6-12-2020

Drive Health: Road Condition Detection

Peter Ferguson

Brian Walker

SANTA CLARA UNIVERSITY

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Date: June 12, 2020

I HEREBY RECOMMEND THAT THE THESIS PREPARED UNDER MY SUPERVISION BY

Peter Ferguson
Brian Walker

ENTITLED

Drive Health: Road Condition Detection

BE ACCEPTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

BACHELOR OF SCIENCE IN COMPUTER SCIENCE AND ENGINEERING

B. Dezfouli

Thesis Advisor

N. Ling

Department Chair

Drive Health: Road Condition Detection

by

Peter Ferguson
Brian Walker

Submitted in partial fulfillment of the requirements
for the degree of
Bachelor of Science in Computer Science and Engineering
School of Engineering
Santa Clara University

Santa Clara, California
June 12, 2020

Drive Health: Road Condition Detection

Peter Ferguson
Brian Walker

Department of Computer Science and Engineering
Santa Clara University
June 12, 2020

ABSTRACT

Roadways play an essential role in toady's society by contributing to economic growth and development, providing social benefits and fast routes to travel around efficiently. With more and more cars on roads, the quality of the streets is deteriorating faster than before. This decrease in road health contributes to hazards such as potholes and can cause significant damage to vehicles on the road. Currently, the process of improving and monitoring roads' health is done infrequently and is time-consuming for the government. Therefore, many road quality issues are manually reported by the people who drive on them. This requires filling out forms or making phone calls while also remembering the pothole or road hazard location. In this paper, we present *Drive Health*, an Internet of Things (IoT) system developed to monitor the health of roadways and to inform the transit authorities of poor road quality. This device also has the potential to inform the driver on how to be a safer and more efficient driver. Drive Health includes a smart sensor and performs machine learning on accelerometer data to process and analyze the device without using the cloud. If the system determines the data indicates the existence of a pothole, its location is recorded and sent to a web server.

Table of Contents

List of Figures

Chapter 1

Introduction

1.1 Motivation

Roadways are a vital part of the infrastructure of any country. They play an essential role in the economic development and growth of countries, as they allow easy access to more remote regions that would previously require much slower travel with much less cargo. Even more importantly, they allow quick and efficient travel methods across the country at speeds that were previously unheard of. Before trains, a trip across the United States could take four months or more. Once the United States' transcontinental railroad was built, people could go from New York to San Francisco in three days. This was a rather unpleasant form of travel as it did not include many stops nor was very flexible as it required sticking to the train's schedule. But once paved roads began expanding across the country, people were able to travel at their leisure and stop where they wanted to—allowing people from cities to go out and actually see the beautiful landscape that exists rather than viewing it through a boxcar window. Thus, roads have allowed travelers to see things that they previously may never have. This opportunity (as well as the chance of expansion) have caused people to build roads that cover quite a large portion of their countries.

According to a 2015 U.S. Department of Transportation report, California has over 432 thousand miles of public road lanes which 29.4 million registered cars drive an annual 335 million miles on [1]. These numbers have only increased since then, causing the roads to be under more use and thus faster deterioration. The deterioration of road quality causes dangerous road hazards such as potholes and cracks which are exacerbated by weather conditions such as rain, snow, and ice. A study conducted by AAA shows that potholes cause $3 billion of damage to American cars every year [2]. That amount only grows when considered worldwide, and can have more dire effects on the people and local economies in countries where cars and mechanics are in short supply.

To check the roadway conditions, the local transit authority has to collect data on road health periodically, but these projects are difficult for many cities and nearly impossible for most. If the city of San Francisco for instance tried to survey its 1,200 miles of road, it would need to hire a large team of people to complete the survey in a reasonable amount of time. Multiple people have walked every street in San Francisco, and in one particular case, it took a single man eight years to complete [3]. Although it would take a much shorter time, it would still be a quite large project for the city of San Francisco to undertake. Once the roads are mapped, the city then needs to shut each down for repairs which is something that San Francisco can hardly afford to do with its already limited street space and the large number of people that commute there daily.

People have been working on solutions for this problem, but most of the existing solutions depend on manual reporting of potholes or other hazards to the city. These solutions provide the city with valuable information, but are time-consuming, manual, and not always accurate. They are not real time reports and rely on people remembering where the hazard was located, as hazards are not being reported during a drive but after. If people need to choose between swerving around the same pothole that they have been for years, or taking a step to get it fixed, most people continue to swerve around it purely out of habit. A better solution is to automate the detection and reporting of these hazards to the city, as it would involve less human interaction and error. Internet of things provides an opportunity for this. A city employee could simply put Drive Health's Sensing Unit in their car, plug it into their auxiliary power outlet (commonly known as a cigarette lighter) for power and leave it there. The only inconvenience being that the power outlet is commonly used to charge phones or other small electronics while driving. To mitigate this, Drive Health's Sensing Units can be powered by a double USB port system as it has very low power consumption.

1.2 Related Work

Several methods already exist to monitor roadways and report the issues that arise. Most of the reporting comes from people noticing or experiencing a hazard and reporting it to their local public works administration. If they call the city, as most do, they get redirected to the county. This translates to more phone calls and thus more time to forget the location of the hazard, or giving up altogether.

To solve some of these issues, a British based company called Mobilized Construction operating in London and Bristol, England; Cardiff, Wales; Kenya; and Uganda is placing smart sensors in vehicles in an attempt to monitor and improve the quality of their roads.[4]. The smart sensors collect and transfer data automatically without the need for interaction from the driver. The data collected is sent to the cloud, where the company performs data analytics and AI on it. They then use the results to assist in monitoring and assessing the roads where the data was collected. With these findings, they hope that the digitized road networks will accelerate data-driven decision making for road work.

In rural areas, the communities in Kenya and Uganda utilize a smartphone application to gather the necessary data [5]. The app makes use of the phone's built-in accelerometer, GPS, and camera to identify hazards on the road. The phone needs to be securely mounted in the car to follow the car's movements and experience everything that the car does. If the smartphone is not mounted firmly, the data that the accelerometer gathers may include false information from the vehicle turning or stopping that would otherwise not appear. The data collected from the different sensors on the phone creates a topographical map of the road. This map is then sent to a platform to make repair schedules, deploy individuals to fix the detected road hazards, and monitor road performance [6].

1.3 Our Solution

This paper presents a low-cost, modular device that can evaluate how the user drives and reports road hazards. This device is easy to replicate and use as it includes only off-the-shelf components. These off-the-shelf components offer a lower cost while retaining the accuracy and reliability needed to collect the necessary data. Due to the components being off-the-shelf, they are easy to replace if a part of the hardware is damaged. To further decrease the cost and allow smooth reproduction, machine learning classification is performed locally, rather than on a cloud platform. Therefore, the device only sends the GPS location of the data point that was classified as a pothole to a web server for data basing and visualization. The device only sends the location and does not include a timestamp to reduce the risk of the data being used to track a person's whereabouts.

Chapter 2

Project Requirements

Because our system is relatively minimalistic, there are very few necessary requirements for us to build it. However, that makes the small number of requirements we have also extremely important. Without correctly identifying and fulfilling these requirements, our system would not function.

2.1 Critical Functional Requirements

These requirements are necessary for the functionality of our system, and without them the system would not work.

- Machine learning to detect potholes
- Online database to store locations of probable potholes and their probability
- Web app to view stored locations and probability
- Accurate GPS to provide location within 10 meters
- Accelerometer that can detect both large and small potholes

2.2 Suggested Requirements

These requirements are not necessary for the most important features of the system (detecting potholes). However, they would make the system much more useful beyond its basic functionality.

- Power efficiency
- Machine learning Machine learning user's driving safety and efficiency
- Integration with driving apps to report the location of potholes to other users without the device

Chapter 3

Use Cases

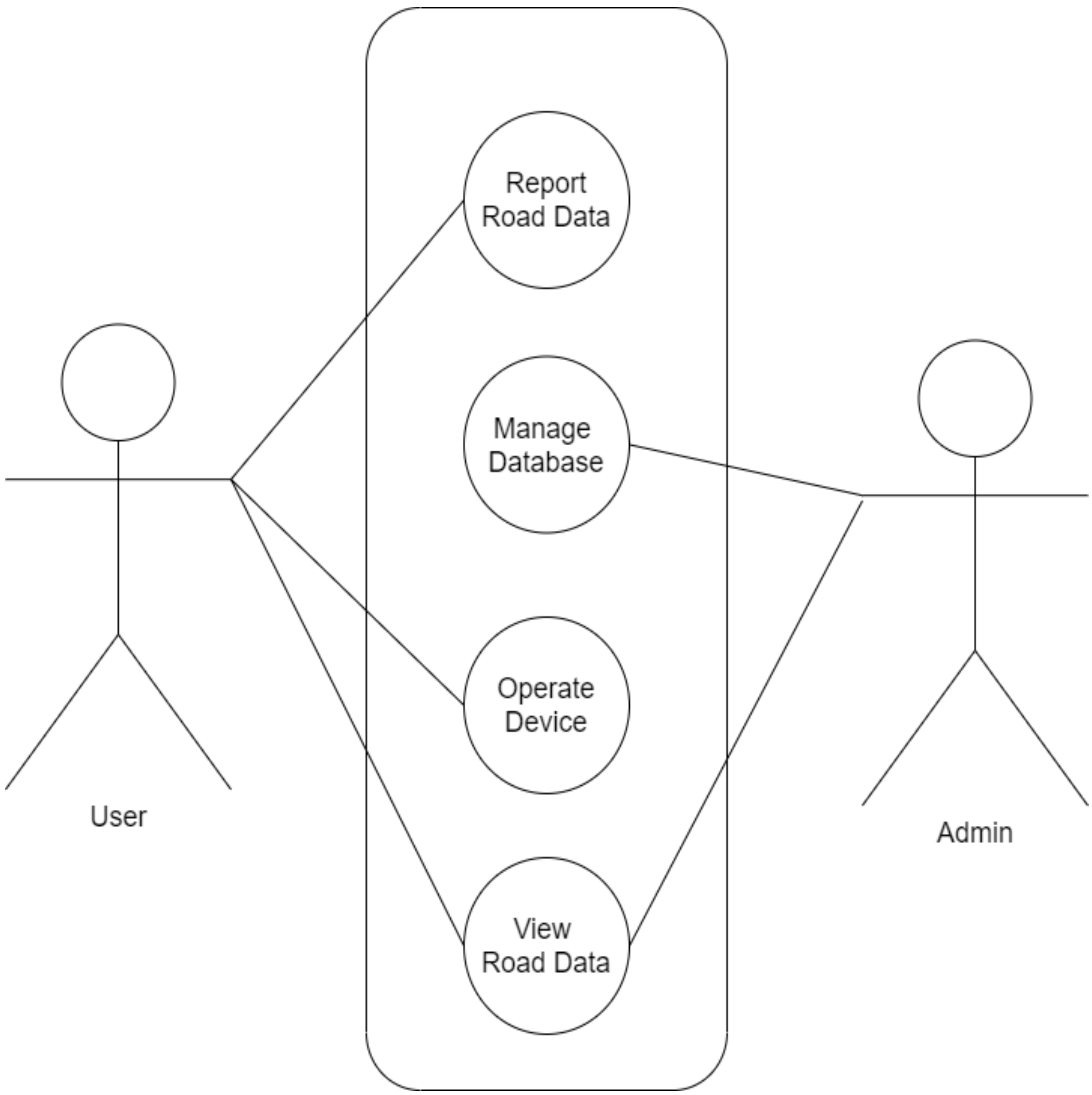

Figure 3.1: Use Case Diagram

3.1 Report Road Data

- **Goal**: User will report data the road data that the device gathers.
- **Actors**: Users
- **Pre-condition**: User hits a pothole and wants to upload the accelerometer data and location
- **Post-condition**: Data is uploaded to the database
- **Exceptions**: None

3.2 Manage database

- **Goal**: Administrator is able to remove information from the database as needed.
- **Actors**: Administrator
- **Pre-condition**: Data is uploaded to the database.
- **Post-condition**: Administrator leaves the data where it is or removes it as needed
- **Exceptions**: None

3.3 Operate Device

- **Goal**: User operates device by placing it in their car and connecting to the internet
- **Actors**: Users
- **Pre-condition**: User has device and car
- **Post-condition**: Device is connected to the internet
- **Exceptions**: None

3.4 View Road Data

- **Goal**: User and administrator operate a webpage
- **Actors**: Users and administrators.
- **Pre-condition**: User/administrator has access to the internet

- **Post-condition**: User/administrator operates webpage and is able to view the location of the potholes that have been detected
- **Exceptions**: None

Chapter 4

User Interface

A very simple and functional UI for the web interface was designed that displays the necessary information. The information that is displayed is the longitude and latitude of a detected pothole that has been reported by the device.

Drive Health: Road Data Potholes

ID	Longitude	Latitude
7	-25.67	22.643
6	5.67	-22.643
5	104.67	-42.643
4	10.2	40.6
3	20.67	-32.643
2	37.3428	-121.936

Figure 4.1: User Interface for Website

Chapter 5

Solution

As aforementioned, the system is split into two main components: the web server and database side and the Sensing Units. In this section the hardware and software used to create the Sensing Units, and the software used to implement the web server are discussed, as well as the reasoning behind the choices. The machine algorithm and its data set and model used for classifying the road data will be covered in the Sensing Unit software section.

The two different Sensing Unit designs explored and prototyped are depicted in Figures 5.1 and 5.2.

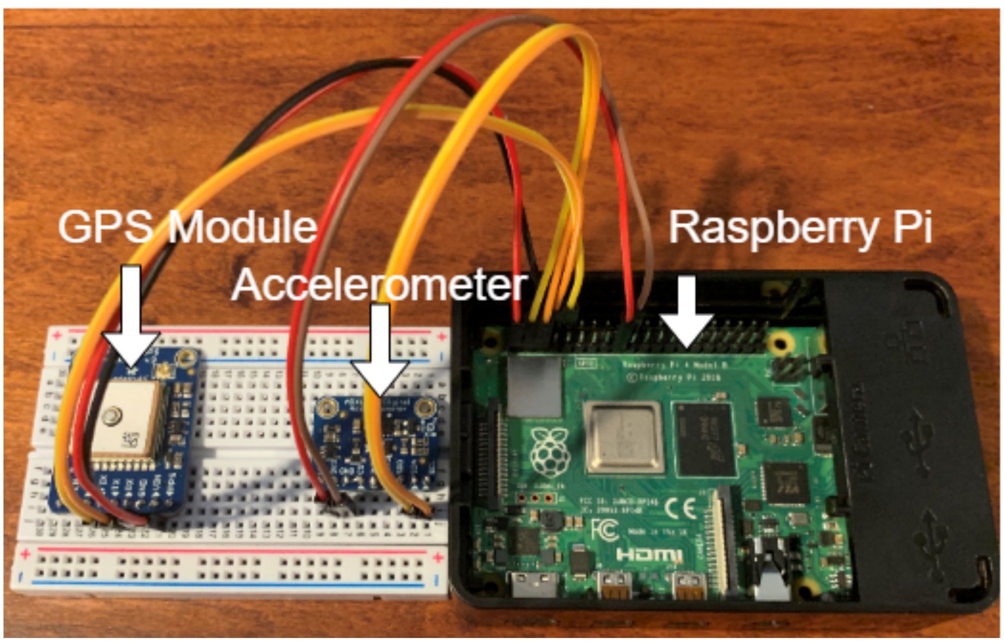

Figure 5.1: Sensing Unit without Arduino

5.0.1 Sensing Unit Hardware

Raspberry Pi

The Raspberry Pi was chosen because, at its core, it is simply a small and easily powered Linux computer. It can be powered directly from a car's auxiliary power outlet, making it extremely simple to use. It is able to communicate

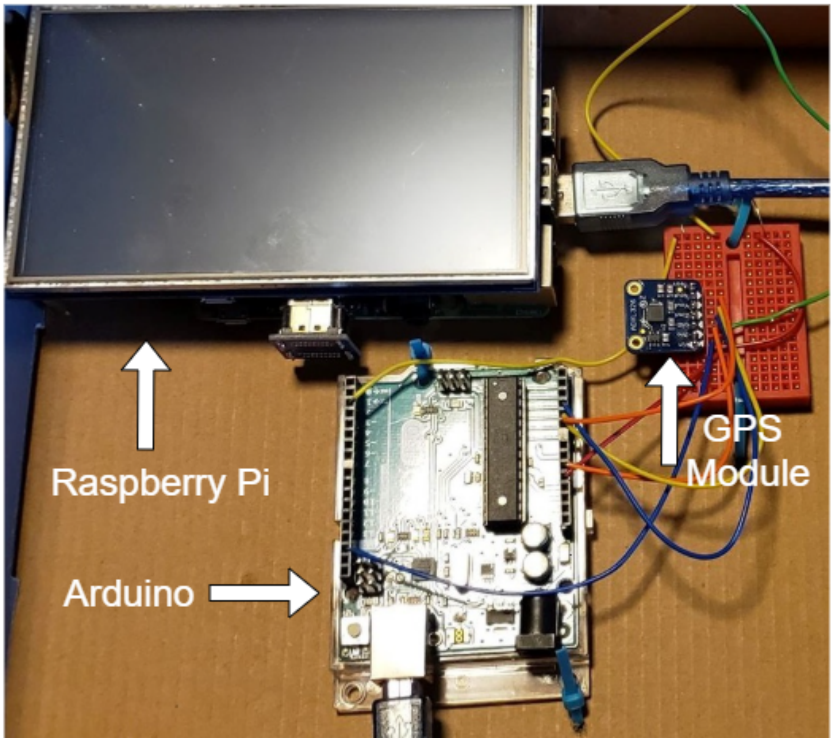

Figure 5.2: Sensing Unit with Arduino

with the Arduino through the USB/Serial port, making reading the data extremely easy.

Arduino

The Arduino was chosen for its simplicity as well as available physical modules that can be bought cheaply online. It can be powered through the Raspberry Pi, because its power requirements are much less than those of the Raspberry Pi. In the first prototype, it was used to operate the accelerometer and filter out specific unnecessary data so that the machine learning algorithm is not continually interrupted. However, after more testing was performed, it was determined that a constant data stream from the sensor was not as large a strain on the Raspberry Pi as expected. Hence, the design was simplified to eliminate the Arduino and connect the accelerometer directly to the Raspberry Pi.

Accelerometer

To ensure the reliability and generality of the design, two different models of accelerometers were used in the Sensing Units: ADXL345[7] and ADXL326[8]. Both accelerometers were found to successfully communicate with the Arduino and the Raspberry Pi, and have flexible working ranges from $\pm 2g$ to $\pm 16g$.

GPS

The GPS chosen was an Adafruit Ultimate GPS. This sensor has the ability to track 22 satellites on 66 channels [9] as well as to update its location at 10Hz. This allows the Sensing Unit to location stamp the potholes as they are being

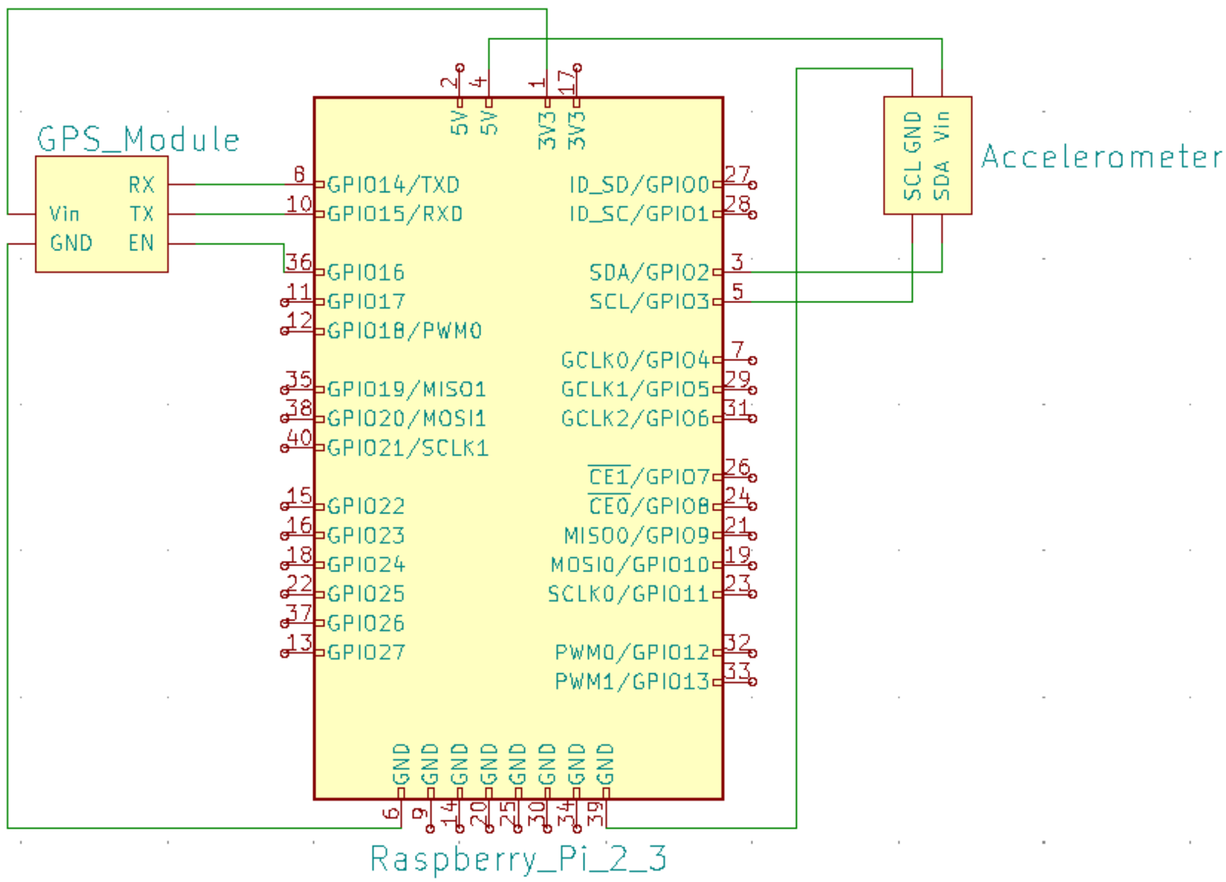

Figure 5.3: Sensing Unit Schematic without Ardunio

recorded.

Figure 5.3 shows the schematic for the finalized Sensing Unit that includes a Raspberry Pi, an accelerometer, and GPS module. The Raspberry Pi uses I^2C to communicate with the accelerometer and a Serial Port Interface (SPI) to communicate with the GPS module.

5.0.2 Sensing Unit Software

Sensing Unit

The Sensing Unit was programmed using Python and the Arduino coding language (a simplified form of C/C++). Python was chosen for ease of use on the Raspberry Pi, and to use libraries already written for the chosen sensors. In the Sensing Unit's first prototype, the serial connection from the Raspberry Pi to the Arduino was extremely easy to monitor using Python.

Machine Learning

Machine learning was used to determine whether or not the data coming from the accelerometer sensor was a pothole. For this problem, a binary classification method was chosen as the requirements were to classify a data point as either a pothole or a ”good” road. Once the binary classification method was decided on, a couple of different algorithms were tested to see what produced the best result based on the data set created. The algorithms tested were Support Vector Machine (SVM), K-Nearest Neighbor (KNN), and Logistic Regression. The data that the models were trained on was gathered using the first prototype of the Sensing Unit. The data set has a total of 430 samples, which consists of 231 potholes and 199 ”good” road data points. Each data point has three features; the X, Y, and Z acceleration data coming from the accelerometer, with the target being whether the data is a pothole (1) or not (0). When training the different models based on the mentioned algorithms, the data set was split into 80% for training the model and the remaining 20% was used for testing the trained model. All three algorithms trained a model in a matter of milliseconds and produced similar accuracy, which is probably due to the small amount of features and lack of complexity of the data set.

To verify and check the accuracy of the model, the stratified k-fold method was used for cross-validation. This method shuffles the data, then splits the data into n folds. Each fold contains approximately the same percentage of each target class as the complete data set. The folded data is then tested and trained on; in this test, 10 folds were used. In the end, the SVM algorithm was used to train the model using the gathered data set. The SVM method was chosen as the algorithm to train the model as it best classifies any outlier data points that may come in from the accelerometer. SVM works by finding the optimal hyper-plane that creates the maximum margin between the two classes. The trained model with the hyperplane can be seen in Figure 5.4. The red squares represent the pothole data points, while the blue dots represent the good road or non-pothole data points. Once the model was trained, it was deployed onto the Raspberry Pi for real-time on-board classification. The algorithms and cross-validation used were implemented by Scikit-learn [10].

5.0.3 Web Server Software

PHP is used in the backend to communicate between the Raspberry Pi and MySQL database in order to store the data received from the Sensing Units into the database. The web interface uses a mixture of HTML, CSS, and JavaScript to display the pothole information stored in the database.

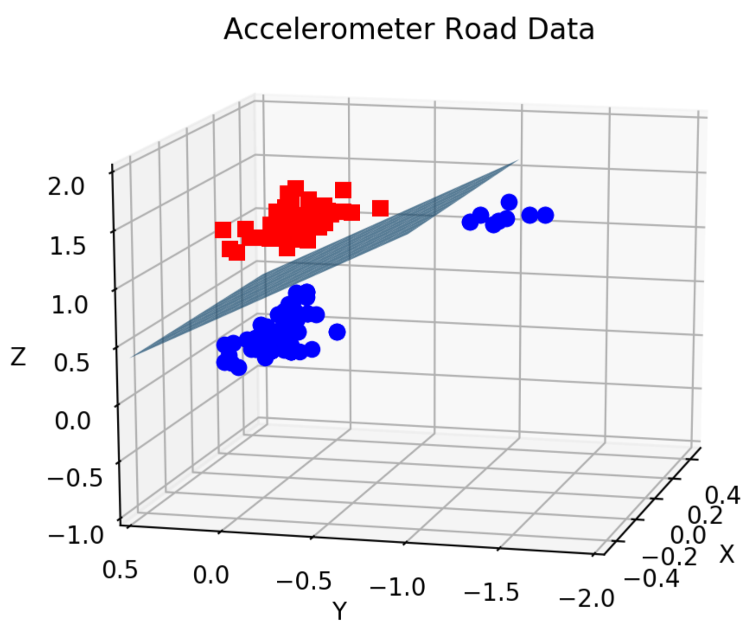

Figure 5.4: SVM Result Graph

Chapter 6

Testing

Gathering the data needed to train the machine learning algorithm was done by driving around the streets surrounding Santa Clara University, having the Raspberry Pi write its gathered GPS and accelerometer data to a local CSV file which was later uploaded to the server for processing.

Once enough data points were gathered, the machine learning algorithm was trained and tested. Figure 6.1 shows the gathered data set with the green dots being good roads while the red dots representing the potholes. As mentioned before, three algorithms were tested for speed and accuracy on the data. The chosen machine learning algorithm produced the prediction model depicted in Figure 5.4 in which the red points are potholes and the blue points are ”good” road.

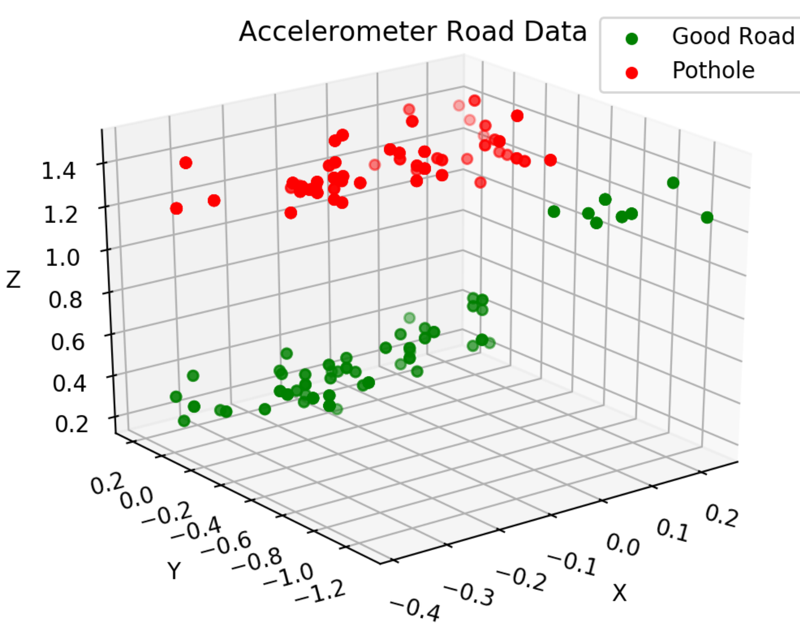

Figure 6.1: Data Set Visualization Graph

Next, the model was downloaded onto a Sensing Unit to test its ability to classify data while on the road. However, the machine learning model had a compatibility error as it was trained using a different version of the machine learning library than the one used on the Sensing Unit. Due to this, the model was re-trained on the Raspberry Pi rather than on a separate machine. This issue can be addressed in the future by ensuring all Python library versions on the Sensing Units are up to date.

The Sensing Unit was secured to the dashboard of the vehicle in order to accurately transfer the movement of the vehicle when driving through potholes. For testing purposes only, a small screen was included on the Sensing Unit that allowed the local visualization of the gathered raw data and resulting classification. Adding the screen increased the power consumption of the Sensing Unit and it required a second power source, for which an external battery was utilized. The data was displayed as an array of three numbers, where each element of the array was an acceleration value in each plane: $[x\_Accel, y\_Accel, z\_Accel]$. With the result of the machine learning classification, a 0 (good road) or a 1 (pothole), being displayed immediately following the array.

Chapter 7

Project Timeline

For this project to run smoothly and on schedule, a timeline was created to act as a reference point for this project. Due to the situation with CONVID-19, many of these goals where changed or pushed back from what is represented here, especially during the Spring quarter. By trying to follow what was planned, there was enough time to adjust to problems that arose over the development process of Drive Health Due to COVID-19 a higher quality product was unable to be produced, but nevertheless the product is functional and works as initially planned. It just lacks a few of the extraneous features that are not necessary for functionality.

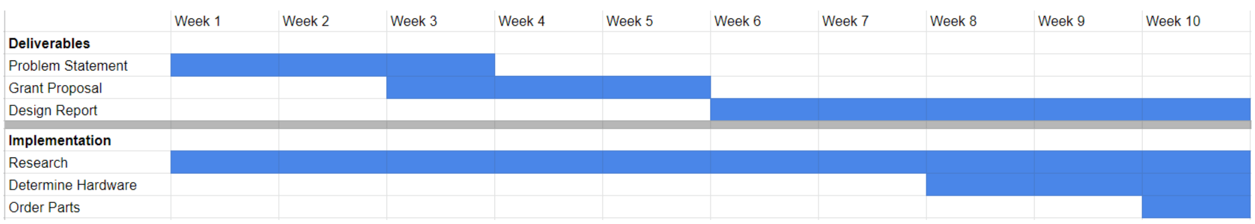

Figure 7.1: Fall Development Timeline

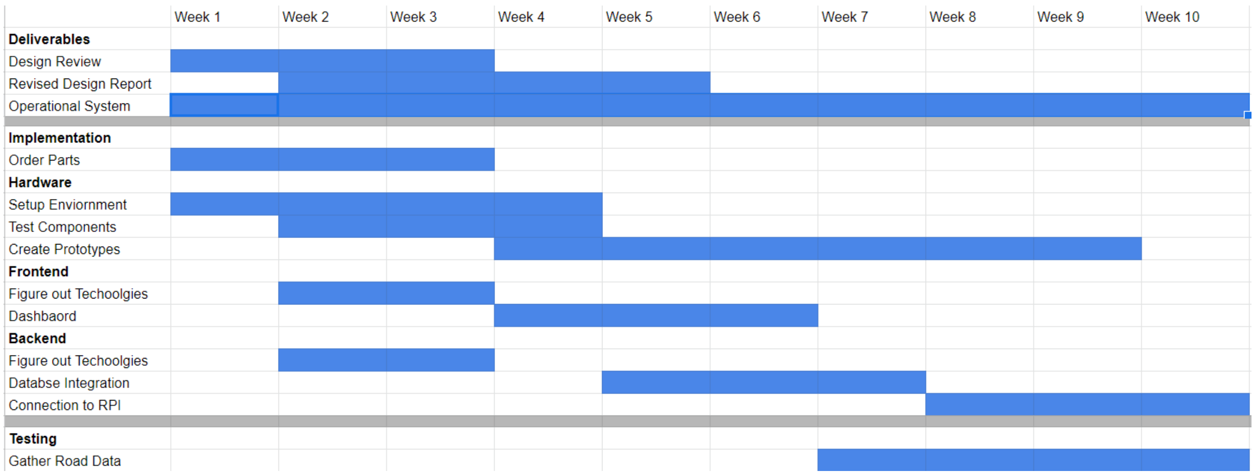

Figure 7.2: Winter Development Timeline

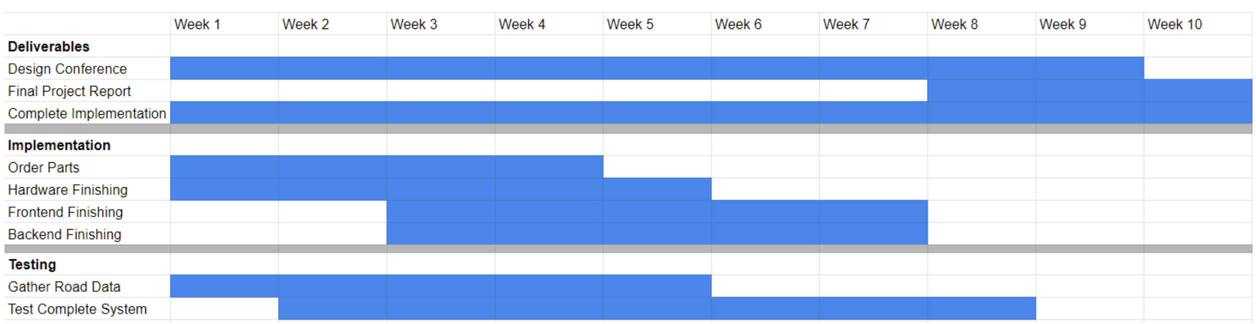

Figure 7.3: Spring Development Timeline

Chapter 8

Risk Analysis

Once work began on Drive Health, risks were identified that would hinder the completion of the project. In the table below the biggest risks and their likelihood are identified. They are also sorted by how much of an impact they would have on the project. The risk that caused the most issue was lack of training data. Because data had to be gathered by going through situations that could potentially damage the vehicles used (by driving into potholes), data gathering had to be done using only team members and their vehicles rather than gathering assistance from friends and family members. An interior risk to gathering training data was that with the screen on the device to check and see that data was being gathered, there was the potential to become distracted by the device and drive unsafely. This was mitigated by bringing a second person along to monitor the device and make sure that it was functioning correctly.

Risk	Consequences	Probability	Severity	Impact	Mitigation
Time	System or device not completed on schedule	0.5	9	4.5	Set deadlines, prioritize main features
Lack Training Data	System does not work, delays in production	0.5	7	3.5	Talk with company to get data, built enough devices and users to gather data
Bugs	System does not work, delays in production	0.99	3	2.97	Comment code, maintain consistent style, read documentation
Hardware Issues	System does not work, delays in production	0.3	5	1.5	Buy from trusted sites, purchase multiple
Project Scope	Delays in production	0.25	6	1.5	Meet required features then add fluff
Loss of dev environment	Loss of productivity	0.05	5	0.25	Take care of personal computer, take care of hardware, create backup of environment

Chapter 9

Societal Considerations

Considering the ethics behind this project is also very important. A major ethical issue that could be asked about Drive Health is whether or not it could be used to track people. Privacy and accountability were two big aspects of this project. The user's privacy is paramount as this Sensing Unit can track a user's movement while in their vehicle. To help reduce the risk of being completely tracked, the Sensing Units are kept anonymous and are not given IDs in the system, as well as not keeping track of when the information is recorded in the system. Social issues have very little impact on the project, as the intent is to only use this to mark and keep track of potholes. Political issues, much like social issues, have little to no influence on Drive Health as it is only related to the condition of roads. Economic considerations can apply, as there is a cost to create the Sensing Units. This cost was kept as low as possible over the course of the project and could potentially be lowered more if a cheaper version of the Raspberry Pi was used. However, this would impact one key feature of the Sensing Units by potentially not allowing machine learning to be done on the device and instead the machine learning would need to be moved elsewhere. This secondary location (a cloud server) would cost significantly more than a singular device as it would need to handle information from many devices at once. Health and safety is one of the main features, as keeping potholes out of roads would mean fewer potential injuries and damage to vehicles (which can cause issues in the future). This this system a city can better keep track of the potholes in their roads and the people are able to have an easier time to get the attention of the city through reporting this data by driving on the roads. Manufacturability was not a large consideration. Sensing Units could feasibly be manufactured, but the desire was to make something that people at home could easily get the parts for and assemble over the course of an afternoon. Sustainability in the scope of Drive Health was brought into play by allowing cities a resource to better maintain their roads with. This means less repairs done on cars, less goods consumed to do the repairs (metal used to manufacture suspension parts), and also a longer overall use time for peoples cars. Sensing Units have little environmental impact. It doesn't produce any waste and uses very few valuable resources in the form of the parts for the internal computers. The usability threshold for Drive Health is fairly low. The most intensive thing is being able to install the operating system onto the Raspberry Pi, which would require a laptop and being able to use a USB flash drive. From an end user standpoint, using the final device is extremely easy and only requires plugging in and pressing a button. Lifelong learning applies to Drive Health as it introduced the team to new aspects of computer engineering in web design. Implementing the website and server to store data gave the developers more of an insight into self education and the process behind learning a new product that isn't being explained to us in a classroom. Compassion played a part through Drive Health attempting to make it possible for cities to fix the roads that they built and promise that they will maintain. This responsibility is sometimes difficult for cities to uphold, and as such the deteriorating roads can cause damage and financial hardships to people. We wish to make an impact and make it so the potential for damage is not there in the first place rather than the onus being on drivers to avoid damaged parts of the road.

Chapter 10

Conclusion

Drive Health attempts to assist cities in solving a challenging problem. Potholes in densely populated cities cause a lot of damage to vehicles. The transit authorities have difficulty fixing these potholes because they can be hard to locate and repairing them requires the shut down of busy roads.

The solution is to create an automated system for identifying potholes, and IoT lends itself to this task well. Since the system's Sensing Units need to be quick to set up and easy to use, they only require the loading of the essential libraries and code before deployment. The hardware components were chosen to be readily available off the shelf in order to ease repairs should they become necessary. SVM was chosen as the machine learning algorithm for classifying the data, because it is more effective at identifying outlying data points than other examined algorithms. After the software was deployed onto the Sensing Unit, gathering data and tracking down the remaining issues was done. Once the bugs were found and removed, final testing was done in order to confirm the entire system worked together.

Dive Health will help reduce the impact that potholes have on people and the damage they cause to motor vehicles.

10.1 Future Work

The Sensing Unit's software can be expanded to evaluate the user's driving. The Progressive Insurance company is currently using a device called the "Progressive Snapshot" [11] which records data on how the user drives. This data is reported back to the company for evaluation and the individual adjustment of the customer's insurance policy pricing. Progressive Snapshot plugs into the On-Board Diagnostics (OBD) port, commonly used by mechanics to get error codes for diagnosing purposes. The OBD port outputs a lot of data, such as the current speed of the car, any error codes that the car's electronic control unit (ECU) is outputting, even the vehicle identification number (VIN). Progressive Snapshot gets total distance driven from the OBD port and has an onboard accelerometer to detect if the user brakes too hard. The Progressive Insurance Company could potentially use the Progressive Snapshot for the same purpose as Drive Health. However, their device is not powerful enough to run the machine learning algorithms necessary. Drive Health's Sensing Units can easily be extended to collect similar data and evaluate driving safety.

To extend the Sensing Units, a backup power supply could be added in order to continue data gathering even if the connection to the auxiliary power outlet is lost.

A program that makes calibration of the accelerometer more efficient than what is currently implemented could also be added. As it stands, calibration is done on the accelerometer by rotating the Sensing Unit and calibrating all axes.

Additions to the machine learning program could be done to detect the severity of the potholes. Currently, the machine learning only indicates the existence of a pothole and not the pothole's depth for instance. This would then allow the transit authority to rank the potholes by severity and attend to the worst ones first.

Another aspect is to improve the design of the website and database in order to allow for a more user friendly visualization of the results. Currently, the results are simple latitude and longitude values stored in a database table which is displayed directly onto the web page. Integration with a world map system would allow the locations to be displayed onto a map for easier viewing.

Linking the Sensing Unit to a smartphone using Bluetooth or a USB cable could enable the real-time communication of the reading to the server and eliminating the need for uploading the results after returning to a WiFi connection. This smartphone connection could also allow integration into apps like Waze, Google Maps, or Apple Maps to potentially provide the location of potholes to other users of the app. Currently Waze allows community reporting, so people know if there is a highway patrolman on the freeway, or if there are road hazards. However, these have to all be manually reported. A Waze-Drive Health integration could for instance automate the reporting. This would increase the likelihood that people would see locations where the road condition is bad enough to cause damage to their cars so they could route themselves around anything too severe.

Bibliography

[1] U.S. Department of Transportation/Federal Highway Administration, "Highway statistics series," Available at https://www.fhwa.dot.gov/policyinformation/statistics/abstracts/2015/, Jul 2019.

[2] AAA, "Pothole damage costs u.s. drivers $3 billion annually."

[3] T. Graham, "Tom graham walks every street in san francisco," Available at https://www.sfgate.com/entertainment/article/Tom-Graham-walks-every-street-in-San-Francisco-3182272.php.

[4] Mobilized Construction, "Road asset management platform," Available at https://www.mobilizedconstruction.com/.

[5] R. Goodier, "Fixing the world's rural roads with a shovel and a phone app," Available at https://www.engineeringforchange.org/news/fixing-the-worlds-rural-roads-with-a-shovel-and-a-phone-app/, Dec 2017.

[6] T. Jackson, "Mobilized construction pilots new way of fixing africa's roads," Available at https://disrupt-africa.com/2016/10/mobilized-construction-pilots-new-way-of-fixing-africas-roads/, Oct 2016.

[7] Adafruit. Adxl345 digital accelerometer. Available at https://learn.adafruit.com/adxl345-digital-accelerometer.

[8] ——. Adxl326 - 5v ready triple-axis accelerometer. Available at https://www.adafruit.com/product/1018.

[9] ——. Adafruit ultimate gps breakout. Available at https://www.adafruit.com/product/746.

[10] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[11] The Progressive Corporation, "Progressive snapshot," Available at https://www.progressive.com/auto/discounts/snapshot/.

Appendix A

Source Code

The source code and data set gathered for this project is available at `https://github.com/Drive-Health/Drive_Health`.