CMPT 353 Final Project

Sensors, Noise, and Walking

By Cyrus Ahn

Introduction

In the current day, almost everyone carries a mobile phone wherever they go. These mobile phones have more functions that just communication, they are equipped with a multitude of sensors which includes the accelerometer used for this project. I gathered various people I know to collect data from their mobile phone accelerometers. Using this data I applied functions that would make the data more readable, and extract useful information. There were also some issues with the accelerometer that would limit the amount of accurate information I could extract. Here I will be explaining the process, thoughts, reasoning, and displaying the data I gathered.

Main Focus

I wanted to answer 2 main questions during this project; when given accelerometer data, can I determine which person provided this data, can I calculate average walking speed only from the accelerometer data.

Gathering data

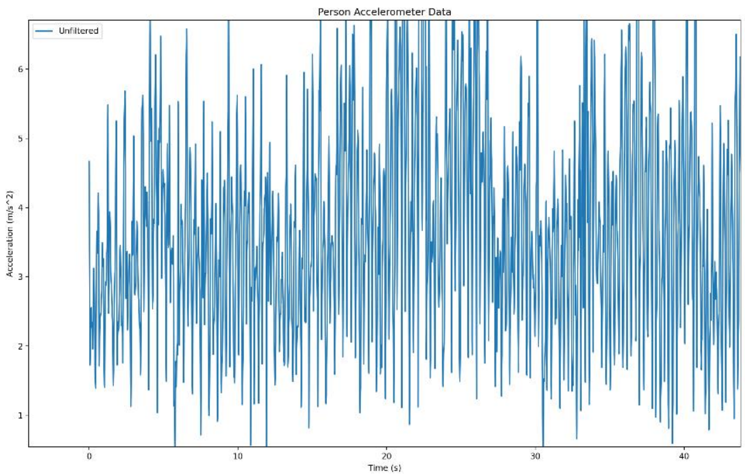
The main method of gathering data was to ask friends, friends of friends, and family. I asked each person to download the app (on both IOS and Android devices) as suggested, “Physics Toolbox Sensor Suite” and to go for a short walk while having the Linear Accelerometer function open in the app. The app has a great function where tapping a button will start recording data, then stopping the recording will automatically create a .csv file with columns time, x, y, z, and total acceleration. Next I scanned through each file to make sure the data made sense. What I looked for was consistent acceleration numbers, consistent time values, and enough data points to use. For some data sets, there were major gaps in time probably caused by not having the app open for the entire walk or the phone going into sleep mode. For example person 3 had data with consistent recordings from 0 seconds to 144.7 seconds, then had a gap where no data was recorded until 230.3 seconds. In order to

time	ax	ay	az	atotal
0.003753	0.59	-0.53	4.6	4.66
0.010559	0.96	-1.1	4.41	4.64
0.021694	1.18	-1.53	3.78	4.24
0.031619	1.22	-1.41	2.86	3.41
0.047089	1.05	-1.4	2.14	2.76
0.050753	0.83	-1.59	1	2.05
0.060741	0.61	-1.61	-0.05	1.72

Example data generated by person 1

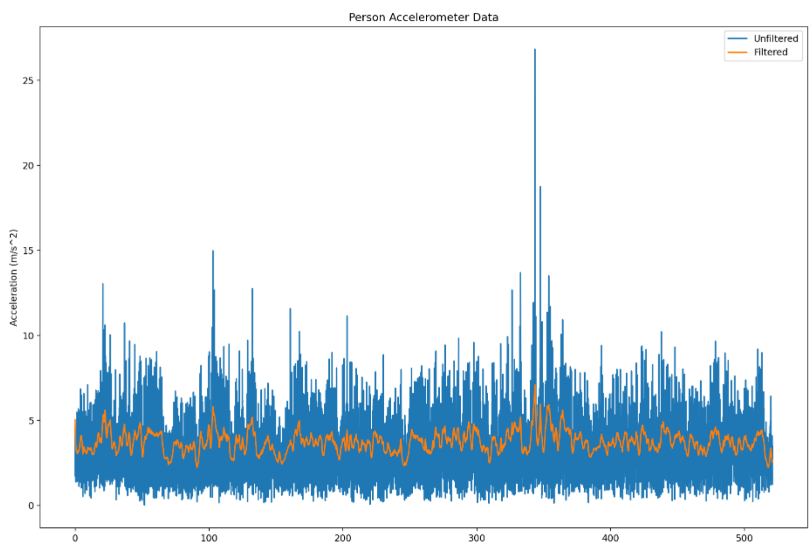
use this data set I chose to split the data set where there was the gap creating 2 separate data sets, person 3 and person 5.

Filtering



The original unfiltered data is almost unreadable. The amount of peaks and troughs that occur within 40 seconds of collecting data is uncountable. This data suggests that this person took about 100 steps in 10 seconds, something that is extremely unlikely.

As suggested, I processed this data through the Butterworth filter to try get rid of as much noise as possible. With the example provided in lecture of the Butterworth filter and some trial and error to find the best values for the filter parameters, I was able to produce a signal that reduces most of the noise. The blue line is the unfiltered signal, the orange line is filtered. The blue line has peaks of over 10 m/s^2 which is an unnatural spike in acceleration.



Analysis

As mentioned in my main focus, the main method to determine which data belonged to each person was finding the steps per minute. With this data I hoped to match the data sets from the same person (person3.csv and person5.csv). To find the total number of steps the person has taken, I counted the number of peaks in the filtered data based on a set condition. I used the signal.peaks method from the scipy library, and set the distance parameter to 3. This parameter ensures that there is at least 3 units of horizontal distance between every peak counted. I wanted to make sure that the small peaks that do not mean anything does not interfere with the larger peaks that are most likely a step made by the person. I also tried using a threshold parameter and width parameter but both had results that were either blank or had a number too big. After finding this value, I divided the number of steps by the total time spent walking to find the steps per minute.

As for my second focus point, I integrated the acceleration data using a cumulative trapezoid function from scipy. The purpose of this function is to use the non constant acceleration data to produce change in velocity at each time point. Since this function was cumulative, I shifted, subtracted, then found the change in velocity at every time point. However this data could not be used to calculate the average velocity of the person. All changes in velocity are positive numbers but that does not make sense. This would suggest that the person walking has been speeding up for their entire walk. In fact all values in the acceleration graph are positive. This resulted from using the absolute value of vector length of ax, ay, az, the data provided by the accelerometer. If we look at the data from person 1 again, the accelerometer did record some negative acceleration mostly in ay. But to calculate the vector length (atotal) the formula is $\sqrt{ax^2 + ay^2 + az^2}$ by squaring all acceleration values, we lose negative values resulting in all positive acceleration.

	time	vector_length	change_vel_cummulative	change_shift	change_in_velocity
0	0.008064	1.102446	0.000000	0.000000	0.000000
1	0.010527	1.118791	0.002735	0.000000	0.002735
2	0.012509	1.134972	0.004969	0.002735	0.002233
3	0.014344	1.150921	0.007066	0.004969	0.002097
4	0.015943	1.166569	0.008919	0.007066	0.001853
...	...	...	...	...	...
109547	544.584493	1.891043	1771.945927	1771.943908	0.002019
109548	544.585720	1.888305	1771.948246	1771.945927	0.002319
109549	544.586523	1.885833	1771.949761	1771.948246	0.001515
109550	544.587278	1.883643	1771.951184	1771.949761	0.001423
109551	544.588148	1.881747	1771.952822	1771.951184	0.001638

time	ax	ay	az	atotal
0.003753	0.59	-0.53	4.6	4.66
0.010559	0.96	-1.1	4.41	4.64
0.021694	1.18	-1.53	3.78	4.24
0.031619	1.22	-1.41	2.86	3.41
0.047089	1.05	-1.4	2.14	2.76
0.050753	0.83	-1.59	1	2.05
0.060741	0.61	-1.61	-0.05	1.72

Results and conclusions

The expected result from this project would be person 3 and person 5 have similar steps per minute, and all other data would be different and distinguishable. For context, the average person takes about 100 steps per minute. The actual result is as shows:

Data set	Steps per minute
Person 1	109.31
Person 2	106.54
Person 3	68.42
Person 4	83.21
Person 5	109.69

The actual result completely disagrees with the expected results. Person 3 and person 5 steps per minute are nowhere close to each other, in fact they are the furthest apart data point in this sample set. Person 1, person 2, and person 5, steps per minutes would suggest that they are the same person when they are all different people. A more likely explanation is person 1, 2 have similar walking speeds close to the average person while person 3 and 4 have a slower walking speed. Person 3 may have also sped up their walking speed near the end of their walk which is reflected in the faster walk speed from person 5.

The expected result from calculating average walk velocity would have been around 1.5 m/s but with the data used, it would not be possible to get this number. Having only positive acceleration makes it impossible to calculate an accurate velocity.

Limitations

A large problem I encountered was collecting accelerometer data from various people. Since I am unable to be with every participant, guide them through the steps, and ensure they keep the app open, many data sets I received were unsuitable for the project. Some had gaps in time starting from 2 seconds and occurring every 10 seconds, some participants used the wrong sensor and submitted other non accelerometer data etc. I also wanted to explore the effects of having the phone on different locations of the body like ankle, hip, and hand, but it was more difficult than anticipated to have people gather data properly in the first place.

Another issue I encountered as I stated before is using only positive acceleration data in all calculations. If I had more time, I would try to find a method to use the x, y, z values with some gps or other sensor to get more accurate acceleration data. A different method will hopefully produce acceleration data with positive and negative values so that average velocity can be calculated.

Something I should have done is gather age, gender, and height data of all participants and use that information to see similarities and differences.

Project Experience Summary:

Calculated steps per minute from accelerometer data with over 10000 points using a combination of Scipy and Pandas library

Organized and guided over 15 participants in collecting accelerometer data using both IOS and Android applications

Filtered raw accelerometer data collected from mobile phones using the Butterworth filter from scipy creating data used for different calculations

Communicated all findings, analysis and results to evaluators the includes diagrams and figures of data