Automatic Pothole and Ghat Complexity Detection System

K Asha
Asst. Professor,
Computer Science and Engineering
TJIT
Bangalore, India

Mamatha K
Asst. Professor,
Computer Science and Engineering
TJIT
Bangalore, India

Ashwini S Patil
Asst. Professor,
Computer Science and Engineering
TJIT
Bangalore, India

Abstract: **With the increased road networks many fatalities on road are increasing, the quality of road is also the point of concern in India. A part of this problem can be solved using the integrated smartphone that we use today. The flexibility and user friendly interface of today's smartphone has encouraged many software developments for mankind like us. This paper describes the ghat complexity and pothole detection of roads. It has been studied that the mobile sensors play an important role in studying the road condition with the company of GPS system. Using Android smartphones and its sensors, 90% of real time data is collected and processed information is provided to the user. This paper also unveils the capability of smartphones to solve real time problems.**

Keywords: Smartphone (android); accelerometer /magnetometer; algorithms; road condition

I. INTRODUCTION

Nowadays we all use Google maps and its application for navigation during travelling, but these applications are not so friendly to tell you any road's condition or its complexity.

Road Quality and Ghats complexity analysis using android phone proposes to utilize the GPS system of phone and different sensors like accelerometer, magnetometer, etc. of android phone, so we could analyze the road and can upload this information of that road on central server so every application user can use this information during traveling.

This information can be helpful to user at the time if there are multiple routes and for destination and he can choose one of the finest and shortest routes.

One of the major problems in developing countries is to collect the road's condition for maintenance purpose. This application can help in monitoring the roads condition and complexity of Ghats.

II. PROBLEM STATEMENT

Pothole detection system is a system that aims at warning the driver about the uneven roads and potholes in its path. We study the different ways in which goal of the system can be achieved. And then we give details about the working of the different subsystems.

This system consists of two components one is mobile node and other is the access point. Access points responsible for storing the information about potholes in its vicinity, taking the feedback from vehicles, updating the information in repository and broadcasting the information to other vehicles.

Whereas Mobile node which is the small device placed in vehicle is responsible for sensing those potholes which it did not have previous information about, locating and warning the driver about the potholes which it has information about, and giving the data about newly sensed pothole to access point.

While deploying the access point we feed in some initial data about potholes to it. Then it keeps on broadcasting the data. Vehicle equipped with the client device catches that data. Now the device has the information about the locations of potholes. The device is responsible for warning the driver about occurrences of pothole. But new potholes may always be formed because of environment or fatigue. So client device also acts as a sensor and finds out the occurrence of newly formed potholes on the road. If it finds out any new potholes it gives data of new pothole to Access point in terms of the feedback. Access points updates this information to its data store and then adds it to the information broadcast.

III. STATEMENT OF SCOPE

A. BUMP DETECTION

The lowest layer of the system is on the application running on the Smartphone. The application collects data from the accelerometer and GPS and then processes this to detect braking and bump events. It then attaches a time and Location tag to this data, and sends it across to the internet web server for further processing. Bump is detected using sensor data gathered from admin phone, details of location of bump is stored on the server side for other users.

B. FINDING GHAT'S COMPLEXITY

As we have seen the data which we get from accelerometer and magnetometer, in that we consider y axis for Ghats detection, here we calculate the angle of 'Y' axis with the north direction by which we can get how much car is turned at right or left side. For this we also consider the previous angle of 'Y' axis with north direction. This helps to count the number of turns in specific alarm, and also we can conclude how much they are tough.

C. EVALUATION OF ROAD AT SERVER SIDE

The REST web service on the server receives the event traces of several Smartphone's along with the time and location tags. Using this information, the web service infers higher level of evaluation such as road is smooth or it is with too much speed bump, Ghats are too complex or they are easy to drive, etc.

D. MAKE DATA AVAILABLE TO OTHER USERS

The web service needs to send over the inferred events to the Smartphone running the application. The Smartphone sends its location, and the internet service responds with events of interests in the vicinity of this location. These events are displayed on a map on the android phone, so that the user of the android application can choose to take alternate routes based on this.

IV. RELATED WORKS

As the smartphone is being used day by day it's more explored for both the real and virtual world. As the smartphones operating system is explored it's getting easier to develop applications for daily life problems, lately. Till now many theories were proposed on bump detection and recording, but they mainly lacked in detecting the exact position of the bump and potholes. This problem has been solved by the smartphones one of the sensor called "accelerometer". Also ideas have been proposed to include accelerometer in one of the cars number of features, to detect the potholes. The road surface roughness and its complexity can be monitored using accelerometer. India as developed many project that study the road conditions using the sensors provided by the smartphones like GSM radio, magnetometer and the GPS, the GPS system is also used to monitor traffic. Pothole patrol system done by massachusetts institute of technology is using hardware and software which is Linux and powered by soekris 4801 embedded systems. External accelerometer was used in this project. The android application was mainly based on the X, Y and Z axes.

The work done in this paper is similar where the co-ordinates are included, but here so algorithms are included like The road bump detection algorithm and also the haversine formula is applied. The formulae and algorithms is applied to the data collected by accelerometer to generate accurate output.

V. APPROACH

The development phase is divided into 3 parts:

- ✓ Accelerometer
- ✓ Location
- ✓ Database

A. ACCELEROMETER

The major element of this project is the Accelerometer. This element is an electromechanical device that measures acceleration. An accelerometer is one of the many hardware features contained in some android phones and has been used to create several interesting Android Apps. Some of these examples include:

- ✓ G-Lock: locks and unlocks the key pad based on movement of the phone
- ✓ Newton's Cradle: demonstrates the physics of Newton's Cradle
- ✓ Raging Thunder: a car racing game.

The accelerometer is one of several sensors available through the android API.

HOW IT WORKS

The Accelerometer measures acceleration on three axes- Azmuth (X), Pitch (Y) and Role (Z). The X axis is horizontal and points to the right side, the Y axis is vertical which points up and the Z axis points towards the outside of the front.

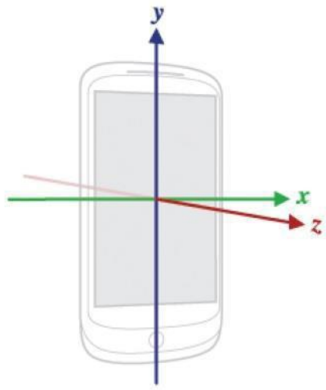


Figure 1: XYZ axes

Any sudden significant changes to one of the Axes should indicate that the car has hit a bump while driving.

The Android API has three classes:

- ✓ SensorManager
- ✓ SensorEvent
- ✓ SensorEventListener

Sensor Manager checks if the device has one or more sensors. The HTC Desire uses BMA150 3-Axis sensor to measure Acceleration and the AK8973 sensor to measure magnetic force. If no sensor is picked up, it returns a null value. SensorEventListenter checks for changes to sensor data while SensorEvent stores the data.

B. LOCATION

The second part of the project is to record the location. One of the better known features of android phones is the Network Location Manager which can record the devices location using either GPS and then display it using Google Maps. The user's location is used in many applications including Satellite Navigation, Location based Social Networking and Augmented Reality browser (e.g. Layer and Wikitude).

HOW IT WORKS

When the application is started the app calls Android's location listener which takes data from the phone's inbuilt

Global Positioning System (GPS) or from Cell tower or Wi-Fi data. The GPS (fine location) is the most accurate. GPS uses a network of 27 Earth Orbiting Satellites (24 active and 3 backups) developed by the US Military. The orbits are arranged so that during anytime and anywhere on Earth, there are at least four satellites "visible" in the sky. The GPS receiver locates four or more of these satellites. It then calculates the distance between itself and each satellite and compares the distances to triangulate its own location.

Figure 2: GPS Satellites in orbit

However despite the accuracy, GPS has some drawbacks. It only works outdoors, has a high battery consumption rate and is slower to return the location. The alternatives to GPS use Cell tower and Wi-Fi Signals to determine location. While they are not as accurate, they work indoors and outdoors, respond faster and require less battery power. This is known as the course location.

CHALLENGES

Another key challenge is identifying user location in a moving vehicle, especially considering GPS slow return rate. The easiest solution is to call the last known location. However the accuracy of this is questionable. One possible solution is to record the entire journey from start to finish. This involves calling the location at regular intervals to determine the path. When an event is recorded, the Application compares the event's timestamp to the route to get the location of the event.

DISPLAYING THE INFORMATION

For ease of programming the application currently displays the data in a digital decimal number for both Latitude and Longitude. The route and location of the Events can be displayed on a map using the Google Maps API which is available to android developers. In order to display data from Google Maps you need to register for a Maps API Key. The set up for the Network Location Provider for Android 2.2 and earlier is awkward as you have to set up the listeners to record all changes. In Gingerbread (Android 2.3), you can simply call a one-time location request using the request SingleUpdate() method.

C. DATABASE

Once the SensorEvent is created, it needs to be stored somewhere. The Android API supports development of a SQLite database. SQLite is "a software library that implements a zero-configuration, self-contained, transactional SQL database engine". The initial design included a list view that read the entries in the database and displayed them using the ListView format. Users could remove events from the list which would delete them from the table.

VI. DATA COLLECTION

As mentioned before the purpose of this experiment is to ascertain whether a vehicle mounted sensor mote can be used to detect potholes. To gather data for this experiment we will fix a MICAz sensor mote equipped with a CrossBow's MTS310CA sensor board on a vehicle. This sensor board contains an ADXL202E dual axis accelerometer. It has a range of $+1g$ to $-1g$. The sensor mote was fixed on the vehicle so that one axis (Y) of the dual axis accelerometer is aligned with the vertical direction and the other (X) aligned with an axis that goes from the back to front of the vehicle. The data gathering apparatus were also carried inside the vehicle. Note that while the BusNet requires a GPS sensor board we have not yet being able to acquire such sensor boards due to export/import restrictions; this restricts us to carry out some of the BusNet related experiment with simulated GPS coordinates. Therefore, for this experiment we used a PDA with GPS capability to get the GPS coordinates while the mote was used to collect the acceleration data.

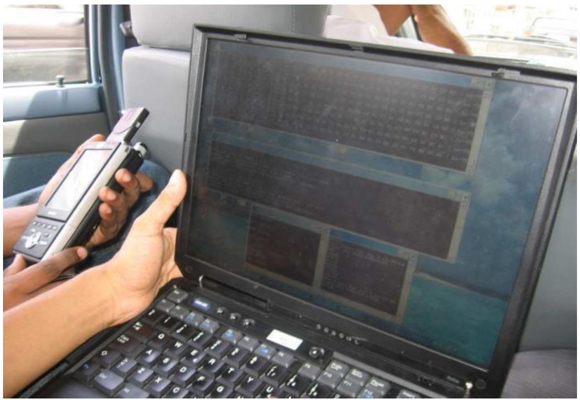


Figure 3: Collecting data

Figure 4 depicts the block diagram of the data collection and pothole detection process and the Figure 1 is a photo taken at one of the data collection runs. There are four main components to the data collection setup. The PDA, sensor mote, data logging terminal, and a human operator. As the vehicle travels along the road the sensor collects the vertical (Y) acceleration, and horizontal (X) acceleration of the vehicle, 100 times per second; the sensor collects 100 samples per second.

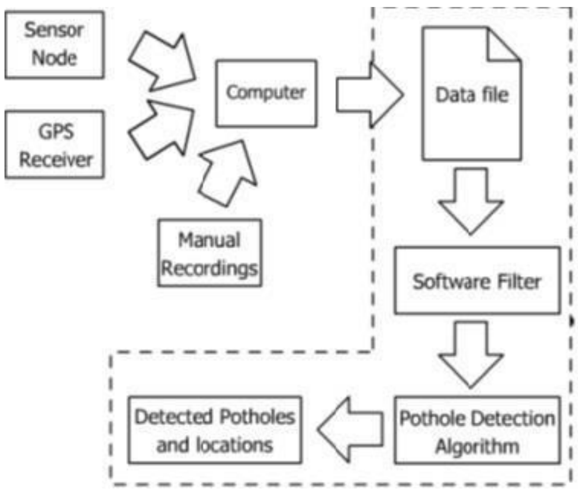


Figure 4: The block diagram of the data collection and the analysis process

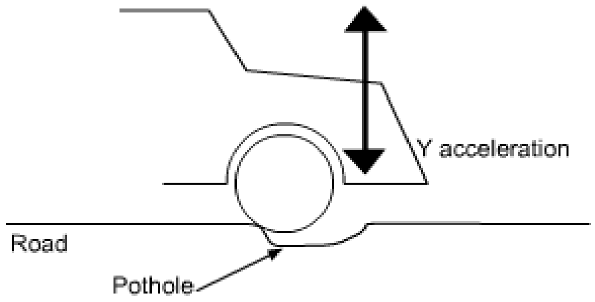


Figure 5: A vehicle going over a pothole

Although the sensor captures both Y and X acceleration, only the Y acceleration is used for this experiment because the vertical movement of the vehicle can be directly mapped to the movement when the vehicle falls into a pothole (Figure 5). The horizontal component of the acceleration also changes when a vehicle goes over a pothole, but we limit the scope of this experiment to analyzing only the vertical component of the acceleration. For each of these samples, the PDA submits the current GPS coordinates to the computer. So GPS coordinates for each sample will be recorded in the data file. In addition to that, for experimental purposes, manual pothole recording needs to be carried out. When the human user feels that the vehicle is going over a pothole, he responds by giving a signal to the data logging terminal. This input is also recorded with the accelerometer readings and the GPS coordinates. Later, it can be verified whether the detection algorithm actually detects a pothole near that location.

VII. ALGORITHMS

A. ROAD BUMP DETECTION LOGIC

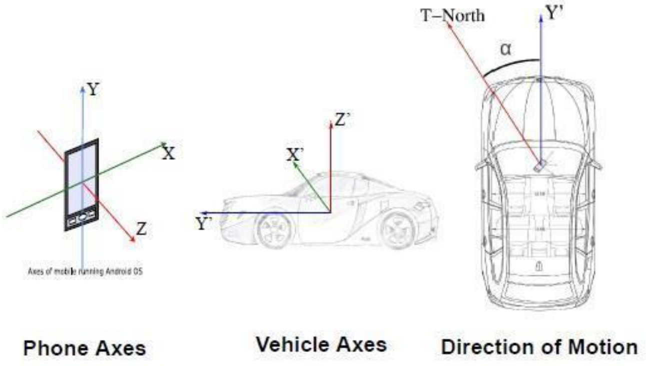


Figure 6: Axes of phone when placed in vehicle

Based on the experiment result, road bump detection logic is designed as follows.

CONDITION 1: Both of the Y-axis or running direction and Z-axis or vertical direction, 50[ms] Standard deviation is large.

CONDITION 2: These sections are appeared with wheelbase time. Here, each variable is defined as follows. A recording order number is defined 'i'. An acceleration data are defined X(i), Y(i), Z(i) for each axis. For Y-axis or running direction and Z-axis or vertical direction, 50[ms] standard deviation is defined SDy(i), SDz(i). For the condition 1, simultaneity index is defined SDyz(i), and it is calculated by equation 1.

SDyz(i) = SDy(i) * SDz(i) --------------------- (equation 1)

Cycle number of wheelbase time is defined Nw. For the condition 2, Bump Index is defined Byz(i), and it is calculated
by equation 2.

Byz(i) = SDyz(i) * SDyz(i + Nw) -------------(equation 2)

Nw is related with vehicle speed. Vehicle speed is defined V[m/s]. Wheelbase is defined Lw[m].Recording cycle is defined H[Hz]. Nw is calculated by equation 3.

Nw = (*Lw/V*) * *H* ------------------------------- (equation 3)

EXPERIMENT RESULT

This logic is applied to data. The 50[ms] standard deviation of Z-axis acceleration or vertical direction *SDz(i)* is drawn in Fig. Simultaneity index *SDyz(i)* is drawn in Fig.7. Bump index *Byz(i)* is drawn below:

STANDARD DEVIATION

$$\sigma = \sqrt{\frac{\sum(x-\overline{x})^2}{N}}$$

$where$

$\sigma = the\, standard\, deviation$

$x = each\, value\, in\, the\, population$

$\overline{x} = the\, mean\, of\, the\, values$

$N = the\, number\, of\, values\, (the\, population)$

B. LOCATION BASED DISTANCE CALCULATION

This uses the 'haversine' formula to calculate the circle distance between two points – that is, the shortest distance over the earth's surface – giving an 'as-the-crow-flies' distance between the points.

Haversine formula:

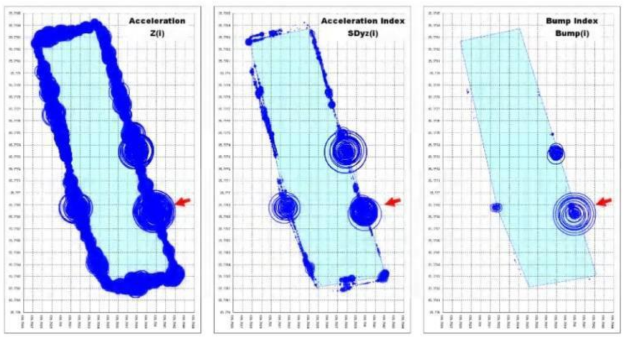
a = $\sin^2(\Delta\varphi/2) + \cos(\varphi1).\sin^2(\Delta\lambda/2).\cos(\varphi2)$

c = 2.atan2(√a, √(1−a))

d = R.c

Where φ is latitude, λ is longitude, R is earth's radius (mean radius = 6,371km)

Note: angles must be in radians to pass to the trigonometric functions.



50[ms] Standard Deviation *SDz(i)* Simultaneity Index *SDyz(i)* Bump Index *Byz(i)*

Figure 7: Deviations

VIII. CONCLUSION AND FUTURE WORK

In this paper we have made an effort to solve the problems related to driver's complexity, and focus on the road while driving. To make driving more comfortable for the driver, the smartphone is the connection between the comforts of driving. Smartphone's sensors like accelerometer, magnetometer collect the real time data, which is given to the server and then deployed as per the section made in the software. The phone is just kept in pocket or on the dashboard and then according to the position of the phone the data is collected.

Our future work would be to develop a system in a vehicle which will auto generate a particular message and send it to a list of registered numbers when met with an accident or any extreme conditions. Our future work also includes experiments with combinations of algorithms and development of self-calibration functionality. In our key ongoing and future work, we are considering also the use of many more different types of smartphones and vehicles as well as different realistic smartphone settings in our project. With these ongoing and future developments, we believe that we will be able to understand more features and aspects on the use of smartphones for the estimation of road roughness condition.

REFERENCES

[1] Viengnam DOUANGPHACHANH, Hiroyuki ONEYAMA, "Estimation of Road Roughness Condition from Smartphones under Realistic Settings" in Department of Civil and Environmental Engineering Tokyo Metropolitan University Tokyo, Japan, 2013. IEEE, 2013.

[2] Artis Mednisy, Girts Strazdinsy, Reinholds Zviedrisy, Georgijs Kanonirs, Leo Selavo, "Real Time Pothole Detection using Android Smartphones with Accelerometers" in Digital Signal Processing Laboratory Institute of Electronics and Computer Science, 2011.IEEE, 2011.

[3] Diallo Abdoulaye Kindy and Al-Sakib Khan Pathan "A Survey on SQL Injection: Vulnerabilities, attacks, and Prevention Techniques" 2011 IEEE 15th International Symposium on Consumer Electronics, pp 468-471.

[4] Girisha Durrel De Silva, Ravin Saranga Perera, Chamath Keppitiyagama, "Automated Pothole Detection Using Wireless Sensor Motes," in University of Colombo School of Computing 35 Reid Avenue, Colombo-07, Sri Lanka.

[5] R. Bajwa, R. Rajagopal, P. Varaiya, and R. Kavaler, "InPavement Wireless Sensor Network for Vehicle Classification," in The International Conference on Information Processing in Sensor Networks (IPSN 2011), 2011.

[6] H. Jol, Ground penetrating radar: Theory and applications. Elsevier Science Ltd, 2009.

[7] Shreyas Balakuntala and Sandeep Venkatesh, "An Intelligent System to Detect, Avoid and Maintain Potholes: A Graph Theoretic Approach" in M S Ramaiah Institute of Technology, Bangalore, India 560-054. 2013.

[8] Sudarshan S Rode, Shonil Vijay, Prakhar Goyal, Purushottam Kulkarni, Kavi Arya, "Pothole Detection and Warning System using Wireless Sensor Networks", Embedded Real-Time Systems Laboratory Indian Institute of Technology Bombay.