A Crowd-Assisted Real-time Public Transport Information Service: No More Endless Wait

Winwin Loon Haw Lim, Joe Tuck Wai Lum, Ian Jer Wei Yeo, Sye Loong Keoh
School of Computing Science, University of Glasgow
Email: {win2_90, ianyeo_1}@hotmail.com, lumtwj@gmail.com, SyeLoong.Keoh@glasgow.ac.uk

***Abstract*—Many passengers have expressed frustration in waiting for public bus endlessly without knowing the estimated arrival time. In many developing countries, requiring bus operators to invest in the installation of a GPS unit on every bus in order to track the bus location and subsequently predicting the bus arrival time can be costly. This paper proposes passenger-assisted sharing of bus location to provide an estimation of bus arrival time. Our scheme aims to exploit the availability and capability of passenger mobile phones to share location information of the travelling buses in order to collect transportation data, at the same time provide an estimation of bus arrival time to the general public. A mobile app is developed to periodically report bus location to the cloud service, and it can detect location spoofing by malicious users. The preliminary results of the field tests suggest that the proposed system is viable and the predicated ETA falls within three minutes of the bus actual arrival time.**

I. INTRODUCTION

Many big cities in developing countries are suffering from serious traffic congestion mainly because the public transportation systems are not efficient. They are often filled with crowd every day, and the current transportation infrastructure is not adequately supported by the government to deal with the overwhelming commuters who need to get to their destination on time. Public buses are the only affordable means of transportation in most cases as there is no rapid transit systems available to ferry commuters efficiently.

However, the public bus network in the developing cities are usually unreliable, and the service frequency is unpredictable most of the time. Many passengers think that it is a complete waste of time to wait aimlessly at bus stops. They do not have any idea when the bus will arrive, and whether it will arrive. In fact, the residents prefer to own motorcars in order to have the convenience of getting around the city. As a result, this causes massive traffic congestion and hence paralysing the road traffic at peak hours.

Therefore, in order to improve the day-to-day commute experience for the residents, the first priority is to improve the public transportation network. Dedicated bus lanes can be implemented in cities, so that commuters who choose to travel on public bus network do not have to suffer from traffic congestion. Additionally, reliability of public buses affecting the passsenger waiting time, estimated time of arrival (ETA), travelling time should be provided accurately [2], so that the passengers have a better control of their journey. Developed cities such as London, Singapore, and Taipei already have information panels mounted at bus stations and bus stops to inform passengers about the ETA of public bus services. However, this requires a massive amount of investment from the government to install GPS tracking devices on all buses and to pay for the heafty amount of monthly mobile data cost.

In this paper, we propose a crowd-assisted sharing of bus location [5] to provide an estimation of bus arrival time. Many research studies and applications such as *Tiramisu* [10], *Moovit*, and *Waze* utilize crowd contributed GPS traces to enhance the commuters' travel experience. A survey conducted in [9] shows that majority of passengers have indicated that they are willing to share their location information while travelling on public buses in order to aid a cloud-based service to provide an estimated time of arrival. Consequently, we aim to exploit the availability and capability of passenger mobile phones to share location information of the travelling buses in order to collect transportation data, at the same time provide an estimation of bus arrival time. This is beneficial to the residents, at the same time, it does not require the public bus operators to invest in the hardware and software infrastructure.

The paper is organized as follows: Section II introduces the smart city initiative in Malaysia, Section III presents the proposed crowd-assisted public bus ETA service. In Section IV, we describe the implementation of the system and present preliminary results. Section V presents related work, and finally we conclude the paper with future work in Section VI.

II. SMART CITY INITIATIVE IN MALAYSIA

Iskandar Malaysia (IM) is the main southern development corridor in Johor, and it is very closely linked to Singapore. By capitalizing on the existing synergies with Singapore, IM aims to create a complementary economic hub to Singapore.In order to create a sustainable and a highly livable city, there are several development plans to transform the region into the southern financial district in Malaysia, a medical hub, a resort for international tourism and an industrial logistic cluster. It is expected that residents in the region will enjoy quality of life, with liveable communities that promote social wellbeing and environmental friendly services.

The main mode of transportation is land, connecting the region to Singapore via the Malaysia-Singapore Second Link, and to Kuala Lumpur via the North-South Expressway. In terms of land public transports, public bus services, taxi and railway services are provided. Currently, there is a plan to build a Rapid Transit System (RTS) between Johor Bahru and Singapore, and several routes for city travel using Bus Rapid Transit (BRT). Nonetheless, travelling within the Iskandar region is still very much dependent on public buses. Although there are more buses that are being put into operation and new routes are being introduced to address the residents' needs,

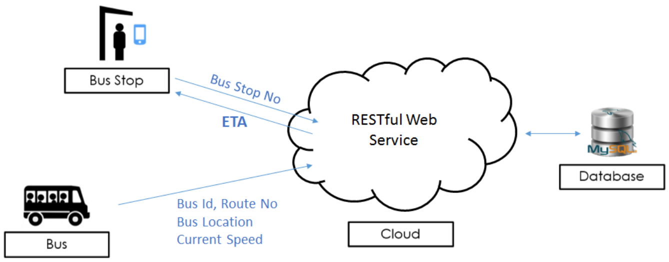


Fig. 1. Architecture for crowd-assisted estimation of bus arrival time

the software infrastructure is still lacking in which a real-time public transport information service is not currently available. For example, a passenger waiting at the bus stop or bus station would not know the ETA of the bus services.

III. Crowd-Assisted Public Bus ETA Service

This section presents the proposed crowd-assisted sharing of bus location information in order to facilitate the estimation of bus arrival time, using one of the routes in Iskandar Malaysia region as a use case. The proposed system relies on the passengers to voluntarily participate in the reporting of their location information to a cloud service while they commute on public buses. This means that passengers possessing a smart phone must be willing to use their mobile data for location reporting. The ETA of bus services are then made available through the cloud service, thus allowing prospective passengers to plan their journey with ease.

A. System Requirements and Challenges

The feasibility of such a crowd-assisted system depends on the following requirements and challenges to be fulfilled:

- Passengers must have a GPS-enabled smart phone that can report bus location periodically in real-time. Additionally, mobile data is needed to transmit the location information to the cloud for processing.
- The location reporting app on the passenger smart phones should consume minimal battery, low data usage and CPU cycles.
- The location reporting app should be run as a background process and it does not interfere with the passenger's interaction with his mobile phone while travelling on the bus.
- The proposed system should be low-cost in that the deployment cost for the bus operators to support this service is low as compared to the cost of installing a GPS tracking unit on all public buses.
- The location information of the bus should be reported in near real-time in order to accurately estimate the bus arrival time for all stops on the route. The ETA should be calculated taking into account the current traffic condition.
- As location data are crowd-sourced, this means that spoofing of bus location is possible, and this should be prevented (and/or detected) in order not to disrupt the ETA of bus services.
- An incentive scheme to reward passengers who participate in the reporting of bus location.

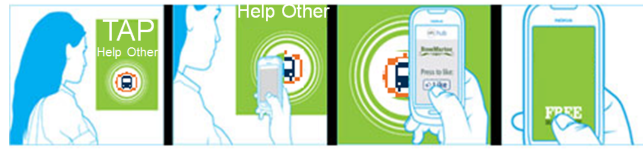


Fig. 2. Using NFC to trigger location reporting on mobile app

B. System Architecture

Figure 1 illustrates the architecture of the proposed crowd-assisted estimation of bus arrival time. It consists of three main components, namely (1) A mobile app running as a background process to report bus location periodically, (2) A mobile app allowing the passengers to check for ETA, (3) A RESTful web service that processes the location data and calculate the ETA based on the current traffic condition and in some cases based on the historical average.

1) Location Reporting: In order to encourage the passengers to use the proposed crowd-assisted system, Near-Field-Communication (NFC) is used to distribute ⟨Bus Service No, Route Info, and Bus Instance ID⟩ to the passenger's smart phone while they are onboard the bus. When the passenger taps his smart phone on the NFC sticker located on the bus, it prompts the passenger to select his destination and subsequently triggers the mobile app to start reporting bus location periodically. As shown in Figure 2, upon tapping the smart phone on the NFC sticker, discount coupons can be given to the passengers as rewards for his participation.

The location reporting is executed every 30 seconds to provide near real-time location updates of the bus service. This is required to increase the accuracy of the bus service ETA, and in some cases, based on the reported location, traffic congestion can be detected when the differences in the reported location in a specified period of time is insignificant.

In the proposed system, with only a single passenger reporting the bus location information, it will enable the bus arrival time to be estimated. As passengers may alight before the bus arrives at its final destination, having multiple passengers reporting the bus location information provides redundancy to enhance the reliability of the proposed system.

The attractiveness of this system is that even though there is no passenger onboard who is reporting bus location information in real-time, if there was a passenger who had reported bus location in the past after the start of the bus journey, the system is able to use historical average to provide a rough estimate of bus arrival time for its remaining route.

2) Calculation of ETA: The bus service route must be defined prior to the deployment of the system. Figure 3 shows an example of bus service between two cities (Kulai-Johor Bahru) in the Iskandar Malaysia region. Currently, the bus service No. 7 (S&S International), 7B (Causeway Link) and BET1 (Trans-Iskandar) are serving this route in both

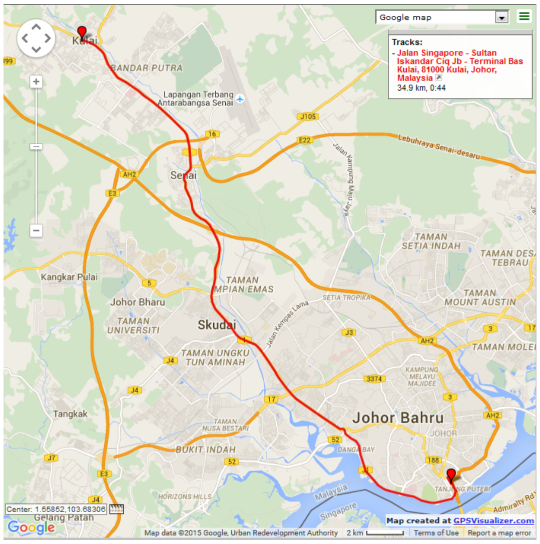


Fig. 3. An example route in Iskandar Malaysia (Kulai-Johor Bahru, MY)

directions. This route information is converted into a polyline, i.e., a set of GPS coordinates (latitude, longitude), so that the distance between a bus stop on the route with the last reported bus location can be calculated, and subsequently deriving the ETA.

In addition, all the bus stops along the route are identified and then mapped to the closest point on the polyline for the route. The distance between consecutive bus stops are calculated based on the route and then stored in the database, so that the ETA can be calculated efficiently.

An ETA computation process is executed every 30 seconds in order to update the time of arrival of all bus services at every bus stop, based on the reported bus location collected from the passengers. As shown in Figure 4, for each bus service route, the ETA computation process determines the latest bus location, t_n and then calculates the distance between the current bus location and the next bus stop, d_0. Using Haversine Formula, the distance between two GPS points (obtained from the route polyline) can be accurately calculated. In order to accurately estimate the bus arrival time at the next few stops, the average speed of the bus in the past five minutes, v is used. As shown in the figure, the distance between two points in the past five minutes are determined, and using the total travelling time between the two points (t_n - t_{n-m}), the system is able to determine the average travelling speed. As a result, the ETA for the next bus stop can be calculated. In the case that there was no reported bus location in the past five minutes, the current speed of the bus is used.

By knowing the distance to the next bus stop, d_0, and the average bus travelling speed, the ETA for the next bus stop can be easily calculated as follows:

$$ETA = t_n + \frac{d_0}{v} \tag{1}$$

As the distance between bus stops on the route are pre-calculated, i.e., $d_1, d_2, ...$ in the example, the ETA for the remaining bus stops can be calculated based on v. It is observed that the velocity of the bus varies throughout the journey, the

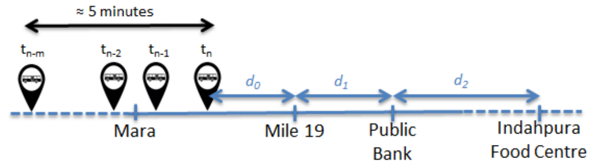


Fig. 4. Calculation of ETA for the Kulai-Johor Bahru, MY service route

system only adopted a mechanism to only calculate the ETA of the next three bus stops using v, while the ETA for the remaining bus stops is then calculated using historical average journey time (c.f., Section III-B3). Based on the location data collected from the passengers, it is possible to pre-compute the average journey time between two bus stops, and this data can be used to better estimate the ETA beyond the next three stops from the current bus location.

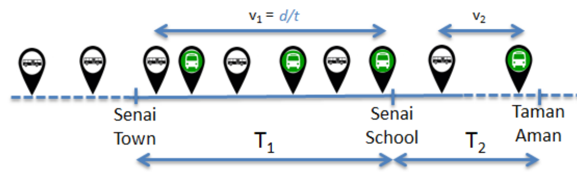


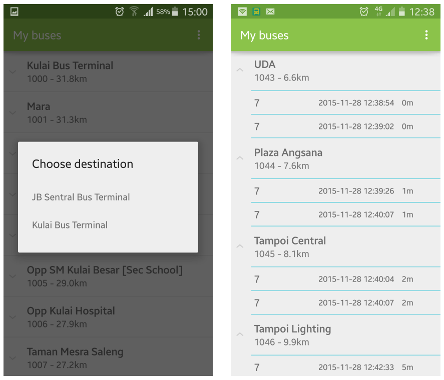
Fig. 5. Calculating the historical average journey time

3) Historical Average Journey Time: A batch processing job is executed once a day to analyze the bus location data reported by passengers, and update the historical average journey time between bus stops for all routes. Figure 5 shows an example of location data collected from two passengers. In order to calculate the average travelling time, the algorithm iterates through the location data and find the closest point to the two consecutive bus stops, where point p must be between stop 1 and 2, i.e., ($P_{Senai\_Town} \leq p \leq P_{Senai\_School}$).

Once the two closest points have been identified, the velocity, v of the bus can be calculated. Using the pre-calculated distance between bus stops, the average travelling time between two bus stops, T, can be derived. These historical average journey time is organized according to the day and time of the journey as the traffic flow can be studied, analyzed and learned to better estimate the bus arrival time.

If we assume that there is no delay in the reporting of bus location by the passengers' smart phones, it is fine to just select two points between the bus stops to compute the average travelling time. However, there could be a case where the cellular network is unstable in rural areas, and hence causing the reported location data to be severely delayed. This will impact the accuracy of the ETA computation due to the data inconsistency between different passengers. Therefore, it is preferable to choose two points from the same passenger when computing the average travelling time.

4) Mobile App: When the passenger taps on the NFC sticker, the location reporting mobile app is automatically triggerred and it prompts the passenger to choose the destination of the bus. This is illustrated in Figure 6(a). In most cases, a bus service number runs in both directions, and it is important that the reported location contains the direction of the bus service in order to accurately compute the ETA for the route. The



(a) Using NFC to trigger location reporting (b) Information on ETA

Fig. 6. Mobile App for reporting of Bus Location in real-time and checking bus service's ETA.

Fig. 7. Validation of location data

mobile app also checks whether the smart phone's GPS sensor is enabled, and that there is Internet connectivity to transmit the bus location data to the cloud service. Subsequently, the mobile app is minimized and run as a background process. When it detects that the passenger has alighted or reached the bus terminal, it automatically stops transmitting location information to the cloud service.

As illustrated in Figure 6(b), passengers can also use the mobile app to check ETA of buses at bus stops. By tapping on the bus stop name, a list of bus services and their ETA will be retrieved from the Cloud service. In the future, other functions such as bus route information, bus stop search, journey planning can be incorporated into the mobile app.

C. Validation of Location Data and Security

This section presents and discusses some challenges in developing the proposed crowd-assisted real-time public transport information service.

1) Validation of GPS coordinates: As illustrated in Figure 7, location reported by passengers may be inaccurate due to the poor mobile cellular network connection. It is thus important to validate the location data, in particular if the received location is behind the last seen bus location, it should be discarded. The accuracy of the GPS sensor on the smart phone also depends on the availability of the GPS signals and could be affected due to poor weather condition.

A further constraint is also imposed on the acceptance of the location data such that the reported location must be within 60-metre radius of the bus route. The example in Figure 7 shows that both reported location are discarded as the first is severely delayed and it is behind the current bus location, while the second is outside the 60-metre radius of the route.

2) Spoofing of Bus Location: Using location spoofing app, e.g., *Fake Location*, any users can spoof the GPS location on their smart phones to falsify the location of public buses in order to disrupt the ETA service. Android OS allows developers to spoof locations by enabling *Android mock locations* for the purpose of conducting location-based application testing. By ensuring that *mock locations* is not turned on, it will definitely increase the difficulty in conducting location spoofing attack. Nonetheless, there is an app called *MockMockLocations* that is available on the XDA Developer site that attempts to prevent apps from detecting "*Allow mock locations*" in developer settings. By running this app, it has the system responds as if the setting is off.

In order to counter-against this security issue, we have developed an additional check on the GPS location on the passenger's smart phone by comparing the GPS location with the location derived from triangulation of cellular towers [6]. The mobile app ensures that both locations are close to each other before it reports the location data to the cloud service. In this way, it is possible to defend against attackers from randomly spoofing bus locations, unless the attackers make an effort to drive along the bus route to perform location spoofing.

3) Encryption of Location Data: The current location reporting mobile app does not encrypt the location data when transmitting it to the cloud service. Without data encryption, anyone can easily spoof location using any computing platform at anywhere and anytime. Worse, attackers can transmit spoofed location even without requiring a GPS sensor, as location data can be manually created and then transmited to the cloud. There is a potential Denial-of-Service (DoS) attack on the system.

Hence, it is advocated that the transmission of bus location should only be done via the mobile app, and that the location data is encrypted with a secret-key shared with the cloud service. A secret-key can be negotiated between the mobile app and the cloud service after the user has successfully installed the mobile app on his smart phone. Although this does not thwart the location spoofing attack entirely, it tries to ensure the following:

- Location reporting can only be done via the mobile app installed on the phone.
- Bus location is obtained from the GPS sensor, ensuring that *Mock Locations* is not enabled, and that the GPS location is equivalent to the location obtained from triangulation of cellular towers.
- Location data is encrypted using an authorized secret-key that is unique per mobile app.

As a result, this would make location spoofing a little bit harder and inconvenient for attackers, as they would need to install the mobile app and then drive along the route in order to spoof the bus locations.

Fig. 8. Data simulator

IV. Implementation and Evaluation

The cloud service was implemented using REST as the interaction with the application are simple requests, such as POST and GET information. It uses a stateless communication protocol that treats every request as an independent transaction. The cloud service was developed using PHP and it is currently hosted on OpenShift by Red Hat.

As for the mobile app, currently only Android is supported as it is the OS of many smart phone manufacturers. iPhone app has not been developed because the current iPhone's NFC chip can only be used with Apple Pay system, and it is not able to read passive NFC tags. Therefore, requiring the passengers to manually input the bus service number and route information when starting the location reporting app is cumbersome.

A. System Testing

1) Field Test: In order to test the accuracy of the ETA using the proposed crowd-assisted mechanism, the developed mobile app was installed on a Samsung S4, Samsung S3, Samsung Note 3 and Google Nexus 5, and carried by testers onboard the bus to collect location data in real-time. Testers boarded the service route No. 7, 7B or BET1 (Johor Bahru-Kulai and vice-versa) at the respective departure point. When the bus departed, the mobile app was turned on to start reporting passenger location to the cloud service. In most cases, two mobile apps were turned on to simulate two passengers onboard throughout the whole journey. The tests were conducted on weekday at peak, non-peak hours and weekend as shown in Table I:

TABLE I. System tests conducted for (Kulai-Johor Bahru) and (Johor-Bahru-Kulai) routes

Route	Kulai-Johor Bahru	Johor Bahru-Kulai
Peak	MON 0800	FRI 1800
Non-Peak	TUE 1800	TUE 1300
Weekend	SUN 1400	SAT 1000

While travelling on the bus, a stop-watch was used to record the actual arrival time at each bus stop along the route. The ETA computed by the cloud service were logged and stored in the database, and displayed on the mobile app. The accuracy of the ETA will be presented and discussed in Section IV-B.

2) Simulation: A Java simulator was developed to simulate the location updates of a travelling bus using the data collected from the field tests in order to refine, improve and validate the ETA calculation algorithm. The simulator greatly helps in re-validating the test cases and hence making the system testing and evaluation more effective.

On the other hand, a data-simulator as illustrated in Figure 8 was used to evaluate the processing of location updates, ensuring that location data that is out of the service route is discarded. In addition, for cases where severely delayed location data is rejected as well.

B. Preliminary Results

Figure 9 shows that the actual bus arrival time falls within the range of predicted ETA computed based on the speed of the bus. As observed, the range of ETA becomes larger towards the end of the route, and this is reasonable as the system is only providing an estimate based on the current traffic condition. At the begining of the route, due to the lack of information on the bus travelling speed, the ETA computed for the first five bus stops were not very accurate. However, this gradually improved as the average speed of the bus in the past five minutes can be used to correctly determine the ETA for the next few stops that were within the next 5-10 minutes. It is observed that the ETA at a particular bus stop was accurate up to +/- 3 minutes. However, in most cases, the bus actually arrived $\approx$ 1 min before the ETA.

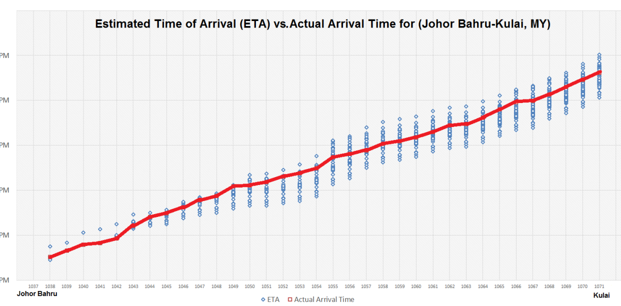


Fig. 9. Comparison between ETA computed using average speed vs. actual arrival time

We also experimented with the computation of ETA purely based on the historical average. As shown in Figure 10, the inaccuracy in the first few bus stops was due to the difficulty in detecting that the bus had departed from the terminal. In the middle section of the graph, it appears that the speed of the travelling bus was faster as compared to the historical average, as this stretch of the route is a motorway, and thus resulting in the early arrival of the bus at the bus stops. As the bus was approaching the final destination, there was a little bit of traffic and hence the actual arrival time was closer to the higher range of the ETA.

The size of the Android app is 20.49 MB. In terms of mobile data utilization, for a 50-minute journey, each passenger had to transmit $\approx$ 120.98 KB of location information to the cloud service. We plan to further improve and optimize the mobile app for better performance and power saving.

V. Related Work

RapidKL in Malaysia has installed a Passenger Information System (PIS) [4], [7] to display departure and ETA of buses

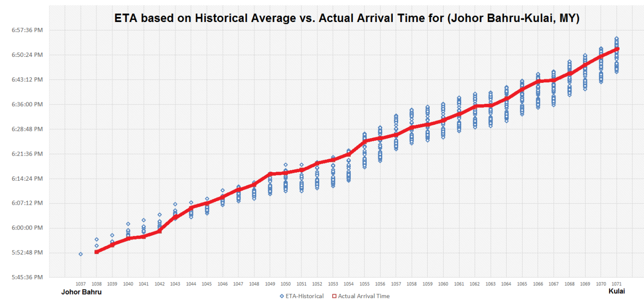


Fig. 10. Comparison between ETA computed using historical average vs. actual arrival time

at stations and bus stops in Klang Valley and Kuala Lumpur. It provides real-time tracking of bus location by installing a GPS unit onboard the bus. However, the prediction is not always accurate due to the traffic congestion and the deviation of departure from the original service schedule [4]. Currently, the data collected from the GPS is also used for fleet tracking and they are not available to the public. Therefore, there is no mobile app that can provide real-time bus arrival information in Malaysia.

In Singapore, the Land Transport Authority (LTA) has improved the accuracy of bus arrival time estimates in July 2015, with 95% of estimates fall within three minutes of actual bus arrival time. Everyday, 50 million data points are collected in this island nation, and these location data obtained from a dedicated GPS unit are transmitted to the cloud to predict bus arrival time based on real-time location data, bus schedules as well as historical travel times. There are multiple mobile apps, i.e., MyTransport, SG NextBus that enable passengers to get quick access to bus arrival information. It is also possible to provide an indication of load of each bus, by keeping track of the number of passengers onboard as each passenger has to tap their EZ-Link card when paying the bus fare.

OneBusAway [3] is a system that uses the published GTFS real-time data feed to provide bus arrival time, information about disruption of services, delays, and location of the buses. It is available for use through a variety of interfaces, including mobile app, webpage and SMS. OneBusAway is designed to be customizable and expandable, and it is now available for use in many big cities in the US including New York City, Atlanta and Puget Sound [1].

[9] proposed a bus arrival prediction system based on sensors on mobile phone and it does not require any GPS traces. Instead, it relies on the cell-tower signals, accelerometers on the phone to determine that the passenger is travelling on the bus, subsequently using the Cell Tower ID reported by the mobile phone to identify the bus route, and finally utiliting both traffic data and history knowledge to predict the bus arrival time. Similarly, EasyTracker [8], [2] is a co-operative transit tracking using the passengers' mobile data to detect that they are travelling on a bus and infer the route of the bus service, as well as to predict the ETA.

VI. Conclusions and Future Work

We have proposed a new way of providing estimation of bus arrival time through crowd-participatory *sensing*, using Iskandar Malaysia as one of the deployment targets. Instead of relying on the bus operators to upgrade its infrastructure, we have provided a low-cost deloyment mechanism to collect, analyze, predict and disseminate transport information to the general public. The data and ETA information can later be integrated with other service providers such as Google Maps and *Moovit* to benefit more commuters.

Analysis and prediction of ETA can be improved further to provide better accuracy. At the same time, big data analytic tools such as Hadoop can be integrated to mine the data collected from passengers.

Finally, the security of the system needs to be analyzed thoroughly and further refined to prevent location spoofing and DoS attacks. An incentive scheme is required to attract more users to participate in the crowd-assisted public bus ETA service, so that discount coupons and rewards can be securely linked to the participating passengers onboard the bus.

Acknowledgment

We gratefully acknowledge the financial support from the School of Computing Science, University of Glasgow. We also thank Darren Yip, Andy Goh, Heimun Sum and Everlyse Tay for implementing the initial prototype.

References

[1] S. J. Barbeau, A. Borning, and K. Watkins. One Bus Away Multi Region Rapidly Expanding Mobile Transit Apps to New Cities. *Journal of Public Transportation*, 2014.

[2] James Biagioni, Tomas Gerlich, Timothy Merrifield, and Jakob Eriksson. EasyTracker: Automatic Transit Tracking, Mapping, and Arrival Time Prediction Using Smartphones. In *Proceedings of the 9th ACM Conference on Embedded Networked Sensor Systems*, 2011.

[3] B. Ferris, K. Watkins, and A. Borning. OneBusAway: A Transit Traveller Information System. In *Proceedings of Mobicase, San Diego, CA, USA*, Oct 2009.

[4] Hazril Bin Md. Isa, Sabri Ismail, and Noor Aqilah Ahmad Tajedi. Passenger Information Systems for Public Transport in Klang Valley, An Overview. In *Malaysian Universities Transport Research Forum Conference, Bangi, Malaysia*, Dec 2013.

[5] Sian Lun Lau and S.M. Sabri Ismail. Towards a Real-Time Public Transport Data Framework using Crowd-sourced Passenger Contributed Data. In *Proceedings of the First International Workshop on Mobile and Context Aware Services (MOCS2015), Boston, MA, USA*, Sept 2015.

[6] Jun Liang Lo. Authenticated Geo-tagged Location Information, University of Glasgow Final Year Project Report, 2015.

[7] Noor Aqilah Ahmad Tajedi, Sabri Ismail, and Hazril Bin Md. Isa. Getting Journey Planner Information into the Mobile Application: Role of GTFS in Klang Valley Area. In *Malaysian Universities Transport Research Forum Conference, Bangi, Malaysia*, Dec 2013.

[8] Arvind Thiagarajan, James Biagioni, Tomas Gerlich, and Jakob Eriksson. Cooperative Transit Tracking Using Smart-phones. In *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*. ACM, 2010.

[9] Pengfei Zhou, Yuanqing Zheng, and Mo Li. How Long to Wait?: Predicting Bus Arrival Time with Mobile Phone Based Participatory Sensing. In *Proceedings of the 10th International Conference on Mobile Systems, Applications, and Services (MobiSys)*. ACM, 2012.

[10] John Zimmerman, Anthony Tomasic, Charles Garrod, Daisy Yoo, Chaya Hiruncharoenvate, Rafae Aziz, Nikhil Ravi Thiruvengadam, Yun Huang, and Aaron Steinfeld. Field Trial of Tiramisu: Crowd-sourcing Bus Arrival Times to Spur Co-design. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2011.