A Wireless-Based Approach for Transit Analytics

Lei Kang Bozhao Qi Suman Banerjee
Department of Computer Sciences
University of Wisconsin-Madison
{lkang, bozhao, suman}@cs.wisc.edu

ABSTRACT

We propose Trellis — an in-vehicle WiFi-based tracking system that passively observes mobile devices and provides various analytics for transit operators. Our infrastructure is fairly low-cost and can be a complementary, yet efficient, mechanism by which such operators collect various information, e.g., popular original-destination stations of passengers, waiting times of passengers at stations, occupancy of vehicles, and more. A key challenge in our system is to efficiently determine which device is actually inside (or outside) of a transit vehicle, which we are able to address through contextual information. While our current system cannot provide accurate actual numbers of passengers, we expect the relative numbers and general trends to be still fairly useful from an analytics perspective. We have deployed a preliminary version of Trellis on two city buses in Madison, WI, and report on some general observations on transit efficiency over a period of four months.

Keywords

In-Vehicle Systems; Mobile Computing

1. INTRODUCTION

Public transit systems carry millions of users in their daily activities throughout the year and are, sometimes, an important part of public infrastructure provided by local governments. Like all systems, public transit has always looked for mechanisms that allow them to improve their services for people in terms of, say, what new routes or stops should be introduced, how do peak and off-peak behaviors be handled, and much more. Traditionally, these decisions are often based on limited surveys — the local Madison Metro Transit would use infrequent volunteers ask people about their transit preferences. However, just as mobile devices have transformed crowd-sourced data collection in a whole range of domains, we believe that transit systems can also benefit significantly from it. In this paper, we advocate a fairly low-cost and simple system through which a transit operator can gather significant user and usage analytics about its operations at a scale and form never possible before.

Examples of transit analytics and Trellis: Transit systems typically need to learn about a lot of usage characteristics. What are the most popular stops at different times of the day; what are wait times for its passengers; how long do they wait at exchange points waiting for the next vehicle; how occupied are different vehicles at different times of the day; and so on. Some of these questions are significantly related to funds allocated to them — in particular, operators sometimes receive government funds based on how many *passenger-miles* they carry annually [11, 3, 14]. Today, these operators use a number of low fidelity methods to collect such information. For instance, farecards swiped inside inside buses may allow the operator to know the stations at which passengers get on (although they might not be able to infer where the passengers get off). Similarly, optional surveys (either in person or over the phone) allow them to collect other statistics. Approaches such as the above tend to provide incomplete data or data with fairly low fidelity.

Our proposed system, Trellis, takes advantage of widely available mobile devices and the popular notion of crowdsourcing from many passengers to quickly gather such information at a significantly larger scale. Wi-Fi-based monitoring system has been widely used in many related scenarios, such as understanding network performance [8], estimating vehicle trajectories [16], and tracking human queues [18]. Trellis is based on similar principles and is fairly simple — it uses a low-end Wi-Fi monitoring unit mounted on the vehicle to determine when a certain passenger gets on and off the vehicle. The approach relies on the fact that many mobile devices typically have their Wi-Fi function turned on, which makes them sufficiently trackable. Obviously systems such as Trellis will miss accounting for passengers who travel without mobile devices or those with their Wi-Fi function turned off, but our experience shows that we can still track general trends in transit behavior quite effectively [1]. We recommend our current version of Trellis to track relative trends in transit systems, as opposed to using them for exact and absolute counts. (We note that in Trellis, we maintain user privacy by simply using consistent hashes on MAC addresses, and not the actual MAC address itself; the latter is dropped immediately.)

[1]Techniques such as randomized MAC addresses may lead to inaccuracies, but we should be able to systematically eliminate all devices that do so, while still keeping *relative* counts somewhat accurate.

HotMobile '16, February 26-27, 2016, St. Augustine, FL, USA

DOI: http://dx.doi.org/10.1145/2873587.2873589

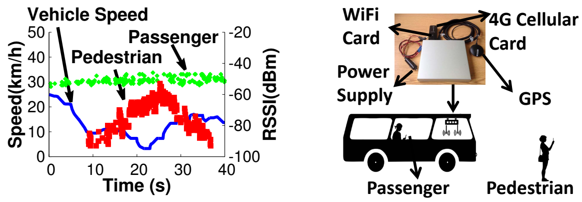

Figure 1: Different RSSI patterns between passenger and pedestrian, and the Wi-Fi monitor installed on vehicle.

We believe a simple and low-cost infrastructure such as Trellis mounted on public transit vehicles can be used to perform transit analytics for a wide range of questions effectively. For the purpose of this paper, we demonstrate how such a system maybe used to answer one specific question (just as an example) – *what are the origin-destination pairs of the user population and how does the popularity of these origin-destination pairs vary for different stations, at different locations, and at different times of the day.* Through our work, we demonstrate how we can build an origin-destination matrix to understand passenger travel patterns using Trellis, which can often identify alternative bus scheduling or routing to improve passenger travel times.

In the end, Trellis provides yet another approach to collect transit information in real-time and can potentially be combined with other existing or complementary approaches.

Trellis approach, some challenges, and preliminary implementation: We implement the Trellis system using off-the- shelf embedded platforms equipped with Wi-Fi interfaces and have deployed it to operate on two city buses in Madison, WI (in collaboration with our partners — Madison Metro Transit). In particular, our functionality is built into an existing system that provides a free Wi-Fi service, called WiRover [13], that is available on these city buses. Given that Wi-Fi services on transit systems are a growing phenomenon, the ability to add a system such as Trellis may not even require new hardware to be installed on these vehicles.

Many aspects of the design of Trellis is fairly intuitive. However, there are some specific challenges that we needed to address. One of them is to reliably determine whether an individual is actually located inside the vehicle (passenger) or outside of it (pedestrian). While one may consider existing localization techniques as that use mobile device Received Signal Strength Indication (RSSI) to infer this information, we have a much simpler mechanism to solve this issue. When a vehicle is moving, typically the signal strength of a passenger's mobile device observed by a vehicle-mounted observer will stay somewhat stable, while that of a pedestrian will fluctuate and eventually disappear (Fig. 1). Hence, by observing device signal strengths coupled with either vehicle location changes or speed of movement, one can easily discern who is inside the vehicle and who is not. This capability is a key building block in the Trellis system.

Contributions. We present, Trellis, a low-cost in-vehicle wireless monitoring system that can track station-to-station passenger movements to assist transit operators for transit user analytics. We develop various simple heuristic algorithms to separate passengers from pedestrians and identify where passengers get on or off a vehicle. To test the efficacy of our system, we have deployed this on two city buses in Madison, WI, over a period of four months and have evaluated how it can be used to infer popular original-destination stations of passengers over time and space. As we continue to work with our partners from Madison Metro Transit (our local transit operator), we continue to evaluate how such a system can be used to identify where to add new bus routes, or when to add faster (non-stop) services between various stations throughout the city at different times of the day and over different days of the week.

2. TRELLIS SYSTEM DESIGN AND IMPLEMENTATION

In this section, we discuss the system design, implementation and deployment.

2.1 System Design

Our system uses a front-end sniffing module to collect Wi-Fi devices' signals and transit GPS information, and uses a back-end modeling module to reconstruct transit schedules and human mobility patterns. The sniffing module collects the data from mobile devices and stores the data into local database. Meanwhile, the sniffing module can send calculated passenger number to remote server in real-time through cellular link, i.e., for the purpose of real time monitoring. Although our system supports real-time communication, we use separated program to send the data from databases to remote back-end server. The back-end server reconstruct public transit schedules and human mobility patterns from the collected data. It further aggregates the data from multiple transit sniffing system instances to provide a more complete view of the transit schedules and human mobility patterns. On top of the abstraction and aggregation modules, we construct origin-destination matrix to analyze transit efficiency in spatial and temporal domains.

2.2 System Implementation

We operate the Wi-Fi monitoring system on the Ubuntu 14.04.1 64bit distribution (with linux kernel version 3.19.0-28-generic), that runs on PC Engines APU platform [1]. APU platform is a mobile embedded platform that is equipped with 1GHz dual core CPU and 4G DDR3 DRAM. We use multi-thread program written in C/C++ to conduct the sniffing tasks. One thread is used to collect the Wi-Fi packets from the specified wireless interface. It also includes a module to check the correctness of received packets by validating the Cyclic Redundancy Check (CRC). Another thread is used to collect the GPS location information from the GPS module. All the data is stored in SQLite database files. There is another thread to send packets back to the data analysis modules, e.g., the number of passengers on bus for real-time demo. There are also bash scripts written to keep the cellular card and sniffer system running when the bus starts or the system aborts due to software or hardware failures. The data analysis modules are written in Java. Each data analysis module performs difference tasks, e.g., transit schedule reconstruction, automatic passenger counting etc.

2.3 System Deployment

We deploy our Wi-Fi monitoring system in two city buses. The bus route is illustrated in Fig. 2. The bus route covers the main campus of the University of Wisconsin-Madison (bottom right) as well as a residential area (top left) accommodates graduate students and visiting scholars. The

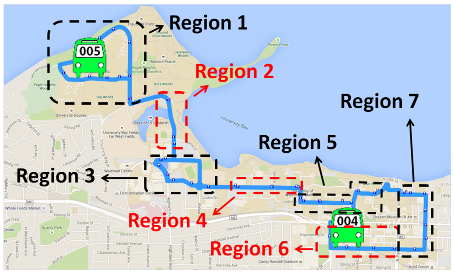

Figure 2: Bus route with labeled bus stops. The map size is around 1.5 mile × 2 mile. Each route traversal takes 45-50 minutes and covers about 8 miles. We divide the route into seven adjacent regions for easy analysis.

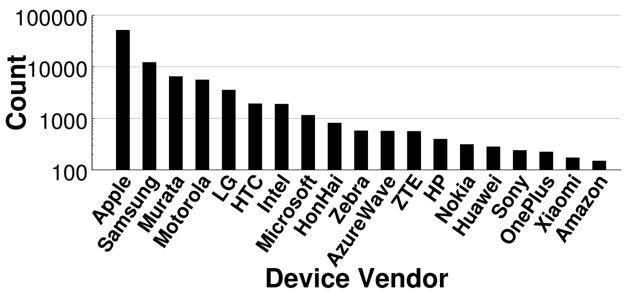

Figure 3: Distribution of devices by vendors in log scale.

city buses are operated by one local bus company. Based on our observation, the scheduling of each city bus is relatively random and the on-road or maintenance dates of each bus is unpredictable. There are usually multiple buses on the same route, while each bus is separated by 7 to 20 minutes based on the time of the day.

2.4 Statistical Properties

We collect data from both buses for around 90 days and 12 hours per day. In total, both buses travel more than 10,000 miles. Among the collected traces, we find 114,227 unique devices. By looking at the Organizationally Unique Identifier (OUI) of the MAC address (the first three octets), we are able to compare the distribution of various vendors. As shown in Fig. 3, Apple dominates all other vendors. Starting from iPhone 5s and iOS 8, Apple introduces randomized MAC address in probe requests under certain settings to protect user privacy. MAC randomization happens only in sleep mode (screen off) where the probe request with randomized MAC sent out roughly every 2-3 minutes. This feature certainly overestimates the number of iPhone users, but it exposes limited impact on statistical transit analytics.

3. PASSENGER TRACKING

In this section, we describe how to reconstruct bus schedules and passenger riding patterns.

3.1 Transit Schedule Reconstruction

For the purpose of public transit analytics, e.g., route design, scheduling, evaluation etc., it is important to track and record public transit when it passes each station. To reconstruct the public transit schedule from collected data, we extract the bus routes and stations from the transit operator's

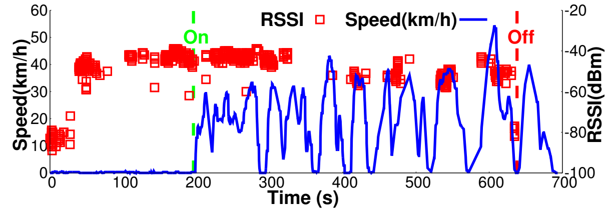

Figure 4: Identify where the passenger get on or off the bus.

website [5]. Each bus station is labeled by an index, GPS location information and direction. By matching the GPS information of the bus stations and that of collected from the sniffing system, we can know when the buses pass each bus station. Location information is not sufficient to accurately localize the bus at any specific time, because there are bus stations that are paired across street as dual way stations. To address this issue, we also need to match the heading direction of bus station with that of calculated from collected GPS. This module essentially provides when the bus arrives each station and how long it stays at that station. This information is important for transit operator to compare the actual operations of the bus with the ideal schedules. It can also be used to accurately identify when one passenger get on/off the bus.

3.2 Passenger Tracking

3.2.1 *Onboard Detection*

To track the passengers, we need to identify when and where they get on and off the bus. The most challenging task is to extract useful information from collected data. First, the RSSI readings are inaccurate and highly fluctuating. Therefore, we cannot use RSSI alone as the indicator to identify if one passenger is on bus. Second, the Wi-Fi signals are opportunistically received. The Wi-Fi signals from mobile device are based on user activities, e.g., screen on or off etc. Even worse, users may turn off Wi-Fi function to save power, which make some applications more challenging, e.g., automatic passenger counting etc.

To identify when and where one particular passenger get on and off the bus, we use multiple RSSI readings at different locations to track the position of the passenger. Essentially if there are consistent high RSSI readings after the bus traveling a certain distance, this device is on the bus with high probability. We will discuss how to find such a RSSI threshold δ in later section.

In Fig. 4, we illustrate the RSSI patterns and the time when the passenger get on and off the bus. We use similar logic to identify the bus stations where the passenger get on and off. We divide the entire bus trip into continuous road segments and each road segment is between two logically nearby bus stations. We identify if the passenger is on bus during this road segment by probing the RSSI readings of received packets. If there is at least a portion of α packets have RSSI readings higher than δ, then this device is on bus in this road segment. For each on bus passenger, it may travel with bus for one or more road segments. The starting bus station of the first such road segments is recognized as the bus station where the passenger get on the bus. The ending bus station of the last such road segments is recognized as the bus station where the passenger get off

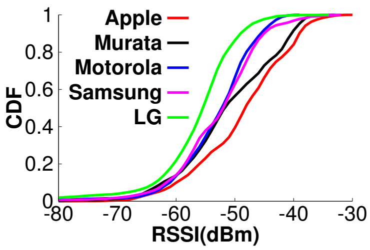

Figure 5: The CDF of on-bus mobile device Wi-Fi signals' RSSI readings.

the bus. It should be noted that we only use this method to identify the bus stations where the passenger get on and off the bus, if there is no packet received during some road segments in between, we still recognize this passenger is on the bus during the trip.

3.2.2 Parameter Selection

After we recognize the road segments the passenger is on bus, we collect the RSSI readings from different devices of various vendors. The cumulative distribution function (CDF) of on-bus RSSI readings are summarized in Fig. 5. This indicates that different thresholds δ should be assigned based on different vendors. Interestingly, the mobile devices from various vendors have huge difference on emitted power (10dB), presumably passengers (no matter what device he is using) are sitting randomly on the bus.

3.3 Origin-Destination Matrix

Another abstraction we build is the origin-destination matrix, which essentially records how many passengers ride from one bus station to another. Let S denotes this matrix and s_{ij} denotes each element in the matrix. s_{ij} refers to the number of passengers get on at bus station i and get off at bus station j. This matrix only builds the spatial relationships between bus stations, while temporary information is also important for transit analytics. We divide the 47 bus stations into seven geographically adjacent regions for easy analysis, i.e., as illustrated in Fig. 2. In the seven regions, there are 11,4,6,7,7,5 and 7 bus stations, respectively. Based on this matrix, we can analyze the region-to-region movement of the passengers. We may also add another dimension, i.e., time domain, to analyze passenger riding patterns in different periods of the day.

4. TRANSIT ANALYTICS

4.1 Automatic Passenger Counting

After reconstruct the transit schedules and passenger riding patterns, we conduct automatic passenger counting to record how many (essentially which) passengers getting on and off at each bus stations. This information is important for transit operators to make transit plans, improve the transit efficiency and seek government funding.

There are several popular methods that current transit operators are using to do this task. First, they are using ticketing system combined with human labor manual counting method. However, most ticketing systems only record how many passengers (assuming they are using traceable tickets instead of cash) get on buses, and cannot record how many passengers get off buses. Human labor counting is expensive and time consuming. Second, they are using camera or infrared sensors. But these systems are very expensive and not easy to deploy. Also existing methods are not able to track individual passenger.

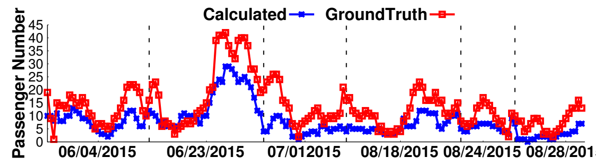

Figure 6: Automatic passenger counting results and ground truth.

Table 1: Ground Truth Data

Date	Time	Duration(mins)	Corr.
06/04/2015	12:15	42	0.72
06/23/2015	11:14	51	0.92
07/01/2015	15:28	48	0.71
08/18/2015	14:26	63	0.62
08/24/2015	11:26	38	0.88
08/28/2015	13:44	49	0.69

Our system provides a low-cost approach to assist or even replace existing counting methods. We evaluate counting accuracy by calculating the correlation between estimated passenger numbers and ground truth. The ground truth data is collected by volunteers who take the bus and count the number of passengers getting on/off the bus at each bus station. We collect the ground truth data in six trips on six different dates. The start time and duration of each trip is illustrated in Table 1. We mannually count the number of passengers getting on/off each bus station and record the numbers in a customized Android app. The Android app is used for recording the number of passengers only and does not serve any other purposes. The ground truth data is then synchronized with the data collect by the sniffing system based on time and GPS location. The date, time and calculated correlation are summarized in Table 1.

While the estimated passenger numbers are strongly correlated with actual passenger numbers (with average of 0.76) the correlation can be lower than 0.62 in some cases. We further analyze this particular case by looking into the actual passenger riding patterns. We summarize the calculated passenger numbers and the ground truth passenger numbers in Fig. 6. Each point in Fig. 6 refers to the number of passengers at each bus station. The difference between calculated and ground truth is the estimation error of our passenger counting system. It is shown that the low correlation is due to some passenger burst, probably caused by students finish one class together and with phone turned off. This present little effects on long term statistical analysis since the burst is short and the group of students get off the bus after only few bus stations.

4.2 Bus Stop Statistics

The strong correlation between estimated passenger numbers indicates our method is sufficient for statistical analysis. For example, our method show that passenger riding is periodic during weekdays. We summarize the average number of

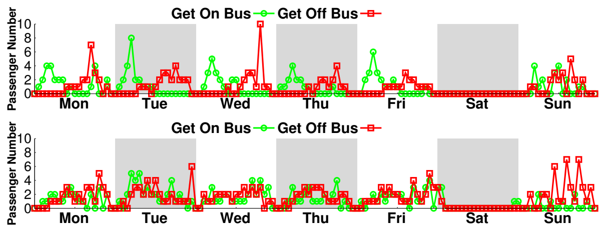

Figure 7: Bus stop in the residential area (top) and the main campus (bottom).

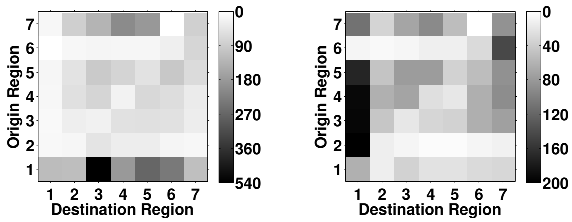

Figure 8: Passenger statistical riding patterns at morning hours (7am-9am), and evening hours(17pm-19pm)

passengers getting on and off two specific bus stations during each hour of one week in Fig. 7. The top one shows the passenger riding patterns in a residential area and the bottom one is in main campus. In residential areas, people are going out for work in the morning and going back home in late afternoon, there are obvious peak in those hours. In main campus, students and staffs are coming for work in morning hours and going back home in late afternoon. Further, undergraduate students live on campus. They travel between dormitories and teaching buildings for different classes during the day, so there are peaks in the number of passengers getting on and off the bus.

4.3 Transit Scheduling Analytics

Transit operators need to make scheduling decisions based on passenger volume and transit occupancy. By analyzing the origin-destination matrix, we can evaluate the transit efficiency and provide suggestions if the transit operators consider to adjust schedules.

We summarize the passenger region-to-region movements during morning hours and evening hours in Fig. 8. Each box in the color map is the number of passengers travel from (seven) different regions to that region. The darker the color, the more the passengers. As can be seen from the figures, nearly half of the passengers travel from region 1 are going to region 1-3 in morning hours(Fig. 8 left). This observation indicates that the bus route can be separated into two segments, while some buses can travel between region 1 and 3 and the rest follow the old schedule but less stop frequency. This can reduce the waiting time of the passengers want to go to region 3 due to lower duty cycle (the route is much shorter) while the rest passengers can have better riding experience due to less travel time. Meanwhile, the cost of the transit operators are reduced as well due to the improved efficiency and less frequent stops. In the evening rush hours(Fig. 8 right), most of the passengers get on the bus from different regions and are riding back to region 1, which means passengers are going back home. This indicates the origin schedule during evening hours is reasonable and efficient.

5. RELATED WORK

5.1 Passenger Counting

Transit operators are required to submit passenger statistics to national transit database [4], so they collect passenger numbers either by manual counting or expensive sensor systems. [7] uses video processing to count the number of passengers getting on/off each bus station. [11] uses passive, non-radiating infra-red technology to detect and count people moving through a door or gate. These system can detect number of passengers are passing a door, but they require expensive hardware and are not able to track individual passengers that are riding between each pair of stations. Meanwhile, the bus passengers are required to tapping IC card when get on and get off the bus in some Asian cities [2, 20]. These system does not count the passengers who are paying by cash. More importantly, tapping the key card when get off the bus may cause extra delays and queues at each bus station. Trellis does not require passenger operations.

5.2 Human Mobility Study

Understanding human mobility [6] enables many applications such as traffic engineering and urban planning. [19] infers human mobility based on multiple data resources, e.g., cellphone and transit data, to avoid baised judgement by single data resources. [12] claims that human trajectories show a high degree of regularity by tracking smartphone locations. [10] infers human mobility by using taxicab location traces. Our work falls in the same category and proposes new applications by performing passenger tracking. However, we propose novel way to conduct public transit analytics by deploying Wi-Fi sniffers on city buses, which separate our work from existing ones.

5.3 Human Tracking by Wi-Fi

[9] uses one pair of fixed Wi-Fi devices to estimate the total of people walking in an area based on power measurements. [18] tracks human queue length by using received Wi-Fi signal features and analyzes the waiting time in the queue. However, it requires customers' smartphones connecting with APs and generating traffics. [16] estimates the trajectory of smartphone holders by using multiple monitors on the road. [17] use mobile phone sensors to estimate people's trajectory, which is fundamentally different from our approach that is using Wi-Fi sniffer to track bus passengers.

6. DISCUSSION

In this section, we discuss the limitations of our system and propose other potential applications.

6.1 Limitations

First, the accuracy of passenger tracking is limited by some unpredictable factors. For example, some passengers are not using smartphones or the Wi-Fi is turned off etc. In these cases, the sniffing system is not able to detect the presence of the passenger. Also, some passengers may use multiple smart devices, e.g., a tablet and a smartphone. In this case, the sniffing system may overestimate the number of passengers. Second, some Apple devices are using randomized MAC address that we are not able to identify. Since randomized MAC address, if triggered, is sent out sparsely in time domain, which makes little effects on our system. But if it actually happens, we may over count the number of Apple users (not our focus though) and may fail to identify the passenger. Our work focuses on providing statistical analysis on transit efficiency to assist public transit operators instead of tracking every single passengers. These limitations exposes challenges for our tasks, but do not affect the practicability of our system.

6.2 Other Applications

Although we focus on transit analytics in this paper, some other applications are possible given the rich data set and well designed abstraction. For example, our system can be used to predict the riding route of individual passenger. Some smartphone applications can use these information to schedule cellular traffic based on link qualities at different locations along the route. Some Wi-Fi related applications can also benefit from accurate predication of passenger's presence [15].

7. SUMMARY AND FUTURE WORK

Our work proposes a passive crowd-sourced approach to infer how passengers use transit systems. The system follows the popular paradigm of tracking mobile devices as identifiable by vehicle-mounted Wi-Fi observers. While our preliminary system demonstrates both feasibility and preliminary usefulness, numerous challenges remain. They include: (i) mechanisms to improve device identification accuracy in the vehicle context; (ii) identification of different analytics capabilities that such a system can provide efficiently; (iii) performing a more rigorous privacy analysis in such vehicular scenarios, even when MAC addresses are obfuscated; and (iv) evaluating other complementary techniques to achieve similar goals and how they can either complement or enhance our proposed system.

8. ACKNOWLEDGMENTS

We would like to acknowledge the anonymous reviewers and our shepherd Dr. Kaushik Veeraraghavan, whose comments helped bring the paper into its final form. This research project is supported in part by the US National Science Foundation through awards CNS-1405667, CNS-1345293, CHE-1230751, CNS-1343363, CNS-1555426 and CNS-1525586.

9. REFERENCES

[1] Apu platform. *http://www.pcengines.ch/apu.htm*.

[2] Ic card in hong kong transit. *http://www.octopus.com.hk/home/en/*.

[3] Infodev automatic passenger counting. *http://www.infodev.ca/vehicles/counting-passengers.html*.

[4] National transit database. *http://www.ntdprogram.gov/*.

[5] Route 80, madison, wisconsin. *https://www.cityofmadison.com/Metro/schedules/Route80*.

[6] F. Asgari, V. Gauthier, and M. Becker. A survey on human mobility and its applications. *arXiv preprint arXiv:1307.0814*, 2013.

[7] C.-H. Chen, T.-Y. Chen, D.-J. Wang, and T.-J. Chen. A cost-effective people-counter for a crowd of moving people based on two-stage segmentation. *Journal of Information Hiding and Multimedia Signal Processing*, 2012.

[8] Y.-c. Cheng, M. Afanasyev, P. Verkaik, P. Benkö, J. Chiang, A. C. Snoeren, S. Savage, and G. M. Voelker. Automating cross-layer diagnosis of enterprise wireless networks. In *In Proc. of ACM SIGCOMM*, 2007.

[9] S. Depatla, A. Muralidharan, and Y. Mostofi. Occupancy estimation using only wifi power measurements. *IEEE Journal on Selected Areas in Communications*, July 2015.

[10] R. Ganti, M. Srivatsa, A. Ranganathan, and J. Han. Inferring human mobility patterns from taxicab location traces. In *Proceedings of the 2013 ACM international joint conference on Pervasive and ubiquitous computing*. ACM, 2013.

[11] H. E. Gerland and K. Sutter. Automatic passenger counting (apc): Infra-red motion analyzer for accurate counts in stations and rail, light-rail and bus operations. *INIT GmbH Innovations in Transportation*.

[12] M. C. Gonzalez, C. A. Hidalgo, and A.-L. Barabasi. Understanding individual human mobility patterns. *Nature*, 2008.

[13] J. Hare, L. Hartung, and S. Banerjee. Beyond deployments and testbeds: experiences with public usage on vehicular wifi hotspots. In *Proceedings of the 10th international conference on Mobile systems, applications, and services*. ACM, 2012.

[14] T. Kimpel, J. Strathman, D. Griffin, S. Callas, and R. Gerhart. Automatic passenger counter evaluation: Implications for national transit database reporting. *Transportation Research Record: Journal of the Transportation Research Board*, 2003.

[15] J. Manweiler, N. Santhapuri, R. R. Choudhury, and S. Nelakuditi. Predicting length of stay at wifi hotspots. In *INFOCOM, 2013 Proceedings IEEE*. IEEE, 2013.

[16] A. Musa and J. Eriksson. Tracking unmodified smartphones using wi-fi monitors. In *Proceedings of the 10th ACM conference on embedded network sensor systems*. ACM, 2012.

[17] A. Thiagarajan, L. Ravindranath, K. LaCurts, S. Madden, H. Balakrishnan, S. Toledo, and J. Eriksson. Vtrack: accurate, energy-aware road traffic delay estimation using mobile phones. In *Sensys*. ACM, 2009.

[18] Y. Wang, J. Yang, Y. Chen, H. Liu, M. Gruteser, and R. P. Martin. Tracking human queues using single-point signal monitoring. In *Proceedings of the 12th annual international conference on Mobile systems, applications, and services*. ACM, 2014.

[19] D. Zhang, J. Huang, Y. Li, F. Zhang, C. Xu, and T. He. Exploring human mobility with multi-source data at extremely large metropolitan scales. In *Proceedings of the 20th annual international conference on Mobile computing and networking*. ACM, 2014.

[20] P. Zhou, Y. Zheng, and M. Li. How long to wait?: predicting bus arrival time with mobile phone based participatory sensing. In *Mobisys*. ACM, 2012.