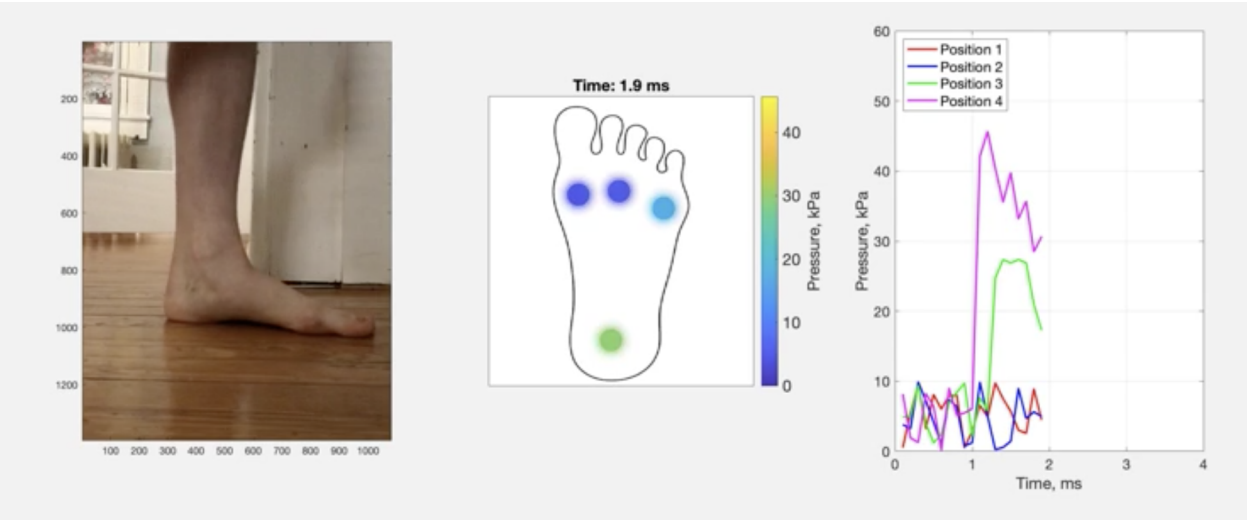
Tufts University

THE FEASIBILITY AND ACCURACY OF THE USE OF HALL EFFECT SENSORS TO MEASURE THE PRESSURE DISTRIBUTION OF THE FOOT



Jeffrey Blitt, Isobel Bohmer, Morgan Burke, Alp Ege, Olivia Martin, Joshua Harvey, Nicole Russo, and Andrew Voss

ME-0070-01 Instruments/Experiments
Professor Felix Huang
05/07/2021

Table of Contents

Executive Summary…… 2

Main Reporting

1. Introduction…… 2
2. Methods……...3
3. Results……7
4. Discussion……10
5. Impact and Conclusion……11

References……13

Appendices

Appendix A……....14

Appendix B……....15

Appendix C……16

Appendix D……17

Milestone Documentation……..19

Executive Summary

Measurement of foot pressure distribution (FPD), known as pedbarography — the study of pressure fields acting between the plantar surface of the foot and a supporting surface — is the go-to method for biomechanical analysis of gait and posture. We can employ this method in a variety of applications including sports, biomechanics, and gait biometrics. Our original motivation of determining a correlation between walking speeds and pressure distribution on the foot directed us to the investigation of how to measure and record this information using limited resources and simple technology. The resulting question can be stated as follows: is it possible to build an FPD sensor out of simple, inexpensive components? This report expands upon and investigates this research question and comments on the repeatability of results. Major topics include the construction of hardware and Hall effect sensor calibration experiments. This is in the hopes of reporting a tried and true method of measuring FPD, by establishing several recommendations for those who would like to replicate this study. Recommendations include the range of values limited by the sensors and the most appropriate material to place in between the magnet and the sensor.

1. Introduction

Originally, our motivation for this project was to explore foot pressure distribution (FPD) at various walking speeds. As college students, our primary mode of transportation is walking, especially around campus. As we began our project, we came to the realization there is no simple and cost-effective tool for measuring FPD. However, FPD can be greatly important in diagnosing biomechanical inefficiencies and guiding the development of therapeutic methods and technologies. For example, diabetic foot ulcerations result primarily from exposure to high load conditions and sensory losses on the foot surface [1]. Ko et al. [1] use a Tekscan High Resolution Floor Mat System to measure the foot pressure distribution of diabetic and non-diabetic individuals in order to suggest modified walking patterns that alleviate plantar stress and prevent foot ulcerations in patients with diabetes. Additionally, Hessert et al. [2] analyze how the pressure distribution on the foot at normal walking speed changes with age. They find that elderly subjects have lower pressure on the medial side of the foot than younger subjects, which may contribute to instability and lower propulsion during walking. These studies illustrate the importance and value of measurements of FPD over a variety of applications, but unfortunately, state-of-the-art insole pressure sensors can cost between $5,000 - $10,000 [3], which may prohibit access to this research topic. In this project, we propose the use of Hall effect sensors to measure the pressure exerted on different areas of the foot during normal walking. Hall effect sensors use the Lorentz force to measure the strength of the local magnetic field [4]. In order to relate the strength of a magnetic field to a force, we sandwich a deformable material in between a Hall effect sensor and a magnet. We develop a calibration curve for the sensor, relating the force

applied by the foot to the output voltage of the Hall effect sensor. This approach is highly desirable because individual Hall effect sensors cost less than $0.50 [5].

2. Methods

A large portion of our work was dedicated to determining the appropriate hardware needed to sense a sizable load (i.e., a human body mass). The first attempt was using a single force sensitive resistor (FSR) wired to an Arduino. The sensor had a maximum limit of a 10kg load, and real-time data visualization confirmed that the sensor maxed out with a simple pinch. We then moved on to strain gauges, similar to the ones found in bathroom scales. One issue with this sensor was the complexity of wiring and collecting data from a single strain gauge. The specific strain gauges that we bought were only operable when placed in a Wheatstone Bridge configuration, with a corresponding HX711 AD Module. While we were able to see some evidence of force measurement from this sensor, it did not provide sufficient resolution and was too bulky to be integrated into the desired format. Hall effect sensors turned out to be the best hardware option. These sensors work by measuring the voltage across a current-carrying semiconductor when exposed to a magnetic field [4]. The specific product we used was the Hall Effect Sensor - AH1815 (Non-Latching) from Sparkfun.[1] The theory underlying the operation of these sensors is outlined below.

2.1 Fundamental Theory

Hall effect sensors utilize the Lorentz force to measure the strength of a magnetic field:

$$\mathbf{F} = q\mathbf{E} + q\mathbf{v} \times \mathbf{B},$$

where q is a charge moving with a velocity **v** in an electric field **E** and magnetic field **B,** and **F** is the force exerted on the charge. Within the sensor, a current is passed through a thin semiconducting sheet. The sensor output is measured as the potential difference across the semiconductor perpendicular to the direction of current. If there is no magnetic field, then the potential difference is zero. However, if there is a magnetic field applied perpendicular to the direction of current, then the electrons will be deflected to one side of the semiconductor via the Lorentz force, which will create a potential difference across the sheet. A visual of this system is shown in Fig. 1.

[1] Link to Hall Effect Sensor Non-Latching: https://www.sparkfun.com/products/14709

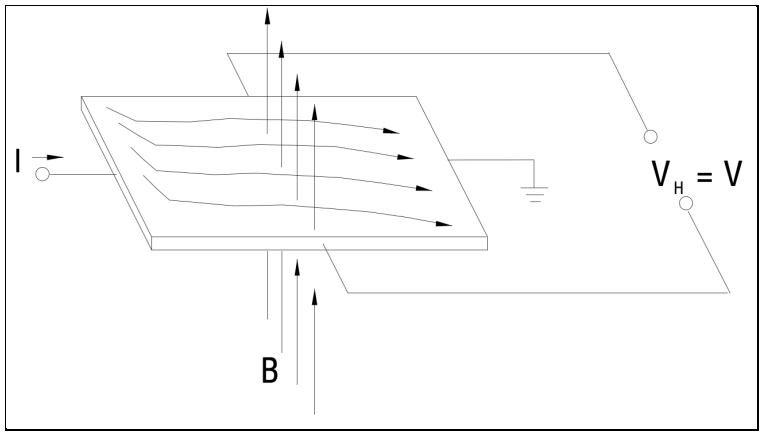


Figure 1. Diagram of a Hall effect sensor [4]. **B** is the magnetic field, **I** is the current through the semiconductor, and $\mathbf{V_H}$ is the output voltage.

Assuming that **B** is always applied perpendicular to the face of the semiconductor, the potential difference $\mathbf{V_H}$ is proportional to the strength of the magnetic field **B** times the current **I** [4]:

$$\mathbf{V_H} \propto \mathbf{I} \times \mathbf{B}$$

When the current is held constant, the output voltage can be used to measure the strength of the magnetic field. The magnetic field drops off at a rate of $1/d^2$ with the distance d normal to the North pole face of a magnet [4]. Therefore, if a magnetic is moved near a Hall effect sensor perpendicular to the direction of current, one expects for the measured potential difference to drop off at a rate of $1/d^2$:

$$V_H \propto \frac{I}{d^2}$$

The second theoretical concept used to develop the force sensing device was that the force applied to a material can be related to its deformation upon loading. Although, the relationship between force and deformation is dependent upon the material of choice. We combined this idea with the properties of the Hall effect sensor by placing a deformable material in between the Hall effect sensor and a magnet. This device was placed under the foot, so that when the person takes a step, the material deforms and the distance between the magnet and the Hall effect sensor decreases, producing an increase in the output potential. If the relationship between the force applied to the material and the deformation is known (e.g., if the material is linear elastic), then the force can be extracted from readings of the output voltage.

2.2 Physical Setup

The physical hardware consisted of layers of various compressible materials sandwiched in between the Hall effect sensor and a magnet. Materials were evaluated such that they would optimize the voltage readings from the Hall effect sensor. The material combination had to be both deformable, such that there would be a significant change in the distance between the

magnet and the sensor upon loading, and stiff, so that the material would not reach its maximum deformation under the weight of a human body.

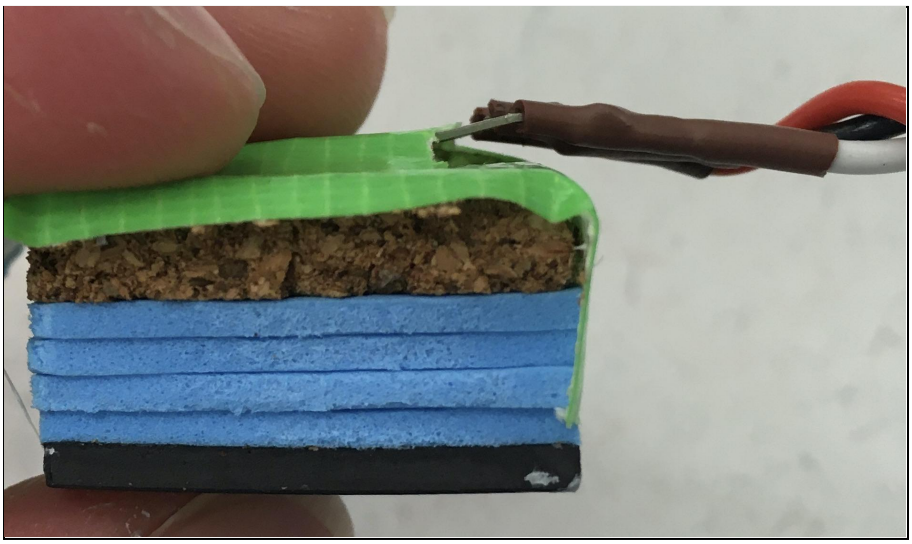
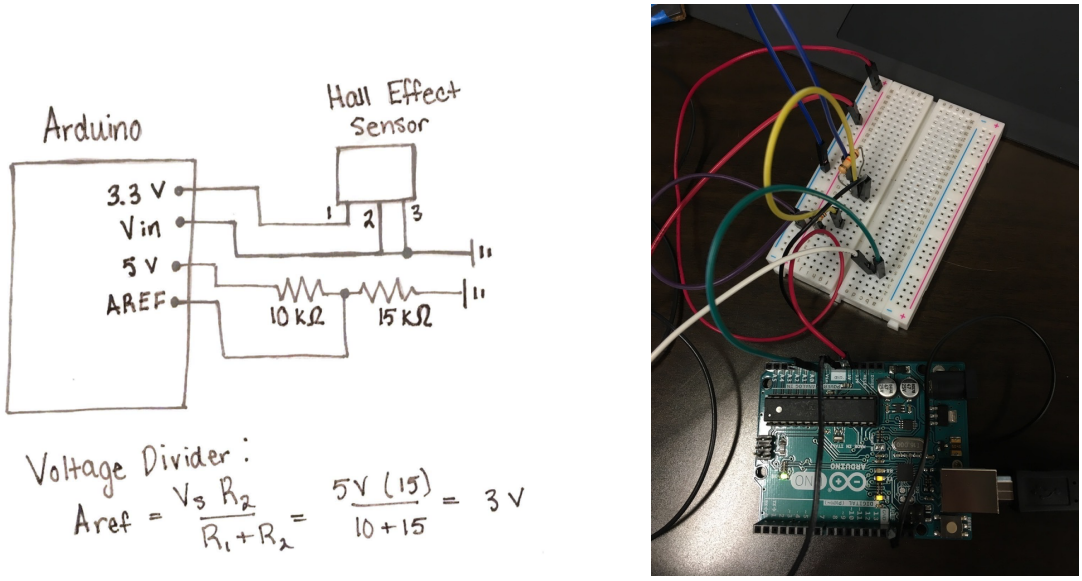


Figure 2: Pictured here is one of the multiple sensor configurations we tried. The sensor is taped to the top of a piece of corkboard, which is layered on top of four layers of EVA foam and a square magnet.

The Hall effect sensor was connected to the 3.3V output of the Arduino and an analog input pin was used to read in the output signal from the sensor (see Fig. 3). During initial testing, we noticed that the sensor had very low precision. The output signal was only varying by 5-15 mV when the sensor was stepped on, and the Arudino is a 10-bit ADC that expects a 5V input (4.88 mV/step). In order to increase the precision, a voltage divider was used to reduce the 5V Arduino output to ~ 3V, and the reduced voltage was connected to the AREF pin on the Arudino. The AREF pin changes the value that the Arduino expects as analog input. In the new setup, the Arudino had 10 bits spanning 3V (2.9 mV/step). Code was written using the Arduino IDE application[2] to read in the input value from the Hall effect sensor (Analog 0 pin) and print the result to the serial port. Microsoft Data Streamer for Excel was used to read the data from the serial port into Excel in real time (see Milestone 6 for more details on multi-port data logging).

[2] See Appendix A for Arduino Code used to read in data.



(a) (b)

Figure 3. (a) Circuit diagram of the Hall effect sensor connected to the Aduino. **(b)** Photo of the physical setup. A voltage divider was used to reduce the voltage of the 5V Arduino output to ~ 3V, which was sent to the AREF pin.

2.3 Test Material Properties

The precision and range of the force sensor were expected to depend highly upon the material placed in between the sensor and the magnet. If, for example, the material was very pliable, the sensor would have good precision, but it would have enough range to measure human body weight. Conversely, if the material was very stiff, the sensor would have sufficient range, but it would have too low precision. Our goal was to find a material (or a combination of materials) that would balance these effects. We tested two different materials described in Table 1. The relationship between the output voltage and the compression distance is expected to be nonlinear (see Section 2.1). The relationship between the applied force and the compression distance is unknown but likely nonlinear as well. Therefore, in order to know the relationship between the applied force and the output voltage, we developed a calibration curve. We measured the output voltage when various objects of known mass were placed on the sensor. However, one major problem encountered during this process was that most heavy masses are made out of metal (iron or stainless steel) that possess some degree of magnetism. In order to mitigate error introduced into the calibration curve, we opted to use non-metal materials. However, this reduced the selection of objects sufficiently heavy for experimentation and, consequently, reduced the number of data points obtained.

Table 1. Description of Test Materials

Material Number	Description
1	4 layers EVA foam (1.8 mm thickness/layer) + 1 layer corkboard (⅛" thickness/layer)
2	1 layer rubber (silicone ⅛" thickness/layer)

3.3 Preliminary Research + Sensitivity Analysis

A preliminary study was done to investigate the precision of the Hall effect sensor. The sensor was used to record the force at the heel of the foot and at the ball of the foot at two different walking speeds. These results are presented in Section 3.3.

3. Results

3.1 Sensor Calibration[3]

The calibration curve for material 2 is plotted in Fig. 4. The response can be approximated as linear at low masses and within small regions of interest. However, as the mass increases and the region of interest expands, the response decays.

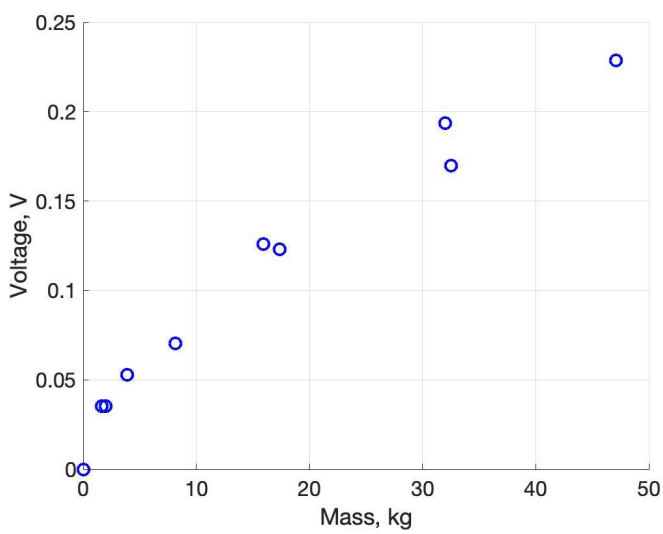


Figure 4. Calibration curve for material 2 (rubber). A linear response is a reasonable approximation at very low loading (1-4 kg) or within a small region of interest, but the nonlinearity progresses as the mass and the region of interest increase.

[3] See Appendix B for calibration dataset and Appendix C for calibration plotting code.

The calibration curve for material 1 is shown in Fig. 5. The range of masses tested for this material only extended up to 9 kg. The reason for this is that as the loading increased, the starting voltage (the voltage recording with no loading) increased due to permanent deformation of the EVA foam. Therefore, beyond 9 kg, the measurements were unreliable, and it became clear that material 1 was not well-suited for this application. Similar to material 2, the response was approximately linear for low loading conditions (1-4 kg), but the nonlinearity increased with added mass. Despite its poor spring-back performance, the large deformation of material 1 reduced the effect of quantization and produced greater precision than material 2.

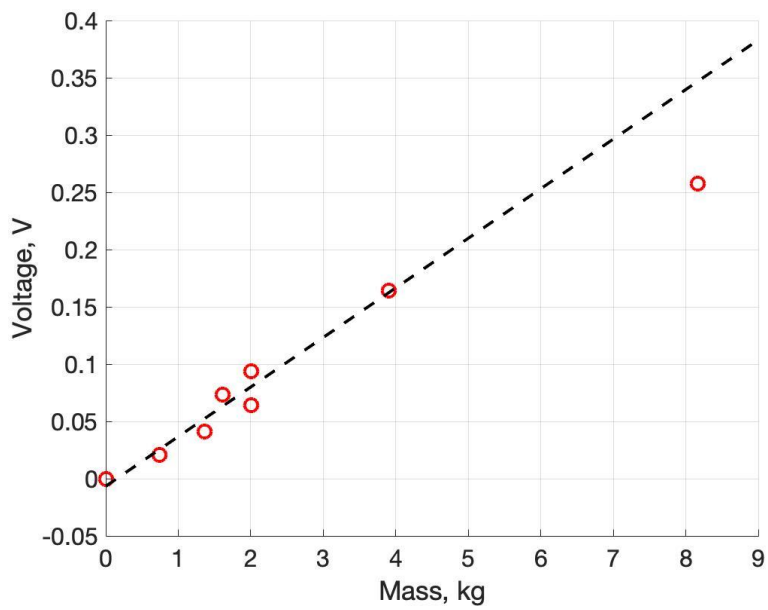


Figure 5. Calibration curve for material 1 (EVA cork + foam). A linear regression was fit to the first seven data points. A linear response is a reasonable approximation at very low loading conditions (R-squared = 0.9586 for first seven data points).

3.2 Response Testing[4]

The results of Section 3.1 indicated that we should expect large deformation from material 1, reducing the effect of quantization error but also producing insufficient spring-back after unloading. These trends were also analyzed by using both materials to measure the pressure at one point on the foot during a footstep. The results of this analysis are shown in Fig. 6. As expected, material 1 showed greater deformation upon loading but did not return to its original form after unloading. Material 2 reached a maximum output voltage of about half that of material 1 but exhibited better spring-back characteristics (although still not perfect). Despite the relative advantages and disadvantages of the different materials, both showed similar trends upon

[4] Code for this section can be found in Appendix D.

deflection. In Fig. 5, the onset of force was determined for each trial and used to align the curves. Tests with both materials showed a gradual onset of force occurring over ~ 0.1 sec and a sharp decay in force upon foot liftoff. The time at which the voltage peaked and the overall duration of the applied force agreed relatively well (see Table 2).

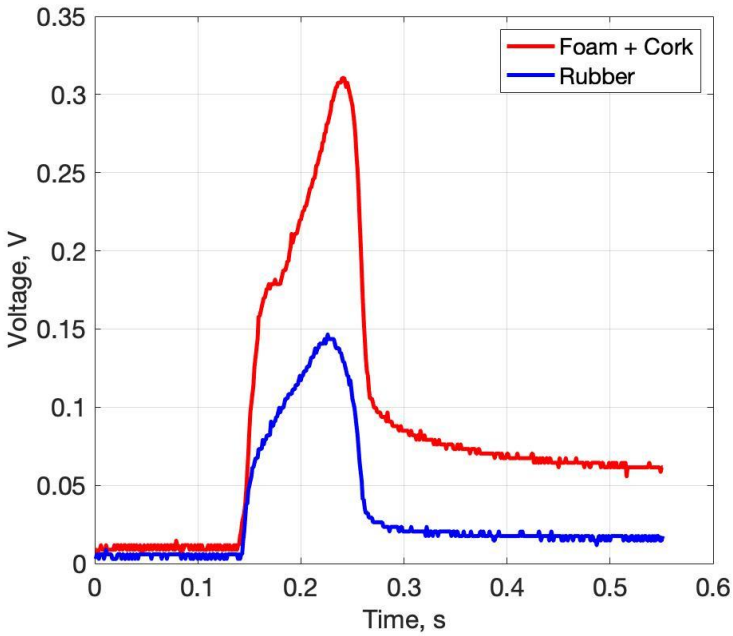


Figure 6. Voltage recorded by the Hall effect sensor during a foot step using material 1 (EVA foam + cork) and material 2 (rubber). Material 1 did not spring back to its original shape immediately after loading (0.053 volts deformation) but had a large range (0.31 volts). Conversely, material 2 was able to better spring back (0.015 volts deformation) but had a lower range (0.15 volts).

Table 2. Response characteristics of test materials during a footstep

Material number	Voltage range	Change in baseline voltage before/after loading	Time of peak	Ground contact time
1	0.31 volts	0.053 volts	0.240 sec	0.113 sec
2	0.15 volts	0.015 volts	0.226 sec	0.127 sec

3.3 Preliminary Research + Sensitivity Analysis

As seen in Fig. 7, the Hall effect sensor was able to distinguish between the force applied to the sensor at two different walking speeds and between the force applied at the heel/ball of the foot.

The ball of the foot showed a gradual onset of force, whereas the heel showed a sudden spike in force.

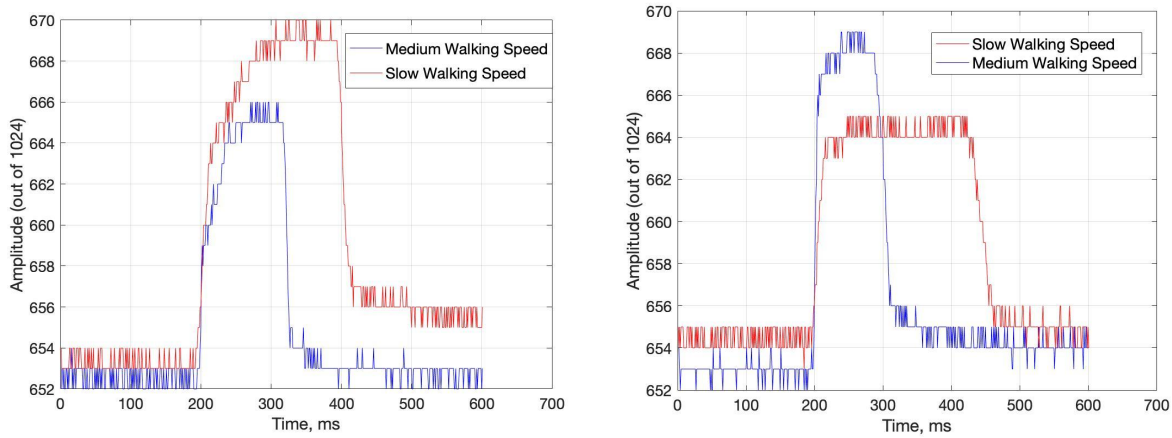


Figure 7. Measurement of the Hall effect sensor output voltage during a footstep when placed under the **(a)** ball of the foot and **(b)** heel of the foot at slow walking speed (blue) and moderate walking speed (red). The slow walking speed was ~ 1-1.5 ft/s, and moderate walking speed was ~ 4 ft/s (see Milestone 6 for more details on collection of walking speed data)[5].

4. Discussion

Although material 1 reached a greater maximum voltage and had greater precision than material 2, there was too much permanent deformation after loading to yield consistent and reliable measurements. This was evidenced by the fact that the change in the baseline voltage before and after a footstep was 17% of the total measurement range for material 1 but only 10% of the total measurement range for material 2 (see Table 2). In order to achieve ideal performance, the material of choice must have sufficient stiffness and linear elasticity.

Despite differences in the elastic properties of the two materials, they were able to capture similar traits in the onset and duration of force during normal walking (Fig. 6). The total contact time with the ground varied by only 0.014 seconds (~ 12%) between the two trials. These results can be validated against measurements from literature. According to Chan et al. [6], a person walking at normal speed averages 120 footsteps per minute (2 steps per sec). During each footstep, the "initial contact reaction" segment makes up the first 0.05 sec, the "mid-stance" segment makes up the second 0.05 sec, the "terminal stance" makes up the third 0.05 sec, and the "pre-swing" makes up the last 0.05 sec before liftoff [6]. Since we only recorded the force at the ball of the foot, we expect (according to Chen et al.) that the duration of the applied force should fall somewhere between 0.05 - 0.15 sec, which it did for both trials (see Table 2). Both trials also

[5] This study was conducted before the study in Section 3.2. We tuned the material and data collection procedure after this analysis, which is why we were able to get better precision in Fig. 6.

showed a gradual onset of force with a sudden and sharp release, suggesting that the Hall effect sensors are reliable and have sufficient precision (even when using a relatively stiff material like rubber in between the sensor and the magnet).

Lastly, when used to measure the force at one location of the foot during a footstep, the Hall effect sensor was able to distinguish between changes in the walking speed and changes in the location of the sensor (ball/heel of the foot). The ball of the foot showed a more gradual onset of force than the heel, and at slower walking speeds, the force was applied over a longer period of time. These results suggest that Hall effect sensors have sufficient precision to be incorporated into a full-foot pressure sensing device.

5. Impact and Conclusion

5.1 Final Messages

In this project, we showed that placing a deformable material in between a Hall effect sensor and a magnet produces a sensor with enough range and precision to reliably measure the pressure distribution on a human foot. The challenging component in incorporating Hall effect sensors into a force measurement system is finding a material with the correct stiffness and elasticity properties to sandwich between the sensor and the magnet. We found that EVA foam had good precision but did not return to its original shape when the load was removed. Conversely, the rubber (silicone) exhibited less plastic deformation than the foam, but it achieved lower precision. Regardless of the material used, the senor captured similar qualities of the foot strike, such as the total ground contact time and the time at which the sensor felt maximum force.

5.2 Limitations

This investigation led us down many possible routes for collecting the pressure distribution of the foot. The main limitation we faced with all types of sensors such as the FSRs, the strain gauges, and the hall effect sensors, was their robustness. We continued to have early issues in hardware development as our sensors became less accurate or maxed out with high levels of applied force. For example, force sensitive resistors (FSRs) maxed out with a light pinch. There is a clear need for robust sensors on the commercial market which can handle loads equivalent to body mass. Although the Hall effect sensors were the method of choice, the changes in voltages with the pressing of a foot were still relatively small and gave a narrow margin for the comparisons of slight changes in force. One other limitation we came across in our implementation of the sensors was the plastic deformation of the material cushioned between the magnets. Most materials will experience some form of plastic deformation, and not return to the original thickness after being loaded, altering the voltages recorded in future trials. We would also like to acknowledge the limitation of time and COVID-19 restrictions. A portion of our

group was virtual at times, limiting the amount of hands-on work and diagnostics. With more time and resources, we could have honed in on our sensors after facing several trials and errors with other hardware.

5.3 Recommendations

If one were to continue this investigation, we suggest sticking with the Hall effect sensors as the primary method. Compared to the force sensitive resistors and strain gauges, the Hall effect sensors were able to handle greater loads and were more manageable to sync with Arduino for real-time visualization. Likewise, we suggest using a material that exhibits greater linear elasticity and spring-back properties as the cushion in between the magnet and the sensor. A fair portion of our work was trial and error to determine which material (yoga mat, cloth rag, corkboard, rubber, EVA foam) provided the best resistance. We also recommend that additional data points be collected to generate a more robust calibration curve that includes error bars.

References

[1] Ko, M., Hughes, L., & Lewis, H. (2012). Walking speed and peak plantar pressure distribution during barefoot walking in persons with diabetes. *Physiotherapy Research International*, *17*(1), 29–35. https://doi.org/10.1002/pri.509

[2] Hessert, M. J., Vyas, M., Leach, J., Hu, K., Lipsitz, L. A., & Novak, V. (2005). Foot pressure distribution during walking in young and old adults. *BMC Geriatrics*, *5*, 8. https://doi.org/10.1186/1471-2318-5-8

[3] XSENSOR Technology Corporation. (2018). *XSENSOR vs. Competitors*. XSENSOR. https://documen.site/download/6sv_pdf

[4] Honeywell Sensing. (n.d.). *Hall Effect Sensing and Application*. Honeywell. https://sensing.honeywell.com/hallbook.pdf

[5] *Amazon.com: Gikfun A3144/OH3144/44E/AH3144E Hall Effect Sensor Magnetic Detector for Arduino (Pack of 20pcs) EK1325: Home Improvement*. (2021, May 7). Amazon. https://tinyurl.com/hk538kc3

[6] Chan, C. W., & Rudins, A. (1994). Foot Biomechanics During Walking and Running. *Mayo Clinic Proceedings*, *69*(5), 448–461. https://doi.org/10.1016/S0025-6196(12)61642-5

Appendices

Appendix A

```
/*
 AnalogReadSerial
 Reads an analog input on pin 0, prints the result to the Serial Monitor.
 Graphical representation is available using Serial Plotter (Tools > Serial
Plotter menu).
 Attach the center pin of a potentiometer to pin A0, and the outside pins
to +5V and ground.
 This example code is in the public domain.
 http://www.arduino.cc/en/Tutorial/AnalogReadSerial
*/
// the setup routine runs once when you press reset:
void setup() {
 // initialize serial communication at 9600 bits per second:
 Serial.begin(9600);
 analogReference(EXTERNAL);
}
// the loop routine runs over and over again forever:
void loop() {
 // read the input on analog pin 0:
//  analogReadResolution(12);
 int sensorValue = analogRead(A0);
 int sensorValTwo = analogRead(A1);
 int sensorValThree = analogRead(A2);
 int sensorValFour = analogRead(A3);
 int sensorValFive = analogRead(A4);
 int sensorValSix = analogRead(A5);
 // print out the value you read:
 Serial.println(sensorValue);
  delay(1);        // delay in between reads for stability
}
/* To read and record these values in excel:
*  1) Make sure no serial communication is open with the IDE
*  2)Enable Microsoft Data Streamer in Excel
*  3)From 'Connect a Device' select the port the Arduino is using
*  4)Use 'Start Data' to verify serial connection is working as intended
*  5)Use 'Record Data' to write values to a .csv file
*/
```

Appendix B

Table 3. Material 1: Mass Applied to Hall Effect Sensor vs. Voltage Reading

Mass Applied (kg)	Voltage Reading (out of 1024)
1.616	12
2.004	12
3.9	18
8.168	18
15.9	43
17.39	42
32	66
32.49	58
47.1	78

Table 4. Material 2: Mass Applied to Hall Effect Sensor vs. Voltage Reading

Mass Applied (kg)	Voltage Reading (out of 1024)
0.74	7
1.36	14
1.616	25
2.004	32
2.002	22
3.9	56
8.168	88

Appendix C

Matlab Code for Calibration Plots

```
load('foamCorkWeight.mat');
load('rubberWeight.mat');

% Convert units to voltage (3 Volts, 10 bits)
foamCorkWeight(:,2) = foamCorkWeight(:,2) * 3/1024;
rubberWeight(:,2) = rubberWeight(:,2) * 3/1024;

% Plot calibration for material 1
figure
scatter(foamCorkWeight(:,1), foamCorkWeight(:,2), 80, 'LineWidth', 2, ...
   'MarkerEdgeColor', 'red')
set(gca, 'fontSize', 16)
xlabel('Mass, kg')
ylabel('Voltage, V')
grid on
hold on

% Compute Pearson's R-Squared to determine goodness-of-fit
[B,BINT,R,RINT,STATS] = regress(foamCorkWeight(1:7,2), ...
   [foamCorkWeight(1:7,1), ones(length(foamCorkWeight(1:7,1)),1)])

pts = [0:1:9];
plot(pts, pts * B(1) + B(2), '--k', 'LineWidth', 2)
xlim([0 9])
STATS(1) % Pearson's R-squared

% Plot Calibration for material 2
figure
scatter(rubberWeight(:,1), rubberWeight(:,2), 80, 'LineWidth', 2, ...
   'MarkerEdgeColor', 'blue')
set(gca, 'fontSize', 16)
xlabel('Mass, kg')
ylabel('Voltage, V')
grid on
hold on
```

Appendix D

Matlab Code for Response Testing

```
%% Material 1 (cork + foam)
T = readtable('footstep.csv'); % read in data in table format
data = T.Var1; % extract data from table into an array

% locate footstep
find_step = find(data < 600); % manual threshold for detecting force onset
time_step = find_step(1); % find first point of crossing threshold

% Extract relevant data around footstep
% In the current setup, voltage decreases with the onset of force.
% In line 12, we invert this to show an increase in voltage with force.
% We believe this is a more intuitive way to display the data and does
% not affect the results;
dataSec = max(data) - data(time_step - 150:time_step + 400);
dataSec = dataSec * 3 /1024; % Convert to voltage

timeSec = [1:length(dataSec)] * 0.001; % Time vector
plot(timeSec, dataSec, 'r', 'LineWidth', 2.5, 'DisplayName', 'Foam + Cork')
set(gcf,'position',[100,100,500,420])
set(gca, 'fontSize', 16)
xlabel('Time, s', 'fontSize', 16)
ylabel('Voltage', 'fontSize', 16)
grid on
hold on

%% Material 2 (Rubber)
T2 = readtable('rubberpadtest.csv'); % load data
data = T2.x__; % read into vector

% detect onset of footstep
find_step = find(data < 600);
time_step = find_step(1);
% locate data of interest
dataSec2 = max(data) - data(time_step - 150:time_step + 400);
dataSec2 = dataSec2 * 3 /1024; % transform to voltage

timeSec2 = [1:length(dataSec2)] * 0.001; % time vector
```

```
plot(timeSec2, dataSec2, 'b', 'LineWidth', 2.5, 'DisplayName', 'Rubber')
set(gcf,'position',[100,100,500,420])
set(gca, 'fontSize', 16)
xlabel('Time, s', 'fontSize', 16)
ylabel('Voltage', 'fontSize', 16)
grid on
legend

%% Approximate length of applied force and peak time

% Material 1
peaks1 = find(dataSec == max(dataSec));
peakTime1 = timeSec(peaks1(1)); % Get first peak (if more than one)

afterStep1 = find(dataSec > 0.025); % manual threshold for detecting force onset
onsetStep1 = time(afterStep1(1));
duringStep1 = find(dataSec > 0.1); % manual threshold for detecting force end
endStep1 = time(duringStep1(end));

lengthStep1 = endStep1 - onsetStep1;

% Material 2
peaks2 = find(dataSec2 == max(dataSec2));
peakTime2 = timeSec2(peaks2(1)); % Get first peak (if more than one)

afterStep2 = find(dataSec2 > 0.025); % manual theshold for detecting force onset
onsetStep2 = time(afterStep2(1));
duringStep2 = find(dataSec2 > 0.05); % manual theshold for detecting force end
endStep2 = time(duringStep2(end));

lengthStep2 = endStep2 - onsetStep2;

%% Compute Permanent Deformation Before/After Step

% Material 1
max1 = max(dataSec);
endVal1 = dataSec(end);
initialVal1 = dataSec(1);
deformation1 = endVal1 - initialVal1; % Deformation
```

```
% Material 2
max2 = max(dataSec2);
endVal2 = dataSec2(end);
initialVal2 = dataSec2(1);
deformation2 = endVal2 - initialVal2; % Deformation
```

Milestone Documentation

Throughout the semester, we documented our progress in the form of six key milestones. Below is a compilation of hand-picked sections of the milestones. They include original motivations, necessary equipment, and other valuable information that we feel adds to this report.

Project Milestone II - February 12th, 2021

How does walking speed affect the pressure distribution on one's foot?

We should likely have real-time sensors to validate the procedure, then collect more detailed,higher resolution data later on. Walking and running may look very different in terms of force distribution and could be compared/contrasted. We can use Deeplabcut to visually analyze walking speed and build a force sensor grid to collect pressure data. We could also use accelerometers or distance sensors to measure velocity. This question seems to be the most feasible.

Upon a brief review of literature, the predominant consensus appears to indicate that increases in walking speed correlate to overall increased pressure on the foot, also described as Peak Plantar Pressure (PPP) [1, 2]. Thus, armed with the knowledge that overall pressure increases with walking speed, perhaps the question can be refined to revolve around what parts of the foot are used more frequently while moving at various speeds. For example, a Mayo Clinic study found that "Typically, long-distance runners initially contact the ground heel first or with the foot flat,whereas sprinters commonly land on the midfoot" [3]. Thus, by using a system of distributed pressure sensors in the sole during movement at various speeds, the part of the foot most clearly associated with various running velocities should be easily determined.

Project Milestone III - February 24th, 2021

Collection:
Data collection will be performed using a standardized pressure sensing pad (incorporated into a sock or other flexible platform), which will connect to a shared microcontroller unit used for data processing and storage (attached to the ankle). Two methods of velocity measurement are being investigated. Preferably, the team will request permission to use Prof. Koushyar's biomechanics

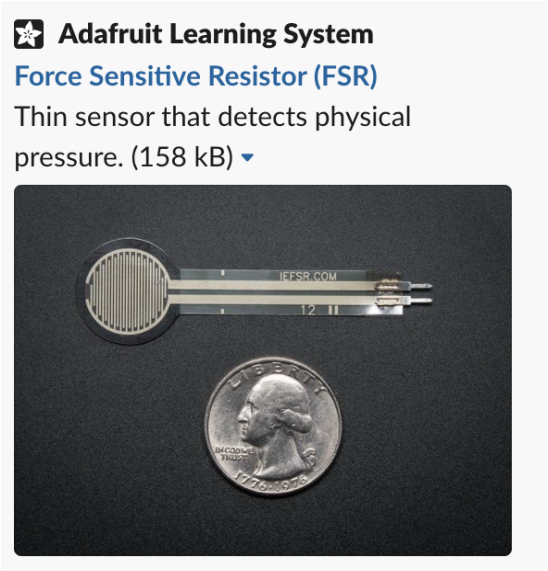
treadmill. If that is not an option, the team will use camera-based motion capture. The team anticipates logging and storing data on an SD card.

Processing and Visualization:
The team plans to use matlab to generate visualizations that display an individual's pressure distribution, comparisons of pressure distribution with varying walking speeds, and higher order regressions of the collated raw data.

Statistical Tests:
After data collection, polynomial regressions can be performed on the data to determine trends relating speed and distribution of foot pressure.
We can also use a paired t-test to study whether the pressure values recorded on the foot at one walking speed are significantly different from the pressure values recorded at another walking speed.

Project Milestone IV - March 19th, 2021

Revised Statement: Using the force sensitive resistors (FSR) to measure the pressure on different parts of the foot and analyse data into a visual representation.



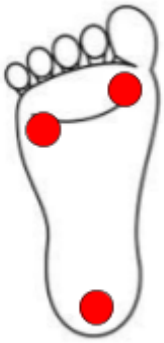
I. Hardware Development

- Obtained two FSRs from Professor Huang, in the process of obtaining more (one already broke)
- Soldering long wires to the ends of the sensors to make them more versatile to plug into microcontroller → 3 foot long leads

II. Helpful Documentation:

DataSheet/Tutorial(Wiring/GettingStarted) :

https://learn.adafruit.com/force-sensitive-resistor-fsr/using-an-fsr
Saving Data from the Serial Monitor of the Arduino:
https://forum.arduino.cc/index.php?topic=367920.0



III. "Plan of Attack"

Microcontroller of Choice: Arduino Uno

Synthesis of Data: MATLAB

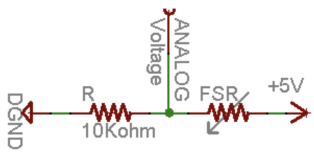
Our rough procedure includes using three FSRs on separate locations of the foot. These being the ball, heel, and flat spot of the foot (see diagram). The FSR would be attached to the Arduino and input the force reading into the program. After collecting this data for different trials and participants, we will transfer the data to MATLAB and make graphs, visual representations possibly.

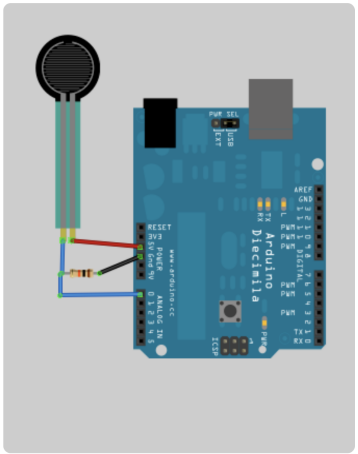
IV. PROOF OF CONCEPT:

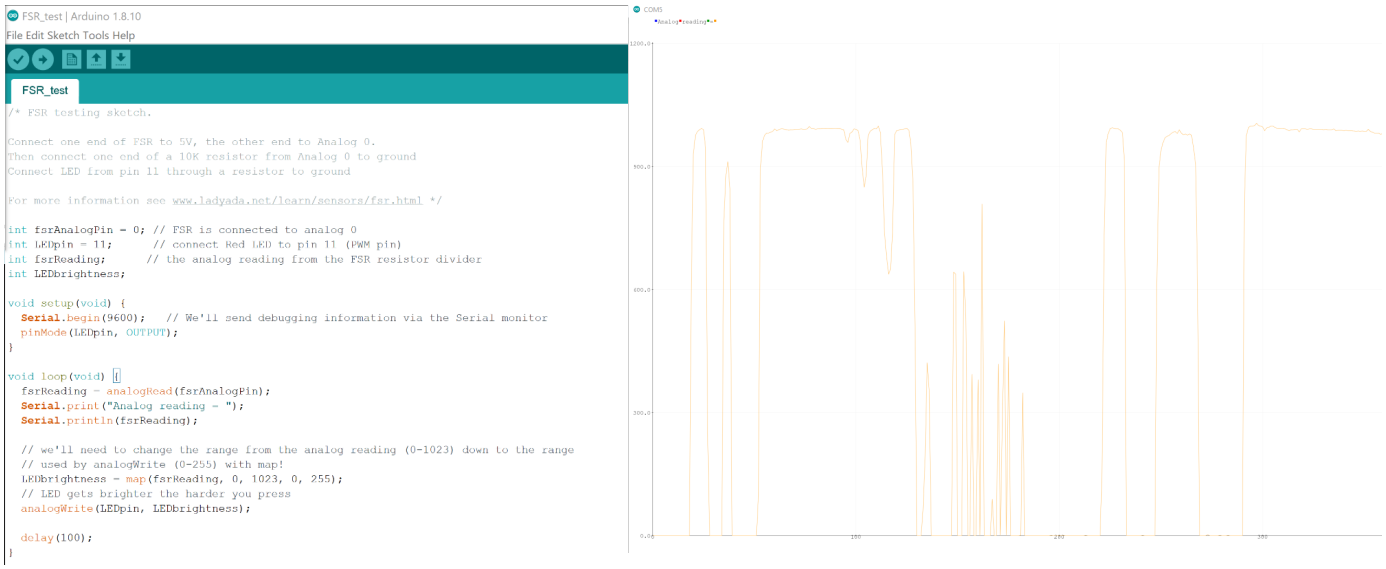
Our goal for this week was to test the compatibility of the FSR and whether the mass of someone's foot could be accurately read. The main concern was having the sensor max out as the datasheet specs suggested a max weight reading of about 20 lbs.

Wiring: We attached the FSR to an Arduino Uno as well as attach an LED to give an in-time visual representation of whether the sensor is picking up data. The resistor is 10K.

Circuit Diagram:





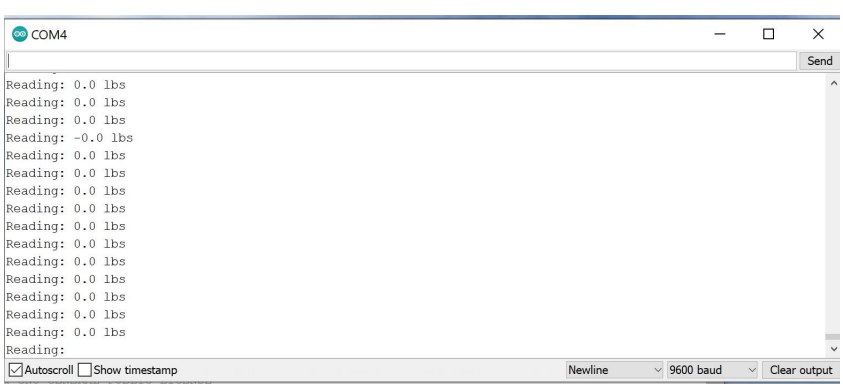


Considering that the max weight reading of the sensor is only 20 lbs, we suspect that the sensor is reaching its maximum threshold. If this is the case, we will be unable to sense force differences across the foot.

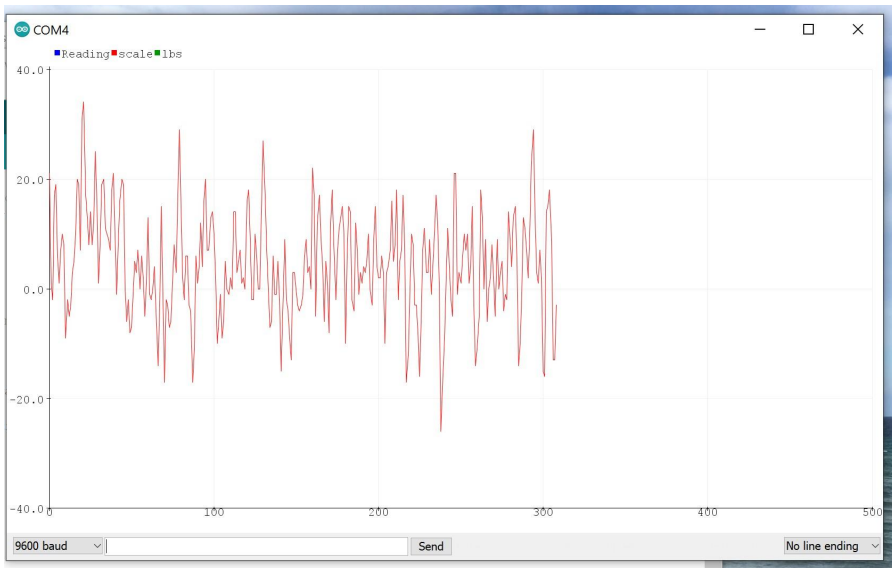
Project Milestone V - April 7th, 2021

Live Visualization of Data Collection - Strain Gauges

This data was collected during an initial calibration test using the sensors in a full wheatstone bridge. The first output readings were all at zero, however we were able to change the calibration value in order to detect values.



Here is the live visualization of our data - pressure vs time. We were able to detect some peaks, however we struggled to get level of distinction that we need, and the pressure required to detect the change was very large.



This photo shows the full set up of the four load cell wheatstone bridge.

Project Milestone IV - April 23rd, 2021

Position Tracking

In order to track the walking speed of the person stepping on the force sensor, we used the MATLAB Image Processing and Computer Vision toolbox. A Matlab example code ('MotionBasedMultiObjectTrackingExample.m')[6] was used as a starting point for our speed detection code with the following modifications:

1. The sensitivity of motion detection was decreased to prevent the code from detecting new objects due to the shakiness of the video.
2. The minimum size of a moving object was increased to, again, prevent minor disturbances or movements from getting picked up.

6 *MATLAB Computer Vision ToolBox: Motion-Based Multiple Object Tracking*. Natick, MA: MathWorks, R2021a.

3. The number of moving objects displayed on screen (yellow box in Fig. 1) was limited to one.

This code works by subtracting out the background of the image and using blob analysis to detect changes to groups of pixels that reflect a moving object--referred to as a track. A Kalman filter is used to predict the next location of each track, and the predictions are used to assign tracks in a previous frame to the tracks in the new frame. The centroid of each track is recorded (in units of pixels). We saved the x-location of the centroid in each frame in order to compute the velocity of the object.

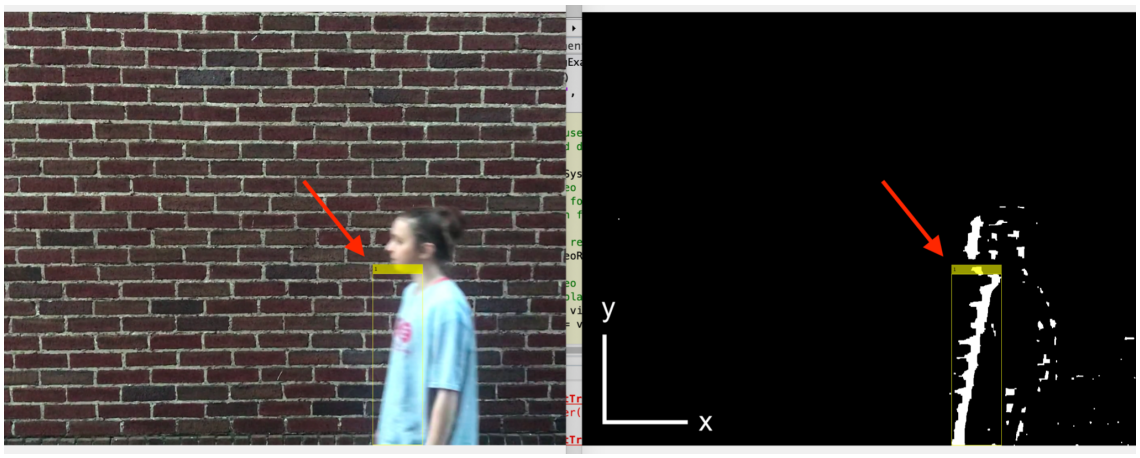


Figure 1. *Visual display of motion tracking code. A rectangular box (yellow) captures the position of the moving object.*

A linear regression model is fit to a curve of position (pixels) vs. time (frame number), and the slope reflects the walking speed (pixels/frame) (see Fig. 2). The coefficient of determination of this regression tells us the consistency of the person's walking speed (i.e. if the walking speed is constant, R^2 will be close to one). Units of pixels/frame are transformed into units of m/s by knowing: (1) the horizontal distance contained within the video frame (measured in meters before data collection) and (2) the video frame rate and duration. In Fig. 2, a video of a person walking at a casual pace was recorded, and their average walking speed was found to be approximately 1.36 m/s (or 3 mph), which is very reasonable.

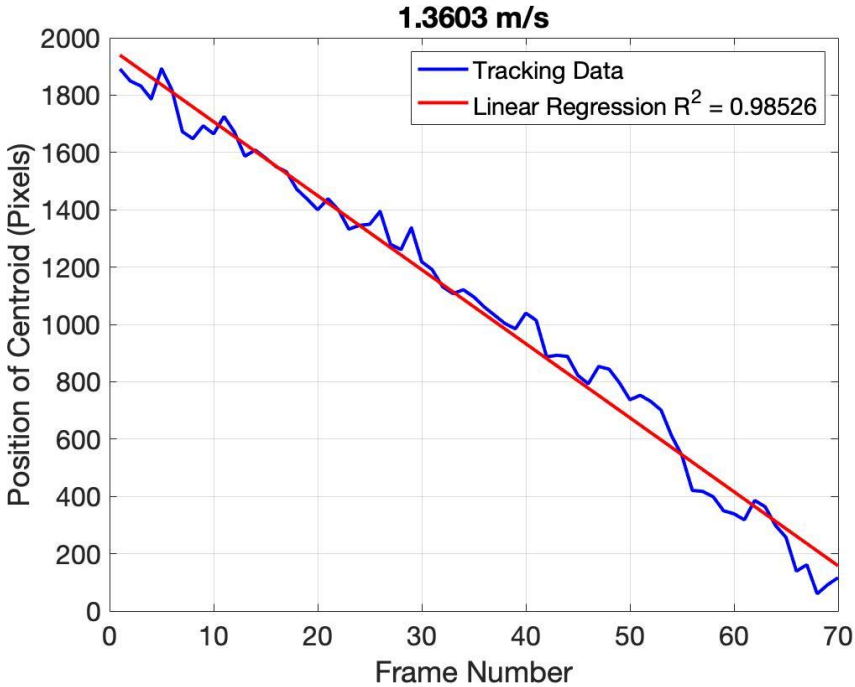


Figure 2. *Position (pixel location) of the moving object (blue) as a function of the frame number. A linear regression (red) is fit to this curve, and the slope reflects the average walking speed. The coefficient of determination is close to one, indicating that the person's walking speed was relatively constant throughout the data collection period.*

Two additional trials were conducted--one at a very fast walking speed (Fig. 3) and one at a very slow walking speed (Fig. 4)--in order to determine if changes in walking speed could be detected.

The MATLAB motion detection code and the plotting/regression code can be found at the end of this document.

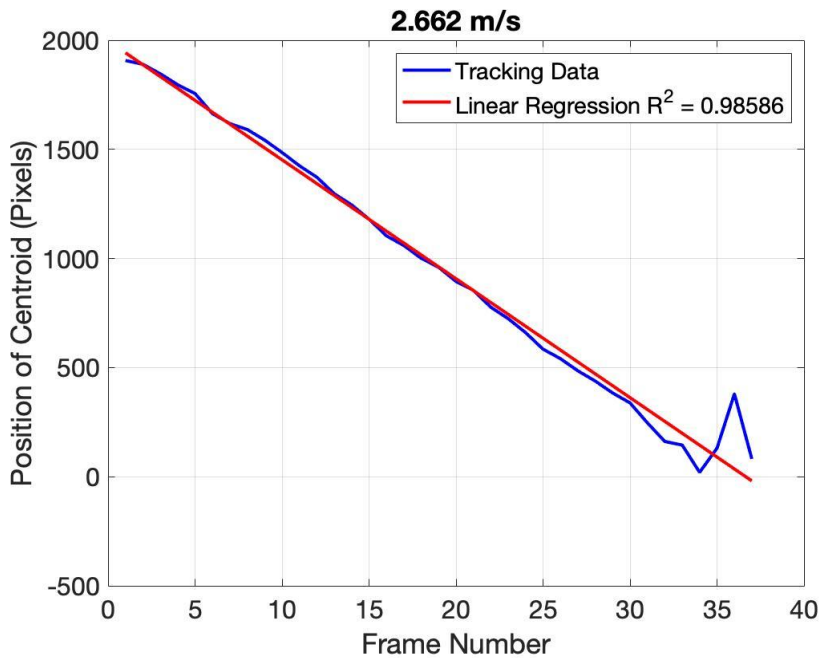


Figure 3. *Position tracking at fast walking speed.*

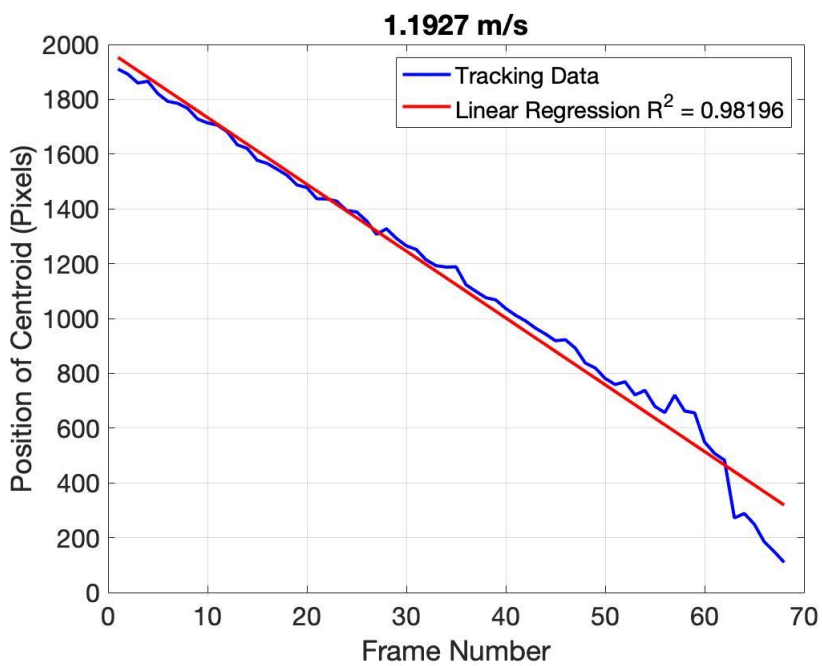


Figure 4. *Position tracking at slow walking speed.*

Figs. 3 and 4 were generated to test the position tracking algorithm. Figs. 5 and 6 are plots from the real data collection period. (Note that Figs. 3 and 4 are in m/s and Figs. 5 and 6 are in ft/s). Note that Figures 2, 3, and 4 represent data for a subject walking from the right side of the frame to the left, while Figures 5 and 6 show a subject walking from left to right.

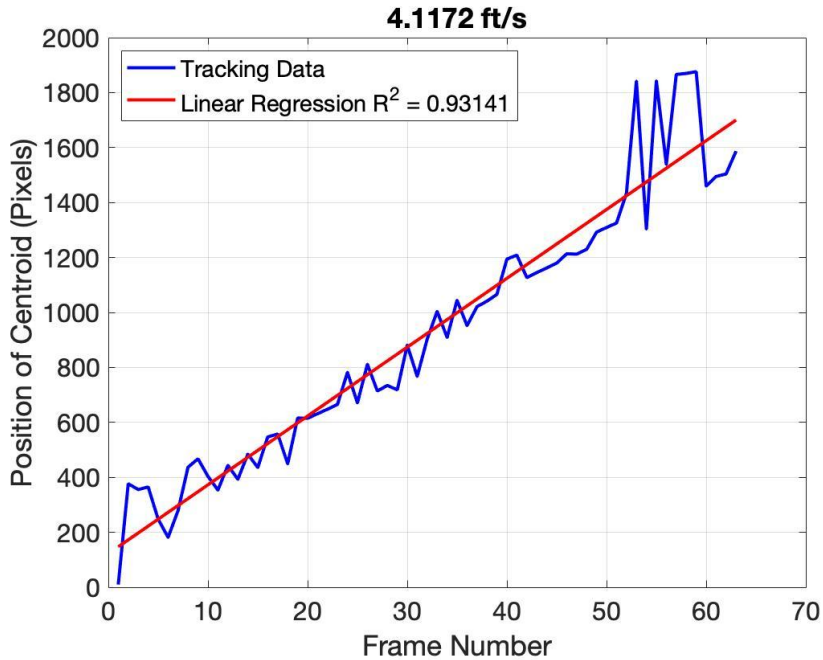


Figure 5. *Position tracking at moderate walking speed.*

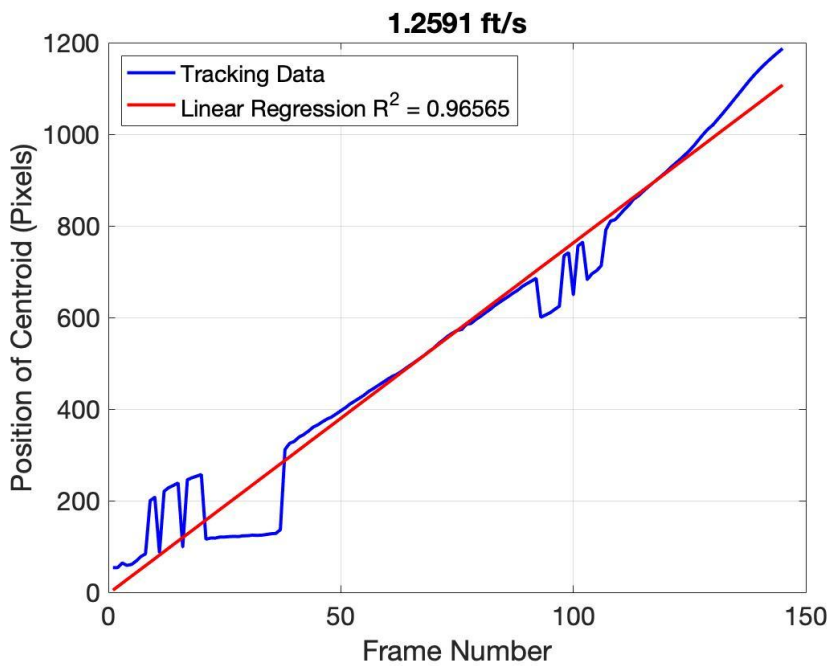
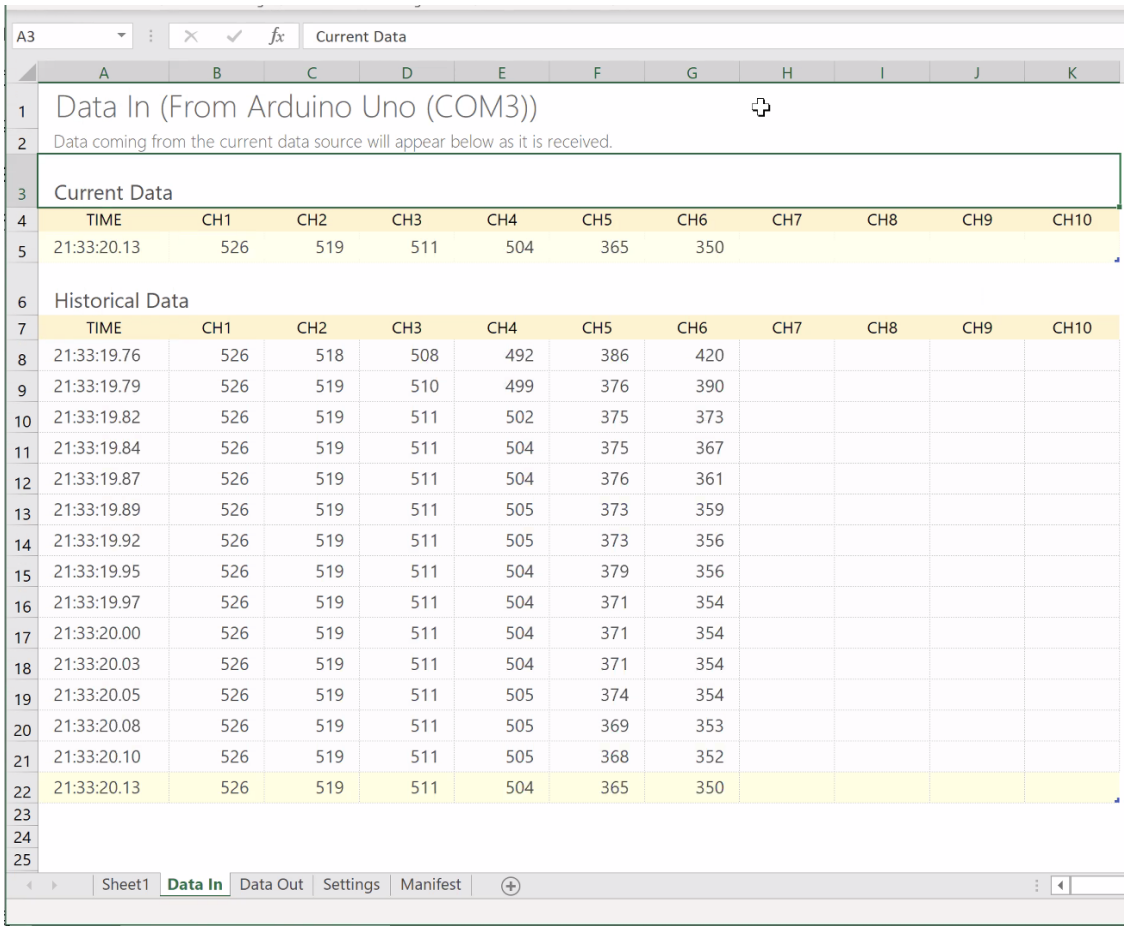


Figure 6. *Position tracking at slow walking speed.*

Data Recording

For this project, we needed to find a method to record data from 6 Arduino analog inputs simultaneously and export it in a format compatible with MATLAB. We were able to do this using Microsoft Data Streamer for Excel. This Excel add-in reads in and records data printed to the Arudino serial monitor. Fig. 5 is a screenshot of the Excel data streamer reading in data from 6 Arduino inputs (CH1 - CH6). This data can be easily exported as a .csv file for analysis in MATLAB.



Data In (From Arduino Uno (COM3))

Data coming from the current data source will appear below as it is received.

Current Data

TIME	CH1	CH2	CH3	CH4	CH5	CH6	CH7	CH8	CH9	CH10
21:33:20.13	526	519	511	504	365	350				

Historical Data

TIME	CH1	CH2	CH3	CH4	CH5	CH6	CH7	CH8	CH9	CH10
21:33:19.76	526	518	508	492	386	420				
21:33:19.79	526	519	510	499	376	390				
21:33:19.82	526	519	511	502	375	373				
21:33:19.84	526	519	511	504	375	367				
21:33:19.87	526	519	511	504	376	361				
21:33:19.89	526	519	511	505	373	359				
21:33:19.92	526	519	511	505	373	356				
21:33:19.95	526	519	511	504	379	356				
21:33:19.97	526	519	511	504	371	354				
21:33:20.00	526	519	511	504	371	354				
21:33:20.03	526	519	511	504	371	354				
21:33:20.05	526	519	511	505	374	354				
21:33:20.08	526	519	511	505	369	353				
21:33:20.10	526	519	511	505	368	352				
21:33:20.13	526	519	511	504	365	350				

Figure 5. *Screenshot of the Excel data streamer reading in data from 6 Arduino analog inputs.*

Data Visualization

While waiting for our sensors to come in the mail, we started developing our data visualization system. We initially developed it using synthetic data, but we will substitute more real data in place of the synthetic stand-in once we have collected more sets of reliable data.

We are interested in answering two different questions:

1.) How does the force distribution on the foot change with the walking speed?
2.) How does the force distribution on the foot change throughout a footstep?

Visualization 1:
This video plots the force distribution on the foot at different walking speeds. It was generated in MATLAB using the code at the end of the document entitled 'plotFoot.m'.
The code displays a generic image of a foot and overlays it with large scatter plot markers to represent the sensors. The walking speed is displayed at the top of the plot.
At each walking speed, 20 synthetic data points were generated for each sensor to represent the maximum pressure experienced by the sensor throughout the foot step. Noise was added to the synthetic data to replicate experimental error. In this synthetic data set, one can see that at low walking speeds, the pressure is greater at the heel of the foot. As the walking speed increases, the pressure moves up the foot.

Visualization 2:
This video plots the force distribution on the foot throughout one footstep at one walking speed. The video on the left shows the foot contacting the ground. The plot in the center shows the pressure on the foot as it changes in time. The plot on the right shows the pressure at each position as a function of time (the same data in the central plot).

Both visualization videos can be found here:
https://tufts.box.com/s/5ieaicm2m098obwy6sva4yikdkk48wlo

Visualization 3:
This Matlab code detects and plots the peaks in the collected data. The data are voltage vs time, which has a squared relationship with the distance between the Hall Effect Sensor and the magnet. This makes in-depth analysis of the shape of the peaks difficult, and for the next stage of our data analysis, we will determine a relationship between the voltage shown here and the force applied on the sensor during the step. Also, the timing of the peaks is not meaningful, as they are plotted based on the data surrounding the peaks, and not aligned with the timing of the steps taken. This being said, it is meaningful to note that the shape of the heel strikes is different from the shape of the ball strikes; the ball strike has more of a ramp-up shape, while the heel strikes have a very squared off shape.

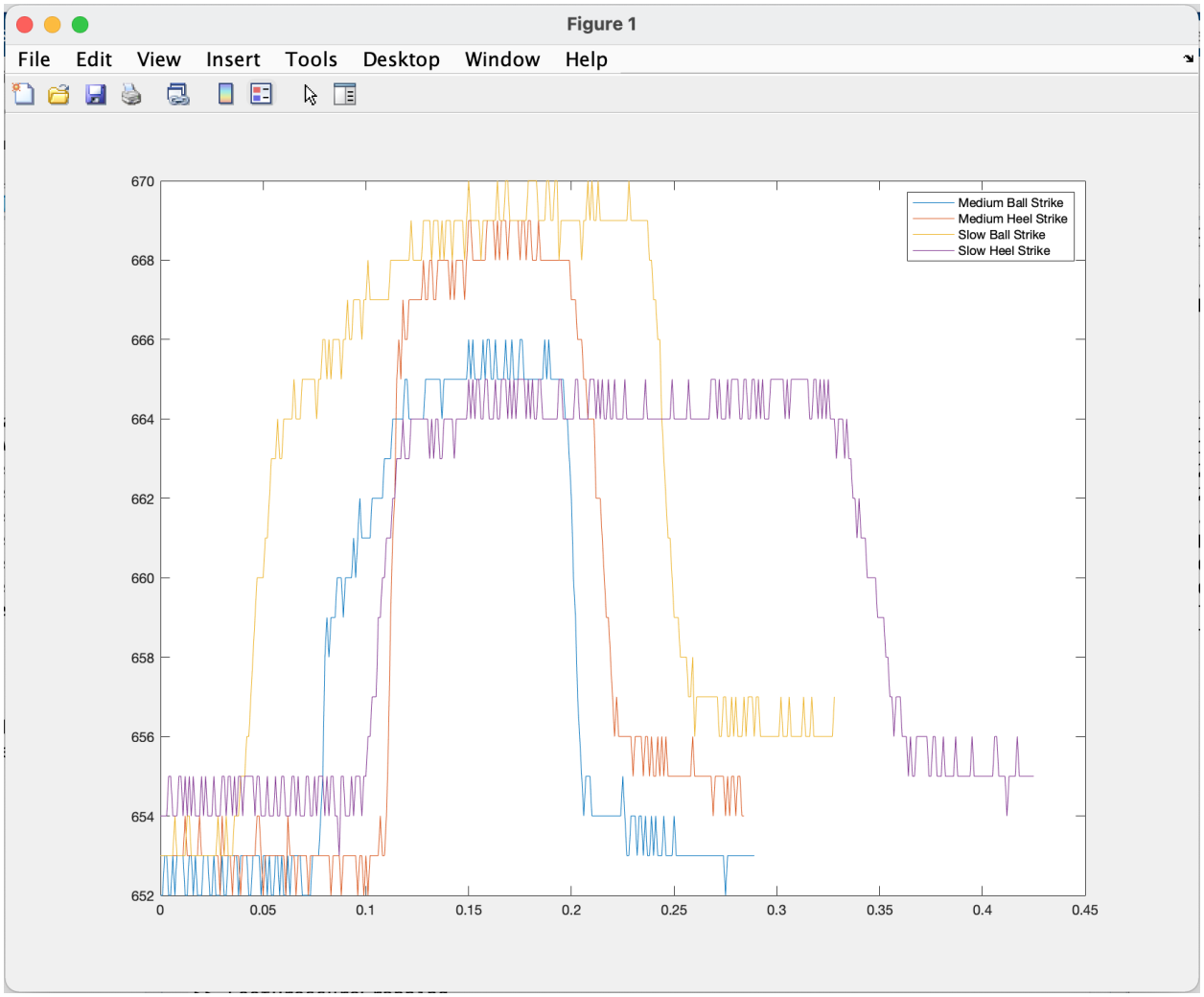


Figure 1
File Edit View Insert Tools Desktop Window Help
Medium Ball Strike
Medium Heel Strike
Slow Ball Strike
Slow Heel Strike
670
668
666
664
662
660
658
656
654
652
0
0.05
0.1
0.15
0.2
0.25
0.3
0.35
0.4
0.45

Visualization 3 Matlab code:

```
clear
close all

% Read in the data from the csv files
walkOneBall = readmatrix('testdata2ball.csv');
walkOneHeel = readmatrix('testdata3heel.csv');
walkTwoBall = readmatrix('testdata4ball.csv');
walkTwoHeel = readmatrix('testdata4heel.csv');

% Detect the peaks in the data and create new vectors of the data
% surrounding the peak
peak1 = find(walkOneBall == max(walkOneBall));
oneBallPeak = walkOneBall((peak1(1)-150):(peak1(end)+100));

peak2 = find(walkOneHeel == max(walkOneHeel));
oneHeelPeak = walkOneHeel((peak2(1)-150):(peak2(end)+100));

peak3 = find(walkTwoBall == max(walkTwoBall));
twoBallPeak = walkTwoBall((peak3(1)-150):(peak3(end)+100));

peak4 = find(walkTwoHeel == max(walkTwoHeel));
twoHeelPeak = walkTwoHeel((peak4(1)-150):(peak4(end)+100));

% Create a vector of times, ms time step between readings
time1B = [0:0.001:(0.001*(length(oneBallPeak)-1))];
time1H = [0:0.001:(0.001*(length(oneHeelPeak)-1))];
time2B = [0:0.001:(0.001*(length(twoBallPeak)-1))];
time2H = [0:0.001:(0.001*(length(twoHeelPeak)-1))];

% Plot the voltage data of the detected peaks vs time
plot(time1B, oneBallPeak)
hold on
plot(time1H, oneHeelPeak)
plot(time2B, twoBallPeak)
plot(time2H, twoHeelPeak)

legend('Medium Ball Strike', 'Medium Heel Strike','Slow Ball
 Strike','Slow Heel Strike')
hold off
```