeled their interchange case as the ground truth in the same way as in Section 6.1. Fine-grained interchange case inference is performed every 5 s.

Table 5 shows the confusion matrix of fine-grained interchange case inference. As is shown, our approach yields inference accuracies of above 82% for all the 10 interchange cases. The highest accuracy peaks at 94.6% in WDS, followed by DEL (91.1%), WTP (93.2%) and WDE (90.7%). The lowest inference accuracy is seen with DES, which is mistaken with DEL (7.9%), UES (0.4%) and WAI (9.6%). The main reason for such errors is that the height gradient may fluctuate at different phases of the same interchange case. For example, the height gradient is small at the start and end of going down on an escalator (DEL), *i.e.*, when passengers step onto or leave the escalator. Such height gradient patterns are similar to those at the start and end of UES (passenger going up on an escalator). The small height gradient also leads to confusion with WAI (passenger waiting). For DEL, the main misjudgements are due to passengers' occasionally shaking their smartphones with arms on escalators, which induces a larger height gradient.

Effectiveness of CRF model. To show the effectiveness of the CRF model, we compare the inference accuracy of CRF to that of two other classic models: (1) HMM, a statistical model known for its applications to temporal pattern recognition; (2) SVM, which can efficiently perform non-linear classifications using the kernel trick, implicitly mapping their inputs into

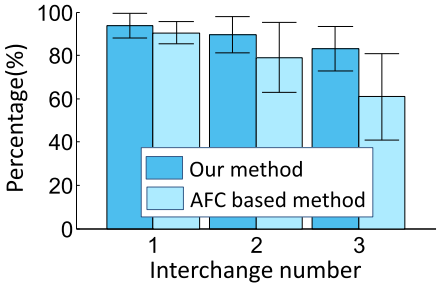

Fig. 11. Baseline comparison.

Table 5
Confusion matrix of 10 fine-grained interchange cases defined in Section 5.

Actual	Inferred									
	UEL	DEL	UES	DES	WUS	WDS	WAI	WTP	WDE	WUE
UEL (%)	86.3	3.0	4.3	0.0	0.0	0.0	0.0	0.0	2.1	4.3
DEL (%)	0.0	91.1	2.1	2.7	0.0	1.3	0.6	0.0	2.2	0.0
UES (%)	1.6	2.2	88.9	0.0	2.6	1.7	1.5	0.0	0.0	1.5
DES (%)	0.0	7.9	0.4	82.1	0.0	0.0	9.6	0.0	0.0	0.0
WUS (%)	0.0	0.0	7.5	0.0	84.3	0.0	0.1	8.1	0.0	0.0
WDS (%)	0.0	0.0	0.0	0.0	0.0	94.6	0.0	3.3	2.1	0.0
WAI (%)	0.4	0.8	3.2	2.9	3.5	1.7	83.3	3.9	0.0	0.3
WTP (%)	0.0	0.0	0.0	0.0	4.9	1.9	0.0	93.2	0.0	0.0
WDE (%)	0.0	5.8	0.0	0.0	0.0	2.7	0.0	0.8	90.7	0.0
WUE (%)	2.5	0.0	0.0	0.0	3.3	0.0	0.0	5.4	0.0	88.8

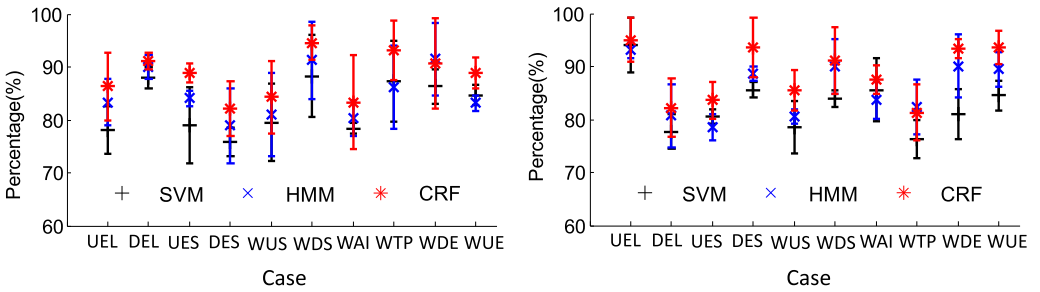

Fig. 12. Comparison of inference model.

high-dimensional feature spaces. Fig. 12 shows the comparison of model performances over the 10 interchange cases. As is shown, CRF outperforms both the HMM and SVM in terms of average precision and recall, and yields limited error bars ($\leqslant 10\%$), which verifies the advantages of the CRF model.

6.4. System overhead

Considering our application targets at continuous passenger tracking in the whole metro trip, it is crucial to evaluate the power consumption on the energy-constrained smartphones. Four popular commercial smartphones have been selected to test the system overhead and their configurations are summarized in Table 6.

In each testing phone, we install a battery logger to track its power consumption during the metro trips. Fig. 13 shows the results. As we can see, only less than 2% battery is consumed every ten minutes. In specific, the remaining battery capacity drops down as time goes by, and ends in 89.5% Galaxy S6, 92.4% for HUAWEI 4C, 90.8% for HTC Desire, and 93.2% for Lenovo after one hour, which demonstrates our system is power efficient and feasible for continuous passenger' tracking during the entire trips.

7. Case studies

The smartphone-based interchange time measurement approach is suitable for crowdsourcing to users. This section describes a set of case studies and findings in terms of metro interchange time across 34 interchange stations within 3 major cities in China collected by crowdsourced volunteers for around 3 months.

7.1. Macro views of interchange time

This subsection describes findings of the total interchange time during an entire metro trip.

Table 6
The configurations of smartphones.

Brand	CPU (GHz)	RAM (GB)	ROM (GB)	Power capacity (mAh)
Galaxy S6	8-cores 2.1	3	32	2550
HTC Desire A6	8-cores 1.7	2	16	2600
HUAWEI 4C	8-cores 1.2	2	8	3100
LenovoK80M	4-cores 1.8	4	64	4000

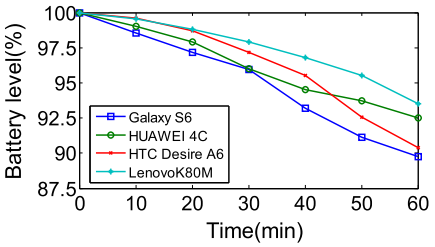

Fig. 13. Power consumption.

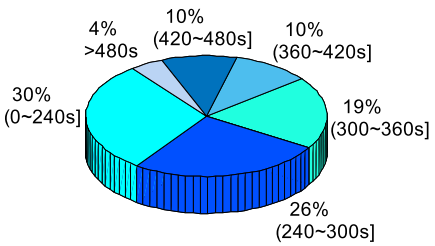

Fig. 14. Distribution of interchange time.

Fig. 15. Stations with interchange time over 360 s.

7.1.1. Distribution of interchange time

We calculate interchange times during a trip by continuously inferring the current trip state. State inference is performed every 20 s, and interchange times are calculated by merging the durations of consecutive *Interchange* states. Fig. 14 shows the average interchange times for all metro interchange stations collected in our testing dataset. As is shown, around 75% interchange times take less than 360 s, while about 4% take more than 480 s. We also mark the interchange stations whose transfer times exceed 360 s with red nodes in the three cities as shown in Fig. 15. The names of interchange stations are reported by the users when we collect their trip data. Long average interchange times are expected at 12 interchange stations among 34 interchange stations tested in our case study. The main reasons for long interchange times are (1) long transfer distances between certain subway lines and (2) crowded passages, especially at the entrances of elevators, escalators and stairs, which will be shown in Sections 7.2.3 and 7.2.4.

7.1.2. Fluctuation of interchange time

Fig. 16 shows the interchange times during different hours of a working day at two major interchange metro stations (Shenzhen North Station and Shanghai People's Square Station). The average interchange time of the two stations is above 360 s (6 min). The boxes in the figure represent the range between the 25th and 75th percentiles, and the whiskers show the highest and lowest values within 1.5 times the inter-quantile ranges. Data points beyond the whiskers are marked by '+'. As expected, most passengers experience longer interchange times during peak hours. The median interchange time at Shenzhen North Station is 9.5 min during 8:00 to 9:00, and 17:00 to 18:00, while that at People's Square Station is around 8.3 min. The median interchange times at both stations drop to around 6 min during off-peak hours. This variability of interchange time within a day might help passengers plan their metro trips and also provide guidance for traffic control.

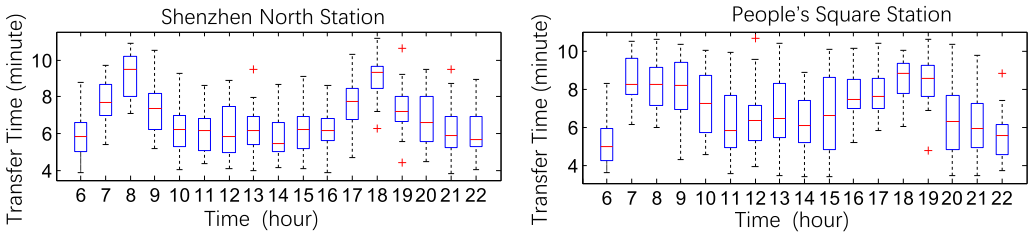

Fig. 16. Interchange time variance of day.

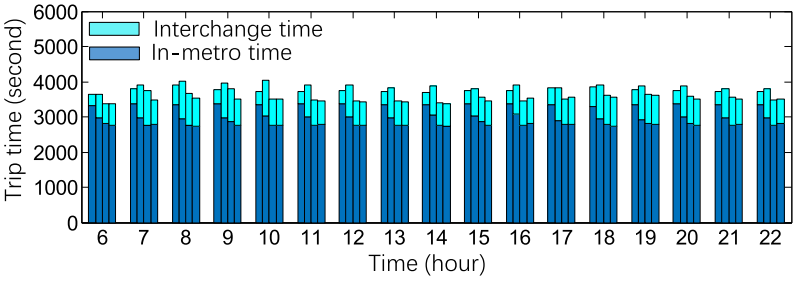

Fig. 17. *Interchange* and *In-Metro* times of 4 routes for the same origin-destination pair. (In each group, the 1st bar indicates to Line 2 → Line 1, and the 2ed bar indicates to Line 2 → Line 4 → Line 1, and the 3rd bar indicates to Line 7 → Line 4 → Line 1, and the 4th bar indicates to Line 7 → Line 12 → Line 1.)

7.1.3. Interchange time and In-Metro time

Given a fixed origin and destination station, there can be multiple route choices, which vary in interchange and in-metro times. As an illustration, we plot the *Interchange* time and *In-Metro* time at different hours of a day for 4 common metro routes from Shanghai Longyang Road Station to Shanghai Xinzhuang Station in Fig. 17 (Route 1: Line 2 → Line 1; Route 2: Line 2 → Line 4 → Line 1; Route 3: Line 7 → Line 4 → Line 1; Route 4: Line 7 → Line 12 → Line 1). As is shown, the total times of route 3 and 4 are shorter among the 4 routes, but their *Interchange* times are longer than those of route 1, whose trip duration is longer with the shortest *Interchange* time. Since long interchange times may cause an uncomfortable experience, some passengers may prefer longer trips with short interchange times. Therefore, by crowdsourcing interchange time measurements and aggregating information for different routes, our method can provide route recommendations to passengers based on different *Interchange* and *In-Metro* times.

7.2. Micro views of interchange time

This subsection provides micro views for the fine-grained interchange times during metro transfers.

7.2.1. Analysis of waiting time for metro

We divide *Interchange* times into two parts: the time taken by passengers waiting on the platform during interchanges, and the remaining transfer time, which starts with departing from a feeder train and ends with reaching the waiting platform. While the transfer times from a feeder train to the next platform are station-dependent, the waiting times on the platform vary and can be minimized by adjusting the metro schedules. For instance, if the transfer time $T_{transfer}$ is known, metro officials can modify the arrival time of the feeder train T^{feeder}_{arrive}, and the arrival time of the receiver train $T^{receiver}_{arrival}$ to minimize waiting time on platform T_{wait} by Eq. (9):

$$T_{wait} = T^{receiver}_{arrival} - T^{feeder}_{arrive} - T_{transfer} \tag{9}$$

We calculate the waiting time T_{wait} on the platform by extracting the last continuous durations of waiting cases (WAI) of the interchange time, and the rest interchange time is the transfer time $T_{transfer}$. Fig. 18 plots both T_{wait} and $T_{transfer}$. As is shown, most passengers spend 18% to 50% of their time waiting at the platform during the whole interchange process, which is frustrating if the passenger is in a hurry. According to the obtained $T_{transfer}$, metro officials can dynamically re-schedule the metros to optimize T_{wait}.

7.2.2. Decomposing interchange time

We detail the composition of interchange times based on our fine-grained interchange case inference method. We collect measurements of interchanges during both peak and off-peak hours at two major interchange stations. A total of 644 pieces of interchange measurements were collected at each station, with half collected during peak hours (7:00–9:00 and 17:00–

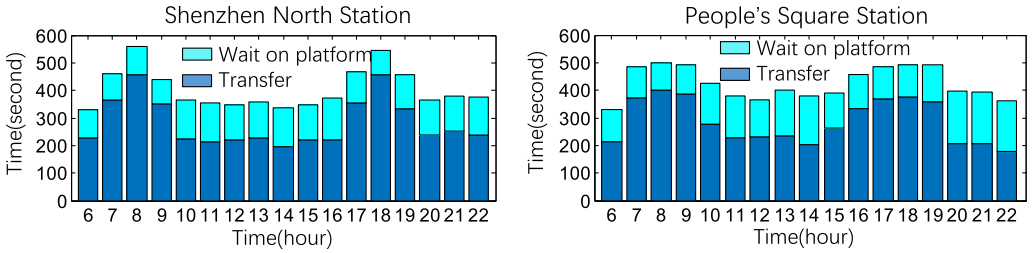

Fig. 18. Comparison of T_{wait} and $T_{transfer}$ during *Interchange* at different hours of a day.

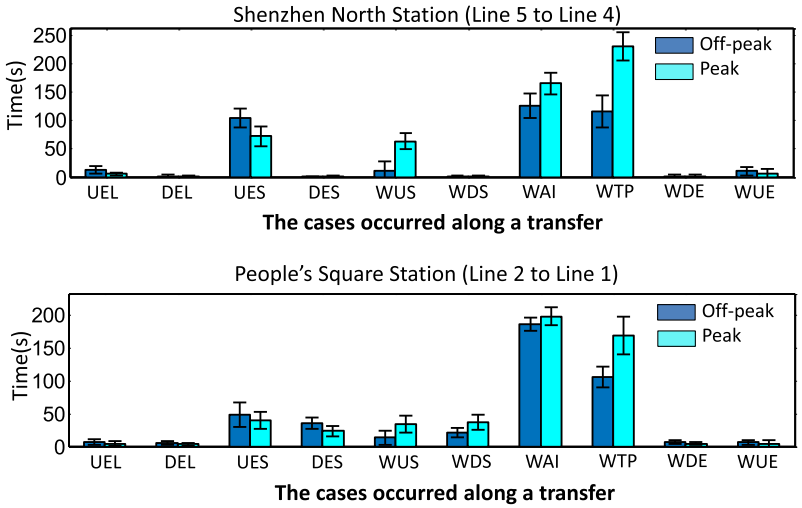

Fig. 19. Time distributions of the 10 interchange cases during peak and off-peak hours.

19:00) and the other half during off-peak hours. The detailed time of 10 interchange cases were inferred by the approach described in Section 5.

Fig. 19 shows the average time costs of 10 interchange cases. At Shenzhen North Station, the average times for UEL, UES, WUS, WAI, WTP and WUE is long but that of DEL, DES, WDS and WDE are short. This is because the usual interchange direction is going upwards to change from metro line 5 to metro line 4 at Shenzhen North Station. The average time of UES during peak hours is shorter than that during off-peak hours, while the average time of WUS during peak hours is longer than that during off-peak hours. This result shows that passengers tend to take escalators to go upwards during off-peak hours but walk upstairs during peak hours (probably because the escalators during peak hours are too crowded). At People's Square Station, however, both upward cases (UES, WUS, WUE) and downwards cases (WDS, WTP, WDE) take up large portions of the interchange times. This is because there are both upwards and downwards interchange directions at the People's Square Station. The time costs of UES and DES decrease from off-peak times to peak times, and increase in WUS and WDS, demonstrating that more passengers in Shanghai are willing to walk upstairs/downstairs over taking an escalator.

7.2.3. Decomposing waiting time

As the waiting times rise dramatically from off-peak hours to peak hours, we further analyze the detailed composition of waiting times during interchanges. We classify waiting times into three categories: at the entrance of elevators, at the entrance of escalators, and on the platform. To distinguish these three categories of waiting, we collect the waiting times before UEL and DEL as the waiting time at the entrance of elevators. The waiting times before UES, DES, WDE and WUE are counted as the waiting time at the entrance of escalators. The rest of the waiting times are regarded as the waiting time on the platform.

Fig. 20 plots the distributions of the three categories of waiting time. As is shown, the waiting times on the platform take up the majority of the total waiting time.

Even though there are fewer metros during off-peak hours, the overall waiting time is still shorter than during peak hours. Specifically, the waiting times at the entrances of elevator/escalator grow during peak hours, indicating there may be large crowds at the entrances during peak hours, which is a sign for metro officials to control passenger flow.

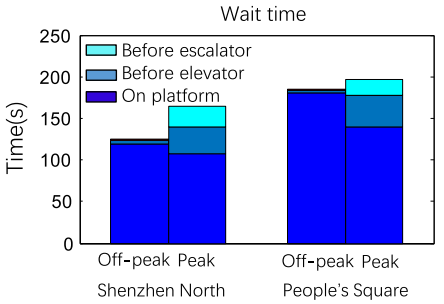

Fig. 20. Decomposed waiting time during interchange.

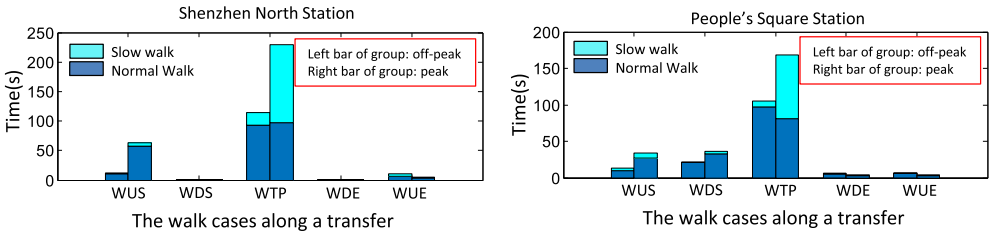

Fig. 21. Fine-grained walking time analysis.

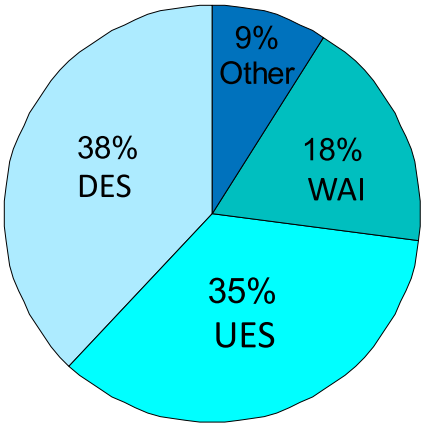

Fig. 22. Fine-grained slow walk time analysis.

7.2.4. Decomposing walking time

We classify the walking status based on walking speeds into slow walking (with period of [1.2–2 s]) and normal walking (with period of [0.5–1.2 s]).

Fig. 21 shows the slow walking times and normal walking times for the 5 walking cases during interchanges. The times of slow walking remain stable from off-peak hours to peak hours for WUS, WDS, WDE and WUE, but dramatically increase for WTP. The results indicate that walking speeds of going upstairs/downstairs (WUS/WDS) and taking elevators (WDE/WUE) are not significantly affected during peak hours. However, walking speeds in the passage drop significantly during peak hours, indicating potential crowdedness in the passage.

To further investigate the destinations (*i.e.* the next interchange case) after slow walking in the passage, we look into the subsequent cases after slow walking in WTP. Fig. 22 summarizes the distributions of the subsequent cases after slow walking in the passage. From the plot we learn that most WTP (slow walking through the passage) occur before UES (going up by escalator, 38%), DES (going down by escalator, 35%) and WAI (waiting on a platform, 18%). The results show that passengers tend to get blocked at the entrances of escalators and sometimes even on the platform. Such results enable metro officials to monitor the situation of metro stations.

8. Conclusion

In this paper, we propose a smartphone-based interchange time measurement approach for underground public transportation such as metros. We utilize carefully selected sensor modalities on smartphones to record ambient contextual features, design a two-tier classifier to infer interchange states during a metro trip, and further differentiate 10 fine-grained cases during interchanges. We implemented our system as an Android application and evaluated its performance across 14 metro lines in 3 major cities in China within 6 months. Experimental results show that our method yields an overall F1-measurement of 91.0% for interchange state inference and a time error of 2 min with an inference interval of 20 s. We also achieve an average accuracy of 89.3% in distinguishing 10 fine-grained interchange cases. From the case studies with crowdsourced data collected in 3 months, we obtain detailed views of interchange times from both macro and micro perspectives, such as the interchange stations with long interchange times, distributions of interchange times within a day, portions of waiting times during interchanges, interchange direction, usage of facilities (stairs/escalators/lifts), and the root causes of long interchange times. Future directions include more energy-efficient sensing frameworks for long-term crowdsourced interchange time measurements, improving time measurement accuracy, metro schedule optimization and route recommendation based on real-time, fine-grained interchange time measurements at an urban scale.

References

Aguiléra, V., Allio, S., Benezech, V., Combes, F., Milion, C., 2014. Using cell phone data to measure quality of service and passenger flows of paris transit system. Transport. Res. Part C: Emerg. Technol. 43, 198–211.

Beauregard, S., Haas, H., 2006. Pedestrian dead reckoning: a basis for personal positioning. In: Proceedings of the 3rd Workshop on Positioning, Navigation and Communication, pp. 27–35.

Bierlaire, M., Chen, J., Newman, J., 2013. A probabilistic map matching method for smartphone gps data. Transport. Res. Part C: Emerg. Technol. 26, 78–98.

Briand, A.S., Côme, E., Trépanier, M., Oukhellou, L., 2017. Analyzing year-to-year changes in public transport passenger behaviour using smart card data. Transport. Res. Part C: Emerg. Technol. 79, 274–289.

Carrel, A., Lau, P.S., Mishalani, R.G., Sengupta, R., Walker, J.L., 2015. Quantifying transit travel experiences from the users perspective with high-resolution smartphone and vehicle location data: methodologies, validation, and example analyses. Transport. Res. Part C: Emerg. Technol. 58, 224–239.

Chen, Z., Zou, H., Jiang, H., Zhu, Q., Soh, Y.C., Xie, L., 2015. Fusion of wifi, smartphone sensors and landmarks using the kalman filter for indoor localization. Sensors 15, 715–732.

Elhamshary, M., Youssef, M., Uchiyama, A., Yamaguchi, H., Higashino, T., 2016. Transitlabel: a crowd-sensing system for automatic labeling of transit stations semantics. In: Proc. ACM MobiSys, pp. 193–206.

Elkan, C., 2001. The foundations of cost-sensitive learning.

Fan, Y., Guthrie, A., Levinson, D., 2016. Waiting time perceptions at transit stops and stations: effects of basic amenities, gender, and security. Transport. Res. Part A: Policy Pract. 88, 251–264.

Feng, T., Timmermans, H.J., 2013. Transportation mode recognition using gps and accelerometer data. Transport. Res. Part C: Emerg. Technol. 37, 118–130.

Gu, W., 2017. Non-intrusive blood glucose monitor by multi-task deep learning: Phd forum abstract. In: Proceedings of the 16th ACM/IEEE International Conference on Information Processing in Sensor Networks. ACM, pp. 249–250.

Gu, W., Jin, M., Zhou, Z., Spanos, C.J., Zhang, L., 2016a. Metroeye: smart tracking your metro trips underground. In: Proceedings of the 13th International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services. ACM, pp. 84–93.

Gu, W., Jin, M., Zhou, Z., Spanos, C.J., Zhang, L., 2016b. Metroeye: towards fine-grained passenger tracking underground. In: Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct. ACM, pp. 77–80.

Gu, W., Shangguan, L., Yang, Z., Liu, Y., 2016c. Sleep hunter: towards fine grained sleep stage tracking with smartphones. IEEE Trans. Mobile Comput. 15, 1514–1527.

Gu, W., Yang, Z., Shangguan, L., Ji, X., Zhao, Y., 2014a. Toauth: towards automatic near field authentication for smartphones. In: 2014 IEEE 13th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom). IEEE, pp. 229–236.

Gu, W., Yang, Z., Shangguan, L., Sun, W., Jin, K., Liu, Y., 2014b. Intelligent sleep stage mining service with smartphones. In: Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing. ACM, pp. 649–660.

Hemminki, S., Nurmi, P., Tarkoma, S., 2013. Accelerometer-based transportation mode detection on smartphones. In: Proc. ACM SenSys, pp. 1–14.

Higuchi, T., Yamaguchi, H., Higashino, T., 2015. Tracking motion context of railway passengers by fusion of low-power sensors in mobile devices. In: Proc. ACM ISWC, pp. 163–170.

Lafferty, J.D., McCallum, A., Pereira, F.C.N., 2001. Conditional random fields: probabilistic models for segmenting and labeling sequence data. In: Proc. ACM ICML, pp. 282–289.

Lee, G., Han, D., 2014. Subway train stop detection using magnetometer sensing data. In: Proc. IPIN, pp. 766–769.

Liu, D.C., Nocedal, J., 1989. On the limited memory bfgs method for large scale optimization. Math. Program. 45, 503–528.

Lou, H.L., 1995. Implementing the viterbi algorithm. Signal Process. Mag. 12, 42–52.

Lu, X., Zou, H., Zhou, H., Xie, L., Huang, G.B., 2016. Robust extreme learning machine with its application to indoor positioning. IEEE Trans. Cybernet. 46, 194–205.

McCallum, A., Freitag, D., Pereira, F.C., 2000. Maximum entropy markov models for information extraction and segmentation. In: Proc. ACM ICML, pp. 591–598.

Misra, A., Gooze, A., Watkins, K., Asad, M., Le Dantec, C., 2014. Crowdsourcing and its application to transportation data collection and management. Transport. Res. Rec.: J. Transport. Res. Board, 1–8.

Ohashi, H., Akiyama, T., Yamamoto, M., Sato, A., 2014. Automatic trip-separation method using sensor data continuously collected by smartphone. In: Proc. IEEE ITSC, pp. 2984–2990.

Pachi, A., Ji, T., 2005. Frequency and velocity of people walking. Struct. Eng. 83.

Parbo, J., Anker Nielsen, O., Giacomo Prato, C., 2015. Passenger perspectives in railway timetabling: a literature review. Transport Rev., 1–27

Pelletier, M.P., Trépanier, M., Morency, C., 2011. Smart card data use in public transit: a literature review. Transport. Res. Part C: Emerg. Technol. 19, 557–568.

Qiu, Z., Zou, H., Jiang, H., Xie, L., Hong, Y., 2016. Consensus-based parallel extreme learning machine for indoor localization. In: 2016 IEEE Global Communications Conference (GLOBECOM). IEEE, pp. 1–6.

Sankaran, K., Zhu, M., Guo, X.F., Ananda, A.L., Chan, M.C., Peh, L.S., 2014. Using mobile phone barometer for low-power transportation context detection. In: Proc. ACM SenSys, pp. 191–205.

Sels, P., Dewilde, T., Cattrysse, D., Vansteenwegen, P., 2016. Reducing the passenger travel time in practice by the automated construction of a robust railway timetable. Transport. Res. Part B: Methodol. 84, 124–156.

Shangguan, L., Yang, Z., Zhou, Z., Zheng, X., Wu, C., Liu, Y., 2014. Crossnavi: enabling real-time crossroad navigation for the blind with commodity phones. In: Proceedings of ACM International Joint Conference on Pervasive and Ubiquitous Computing, pp. 787–798.
Sokolova, M., Lapalme, G., 2009. A systematic analysis of performance measures for classification tasks. Inform. Process. Manage. 45, 427–437.
Stockx, T., Hecht, B., Schöning, J., 2014. Subwayps: towards smartphone positioning in underground public transportation systems. In: Proc. ACM SIGSPATIAL, pp. 93–102.
Sun, Y., Xu, R., 2012. Rail transit travel time reliability and estimation of passenger route choice behavior: analysis using automatic fare collection data. Transport. Res. Rec.: J. Transport. Res. Board, 58–67.
Thiagarajan, A., Biagioni, J., Gerlich, T., Eriksson, J., 2010. Cooperative transit tracking using smart-phones. In: Proc. ACM SenSys, pp. 85–98.
Vlahogianni, E.I., Barmpounakis, E.N., 2017. Driving analytics using smartphones: algorithms, comparisons and challenges. Transport. Res. Part C: Emerg. Technol. 79, 196–206.
Wu, J., Liu, M., Sun, H., Li, T., Gao, Z., Wang, D.Z., 2015. Equity-based timetable synchronization optimization in urban subway network. Transport. Res. Part C: Emerg. Technol. 51, 1–18.
Xiao, J., Zhou, Z., Yi, Y., Ni, L.M., 2016. A survey on wireless indoor localization from the device perspective. ACM Comput. Surveys 49, 25:1–25:31.
Xiao, W., Ni, W., Toh, Y.K., 2011. Integrated wi-fi fingerprinting and inertial sensing for indoor positioning. In: IPIN, pp. 1–6.
Xu, X.y., Liu, J., Li, H.y., Hu, J.Q., 2014. Analysis of subway station capacity with the use of queueing theory. Transport. Res. Part C: Emerg. Technol. 38, 28–43.
Yang, Z., Shangguan, L., Gu, W., Zhou, Z., Wu, C., Liu, Y., 2014. Sherlock: micro-environment sensing for smartphones. IEEE Trans. Parallel Distrib. Syst. 25, 3295–3305.
Youssef, M., 2015. Towards truly ubiquitous indoor localization on a worldwide scale. In: Proceedings of the 23rd SIGSPATIAL International Conference on Advances in Geographic Information Systems. ACM, p. 12.
Yu, K., Zhu, H., Cao, H., Zhang, B., Chen, E., Tian, J., Rao, J., 2014. Learning to detect subway arrivals for passengers on a train. Front. Comput. Sci. 8, 316–329.
Zhang, Y.S., Yao, E.J., 2015. Splitting travel time based on afc data: estimating walking, waiting, transfer, and in-vehicle travel times in metro system. Discrete Dynam. Nat. Soc. 2015.
Zhao, J., Frumin, M., Wilson, N., Zhao, Z., 2013. Unified estimator for excess journey time under heterogeneous passenger incidence behavior using smartcard data. Transport. Res. Part C: Emerg. Technol. 34, 70–88.
Zhou, Y., Kang, Z., Zhang, L., Spanos, C., 2013. Causal analysis for non-stationary time series in sensor-rich smart buildings. In: 2013 IEEE International Conference on Automation Science and Engineering (CASE). IEEE, pp. 593–598.
Zhou, Y., Ninghang, H., Spanos, C.J., 2016. Veto-consensus multiple kernel learning. In: Thirtieth AAAI Conference, AAAI.
Zhou, Y., Spanos, C.J., 2016. Online learning of contextual hidden markov models for temporal-spatial data analysis. In: Control and Decision Conference (CDC). IEEE.
Zhou, Y., Spanos, C.J., 2016. Causal meets submodular: subset selection with directed information. In: The Thirtieth Annual Conference on Neural Information Processing Systems (NIPS), NIPS.
Zou, H., Chen, Z., Jiang, H., Xie, L., Spanos, C., 2017a. Accurate indoor localization and tracking using mobile phone inertial sensors, wifi and ibeacon. In: 2017 IEEE International Symposium on Inertial Sensors and Systems. IEEE.
Zou, H., Huang, B., Lu, X., Jiang, H., Xie, L., 2016. A robust indoor positioning system based on the procrustes analysis and weighted extreme learning machine. IEEE Trans. Wireless Commun. 15, 1252–1266.
Zou, H., Zhou, Y., Jiang, H., Huang, B., Xie, L., Spanos, C., 2017b. Adaptive localization in dynamic indoor environments by transfer kernel learning. In: 2017 IEEE Wireless Communications and Networking Conference (WCNC). IEEE, pp. 1–6.