David Tran (301223841)
Ethan Jung (301225663)
Edwin Li (301223830)

CMPT 318 Report

Overview

In our project we decided to take a look at the topic of sensors, noise and walking. We used the recommended app, Physics Toolbox Sensor Suite, to provide us with measurements of our acceleration. The goal of our project was to compare to see if the velocity of someone's walking speed is related to their height.

Data Collection

We started off by measuring walking acceleration using the app in three different positions. The acceleration was recorded with the phone being held in hand, in the pocket, and on the ankle. This was done to determine which method of measuring acceleration was the most accurate. We made sure that we walked in a straight line for around 30 seconds with the phone placed sideways so that we could use the x-acceleration to measure the acceleration without needing to utilize the y and z-acceleration. This was because without a gyroscope we could not measure the orientation of the phone so it would be difficult to accommodate the other axis of acceleration.

Extracting and Transforming the Data

In order to be able to analyze the data, we had to do a few things to the data before proceeding. The data that was provided by the app was noisy so in order to work with it we needed to clean it. The method we used to clean the data was with a Butterworth Filter to remove the smaller frequency bumps. The filter parameters were set to 10 samples per cycle. Once the smoothed data was obtained we made sure not to overwrite the raw data so that we can compare the accuracy of the smoothed data to the raw data. The data was also trimmed by cutting the first and last six seconds of the walk to remove extraneous movement such as taking the phone out of the pocket and the pauses at the start and end.

est_distance	est_velocity	name	steps	steps_per_sec	version	true_distance
386.226956254	11.5553780593	david_ankle	71	2.1242856715	raw	53
295.4736028564	8.8401628428	david_ankle	56	1.675492924	smooth	53
344.6173925003	9.4376938929	david_hand	73	1.9991784198	raw	41
358.6124986033	9.8209639492	david_hand	60	1.6431603451	smooth	41
239.4716136138	6.8182795289	david_pocket	105	2.9906861489	raw	71
231.8520306194	6.60133337	david_pocket	98	2.7913070723	smooth	71
235.0029672004	5.1825552365	edwin_ankle	113	2.4921706144	raw	82
54.1926855561	1.1951193198	edwin_ankle	46	1.0145119315	smooth	82
107.7369721739	3.016405974	edwin_hand	35	0.9799255257	raw	82
98.7778586952	2.7655698602	edwin_hand	33	0.9239297813	smooth	82
254.5055203803	6.2960572046	edwin_pocket	118	2.91956355	raw	82
269.4938153013	6.6668435124	edwin_pocket	105	2.5979167182	smooth	82
45.8100803307	1.2279876781	ethan_ankle	77	2.0643985093	raw	50
53.0629888916	1.4224095669	ethan_ankle	56	1.5013807341	smooth	50
568.4730873813	16.27789959	ethan_hand	34	0.9737377209	raw	50
567.1063609435	16.2387641653	ethan_hand	30	0.859180342	smooth	50
122.7885306243	4.0600644984	ethan_pocket	77	2.5464647133	raw	66
123.7641480924	4.0923237805	ethan_pocket	73	2.4141808321	smooth	66

Figure 1: acceleration_analysis.csv

The collection of acceleration files gathered is saved as .csv files which makes them easy to use with Pandas DataFrames. Each of the acceleration recordings comes in a separate file so they were placed in a folder where they would be iterated through. The important data from each file was extracted using the process as described below.

The velocity was calculated by using the formula, $v = v_i + at$, where v_i is the previous velocity, a is the current acceleration, and t is the difference in time between the previous point and current point. The distance was calculated using a similar formula, $d = d_i + vt$, where t is the difference time, d_i is the previous distance, and v is the velocity. This was done was with the raw and smoothed data. In addition to the distance and velocity we also calculated the number of steps which was estimated by counting the peaks of the acceleration. Dividing the total steps to the total time resulted in obtaining the steps per second. The results for each set of data was saved into a file called acceleration_analysis.csv as you can see above in Figure 1.

The *est_distance* column is the cumulative distance obtained from summing the results from the distance formula above. The *est_velocity* column is obtained from taking the estimated distance and dividing it by the total time. The number of steps and the frequency of steps per second are also recorded in the columns *steps* and *steps_per_second* respectively. These values were obtained using a function that detects local peaks taken from a GitHub repository [2]. As stated earlier, the raw and smoothed data were both recorded, as shown by the *version* column. Finally, the *true_distance* column is the actual number of meters traveled by each person during the accelerometer data. This value was calculated using a tool which measured the distance between two points on Google Maps [1].

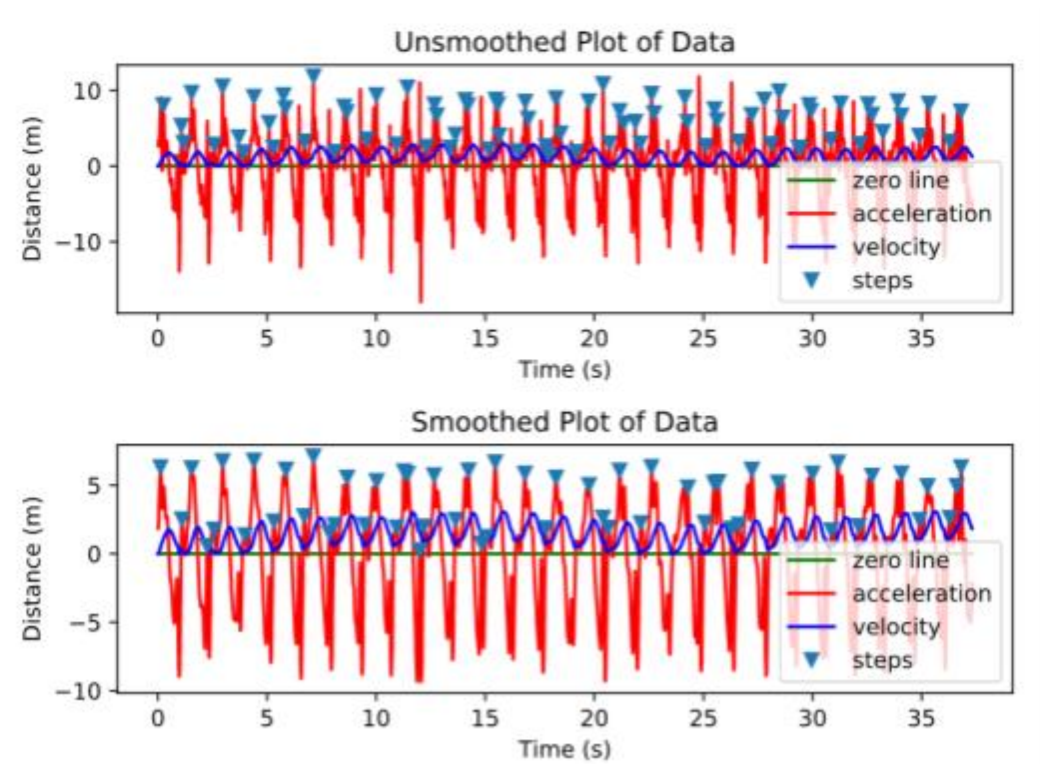

Figure 2: raw and smoothed graphs of ethan_ankle

In addition to recording the final results on the acceleration_analysis.csv, the raw and smoothed acceleration, velocity, and steps taken were plotted on a graph (Figure 2). There is one plot for each of the positions and people. Each triangle on the graph represents a single step taken. The function was set to look 20 points ahead to determine whether a candidate's acceleration peak is the actual peak which would count as a step. Afterwards, only peaks that had a positive acceleration were taken into account to discard any peaks that may have been formed from sudden shifts in acceleration when slowing down.

Finally, to visualize the accuracy of each of the phone locations, we separated the data points by the person, and then plotted the difference between the estimated distance and true distance (Figure 4, below). Once we obtained this data it was time to analyze the results and determine which was the best method to measure speed.

Data Analysis

Looking at Figure 2 above, we can see that smoothing reduces a large amount of noise by seeing that there are more evenly spaced out steps recorded and less extreme jumps in acceleration compared to the results of the raw graph data. The large peaks represent a step taken or the foot moving forward and the foot moving in the opposite direction is represented by the lower peaks.

However, some of the results from the raw data were very inaccurate, possibly due to factors such as the phone's ability to record. Even with the smoothed data we were unable to return a reasonable graph which an example would be Figure 3 below. The graph depicts the velocity constantly increasing to upwards of 15m/s which is unlikely to be accurate walking speed.

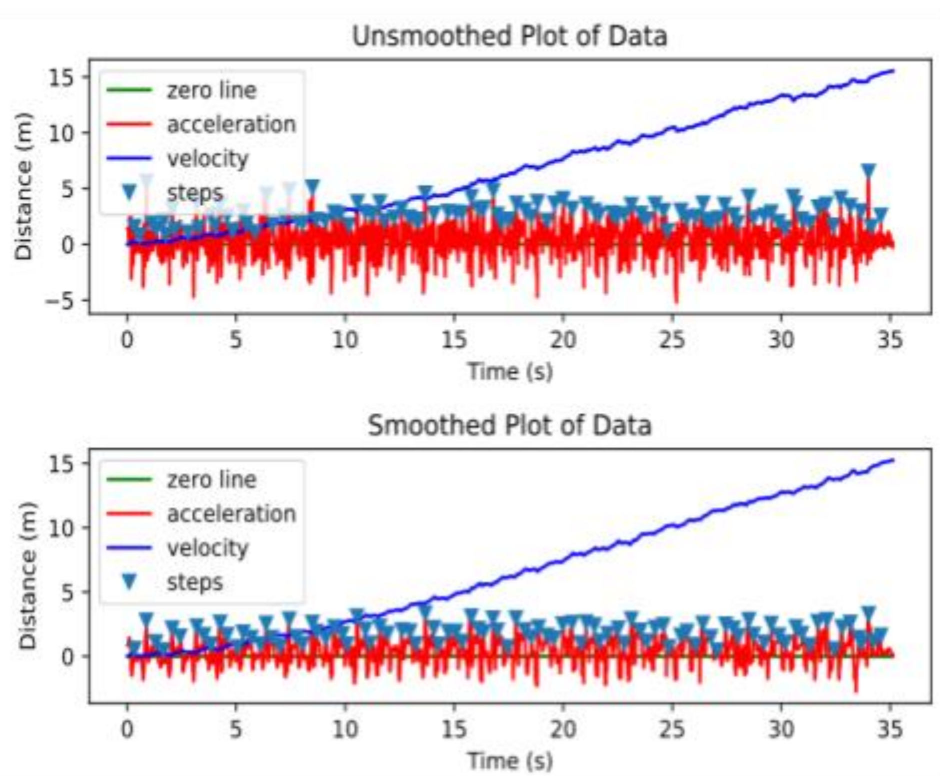

Figure 3: raw and smoothed graphs of david_pocket

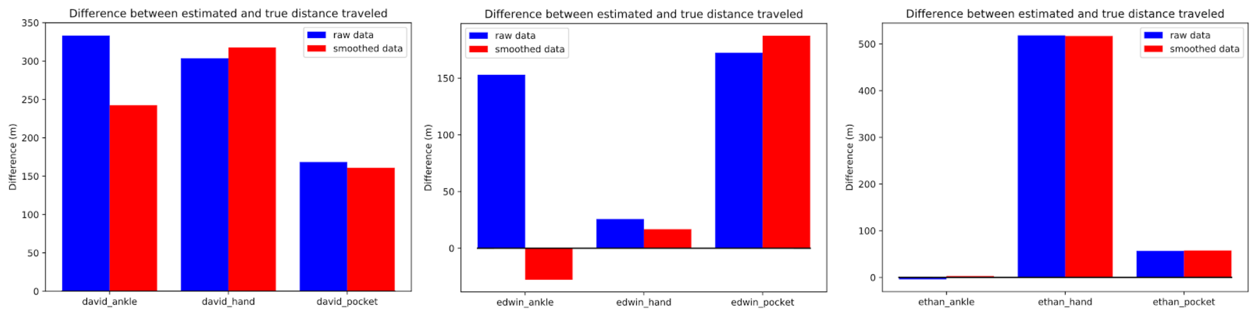

Figure 4: Differences between the estimated and true distances

From Figure 4, the filtering provided by the Butterworth Filter increases the accuracy of estimated distance in most cases, with a couple exceptions.

Determining which position the phone should be in to get the greatest accuracy, however, is more difficult. The differences between the estimated distance and true distance for David is inaccurate at all three possible locations, with the smallest difference having an estimated distance twice the true distance. Edwin has a close difference of 12m with the phone positioned in his hand, and a 28m difference with the phone on his ankle. The difference with the phone in Edwin's pocket is inaccurate even with smoothing. The ankle data is very accurate for Ethan, with a difference of 3 meters. The pocket data and hand data are both inaccurate, with the smallest difference having the estimated distance nearly twice the true distance.

With all the distances being wildly varying and no consistent results among the three we decided to take a look at the number of steps that were taken. In our csv in Figure 1 at the top, the number of steps that were taken were reasonably close to the distance travelled. Usually the number of steps were greater than the distance travelled which would makes sense since a person would normally walk less than a meter with one step.

By looking at the results of the steps, the position to match the pattern of being slightly higher than the distance travelled was the smoothed pocket data. We would proceed to use the pocket data to determine the number of steps per second with the most accuracy. With this change we had to change our initial goal of "if the velocity of someone's walking speed is related to their height" to "if the number of steps a person takes is related to their height".

Step Data Collection, Extraction, and Transformation

Data gathering is similar to the previous method of gathering but this time we had to ask participants to help. This time the data gatherers would only need to record the acceleration with the phone placed in their pockets. In addition to recording the acceleration the participants would also have to provide their heights so we can compare their steps to their height.

Transforming and extracting the data is similar to before except we only use the smoothed data and peak functions to obtain the steps per second. Once the steps per second were obtained for each of the recordings, they were plotted against their related height as seen in Figure 5 below. In addition to the plot we also plotted the linear regression that the points gave us.

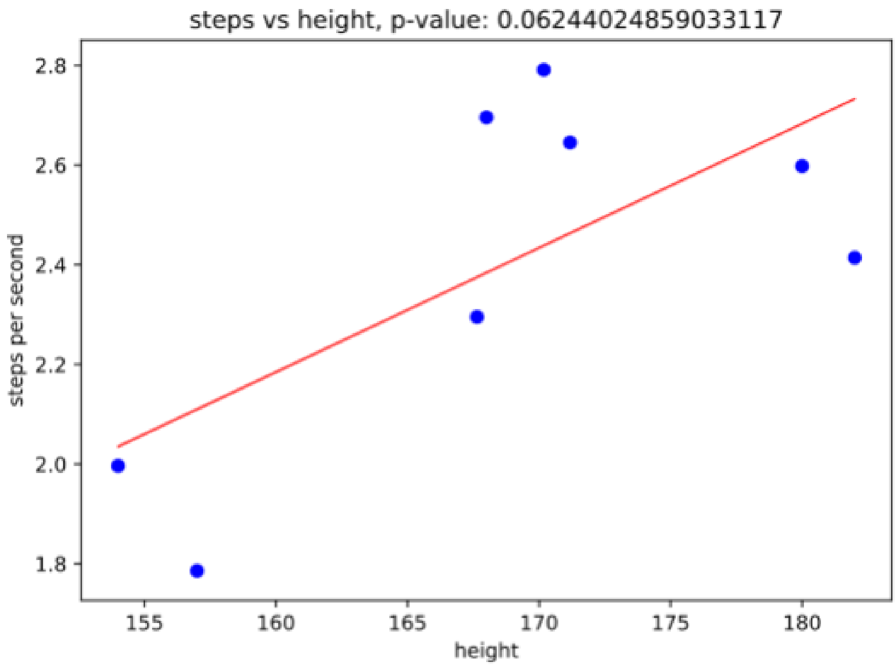

Figure 5: Comparison between steps per second and height

Step Data Analysis

To start the analysis our H_0, null hypothesis, would be "the steps per second are unrelated to the to the height" while the H_a, alternative hypothesis, would be "the steps per second and heights are related". The p-value we received from our linear regression was 0.062 which is relatively close to 0.05 but still greater so we could not reject the null hypothesis, or in other words we cannot falsify the claim that steps per second are unrelated to the height.

Even if the p-value provided us with a value that was less than 0.05, our results may not be entirely accurate. This is because we only have eight data points which is not reliable in representing the population. Although despite this the data points we received actually do appear to have some pattern since the people with lower heights generally had a lower step per second count than the taller people.

Suppose that the p-value did not get rejected and we had enough data points. If that were the case we could put it in a linear regression model. We can then use the model to predict the height of a person based on their steps per second or the other way around.

Limitations

Due to the limitations of recording accelerometer data on different phone and the noisy quality of our data, it is reasonable for large inaccuracies to exist. For example, many pockets do not allow for the phone to be placed in such a way that the x-axis is parallel with the direction of travel. In order to solve the issue of an angled phone, a gyroscope can be used simultaneously together with an accelerometer to record the orientation of the phone together with the x, y and z accelerations.

In our sensor readings, we solely used the x-acceleration due to the lack of information, and did not remove the acceleration from gravity from the y-axis. Additionally, a walking motion results in the phone device being swung in a pendulum-like motion. Due to this motion, the horizontal acceleration is moved away from the x-axis onto the y- and z-axes. The phone orientation obtained from the gyroscope can be applied to the three-dimensional acceleration to obtain a smooth horizontal acceleration graph using the formula from Maria Yousefian's thesis [3].

We lacked a good number of participants for our data as having only eight data points to use for the linear regression provided us with results that may not be accurate. The participants recording the accelerometer data had to stand still for a few seconds before and after walking which resulted in us trimming six seconds from the beginning and end of the accelerometer data, the start of the acceleration may be trimmed, resulting in a lower estimated initial velocity.

References

[1] Daftlogic.com, "Distance Calculator" [Online]. Available: https://www.daftlogic.com/projects-google-maps-distance-calculator.htm

[2] "Peak Detection in Python" [Online]. Available: https://gist.github.com/sixtenbe/1178136/d52dfaaf987c56bec20bb64d35f3fb35d39e1f80

[3] Yousefian, Maria, "Design and Implementation of a Smartphone Application for Estimating Foot Clearance during Walking" [Online]. Available: http://summit.sfu.ca/item/17204

Project Experience Summaries

David Tran, 301223841

The project was an opportunity to use the skills I have accumulated in this class. I used techniques such as data manipulation in the Pandas DataFrame to create new data that is readable and useable as well as plot it for later analysis. It was unique in this situation from standard exercises since I had collection of data instead of a single .csv file to work with. In order to use multiple .csv files, I had to extract the important details from the files so that they could be compared. The code was made in a way so that it would be entirely automated so we did not have to look at or edit the .csv file, all we would need to do is insert it to the correct folder and have the program generate an easy to read visual that could be analyzed. In addition to adding features to the code I also tasked with modularizing it so it would be easy to maintain for the rest of the group.

During the analysis portion of the project I used the data to finalize what we would use as a means of measuring the velocity. Once the method was finalized I revised the initial goal and created the null hypothesis and alternate hypothesis that would be tested.

Ethan Jung, 301225663

The project on accelerometer analysis proved to be an interesting learning experience where I had to learn how to manipulate the initial data sets into something meaningful. I recorded the initial accelerometer data set, and decided to try to extract the distance traveled. To begin, I created data_smoother.py and smoothed the x-acceleration points. After trawling through the internet, I discovered how to convert acceleration into velocity, and velocity into distance. I created sensor_analysis.py and implemented a rough copy of an acceleration to distance converter, along with a plot of the data.

After looking at the smoothed acceleration plot, I could clearly see that peaks were forming in accordance with each step I took. In an attempt to count the number of steps, I first attempted a Fourier transformation. However, that did not work out, so I resorted to testing peak detection functions from different libraries. During this, I received another data set from a teammate, who had a much noisier graph. As a result, I required a peak detection function that allowed me to set the number of points in between peaks. After a while, I settled on the current function, filtered the returned peaks so that all the peaks would be greater than 0, and pushed the code to a repository so that it could be refactored for modularity and then analyzed.

Edwin Li, 301223830

I provided the visual analysis for the comparison of data from the phone position in the pocket, hand, and ankle, against the difference between the true and estimated distance from for each data set. As well as the visual analysis of height against steps per second to see if there was a plausible relationship between the two variables. Incorporating code and data that was not mine was the experience I received from this project. Because my analysis was based on data that was already calculated by another member, I had to ensure the structure and correctness of the data before continuing with the analysis, which required extensive analysis on the code itself that was used to process the raw data.