Real-time Traffic Tracking Sensor Web System

Prakash Chourasia
Georgia State University
Department of Computer Science
Atlanta Georgia, USA
E: pchourasia1@student.gsu.edu

ABSTRACT

This article describes how the mobile internet is changing the face of the transportation cyber physical system at a rapid pace. In the last five years, cellular phone technology has leapfrogged several attempts to construct dedicated infrastructure systems to monitor traffic. Today, GPS equipped smartphones are progressively morphing into an ubiquitous traffic monitoring system, with the potential to provide traffic information in real time for the entire transportation network.

In this paper, we will develop a real-time traffic tracking sensor web system. Such a system can leverage smart phone platform and its sensors (such as GPS, accelerometer and compass) to track real-time traffic situations and report to a sensor web database. The crowd-sourced data can then be visualized in Google maps (like the traffic map overlay on Google map).

Categories and Subject Descriptors

C.2.4 [Computer-Communication Networks]: Distributed Systems—Client/server,Distributed applications.

General Terms

Design, Experimentation, sGPS, Accelerometer, latitude, longitude, API keys.

Keywords

Mobile Platform, Mobile Sandbox, Participatory Sensing, Opportunistic Sensing, Smart Phones

1. INTRODUCTION

Mobile phones are proliferating with more than 4 billion phones in use worldwide [1]. Programmable smartphones constitute a significant and growing fraction of these phones. For instance, 172 million of the 1.2 billion phones sold worldwide in 2009 were smartphones [10]. While smartphones today are used largely in isolation, operating as individual units serving their respective users (e.g., enabling users to check their email), there is a nascent interest in community applications, which leverage the resources of a potentially large and distributed set of mobile smartphones. One such class of community applications that has received much research attention recently is community sensing [3, 4, 7, 8, 11]. Community sensing using mobile smartphones is motivated by the observation that such phones include not only computing and communication capabilities but also a range of sensing capabilities, such as provided by the microphone, camera, GPS, and accelerometer, among other sensors. The idea, then, is to orchestrate the computing, communication, and sensing capabilities of a population of mobile phones, which happen to be at the right place at the right time, to enable large-scale sensing purely through software running on this existing hardware base. A community sensing application could either be participatory, involving explicit user action (e.g., taking photographs), or opportunistic, operating without user involvement (e.g., recording a GPS trace) [5, 9].

Mobile devices, such as cellular phones and music players, have recently begun to incorporate diverse and powerful sensors. These sensors include GPS sensors, audio sensors (i.e., microphones), image sensors (i.e., cameras), light sensors, temperature sensors, direction sensors (i.e., compasses) and acceleration sensors (i.e., accelerometers). Because of the small size of these “smart” mobile devices, their substantial computing power, their ability to send and receive data, and their nearly ubiquitous use in our society, these devices open up exciting new areas for data mining research and data mining applications. The goal of our WISDM (Wireless Sensor Data Mining) project [6] is to explore the research issues related to mining sensor data from these powerful mobile devices and to build useful applications. In this paper we explore the use of one of these sensors, the accelerometer, in order to identify the activity that a user is performing—a task we refer to as activity recognition. We have chosen Android-based cell phones as the platform for our WISDM project because the Android operating system is free, open-source, easy to program, and expected to become a dominant entry in the cell phone marketplace (this is clearly happening).

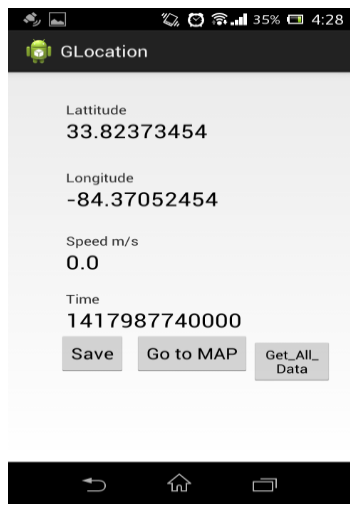
Our project currently employs several types of Android phones, including the Nexus One, HTC Desire, Sony Xperia and Motorola Backflip. These phones utilize different cellular carriers, although this is irrelevant for our purposes since all of the phones can send data over the Internet to our server using a standard interface. However, much of the data in this work was collected directly from files stored on the phones. All of these Android phones, as well as virtually all new smart phones and smart music players, including the iPhone and iPod Touch [2], contain GPS.

Our work makes several contributions. One contribution is the data that we have collected and continue to collect, which we plan to make public in the future. This data can serve as a resource to other researchers, since we were unable to find such publically available data ourselves. We also demonstrate how GPS data can be transformed such that can be used by application for real time trafficking. We believe that our work will help bring attention to the opportunities available for mining wireless sensor data and will stimulate additional work in this area. The remainder of this paper is structured as follows. Section 2 describes architecture and the process for data collection, data preprocessing, and data transformation. Storing and retrieving data. Section 3 describes our experiments and results. Related work is described in Section 4 and Section 5 summarizes our conclusions and discusses areas for future research.

2. Architecture & Data Collection

2.1 Data Collection

The data collection was controlled by an application we created that executed on the phone. This application, through a simple graphical user interface, permitted us to record the user’s name, latitude, longitude, speed and time stamp at which data collection performed and saved in SqLite. The application permitted us to control what sensor data (e.g., GPS) was collected and how frequently it was collected. In all cases we collected the GPS data every 10s, so we had 6 samples per minute.



2.2 Data Transformation

A back end server will aggregate data from a large number of mobile devices and push the data to online database for data assimilation, which will combine the cell phone data with other information to produce the best estimate of the current state of traffic. The map data server will provide the map data which is required for the network based traffic flow models.

Phone tracks car location with GPS and calculate speed (note that speed calculation is tricky)

- If GPS signal is blocked, then discard data.

o Phone periodically uploads location and speed records with timestamp to online database

 If there is no data connection, then the data shall be buffered locally.

o Online database server manages merging similar records, removing outdated records and compute route speed.

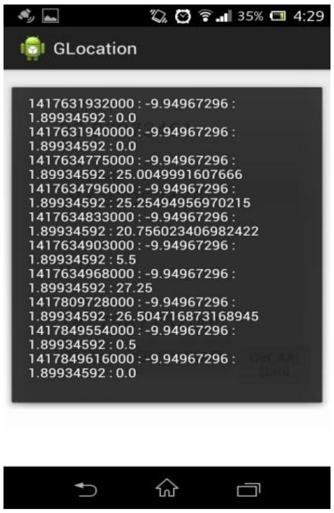
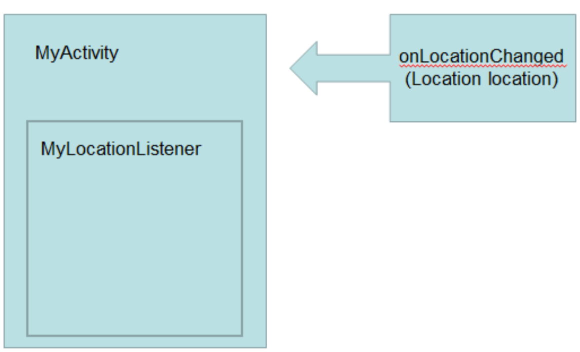


Table 1. Table in SqlLite

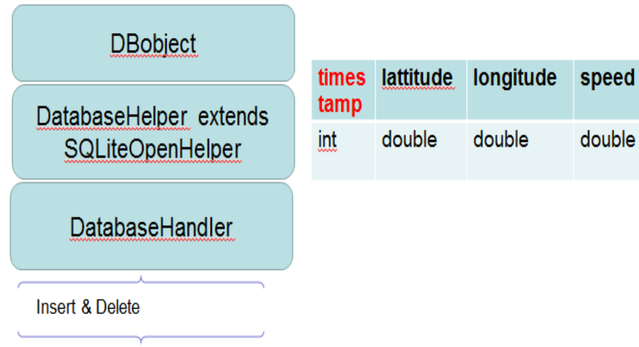
timestamp	lattitude	longitude	speed
Int	double	double	double

2.3 Architecture

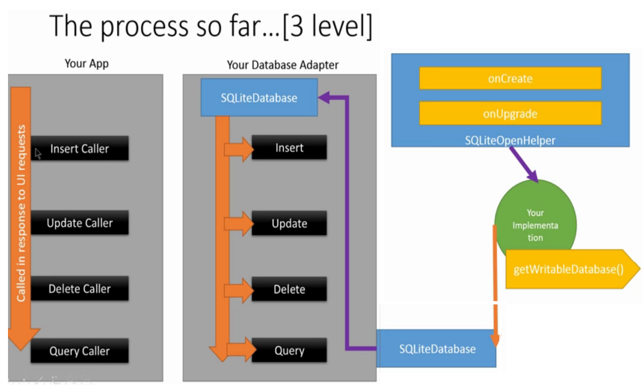
There is main activity page which implements locationlistner interface, where we used onlocationchanged function to update GPS data in every 10 sec.



In Database handler class we extends `SQLiteOpenHelper` for creating database and table and then performing Insert and Update operation in SqLite database.

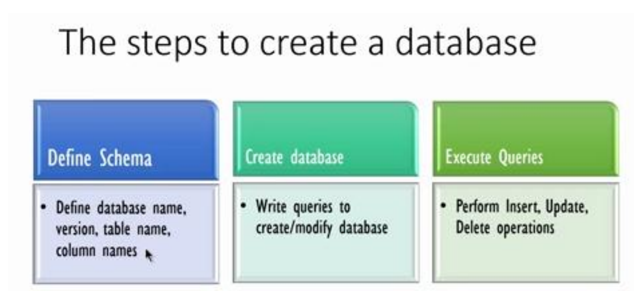


onCreate and onUpgrade methods are provided by SQLiteOpenHelper. We used 3 level desing and thus using adapter for performing operations with database.



'onCreate' method is called when the database is created for first time. Creation of table and initial dta inside table is put inside the method.

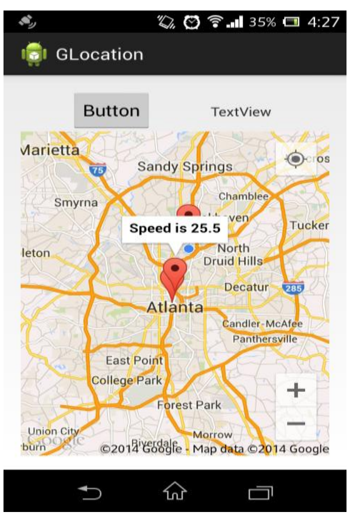
'onUpgrade' is called when database needs to be upgraded. We can use this method to drop tables, add tables or do anything else it needs to upgrade to the new schema version.



3. EXPERIMENT AND RESULTS

A prototype system architecture was implemented to test the system which consists of four layers: GPS-enabled smartphones in vehicles (driving public), a cellular network operator (network operator), cellular phone data aggregation and traffic estimation (Nokia/Berkeley), and information dissemination (Info Consumers). On each participating mobile device (or client), an application is executed which is responsible for the following functions: downloading and caching data from server, detecting trip traversal, and filtering measurements before transmissions to the service provider. To determine trip line traversals, the device checks if the line between the current GPS position and the previous GPS position intersects with any of the trip lines in its cache. Upon traversal, the mobile device creates an update. The update comprises of a speed reading, time-stamp, and the latitude and longitude. These updates are transmitted to the server over a secure channel.

Note that all data packets transmitted from the mobile device, regardless of the application (traffic, email, etc), must contain the mobile device identification information for billing by the network provider. Thus, in the Mobile Century system, an ID proxy server is used to first authenticate each client to prevent unauthorized updates, then remove the mobile device identification information from the data packets. It then forwards the anonymized updates to the server.



We used Google API Map services V2 for using Maps in our android application.

4. RELATED AND FUTURE WORK

Combine with phone compass or other information to identify the turn point of path. Identifying turning point is critical for speed calculation in local roads. Displays weekly archived traffic data in google maps – user can inquire the archived traffic condition at certain historical time and day of a week. Tab controller to switch between a live map and archived data map. Sliding bar that allows user to slide the time and day (such as 1PM Monday) to see the achieved traffic data.

Later we will try to combine accelerometer information to determine walking/sitting/driving state and do not upload data from someone who carries the smart phone while walking or biking. Phones could also form ad hoc network through its wifi or bluetooth to exchange the traffic information, so that phones without Internet link can also exchange and get local traffic information.

Activity recognition has recently gained attention as a research topic because of the increasing availability of accelerometers in consumer products, like cell phones, and because of the many potential applications. Some of the earliest work in accelerometerbased activity recognition focused on the use of multiple accelerometers Placed on several parts of the user's body. In one of the earliest studies of this topic, Bao & Intille [12] used five biaxial accelerometers worn on the user's right hip, dominant wrist, nondominant upper arm, dominant ankle, and non-dominant thigh in order to collect data from 20 users. Using decision tables, instance-based learning, C4.5 and Naïve Bayes classifiers, they created models to recognize twenty daily activities. Their results indicated that the accelerometer placed on the thigh was most powerful for distinguishing between activities. This finding supports our decision to have our test subjects carry the phone in the most convenient location—their pants pocket. Other researchers have, like Bao & Intille, used multiple accelerometers for activity recognition.

5. CONCLUSION

In this paper we described how a smart phone can be used to perform and monitor real time traffic data, simply by keeping it in ones pocket. We further showed that traffic can be highly accurate, with most activities being recognized correctly over 95% of the time. In addition, the traffic can be recognized quickly, since each example is generated from only 10 seconds worth of data. We have several interesting applications in mind for activity recognition also and plan to implement some of these applications in the near future. Our work would not have been possible without establishing Android-based data collection platform, and we view this software and hardware architecture, where data is transmitted by the phone to our Internet-based server, as a key resource produced as a consequence of this work. By having this in place we will be able to mine other mobile sensor data much more quickly.

This platform, as well as the data that we collected, will ultimately be made public. We plan to implement human activity recognition in several ways. The straightforward improvements involve: 1) learning to recognize additional activities, such as bicycling and car-riding, 2) obtaining training data from more users with the expectation that this will improve our results, 3) generating additional and more sophisticated features when aggregating the raw time-series data, and 4) evaluating the impact of carrying the cell phone in different locations, such as on a belt loop. In addition, in the near future we plan to significantly enhance our online database plateform so that we can generate results in real-time, whereas currently our results are generated off-line and are not reported back to the mobile phone and the user. We plan to provide real-time results in two ways. The first way minimizes the intelligence required on the phone by having the phone transmit the data to the Internet-based sever over the cellular connection, as usual, with the server applying the activity recognition model and transmitting the results back to the phone. In one variant, the phone will send the raw accelerometer data and in a second variant the phone will perform the data transformation step and only transmit the data when an example is generated. The second method involves implementing the activity recognition model directly on the cell phone. Given the computational power of these devices, this is certainly a feasible option. One key advantage of this method is that it removes the need for a server, which makes the solution perfectly scalable, and ensures the user's privacy, since the sensor data is kept locally on the device. The work described in this paper is part of a larger effort to mine sensor data from wireless devices. We plan to continue our project, applying the GPS and accelerometer data to other tasks besides activity recognition and collecting and mining other sensor data, especially GPS data. We believe that mobile sensor data provides tremendous opportunities for data mining and we intend to leverage our Android-based data collection/data mining platform to the fullest extent possible.

6. REFERENCES

[1] 4.1 Billion Mobile Subscribers Worldwide Helps Reduce

Digital Divide (Slightly), March 2009.

www.moconews.net/entry/419-4.1-billion-mobilesubscribers-mobile-helping-reduce-digital-divide-sli/

[2] Worldwide Mobile Phone Sales in 2009, February 2010.

http://www.gartner.com/it/page.jsp?id=1306513.

[3] E. Agapie et al. Seeing our signals: Combining location

traces and web-based models for personal discovery. In

HotMobile, 2008.

[4] [7] M. Bilandzic et al. Laermometer - a mobile noise mapping application. In NordiCHI 2008, 2008.

[5] D. Burke et al. Participatory Sensing. In World Sensor Web

Workshop, 2006.

[6] WISDM (Wireless Sensor Data Mining) Project. Fordham

University, Department of Computer and Information Science, http://storm.cis.fordham.edu/~gweiss/ wisdm/

[7] R. Honicky, E. Brewer, E. Paulos, and R. White. N-smarts: Networked suite of mobile atmospheric real-time sensors. In NSDR, 2008.

[8] A. Krause, E. Horvitz, A. Kansal, and F. Zhao. Toward Community Sensing. In IPSN, 2008.

[9] N. D. Lane, S. B. Eisenman, M. Musolesi, E. Miluzzo, and A. T. Campbell. Urban Sensing Systems: Opportunistic or Participatory? In HotMobile, 2008.

[10] Worldwide Mobile Phone Sales in 2009, February 2010. http://www.gartner.com/it/page.jsp?id=1306513.

[11] [37] N. Ramanathan, S. Reddy, J. Burke, and D. Estrin. Participatory sensing for project surya. In SenSys 2007 Workshop on Sensing on Everyday Mobile Phones in Support of Participatory Research, 2007.

[12] Bao, L. and Intille, S. 2004. Activity Recognition from UserAnnotated Acceleration Data. Lecture Notes Computer Science 3001, 1-17.