

Title: "Crowd-Sourced Bus Tracking System"

Project ID : 12

April 11, 2018

1 Objective

To develop a bus tracking system that is crowd sourced and requires minimum human intervention to provide accurate and timely location of buses to the students. The students inside the bus will share location and other students waiting for bus can get updated location of bus on their mobile phones.



Figure 1: Objective

Many students face transportation problem in the institute. Most of the hostels are away from the academic area and students need to make multiple visits to the department mostly before and after lunch. The Tum Tum (Bus service) is inadequate, and unreliable especially in peak hours. Students have to wait for around 5 to 15 minutes with no knowledge of the buses. If they have proper information of the buses, they can timely leave the department/hostel and reach the bus stop and avoid unnecessary waiting. Moreover, if the buses are not available, they can look for other modes of transport like cycle, auto rickshaw

2 Names of team members

Name	Roll No	Email	Efforts
Vinay Koonu Teja	163050054	vinaykoonu@cse.iitb.ac.in	High
Ashish Kumar Jaiswal	163050055	ashishiitb@cse.iitb.ac.in	Medium
Trishal Patel	163050057	trishal@cse.iitb.ac.in	Medium
Maj Amit Pathania	163054001	sujanian.amit@gmail.com	Medium

3 Architecture diagram

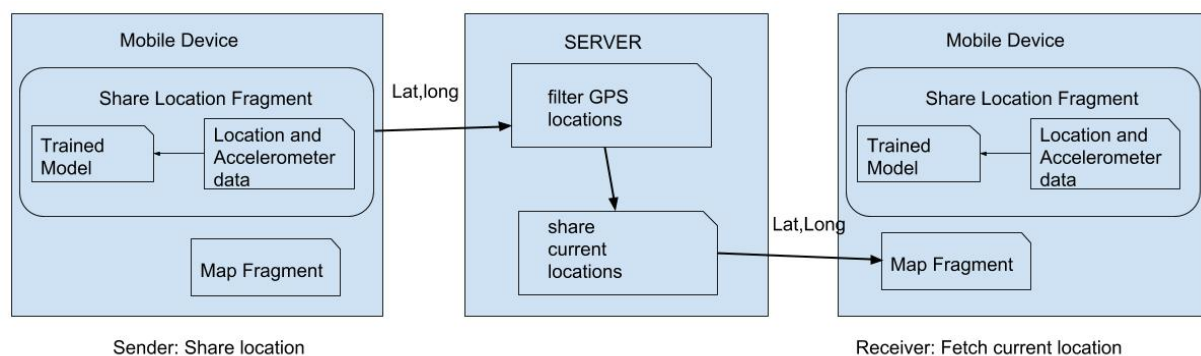


Figure 2: Architecture

4 Hardware and software pre-requisites

Hardware : Web server (django based) to collect data from cell phones and process information. The other users will fetch the current bus location from this server using their mobile internet/wi-fi. Server is hosted at <http://dummy.pythonanywhere.com> which is free web hosting website.

Software: LibSVM library to train model and automatically predict when user is inside bus using that trained model. Python packages for processing received data at server end.

5 Progress after the midsem demo

5.1 Up Till MidSem:

- Built Prototype App to collect data
- Built Main App "Bus tracker" that had UI implemented.

5.2 After MidSem: Setup Backend Server

- Server stores recent locations sent by different users.
- Server sends latest locations to the users.

5.3 After MidSem: Android Application

- Implemented “Share location” which allows user to send location to server.
- Application shows live locations received from the server on Google Map interface within the app.
- Collected data and trained SVM model to share predict if the user is in bus or not.
- Implemented an algorithm that detects is the user is on IITB road or not.

6 Usage of the app

The app is having four tabs: **map,share,record and train**

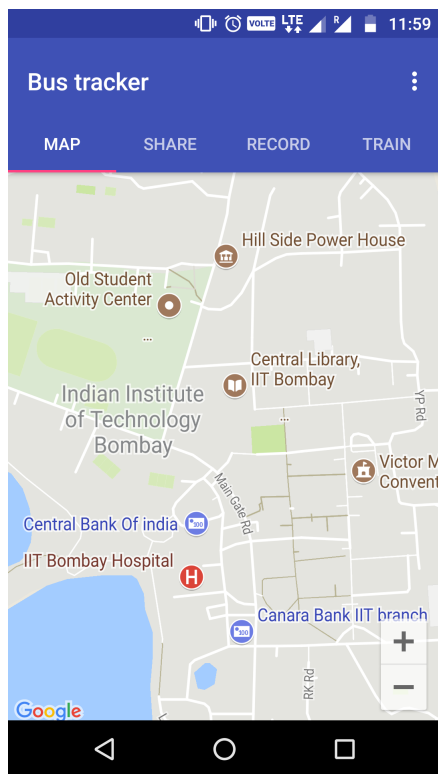


Figure 3: Home screen

6.1 Record Tab

In Record tab, user can record accelerometer and GPS data with labels for training. The labels are "Standing", "walking", "Running" and "Bus". These values will be stored in csv file.

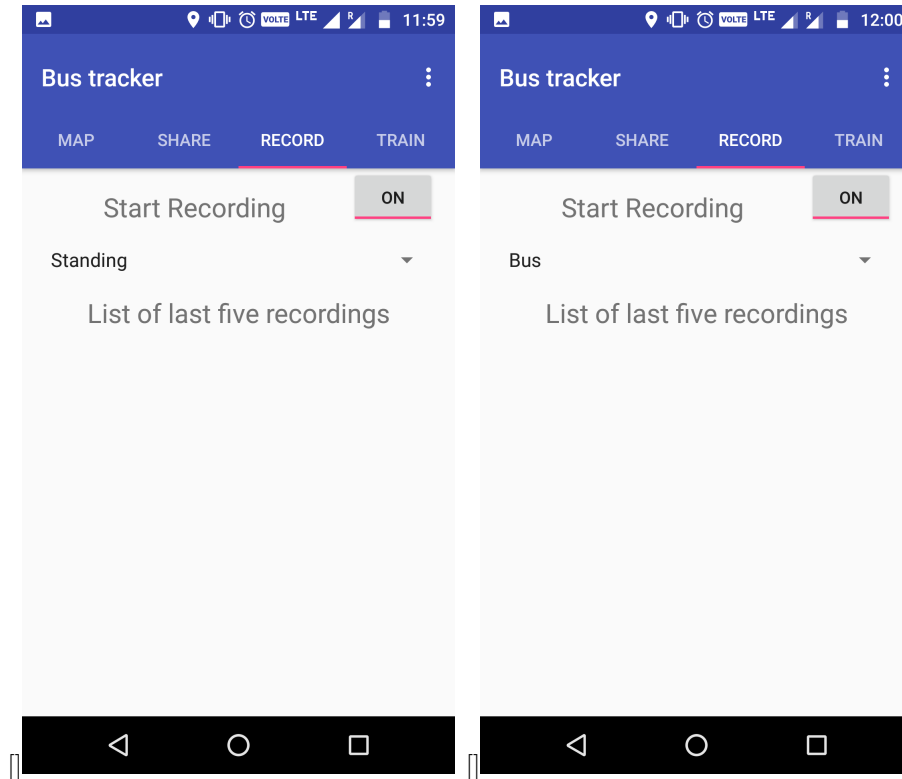


Figure 4: Record Tab

6.2 Train tab

In Train tab, user can pick the recording file and train the model using LibSVM library. We have considered acceleration along x,y and z axis (from accelerometer) and speed (from GPS sensor) as features for training the model. The trained model will be stored in csv file.

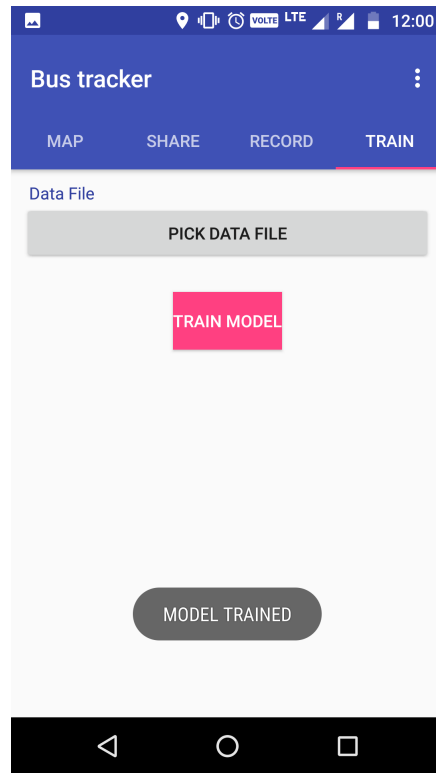


Figure 5: Train tab

Train dataset: 379

Given Labels: Standing, walking, bus

Used Binary encoding: Bus: +1, Others(walking, standing): -1

Features: acclx,acclz,acclz,speed

svm_type: c_svc

kernel_type : rbf

gamma: 0.166667

nr_class : 2

rho :-0.976288

6.3 Share tab

In share tab, user can select whether he want to share his location or not with the server. First app will check whether user he present near the road inside IIT Bombay. For this, we have mapped the complete latitude longitude of roads inside IIT Bombay. When app detects that user is within 5 meters range to road. It will predict whether user is inside bus or not using SVM predict function and model trained in previous step. Once it is learned that user is

inside bus using Machine learning, the app will send user device's unique ID and GPS coordinates. Unique ID ensures that we can consider only one reading per phone. The app will continue sending location update to server periodically whenever user is inside bus.

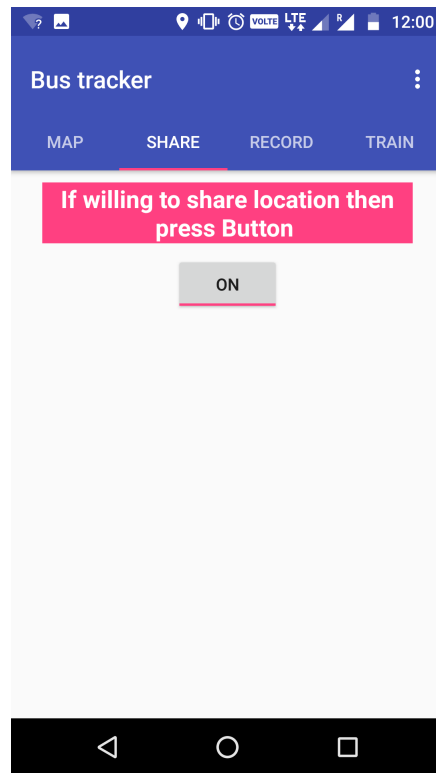


Figure 6: Share Tab

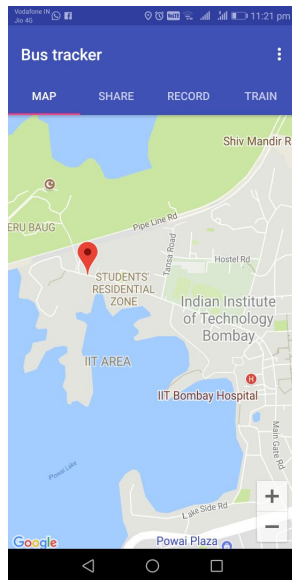
6.4 Map tab

In Map tab, user can view the current location of buses on the road. For this, user will send periodic requests to server and will pull updates from the server.



Figure 7: Map Tab

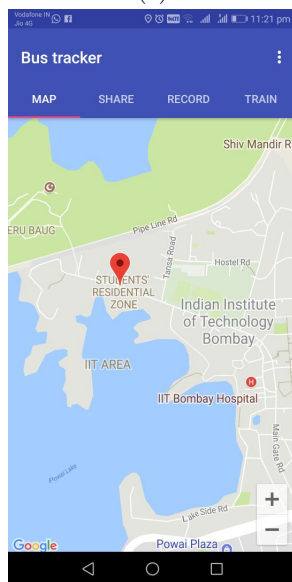
The moving bike(bus) being displayed on map:



(a)



(b)



(c)



(d)

7 Link to source code and executable

<https://drive.google.com/open?id=1LZMEqkxa7NRnn4sDqv6dvVLxY-LRx61G>

8 Unfinished tasks

We had planned to train the model on IIT buses but due to bus strike, we had implemented our project using cycle to verify our concept. We can extend the same concept in reality to actual buses.