**Boston University**

**Electrical & Computer Engineering**

**EC464 Senior Design 2**

**Final Testing Report**

**EZRider**

**by**

**Team 27**

**Team Members:**

Aymane El Jerari

Luke Bacopoulos

Jason Wang

Haoming Yi

Balaji Sathyanarayanan

**Equipment and Setup**

We purchased 5 vehicle dashboard mounts for accelerometer data collection. One of the dashboard mounts was placed on the table for testing purposes. A smartphone was attached to the mount to simulate the capabilities of our Flutter application.

Our application collects accelerometer data and GPS data along with capturing images. The data collection process is initiated by the press of a button (‘Record’). The user clicks on the ‘Stop Recording’ button, stores the data locally, and then uploads the data to our Firebase storage.

The equipment and setup are divided into two parts: the smartphone will be placed in its designated dashboard mount with the application opened; a laptop displaying our Firebase storage and the Python notebooks and scripts implementing the Z peak algorithm with iterative threshold selection method.

The application displayed the accelerometer data with the help of the smartphone’s built-in accelerometer. The data was stable (no observable numerical variations in the x, y, and z axes). The device was subjected to manual forces (shaking, pushing, pulling, etc.) while being fixed on the mount. Consequently, we could observe new accelerometer values on the application. This situation is representative of what would occur when the vehicle goes over a pothole on the road. The smartphone collects accelerometer data after the application user approves the data collection process. The data is uploaded to Firebase storage. We used the Z Peak algorithm to predict rough road locations using accelerometer data points. The threshold value for the Z Peak algorithm was decided using the iterative selection threshold method. The Z Peak algorithm identifies rough road locations by checking if the x direction accelerometer values exceed the threshold value or not. The iterative selection method for finding the threshold is based on identifying peak points (maxima) in the x-axis accelerometer values and then working on it. The predicted rough road locations were saved as a dataframe and uploaded to Firebase storage as a json file.

**Detailed Measurements taken**

For the accelerometer and camera data, we measured the success of our application by looking at two factors: if we could collect data automatically using our app, and if data could be successfully uploaded and subsequently downloaded from. We set up an iPhone in a rigid mount, and tested out pressing “start recording”, then “stop recording”, then both “delete data” and “upload data”. Upon checking the Firebase storage, it showed that the app had acted correctly, only saving one json file of data taken in the interval where we had pressed start, stop, and upload. The next step was ensuring the data could be downloaded locally to any computer to be trained, so we took 10 measurements of accelerometer and image data and ran the python script we wrote on two computers. Both computers showed 10 downloaded json files and corresponding images with the same data sets that were in the corresponding files in the bucket.

The rough road locations were predicted using the Z Peak algorithm with the iterative threshold selection method. The predicted locations were displayed on a map accessible through our app. Since it is an unlabeled dataset (unsupervised learning), we do not have a mathematical way to test the accuracy of our model. We visited the predicted rough road locations in person and confirmed the results.

We did not use the deep learning models for the final testing session. The Inception V3 image classification model will be inferenced on the images collected through our application. This will be completed before ECE day. We trained 2 neural networks (image segmentation, Inception V3 image classification) on image datasets found online. We tested the neural networks by analyzing its accuracy on the test data. For example, the Inception V3 net gave 91% accuracy on the test dataset with binary cross entropy loss. The datasets used for training were shown to the graders. A few samples from the dataset (positive and negative samples) were produced and the characteristics of the dataset (number of samples, lighting, size of images, etc.) were highlighted.

We also demonstrated the map functionality with the markers from collected rough roads. Many of the rough roads we have found thus far were around BU’s campus, so it was easy to view and validate them. The map displays the markers of rough roads with enough accuracy to pinpoint their exact location while driving.

**Conclusions**

Our application was able to successfully collect accelerometer and image data and subsequently upload it to our Firebase storage. The data is downloaded from the bucket to any local computer by running a simple python script. We demonstrated extended capabilities (“start recording”, “delete data”, “display map”) of our application and showed our clean user interface with a seamless and simple data collection feature.

The Z Peak algorithm successfully predicts the locations of rough roads. The results are validated by visiting the sites of the potholes.

The datasets used for training the image segmentation and image classification models were shown. The models were successfully trained and produced more than 80% accuracy on the corresponding test datasets.