CENS Mobility Classification Service

Brent Longstaff, Hossein Falaki, Nithya Ramanathan, Hongsuda Tangmunarunkit,
Cameron Ketcham, Joshua Selsky, John Jenkins, Deborah Estrin

CENS, UCLA

Introduction

Mobility is a human activity and transport mode classification service for Android that was developed and maintained by the Center for Embedded Networked Sensing (CENS). It uses accelerometer, GPS, and visible Wi-Fi access point names to classify the user's activity to one of *still*, *walking*, and *driving*. The Mobility service operates on an interval basis (such as one minute or five minutes) that can be configured by the smartphone user. It uses the Android content provider mechanism to allow other applications to read the classification results.

The classifier is a decision tree. First, it uses accelerometer data to determine the user's ambulation state (walking or sedentary). If the user is sedentary, it then it infers whether the user is still or in a vehicle by observing changes in the surrounding Wi-Fi access points and the change in location over time. Mobility uses Android libraries (developed at CENS) to read sensor information, which are included in the app. It uses AccelService to collect acceleration data, and WiFiGPSLocation for GPS and Wi-Fi information. In the following sections we introduce these two services.

AccelService

AccelService is an Android application that continuously runs as a service. The primary user of this service is the Mobility classifier, but it is capable of supporting multiple applications (a.k.a. clients). Each client submits the rate, recording length and interval of its desired accelerometer samples. AccelService will then interface with the Native Android sensor manager API to record accelerometer sensor information such that it can be used by all the clients.

Android's native sensor manager API is asynchronous – applications can register for accelerometer values with a given rate and will be called back when data is available. AccelService offers a much simpler synchronous interface to applications. Every time an application makes a function call (through native Android IPC mechanisms), the call returns immediately and contains the latest batch of data according to the requirements of the application.

WiFiGPSLocation

WiFiGPSLocation is an Android service to simplify duty-cycling of the GPS receiver when a user is not mobile. The WiFiGPSLocation application runs as an Android Service on the phone. It defines a simple interface using the Android Interface Definition Language that any other application can use for energy efficient location tracking. Applications can get the last location of the user through any time they make a call to this service. Unlike the default Android location API, the location API provided by WiFiGPSLocation is synchronous (i.e., a call to getLocation() is guaranteed to return immediately with the last location of the user.)

WiFiGPSLocation continuously scans for visible Wi-Fi access points (APs). The result of each Wi-Fi scan is stored in an in-memory cache along with the corresponding GPS location. If a specific set of Wi-Fi access points has been seen for significant number of times, it is marked as a "significant entry." Next time this same Wi-Fi signature is seen, the service concludes that the user is not moving, and therefore turns off GPS. The GPS receiver is turned back on as soon as the set of visible Wi-Fi APs changes (to a non-significant signature). Applications get access to GPS information as well as Wi-Fi signatures.

Mobility Classifier

Once each mobility sampling interval, the accelerometer data are recorded for one second. This produces up to 50 triaxial acceleration values (the exact number varies based on phone hardware). The classifier is designed to be orientation-agnostic; therefore it only uses the magnitude of acceleration. It calculates the magnitudes of each sample and then calculates the variance across all the samples during each interval. It also uses recent Wi-Fi fingerprints (lists of scanned access points nearby) and location to determine whether the user is changing location. These are used in a decision tree to classify whether the user is still, driving, or walking.

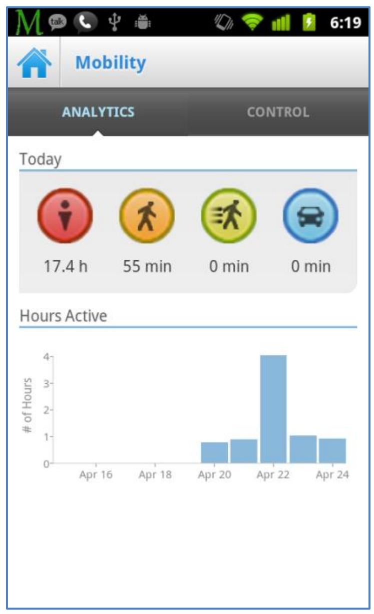
For the location change estimate, the classifier originally used GPS velocity. However, there were two disadvantages to this approach. First, GPS consumes significant energy, which reduces the phone’s battery lifetime. Second, accuracy is poor indoors and wherever GPS satellites are obstructed, which leads to driving being detected as still and vice versa.

To avoid these problems we originally moved to an approach using only Wi-Fi scanning. The WiFiGPSLocation service already scans Wi-Fi to cache GPS locations in order to save power, therefore this new approach imposes no additional energy cost. The classifier compares the most recently scanned Wi-Fi access points to those seen over the previous 10 minutes to determine whether the user is moving or still. If the current Wi-Fi access points are recognized from previous samples, it decides that the user is still. If all or most are new, then the user is presumed to be driving. The Wi-Fi classifier does not attempt to distinguish different types of motion; Mobility still relies on the accelerometer to determine whether the motion is due to walking or driving. Using Wi-Fi increased the accuracy of the transport mode classifier, and allowed us to greatly reduce the use of GPS to save energy and phone battery life. GPS is still necessary in situations where there are no visible Wi-Fi access points, but can be turned off in areas with Wi-Fi coverage, which is the case for most urban areas.

In the most recent version of Mobility, we also tracked the user’s most recent locations (as reported by GPS or network location) and now the classifier will infer motion when the location changes by more than a certain distance threshold.

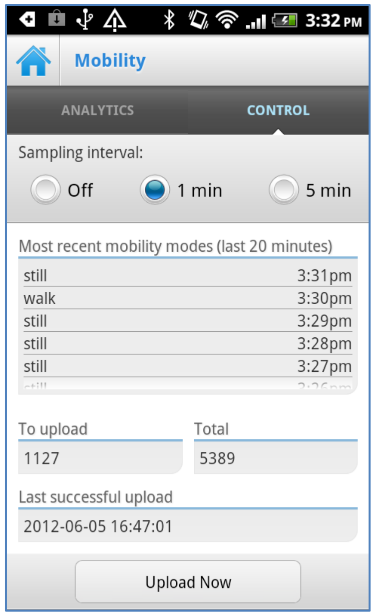
Feedback

Our health application framework (named Ohmage) offers concise feedback to the phone user based on the mobility classification results. The feedback is based on data from past seven days. The following screenshot shows a sample visualization generated on the phone by Ohmage.



Control

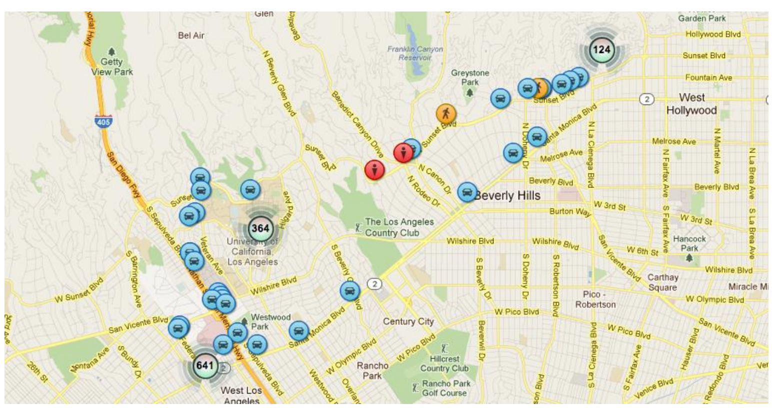
Ohmage also has a control and diagnostic screen, which allows users to adjust the classifier duty cycling interval to 1 or 5 minutes. It also displays classifier results from the previous twenty minutes for diagnostics and detailed feedback.



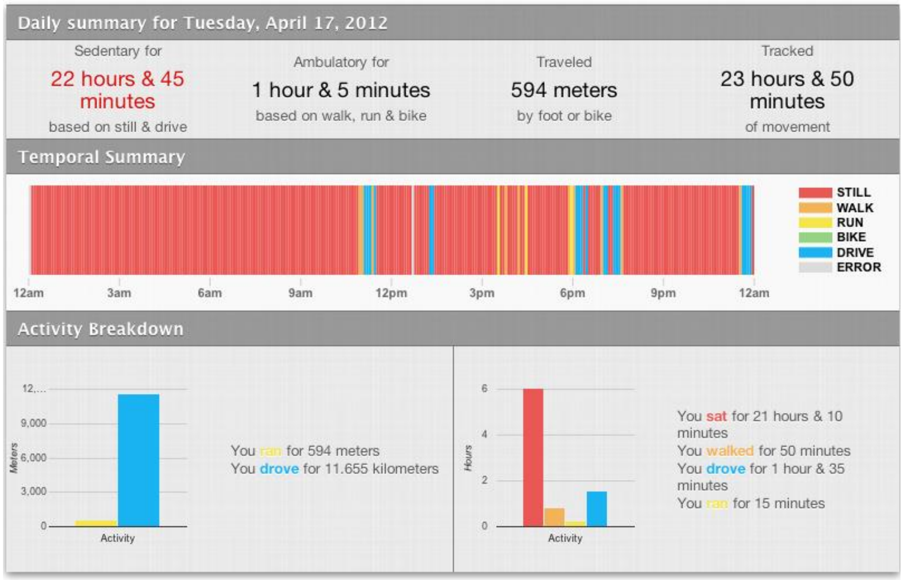
Visualization

In addition to the simple graphs on the phone that serve as a form of feedback to the user, we have implemented sophisticated web-based visualizations for mobility classification results. Our implementation is entirely in Javascript. The browser code gets classification information from the Ohmage server for the user and generates interactive graphics on the client side.

Following are sample screenshots of our existing web-based visualization. The first graph shows the user's mobility modes based on location.

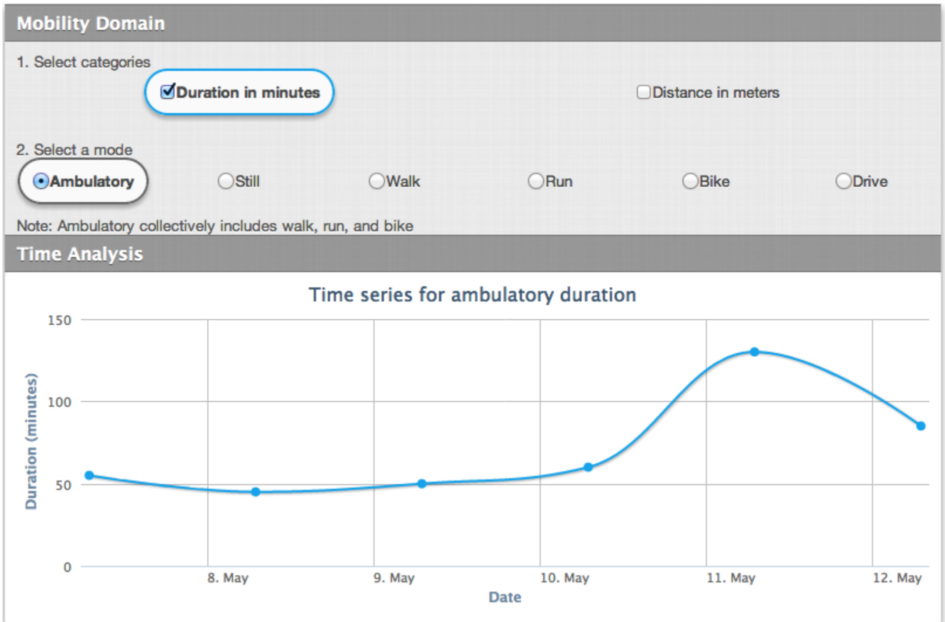


The second screenshot shows a temporal summary of mobility modes within 24 hours. It offers a summary of how active/mobile the user has been during the selected day.



The next graphs allow the users to compare their mobility modes across multiple days.

If a user is interested to compare the time or distance of a specific mobility mode across several days, he/she can use the next visualization, which allows the user to select either time or distance and the mode.



Mobility Domain
1. Select categories
Duration in minutes
Distance in meters
2. Select a mode
Ambulatory
Still
Walk
Run
Bike
Drive
Note: Ambulatory collectively includes walk, run, and bike
Time Analysis
Time series for ambulatory duration
Duration (minutes)
150
100
50
0
8. May
9. May
10. May
11. May
12. May
Date