Travel Data Collection: Developing an Android Smartphone Application

Raul K. Gonzalez

Xin Ye Ph.D.

Civil Engineering

McNair Scholars Program

California State Polytechnic University Pomona

August 29th, 2013
Abstract:

Travel data collection refers to the data collected as someone travels and generates a list of trips, destinations, purpose, modes of transportation, company, departure, arrival times, and route choice. Travel data has been used to create a travel demand model which is a useful instrument to create new transportation policies. It has also put into effect to see how current transportation policies affect air quality and to gain a better understanding of how and why people travel. Our research studies the development of a cutting-edge travel data collection device by taking advantage of the wide availability of smartphones. An Android smartphone application (app) will be developed using Eclipse app development software. This app will detect longitudinal/latitudinal coordinates, exact time for those coordinates, store and then send these collected data to a server to be plotted onto Google Earth, where the trips will be visualized. We will use various software programs to convert the text file stored on the phone’s memory into a format that can be plotted onto Google Earth. Currently, an Android app has been developed that detects and stores coordinates on the internal memory. An R script has been written to convert coordinates stored in comma separated value (CSV) format into a keyhole markup language (KML) file and display coordinates in Google Earth. A server has been set up and the connection between the App and the server is currently being made. This study focuses on the development of a smartphone application, setting up an appropriate server, and ensuring that the application is both battery efficient and runs as a background service.
Development of a Smartphone Application for Travel Data Collection

Introduction:

Travel data collection refers to the data that is gathered as a traveler changes trajectory from one point to the next, during this trajectory a list of the trip, mode of transportation, company, departure/arrival time, as well as route choice is generated. The largest problem that travel data collection faces today is that we have a chance to take an advantage of technological advancements to create better devices for travel data collection purposes and we are not doing so.

According to the National Household Travel Survey (NHTS) the earliest data they gathered is from 1969 (NHTS.gov) There have been many methods developed throughout the years and new methods continue to be developed. As the American Society of Civil Engineers report in 2011, "Methods that have worked in the past are no longer working as well. For example, telephone interview response rates have declined substantially over time, and directories of land telephone lines that have traditionally been used do not capture an increasing population of individuals who only use mobile telephones" (Shalaby & Saneinejad, 2011). This shows an increase demand for a more innovative technology to collect travel data.

Since the turn of the century, mobile technology has been more widely available to the general public. A recent study done by CNN shows that about “half the phones in the U.S. are now Smartphones” (Goldman, 2012) and the National Center for Health Statistics, “indicate that the number of American homes with only wireless telephones continues to grow. More than 3 of every 10 American homes (31.6%) had only mobile phones during the first half of 2011—an increase of 1.9 percentage points since the second half of 2010” (NCHS, 2012). This clearly shows the rise in mobile technology in the average American family household. Having such wide availability in technology has given the necessary tools to conduct better research.
experiments such as travel data collection. Travel data collection refers to keeping track of travel times, destination, purpose of trip, and route choice.

Travel data collection is a rather important asset to transportation research because it has many applications such as understanding traveler behavior, analyzing the effect of current transportation policies on traffic and the quality of the air (Ben-Akiva & Cambridge Systematics Inc., 2003). Having this data available opens up several opportunities to provide better services to the travelers and become more eco-friendly. Travel data is essential to develop a travel demand model and provide the public with better services.

Smartphones have certain characteristics that if taken an advantage of can lead to major contributions to the travel data collection efforts. These characteristics are the Global Positioning System (GPS), internet access, phone, short message system (SMS) and can be conveniently carried. This has lead to more accurate travel data collection studies.

The United States population can be greatly benefited from a study like this. If a smartphone application is released to the general public and travel data is collected, this can lead to creating a dataset that can be a useful asset to developing new transportation policies. According to the National Household Travel Survey, in 2015 there will be a national travel survey utilizing conventional methods such as phone calls which have worked in the past as well as newer, more efficient methods such as email, mobile phones and smartphones.

The purpose of this study is to develop an android smartphone application to expand the current travel data collection knowledge in the field of transportation. This application will report exact place and time after a 12 hour traveling period. This data will be plotted into Google Earth to give us a visual representation of what the daily travel route looks like. From this we should
be able to detect travel patterns and gain a better understanding on traveler’s behavior (i.e. trips, destination/purpose, mode, company, departure/arrival time, and route choice).

*Literature Review:*

Travel data collection has been done since the early 1970s (NHTS.gov). Some of the applications that travel data has include to detect travel patterns, to comprehend traveler behavior (i.e. how and why people travel), as well as to develop a travel demand model which can be a useful tool in creating new transportation policies. Additionally, travel data has also been use to try to influence the traveler to use alternative modes of transportation such as biking, walking, or public transportation. In a study done by UC Berkeley, the University of California Transportation Center creates a smartphone app which analyzes travel data and compares the calories being burnt and compares that to the CO₂ emissions that the user has when using a vehicle. UCTC comments, "First, it was intended to demonstrate how the Quantified self-movement can be leveraged by the travel demand modeling community for data collection...Second, the experiment was designed to investigate whether data collection effort that includes elements of traveler feedback and a direct engagement of subjects via a website can lead to a change in behavior toward a more sustainable mode of transportation." (UCTC, 2012).

In another case by the Massachusetts Institute of Technology takes a look at “current transportation policies and their effect on traffic and air quality.” (Ben-Akiva & Cambridge Sytematics Inc., 2003). These are two examples of what travel data collection has been used for. Travel data is a growing topic, as technology develops, it can lead to improved policies that can affect society on the individual basis as well as on a global basis.

One of the earliest methods of collecting travel data is using travel diaries, where the holder of the diary would write down his/her trips, times and purpose to the best of their abilities
on the diary. The researcher would then collect the diaries to analyze them. The biggest issue with conventional travel diaries is that the travelers would often forget to write down short distance trips. Additionally, it was merely impossible for the users to record their precise location. As the researchers from the University of Tokyo comment, “Problems with these methods [traditional travel diaries] include lack of reporting for short trips, poor data quality on travel start and end, travel times, and destination locations” (Ohmori, Muromachi, et. Al., 2000). This gives an idea for the need of a more innovative way of gathering travel data.

Travel diaries have often been accompanied by reminders via phone calls and/or beepers. As technology develops, the previews issues are taken into consideration and for the most part they are fixed. These calls would often be randomized, not all participants would be reminded to record their information on their travel diaries. Additionally, "In early applications, researchers often relied upon telephones, pagers, or beepers to either remind participants to keep a diary of the studied event... The timeline of reminders and signals were often randomly scheduled and rarely occurred at the times they supposed to occur, i.e., immediately after the study event" (Chen & Fan, 2011). This shows the need for an automated immediate response after the trip has been made so that better data is collected.

The development of Global Positioning System (GPS) significantly helps travel data collection. GPS provides a more accurate data collection when using such tool as an aid to collect travel data. An example of these handheld devices can be found in Guensler and Wolf’s research Development of a Handheld Electronic Travel Diary for Monitoring Individual Tripmaking Behavior, which is one of the earliest methods for GPS related travel data and it is created in the late 1990s. This device breaches the gap between inaccurate data provided by the user and actual coordinates and time. However, these devices are rather costly. The average price
on one of these handheld electronic devices is about $1200 depending on the capacity of the memory (flash disk) and also software had to be purchased in order for the device to work properly (Guensler, and Wolf, 1999). Taking into consideration that the flash disk (or SD cards) are now relatively inexpensive, these devices are still rather inconvenient for travel data collection purposes not only because of their size but also because there are better alternatives to travel data collection such as Smartphones.

The idea of handheld travel data collection devices is quite simple; this provides the traveler a device that not only allows him/her to gain an accurate location and time but it also replaces handwritten biased travel diaries. The handheld electronic devices have the options of the traditional travel diaries (i.e. how many people in the car, who is driving, reason for trip, route choice, and departure/arrival time). Additionally, all these data is stored inside the devices’ memory unit and it can be easily extracted for the researcher to conduct analysis using software.

Travel data collection has been an ongoing collective process with new methods being created to collect these types of data. Recent studies show smartphones as the new and more efficient way to go about travel data collection. The smartphone holds all the capabilities necessary to collect travel data such as the electronic handheld devices and they are much more efficient due to people’s necessity to carry their phones with them at all times. Smartphones are equipped with a GPS sensor, have internet access (either via Wi-Fi or data plan), they can receive calls, and text messages (SMS) which makes Smartphones the perfect tool to collect travel data.

Additionally, recent studies such as Ben-Akaiva, Cambridge Systematics, Chen & Fan, and UCTC show that smartphones will open up the door to more innovative ways to conduct travel data experiments using smartphone technology; especially using smartphone applications
also known as Apps. Chen & Fan state, "Utilizing the computing, communication and sensing capabilities of smartphones [this app] collects real-time information about traveler behavior, physical activity and psychological experiences associated with travel" (Chen & Fan, 2011). For these recent studies the researchers have developed smartphone applications and released them onto a certain population to collect travel data from participants. However, some of the issues that have been reported by several of these institutions include constantly having to refresh the App, as well as battery consumption from the app.

**Methodology:**

The methodology for this experiment includes developing the android smartphone application (App) using open source Eclipse app development software powered by Java. It is essential to our research to figure out a way to conserve battery and run the android application on the background. Using this software, we will run the App in the background. In addition to creating a background service; this application detects, stores and sends longitudinal, latitudinal coordinates, exact time for those coordinates, accuracy, speed and bearing.

Once the data is obtained, it will be converted from the original text file (obtained from the phone) to a comma separated value file (CSV) and then converted into keyhole markup language file (KML) so that the coordinates can be plotted onto Google Earth. We will be using R statistical software to create the KML file, as well as SPSS to conduct statistical analysis.

Ideally, this application will have a built-in survey that will ask the traveler simple questions after making a trip such as purpose (school, work, pleasure, shopping, etc), how many people he/she is traveling with, and mode of transportation (i.e. public transportation, walking, bicycle, car, etc). The data will be ideally automatically uploaded to a server after a certain data
collection period; this will include uploading not only the travel data but also the travel diary data.

Results:

We were successful in creating an Android App that runs in the background and does not interfere with the phone’s functions. Figure one shows what the user interface looks like on the App named Traveler. As the information from the satellite is updated, the data is refreshed and displayed on the interface. Additionally, the data shown is also recorded on the internal memory of the smartphone, Traveler records data by the second.

Figure 1: User interface

In addition to the interface, there is code which aids in obtaining data from the satellite and storing it in the internal memory. Here is the Java code for the entire App, figure two shows the code for the main activity for the App with comments. Figure three shows the class that is created that enables the app to run in the background.

Figure 1: Main Activity

package com.example.traveler;
import java.io.File;
import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.OutputStreamWriter;
import java.util.Date;
import android.location.Location;
import android.location.LocationListener;
import android.location.LocationManager;
import android.os.Bundle;
import android.app.Activity;
import android.content.Context;
import android.content.Intent;
import android.view.Menu;
import android.widget.TextView;

public class MainActivity extends Activity {

    TextView textLong;
    TextView textLat;
    TextView textTime;
    TextView textAccu;
    TextView textSpeed;
    TextView textBearing;

    //Allows the application to start to run in the background automatically when the App is launched
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        startService(new Intent(getBaseContext(), MyServices.class));

        //Obtains the following data from GPS, displays obtained data on interface
        textLong = (TextView) findViewById(R.id.textLong);
        textLat = (TextView) findViewById(R.id.textLat);
        textAccu = (TextView) findViewById(R.id.textAccu);
        textSpeed = (TextView) findViewById(R.id.textSpeed);
        textBearing = (TextView) findViewById(R.id.textBearing);

        //Location listener: 1st parameter is timing in milliseconds, 2nd is displacement
        LocationManager lm =
        (LocationManager) getSystemService(Context.LOCATION_SERVICE);
        LocationListener ll = new myLocationListener();
        lm.requestLocationUpdates(LocationManager.GPS_PROVIDER, 1000, 1/2, ll);
    }
    
    class myLocationListener implements LocationListener{

9 | Gonzalez & Xin
@Override
public void onLocationChanged(Location location) {
    if(location != null)
    {

        //Defining variable for each parameter
        String Output = "";
        double pLong = location.getLongitude();
        double pLat = location.getLatitude();
        double accu = location.getAccuracy();
        double speed = location.getSpeed();
        double bearing = location.getBearing();

        long time = System.currentTimeMillis();
        Date DateTime = new Date(time);

        //Makes a string with the desired data
        Output = DateTime + "," +
                Double.toString(pLong) + "," +
                Double.toString(pLat) + "," +
                Double.toString(accu) + "," +
                Double.toString(speed) + "," +
                Double.toString(bearing) + "\n";

        //Saves string on the internal memory
        File myFile = new File("/sdcard/mysdfile.txt");
        FileOutputStream fOut = null;
        try {
            fOut = new FileOutputStream(myFile, true);
        } catch (FileNotFoundException e) {
            e.printStackTrace();
        }

        OutputStreamWriter myOutWriter = new OutputStreamWriter(fOut);
        try {
            myOutWriter.append(Output);
        } catch (IOException e) {
            e.printStackTrace();
        }

        try {
            myOutWriter.close();
            fOut.close();
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}
} catch (IOException e) {
    e.printStackTrace();
}

textLong.setText(Double.toString(pLong));
textLat.setText(Double.toString(pLat));
textAccu.setText(Double.toString(accu));
textSpeed.setText(Double.toString(speed));
textBearing.setText(Double.toString(bearing));

@Override
public void onProviderDisabled(String arg0) {
    // TODO Auto-generated method stub
}

@Override
public void onProviderEnabled(String arg0){
}
    // TODO Auto-generated method stub

@Override
public void onStatusChanged(String provider, int status, 
Bundle extras) {
    // TODO Auto-generated method stub
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    // Inflate the menu; this adds items to the action bar if it is present.
    getMenuInflater().inflate(R.menu.activity_main, menu);
    return true;
}

Figure 3: Background Service Class

package com.example.traveler;
import android.app.Service;
import android.content.Intent;
import android.os.IBinder;
import android.widget.Toast;
import android.os.IBinder;
import android.widget.Toast;

//Class for background service

public class MyServices extends Service {


@Override
public IBinder onBind(Intent arg0) {
    // TODO Auto-generated method stub
    return null;
}

@Override
public int onStartCommand(Intent intent, int flags, int startIdarg0) {
    //this service will run until we stop it
    Toast.makeText(this, "Service Started...", Toast.LENGTH_LONG).show();
    return START_STICKY;
}

Furthermore, there are two added permissions. The first one being permission for the app to access the GPS signal, and the second one is permission to access the memory unit and record the data gathered by the GPS signal. Once the data is collected it is ideally sent to a server, where it can be extracted and analyzed. The server has been setup utilizing WAMP (Windows Apache MySQL PHP) server on my personal computer with my home router. The URL for the server is http://travelerapp.from-ca.com/.

Once the data is collected, we have to convert it from the original text file, obtained from the smartphone, into a comma separated value file (CSV) where we can begin to analyze the gathered data. It is important for us to change the file format because it is much easier to convert CSV files into keyhole markup language files (KML), in order for the coordinates to be plotted onto a Google Earth Map. Figure four shows an example of the R script that is used to convert the files. Figure five shows a KML file being plotted onto Google Earth. It displays a path from the data collected, starting at the earliest time and ending with the last point. What the code does is it obtains the latitude and longitude and marks them as individual points that can be plotted onto a map, like an X & Y plane but with coordinates. This path in particular was captured using
three different modes of transportation: car, bicycle and walking. This shows that the App works well in various modes of transportation.

```r
setwd("C:\\Users\\User\\Documents\\KML")
points = read.csv("1_meter_parameter.csv")
coordinates = paste(points$longi, points$lati, sep = ",")
filename = file("1_meter_parameter.kml")
head = c("<kml><Document><Placemark>",
"<Style>",
"<LineStyle>",
"<color>fff00000</color>",
"<width>10</width>",
"</LineStyle>",
"</Style>",
"<LineString><coordinates>
"
tail = "</coordinates></LineString></Placemark></Document></kml>"
writeLines(c(head, coordinates, tail), filename)
close(filename)
```

Figure 4: R script to convert CSV file to KML file

![Google Earth map with marked lines](image.png)

Figure 5: KML file plotted on Google Earth

*Discussion:*
Even though we created an Android App that runs in the background and is somewhat battery efficient, there are certain limitations to the App. For instance, after testing out the device on two different android phones, there were some discrepancies between the two. The main one being that the battery reacted differently to our Traveler App. The second one was that the App would stop after certain time of inactivity or searching for the signal and not finding it due to a weak GPS connection.

We also found some gaps in the points collected when we were collecting data by the second. We plotted the data onto a map; we realized that those gaps occur when the satellite signal is being blocked off by being inside a building or being in the woods where there was a poor satellite signal.

This App can work with a data plan, Wi-Fi connected or offline. The only difference is that when it is connected to a network, the App tends to get better accuracy and a faster connection compared to the offline mode, where it is less accurate and takes more time to gain the satellite signal. Another issue is that when the App is connected to a data plan, it uses several megabytes of data. In the future study we will expand on this concept to gain a better understanding of exactly how much data is consumed by the App.

I have also had some difficulties in creating the connection between the App and the server due to a lack of time and not being able to find the best way to connect it. However, I hypothesize that I can begin by making a separate App that uploads some basic information to a server and from there I can apply the same concept to Traveler. I know I have to learn some basic PHP coding in order to create a database in WAMP server so that the data can be uploaded there.

*Future Work:*
For the future, I would like to continue looking for ways to save more battery directly from the App. I would also like to test out the App on a larger controlled population (anywhere between 15-20 participants) and gather travel data from them. Future work will also include the learning GIS software and applying it to travel data. Additionally, I really want to focus on the connection between the Android App and the server, as well as creating the built-in travel diary.
References:


