Athlete Motion Tracking
Final Report

Client: Precision Performance Cycling (Nathan Johnson)

Adviser: Phillip Jones

Team Number: sdmay19-01

Team Members

Software Engineer:	Monte Friestad
Hardware Engineer:	Ryan Hansen
Software Engineer:	Nathan Mazarelo
Hardware Engineer:	Madeline Rogers

Email: sdmay19-01@iastate.edu

Website: http://sdmay19-01.sd.ece.iastate.edu/

0. Executive Summary

The goal is to create an integrated interface between multiple software and hardware systems to decrease the bias when the client is manually analyzing the athlete's body position. The athlete's position is important as improper positioning will create unnecessary injuries and minimize the performance. The client is currently measuring the position based upon their eye, leaving no room for mathematical analysis to create quanitivative reports for athletes (Figure 7).

1. Requirements Specification

A. Functional Requirements

a. Motion Capture

i. Ability to capture video of an athlete

ii. Ability to recognize presence of athlete

b. Pressure System

i. Ability to capture sensor data

ii. Detect force from athlete

iii. Ability to withstand residual moisture

c. Data Analytics

i. Ability to analyze video of an athlete

ii. Ability to analyze sensor data

iii. Data display with analysis

d. User Interface

i. Ability to compare data and readings over multiple training sessions

ii. Data display in the raw form and minimal analysis

iii. Ability to view previous data from any setting

B. High-Level Requirements

a. The system should be able to gather data about the athlete in a variety of ways.

b. The system should analyze the data in an effective, efficient way.

c. The system should be able to display the data to be seen by the user.

C. Use-Cases

a. Use the User Interface to compare two past sessions.

b. Use the User Interface to compare a new and past session.

c. Use the pressure sensor to determine strike time.

d. Use the pressure sensor to determine correlation of rotation and

D. Non-Functional Requirements (tied to the clients and/or target users)

a. Motion Capture
   i. Precision within 2.5%
   ii. Three different camera positions
   iii. 1080p

b. Pressure System
   i. Athlete comfort
   ii. Multi-User Friendly
   iii. Waist mounted
   iv. Wireless

c. Data Analytics
   i. Precision within 2.5%
   ii. Angle overlay

d. User Interface

i. Review of past sessions

ii. Side by Side comparison of sessions

2. System Design & Development

A. Design Plan

There are two very distinct modules in the project. The first is the camera system for the image processing. This module starts with the inputs of the cameras. The cameras take the video and then go to the iPi software. Then the software processes the video. After it has been processed and analyzed the software outputs the data to a CSV file. The GUI then takes this and reads the values to display the interface for the web application. This will be determined first. The design of this system will include the selection of a camera, the number of cameras, and the software to be used.

The second system is the sensors. The pressure sensors are variable resistors that change the voltage output (Figure 9). The greater the force, the greater the resistance. From there, the voltage is converted into a meaningful value, force. The voltage output from the circuit is then interpreted by the Arduino, converted to a force value, and then saved onto a SD card mounted on the Arduino. This system will be completed last. This will allow for the application to be set up and working prior to implementing a second system. The design of this system will include the selection of the sensors, the circuit that outputs voltage readings from the sensors, the ergonomics, the data delivery, and then the placement of the sensors.

The two systems will then be integrated across one user interface. The interface will be implemented second. This will allow for the implementation of the image processing system to be used and worked into the system and then built upon with the pressure sensor system. The design of this system will include the presentation of the analytics, the data analytics, the storage, placement, and further features.

B. Design Objectives, System Constraints, Design Trade-offs

a. Motion Capture

i. Objective: Create a system using markerless motion capture that can track an athlete's movements with precision, either on a bike or from free form movements. The system should have no impact on the athlete's performance and allow them to move freely as if they are not being tracked at all. Multiple views of the athlete are needed so their motions can be viewed from different angles. The video should be recorded in 1080p at a minimum of 60fps. The system should provide consistent data readings and filter out improbable movements. The purpose of this system is to assist with bike fitting as well as capturing irregular free motion movements that could lead to an injury.

ii. Constraints:

Time: This system requires careful setup and calibration to operate properly. Experimenting with different camera placements is necessary to achieve acceptable results. Once calibrated, each frame of the video is processed one by one. This operation requires a large amount of graphics processing power which was not available to us. This resulted in rendering times up to 2 minutes in length for a 10 second video consisting of around 600 frames. This impacted our testing phase as most of the time we were waiting on renders to see if certain tracking points were tracked properly.

Cost: Our budget was less than $200 for the entire system, this budget is unrealistic when considering the system's objectives. When selecting a camera, we only found one that met the 1080p 60fps criteria that was within the budget for 4 cameras. Upon receiving the cameras, we did testing that brought up a few issues. First, the cameras recorded 1080p with an average of 40fps, not at the stated frame rate on their website. This caused problems with the tracking software as frames were consistently dropped, reaching frame rates as low as 12 fps. Second, after around 10 minutes of testing, the cameras stopped working altogether and produced a lined pink and white image. We are not sure of the cause of this, but can take a guess that the price reflected the build quality of these cameras. We rather would have purchased Microsoft Kinects like the ones we borrowed for our initial software testing, even though they were more than double our budget.

Environmental: We tested the motion capture system in six different environments which mostly consisted of classrooms we reserved on campus or our apartments. For the motion tracking to work properly, the subject needs to be wearing dark clothes and the background should be composed of light colors, with white being the best. From our testing, we've concluded that the best environment is one with no windows, a single light source, carpet flooring, white walls, and at least 225 sq. ft of open area. The tracking is heavily dependent on filtering out all light colors and focusing only on the dark ones. Windows provide dynamic light and we saw that tracking in a room with windows greatly impacted the accuracy as the light from the windows were not consistent as time passed. By having a single static light source such as an overhead light, the amount of light in the recording is consistent thought the entire session. Carpet flooring is also essential as we had lots of problems with reflections specifically on tile floors that caused misdetects of the virtual ground plane. By having the 5 properties of a good environment in place, the calibration error is greatly minimized and makes the subject much easier to track. We were able achieve errors of less than 1% when all these properties were considered.

Non-intrusive: The system should have no impact on the athlete's performance and allow them to move freely as if they are not being tracked at all. Our task was to find a way to track human motions without using special body suits like the ones that you see with ping pong balls for example. Determining what software to use was more difficult because most software requires the use of some sort of marker to track movements.

iii. Trade-Offs:

Cameras: We went with a more reliable type of usb webcam that is widely known as the PS3 Eye. The trade off was video quality, going from 1080p down to 480p, but the benefit of this was a consistent 60fps with with an average of 58.9 fps for a recording, which is much better than the results we obtained for the other cameras.

Software: We did not design the tracking software that we used in our project or for our testing. The iPi software we used was the perfect package and it would allow us to do everything that we were targeting. Compared to other tracking software, this one was relatively cheap for what it was capable of. To continue using the software we used multiple computers to keep getting the 30 day free trial. We did this because the client did not have enough money in his available budget for the software. The high price of the software truly reflects how difficult it is to code and develop a piece of software that is capable of markerless motion capture. With the other systems of this project in mind, it would have been unlikely that we would have been able to create a program like iPi ourselves in the given time (Figure 6).

b. Pressure System

   i. Objective: Create a system that can accurately measure the force an athlete applies in four different areas for each foot. The system should be non-intrusive and have no impact on the athlete's performance. The measurements from the sensors should be recorded along with the time of occurrence, saved, then processed along with the motion capture system to accurately depict the exact frame where a force was applied. The purpose of this system is to determine if an athlete is putting too much pressure on a certain area of their foot depending on what motion was being performed.

   ii. Constraints: The PCB should not affect the athlete in any way. The size of the PCB should be as small as possible and be able to be mounted on the area near the heel at the back of the shoe. The entire circuit should be practically weightless.

   iii. Trade-Offs: Extra battery to reduce board size. If going with the other option, the board would have been bigger and would have required the use of a single larger battery. Two batteries are needed for this circuits operation, but is overall better on the weight of the package and the PCBs size.

   The original idea was to mount the PCB to the athletes shoe towards the heel, but it was determined that the weight of the battery would affect the athletes movements. The trade-off was to keep the design as is, but instead find a way to place the PCB and batteries on the athletes ankles, using some sort of strap.

c. Data Analytics

   i. Objective: The objective of the data analytics program is to present the data gathered in a meaningful and efficient manner. This allows the user to interpret the data presented in a style that seems best appropriate for their needs. The program needs not to directly influence the user on what they are doing correctly or incorrectly. The program will give statistical data like angles and distance between body parts. The program will use the data exported by the camera system as its main component for the analysis.

ii. Constraints: The data analytics program should be able to present data to the user with 2.5% precision. This allows the user to be able to consistently use the data presented to them in a reliable way.

iii. Trade-Offs: The increased focus on precision will lead to a less focused attribute of accuracy. The data analysis may not always be an exact one to one measurement, but it will consistently give you the same measurements when used. This allows for a better view of how the data is changing over time rather than a completely accurate list of measurements that is not able to be consistently replicated.

d. User Interface

i. Objective: The objective of the user interface is to create a system that can display and review the information gathered by the cameras and pressure sensors. The user interface is also used to compare the information and data between sessions. This system needs to be easy to use, but also needs to be extendable to include multiple types of information for any given session. Additionally, the tool needs to be able to present multiple sessions of data in order for the user to compare the data from various sessions and view change over time.

ii. Constraints: The constraints with the user interface are the ease of use and the extendability. The interface needs to be ease to use and simple to understand since the client plans on bringing in athletes to use the web application and the athletes should be able to use the application. However, the client wants the interface to be extendable. He wants the application to support multiple types of files and be able to be improved to include more types of files in the future.

iii. Trade-Offs: The main trade off for the user interface came in the form of finding a balance between the constraints. On one hand, the design and interface should be as simple as possible while on the other hand, the client needs to be able to add new information to the user interface in the future. To find a balance, I broke the two constraints into two separate parts. The easy to understand UI is handled by the web app by making the front end user friendly while the back end has acceptable complexity for the client to understand and use while allowing the system to be improved over time.

C. Architectural Diagram, Design Block Diagram -- Modules, Interfaces

The block diagram can be seen in the Appendix under Figure 6.

D. Description of Modules, Constraints, and Interfaces

a. The constraints of the project stem from the multiple different aspects that this project covers. The first constraint comes from the rider comfort. Stickers are often used for image tracking and these can be seen as invasive - meaning this needs to be avoided as these would be placed on important joints that could inhibit the motion.

b. The other limitation needs to be that the sensors are wireless. Sensors when not fitted properly will affect how the rider is positioned on bike making the data misrepresentative and need to withstand hours of racing or training. The next limitation is price on the camera. While the price of camera was not specifically laid out, many cameras that are used for athlete body position are in the tens of thousands of dollars. In addition, the area used for these camera is quite large. The space specified for the system we are producing is not as large as these cameras require. The cameras must also be able to capture an athlete cycling at 150 rpm for cycling and 170 rpm for running. The price for the image capturing software is a final limitation. Maintaining an efficient, inexpensive system for our client is important as it is a small company.

c. The final User Interface needs to be interfacing with the image capturing system and the hardware system. The iPi system creates a CSV file that was created from the camera video. The CSV interfaces with the analytics which interfaces via the Django web application.

d. This also interfaces with the hardware system. The sensors will create a change in voltage. The voltage will be read and converted into force on the Arduino which saves them to the SD card. The card takes them and puts them to a .txt file. This file is read by Django and displayed alongside the image processed video.

3. Implementation

A. Implementation Diagram, Technologies, Software Used

a. Motion Capture

i. Technology/SW Used: Markerless motion capture. iPi Recorder/Studio 4.

b. Pressure System

i. Technology/SW Used: AutoCAD EAGLE was used to design the PCB for the pressure sensor system. The PCB uses through-hole parts and is powered by two 9V batteries. Each PCB is capable of tracking force applied to four separate pressure pads.

c. Data Analytics

i. Technology/SW Used: Matplotlib, a python library, was used as the to main component to display and render the analyzed data into graphs,charts, and other animations.

d. User Interface

i. Technology/SW Used: Django web application was used for the design of the user interface.

B. Rationale for Technology/Software Choices

The final interface will need to be both informative but basic. A coach will do the analysis based on the data we present. Therefore our assumption is the coach is a SME in biking and not web apps which is why this needs to be basic. Further the app needs to allow our client in to the backside to view the data if he so desires. Comparable software options analyze the angles of the athlete for the coach. This project needs a simple interface where the angles are displayed. The coach will make the call whether changed need to happen and should not be included in the web application.

a. Motion Capture

i. iPi: The next step in the processing the color and distance data that we receive, is via an image processing software. The difficulty with determining what software we will use is that typically the software and cameras come together. Therefore, we needed to find software that would allow a different camera. The other limitation in regards to software is the price. Many software applications require subscriptions and renewals. This was something we wanted to avoid for our client making it as user friendly as possible.

We came to the conclusion of using iPi. This software can take the image from a set of cameras and turn it into multiple formats for analysis purposes. Additionally, it allows for easy exportation of data between different sources, which is beneficial for our web application.

b. Pressure System

   i. AutoCAD EAGLE: The pressure system was designed and simulated in AutoCAD EAGLE. The reason I chose this program was that it was heavily recommended by an ETG employee over Multisim/Ultiboard, which I had used in previous courses at Iowa State. EAGLE has a library import function that I used to insert the exact parts and their hole spacings from digikey.com, which is excellent for preventing part hole/spacing issues.. In addition to this, EAGLE files can be directly uploaded to OSHpark.com, which is where the boards were ordered from, so the client would not have to deal with a zip file full of Gerber files. The decision to use EAGLE benefited both our team and the client and I got to learn a new piece of software in the process.

c. Data Analytics

   i. Python: Python was chosen as the main language of development because there are many common libraries for data analytics and presentation. These libraries will be helpful when we need to relay the data back to the user. Python is also a scripting language which allows for easier integration with web applications, which is what the user interface will be. Python provides lightweight use and easy to learn syntax which is very important when the timeline of the project is very short.

   ii. Matplotlib: Matplotlib was one of the main Python libraries chosen to use for the data analytics because it provides fast rendering of the frame by frame data. With our cameras shooting at 60 frames per second , it is crucial that the program be able to process it in a timely manner. It has a plethora of graphical libraries to visually transform our data which is important because we need to present it in a meaningful way. The library is also open source which means no additional costs which was a benefit to our client. These traits accommodate project's needs for presenting the data in a meaningful and and efficient way.

d. User Interface

i. Django: When looking into different tools we could have used to make the web application, Django stood out as a popular and effective tool. In addition to being one of the most popular tools, Django works with many of the libraries that Python has to offer, ideally allowing for seamless integration between the web application and the other tools.

C. Applicable Standards and Best Practices

a. Motion Capture

i. Camera Selection: For the camera, there are many standards on the market and in general, an increase in price produces a higher quality image at a higher frame rate. We researched cameras ranging in price and image quality. Due to the constraints of the client, it was determined that 60 fps is the standard that is needed. This frame rate was determined by finding the minimum required rate then doubling it. This gives a buffer in case there there is an anomaly where the bike goes above 170 RPM quoted by the client.

ii. Software: Current market standard software is available including Dartfish, which has a $70 monthly fee and works with any camera (Figure 8). This software is highly recommended by people in the industry and would likely be the software used if the camera system does not come with its own software. The Dartfish software gives the angles of various parts of the body for athletes on camera. This software takes a video and shows the angles of the athlete's back, legs, and arms over top of the athlete. The Dartfish software does not allow the user to analyze the data any farther than the video overlay. Other software requires the user to wear a specific suit inside of a highly calibrated room. While this software is very precise, it is expensive and inhibits the user by requiring many sensors to be placed on the body. The goal of our project would be to create a less precise version which is cheaper and less inhibitive.

b. Pressure System

i. Sensors: The range of the sensors that could tolerate the force was selected based on the finding the maximum expected force then creating a buffer for error. With this in mind, the force of the sensor was selected based on the stats from running and then selected with a buffer as well for an error.

ii. Standard: Leomo is the current leader for pressure sensors on the market. The baseline sets what is expected for accuracy, precisions, user interface, features, and ergonomics (Figure 10).

c. Data Analytics

i. Standards and Best Practices: the standards for the data analytics were self imposed using best practices. We kept data points consistent among the sessions and athletes.

e. User Interface

i. Standards and Best Practices: the user interface was modeled after the current standards used by the competitors in the market with the added feature of multiple athletes over multiple sections.

f. General Engineering Standards:

i. Engineering standards are very important to the work we accomplish as engineers. Standards allow us to have minimum performance, meet safety requirements, make sure that the product/system/process is consistent and repeatable, and provide for interfacing with other standard-compliant equipment. For this project we will usings standards from IEEE that deal with the development cycle and measurements.

ii. IEEE 12207 - 2017 sets a framework for the how the processes, activities, and tasks will be used during the supply, development, operation, maintenance, and disposal of software product. This standard is relevant to our project because we are making a web application for our client. The web application will need to go through development, operation, and maintenance which are all activities mentioned in the this standard.

iii. EEE 15939 -2017 is standard for system and software engineering disciplines. It sets a model for what measurements are to be required, how the analysis of the measurements are to be applied, and how to tell if the analysis results are valid. This standard is relevant to our project because we will be measuring different motions of an athlete. This will include the angles of their body as well as their pressure data using sensors. In this standard they talk about how the analysis of measurements are to be applied and how to tell if the analysis results are valid. This is very necessary to us as we need to make sure that we are interpreting the measurements in the correct way and then verifying that our interpretations are valid.

4. Testing, Validation, and Evaluation

A. Test plan

a. Motion Capture

There are two main parts, the first part involves calibrating the cameras which is required every time they are moved, the environment changes, or the error is greater than 2.5%. The second part is where you would record the athletes movements and manually look for inconsistencies before exporting the data.

b. Pressure System

The pressure sensor system will be tested for three different components. These are overall reliability, accuracy, and athlete comfort. The reliability was tested by the multiple different movements that it experienced. The multiple different locations required lots of different movement. The circuit was able to withstand the multiple different transportations. The second aspect was accuracy. The accuracy was tested at State Gym. Ryan brought the sensors to the gym where he used different weights to determine the voltage that was read. This was then converted into a force. This generated an equation to take the voltage and turn it into a force. The final test was athlete comfort. This was tested during the film testing. The sensor was placed on the foot of the athlete when walking around. The athlete was able to determine if the sensor was intrusive. Also, the athlete walked across the room and determined if the sensor was in the path. It was laid down and the athlete walked across the room and determined when they stepped on the thin film resistor. When the athlete didn't feel the sensor it was deemed acceptable.

c. Data Analytics

The data analytics program will be evaluated based on its ability to have precise data measurements. The precise measurements will be tested by using unit testing. A particular body position will be chosen to track its angle. The data analytics program will then track the specified body position and output the angle to the user. The consistency of that angle will be tested through multiple tests of the same body position. The accuracy of the angle will not be as important as the precision of the angle. If throughout the attempts the angles is within the required 2.5% precision, then it sufficiently passes this test. The second part of testing will be for integrating will the camera software. The program will need to be able to parse the output of the camera software.

d. User Interface

The test plan for the user interface has two steps. The first step in testing the user interface involved unit testing. The unit testing consisted of ensuring that each of the functions and pieces of code works as expected. This part of the testing was to take part as the code was developed to ensure that not only do new features function properly but also they do not break existing features. The second part of the testing was user level testing. The goal of this testing is to ensure that the web application is easy to use for new or inexperienced users. This second form of testing came later in the project since it does not make sense to do it early when there are little to no features.

B. Unit testing

a. Motion Capture

We tested the cameras by having them record video for an extended period of time. The entire session not one of the cameras turned off, which was impressive since they were on for around 8 hours. The frame rate of the 4 cameras we used had a combined average of 58.9 fps after this session, which is within the 2.5% acceptable error range.

b. Pressure System

The unit testing for this project system includes: Arduino, SD card, SD card reader, circuit, and sensors.

The Arduino was tested using a simple circuit. Signals were sent to an LED and created a pattern for the light to emit light. If this worked as expected, both the Arduino and software understanding worked.

The SD card was tested using the computer. If the SD card worked in a computer reader that the SD card was working.

The SD card reader was tested using basic SD Card software. The initial test was determining if the Arduino was able to pull information about the SD

Card. One it was able to read the card then it was written to using basic code. From there the voltage readings were read to the SD Card.

The circuit was tested using a multimeter. The circuit was hooked up and fed a voltage. The output voltage was measured. Once the output and input voltage was determined to match it was accepted that the circuit was working.

The sensors were tested using a simple circuit again. The simple circuit was hooked up with a multimeter. When the pressure sensor was pressed it was determined that the voltage was changing on the multimeter and confirmed that the sensors were working. From there the entire system was integrated for further testing.

c. Data Analytics

The unit testing for the data analytics program involved testing the functions that calculated the angle of the body part that was being tracked. The predicted angle was given and then the output was tested against it. This was done multiple times to check the precision of the output. If the output was consistently within the 2.5% required precision to what was outputted in the previous tests then it had passed the precision test.

d. User Interface

The unit testing for the user interface involved testing functions and individual pieces of code to make sure they provide the expected output and do not cause issues if they received unexpected input for some reason. These test cases were written to test new pieces and we used throughout the project to ensure that features did not become broken with the addition of new features.

C. System integration testing

a. PCB and Arduino

To integrate the PCB and the Arduino, a simple 1x4 terminal block was soldered onto the PCB. This terminal block takes the four voltage outputs from the MCP6004 op amp and sends that voltage into the Arduino's analog inputs. From here, Arduino code reads the voltage and converts it into a force value. This force value is recorded onto a SD card along with the time the reading was taken. This integration of the PCB and Arduino was verified by using a multimeter to verify that the Arduino was reading the actual voltages that the circuit outputted.

b. Data Formatting

In the data analytics program, one of main points of integration is with the camera software. The camera software will output the data points needed for analysis. The analysis program will read and parse that data for further use. The size and amount of data points will change based on how long the video recording is.

To test that the data analytics program can read any format the data is in, we used various test cases. The test cases included increasing the length per row and and column of the data file, making the rows and columns uneven, and adding unknown characters to represent corrupted data so that the user could be alerted and the data could be resent.

D. User-level testing

The user-level testing was focused on making sure that the users would be able to understand and use the tool with little to no experience. This testing was done in two steps. First, as new features became available in the application, a test of the application was created from the user perspective. The user perspective tests were modeled after the use cases for the application and each use case had at least one test ensuring the user would be able to complete that use case. Once most of the features were available, the user-level testing evolved to include giving the software to individuals who had not seen the tool and asked them to complete tasks on the application. This started by testing the other group members but expanded to include non-engineers and even our client.

E. Evaluation

a. Performance Metrics

i. The precision on the cameras is to be within an acceptable error range of 2.5%.

ii. The pressure sensors need to be able to support athletes who weigh up to 200 lbs. The pressure sensors were calibrated for a maximum force of 200 lbs.

iii. Rendering time should be within 2 minutes for every 10 seconds of video captured captured.

iv. Video capture needs to be recorded at a minimum resolution of 480p at 60 frames per second.

b. Test Cases

i. All black clothed athlete do jumping jacks. When the athlete claps together and when the hands reach their sides, the time should be captured on a separate system. This time should be marked and compared to the iPi software system. The time should be within 2.5% of each other.

ii. An athlete under 200 lbs and over 200 lbs will walk using the sensors. Then run. It should be able to pick the athlete under the limit and is not expected to pick up the force from the 200 pound athlete. The accuracy should still be in place for the athlete under the limit.

iii. Rendering Time will be measured from a 1 minute video. It is expected to happen along a linear path even as the movies get longer.

iv. We looked at the output from the cameras both at about 2 minutes and 15 minutes of use. We looked at the quality of video produced after these times and compared with the required metrics specified at the beginning of the project.

c. Evaluation Results

i. Our cameras saw at best .3% error with our readings, however on average we saw about 3.2% error with our readings when the calibration was set up properly.

ii. The pressure sensors are rated for up to 1000 lbs and we saw effective readings for up to 200 lbs in our testing.

iii. Our client purchased an alienware computer for their own use and we tested the software on their computer. When we used it, it took about

iv. Our first set of cameras claimed to work at 60 fps and 1080p. However, after testing, we found that they could not operate at the specifications. Firstly, they did not work occasionally, and when they did work, they would overhead and drop frame rate, down to 40 fps within 2 minutes of use. The second set of cameras claimed 480p and 60 fps. These cameras were capable of these metrics and still held at these values after prolonged use.

5. Project and Risk Management

A. Task Decomposition & Roles and Responsibilities

i. Monte: Lead the User Interface design portion of the project. His responsibilities lied within taking the data from the analytics and creating a web application that would allow for storage, comparison, and video overlay.

ii. Nathan: Lead the analytical portion of the project. The goals of the analysis program were to present the data in a meaningful and efficient manner.The data was to be extracted from the camera software and parse into data that could be analyzed.

iii. Ryan: Lead hardware designer and system tester. Ryan designed the pressure sensor schematic and board, selected and learned how to use the motion capture software, and performed the testing associated for each system. Ryan had a major part in designing each version of the PCB and choosing the appropriate components to optimize the PCBs size and weight.

iv. Maddie: Focused on the design for the Arduino and selected the pressure sensors. Maddie worked with the client to determine the force required and selected sensors off the specifications the client saw fit.

B. Project Schedule - Gantt Chart

i. Proposed
This project was divided into three different pieces. Phase 1 is the image capturing system. Phase 2 is user interface. Phase 3 is pressure sensor system. Phase 1 will consist of implementing an image capturing system to collect data to monitor the athlete's motion. A program will be developed to analyze the angles of the data collected by the cameras. A web application will then be created to display the motion analysis program and raw data captured by the camera. Phase 2 will consist of creating the hardware to detect the athlete pressure readings using sensors, displaying that data onto the web application, and creating the back end. Phase 3 will have the

final rounds of testing for the image capturing system and hardware sensors to make sure they meet the constraints of the project. Phase 3 will also include the final documentation and presentation of the project to our client.

ii. Actual

The actual Gantt Chart and schedule was pushed back quite severely (Figure 3). The implementation of the cameras set back the project quite a bit. Due to cameras not meeting specs it was difficult to get data that could be used for the iPi and the user interface. This pushed the whole project back by a month. Determining what cameras to select instead was time consuming as making sure the specs were within scope. The Arduino also pushed back the testing due to having some small issues with the integration.

C. Risks and Mitigation: Potential vs. Actual and how they were mitigated

Potential Risks that we saw were the moisture. The concern was that the sweat from the athlete could cause a humidity that could interfere with the circuits along with potentially loosen components.

Actual risks that we ran into was the camera calibration. The room space used was not large enough even though it met specs. The lines of the rooms along with color combination created issues with the calibration. This was mitigated by renting a bigger room that would suffice with our extended requirements. Having carpeting was useful for recording the multiple different test data points.

The colors of the clothing was an additional place that an error was seen. This happened because the clothing was not all dark enough. All black outfits needed to be used to create a fully integrated and smooth recording. This was mitigated by wearing dark clothes as this is recommended by the iPi software.

D. Lessons Learned

The largest takeaway from this project was the planning and ordering components within specs. While many of our components were within the scope of our specifications they did not perform within the specification. Going forward, it was important to understand that contacting the suppliers and other past users is important.

6. Conclusions

A. Closing remarks for the project

For our senior design project, we have been given to opportunity to design a system for tracking the motion of various athletes. Our client Nathan Johnson, owner of Precision Performance Cycling, has given us this task as a way to further the performance of his own clients. To fulfil this task, we will be implementing motion tracking cameras and pressure sensors to be positioned around the athlete to give us optimal readings of motion for further analysis. All data will be be readily available through the interactive web application being built for the users of the motion tracking system. The web application will create an easy and organized means for the athletes and their coaches to analyze the data as they see fit and create optimal changes to their movement based on the information that they are seeing through our web application. This project is intended as a way to give athletes a means of optimizing their performance through analysis of their own motion. Our team is excited to help Nathan and rest of Precision Performance Cycling in their goal of improving and developing their clients.

B. Future work (potential directions)

Additional sensors can be added for further data collection. Potential sensors could be accelerometers. This would get data for the athlete's acceleration as well as position for their body. This would be helpful for following the body as it fatigues throughout the workout. Live video footage is an additional place for improvement. This would allow for the coach to watch their athlete live and see how the terrain affects the performance. Additional areas for pressure sensors will be an area for future work as well. Knowing how the athlete is applying pressure to different areas will be useful. These areas could include the saddle and handlebars. The foundation for the sensor system has been set up and will allow for easy addition of the additional sensors.

References

Project Block Diagram "SDMay10-01Block Diagram." Matlab - Documents, Ryan Hansen.

Microsoft Kinect "Microsoft Kinect - Gaming Breakthrough." Microsoft - Product, www.microsoft.com/.

iPi Software “IPI - Member Network.” IPI - Member Network, www.ipiphoto.com/.

Important Body Points to Track “Body Position Matters - Bikers’ Guide.” Biker’s Guide - Online.

Dartfish GUI “Dartfish User Profile.” Dartfish. General Membership, www.dartfish.com/.

Thin Film Force Sensor Resistor “Force Sensor.” Omega Catalogs and Specs. Omega Online, omega.com.

Sensor Data GUI “Pressure Sensor Biometrics.” Google Image Search.

Typical Athlete Jersey “Outfitted - Biker’s Guide.” Biker’s Guide - Online.

Typical Fit of Bike Attire “Outfitted - Biker’s Guide.” Biker’s Guide -Online.

Appendix

Process Flow

1-2: Successfully understand the needs to define the problem

4-3: new ideas made from prototypes created

5-3: Testing reveals new ideas to solve the problem

5-2: Testing provides further insight that can solve the defined problem

5-1: Users/client learn from testing and want further functionality or greater understanding

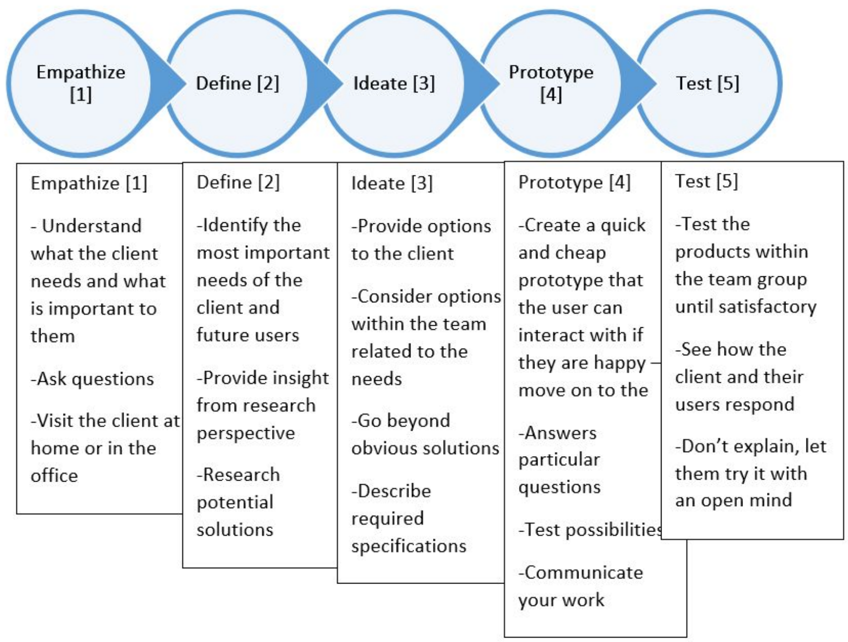


Figure 1. Design Thinking Model

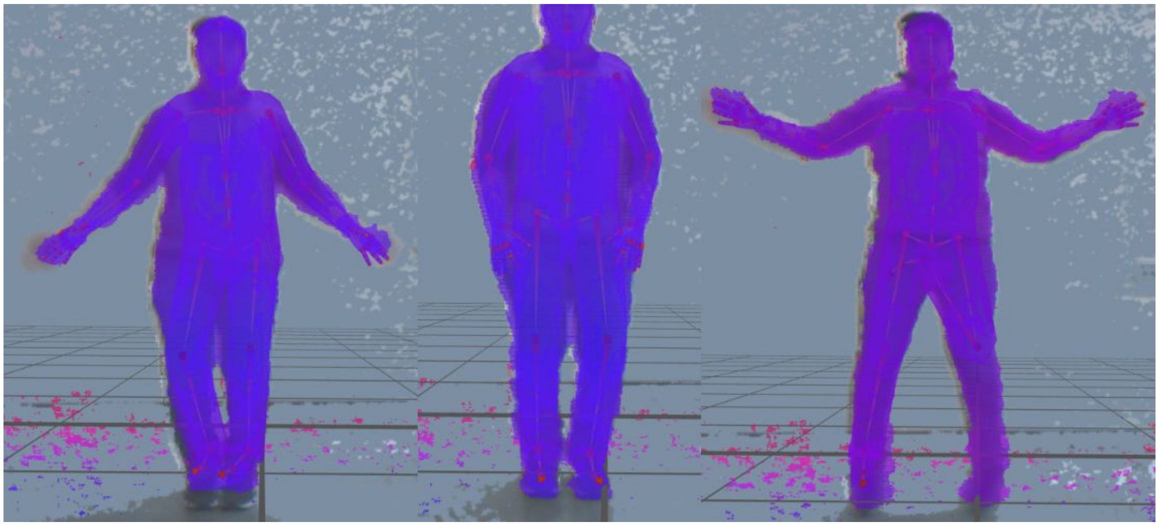
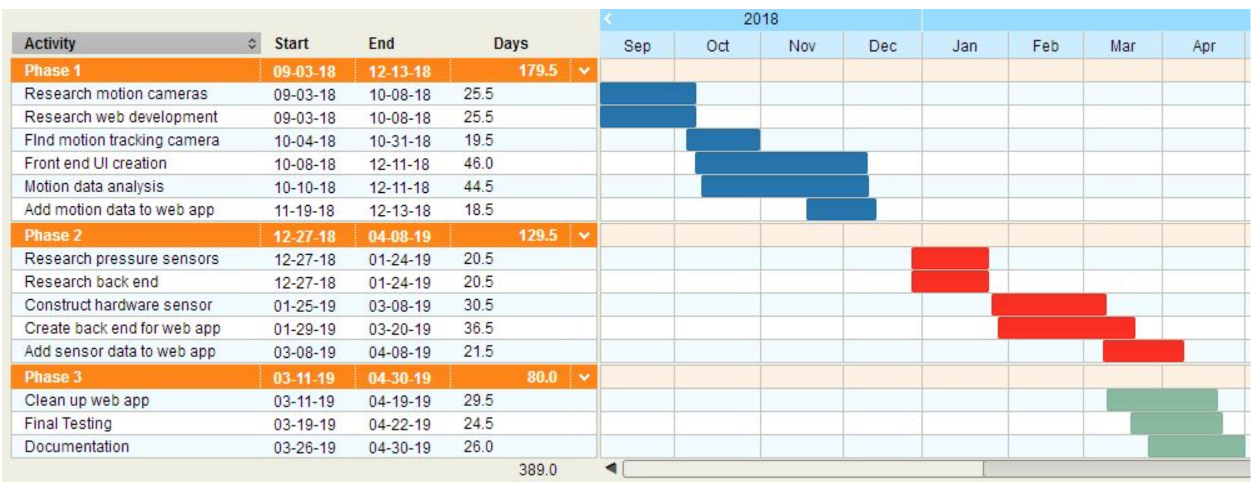


Figure 2. Kinect Motion Tracking Test



Activity	Start	End	Days
Phase 1	09-03-18	12-13-18	179.5
Research motion cameras	09-03-18	10-08-18	25.5
Research web development	09-03-18	10-08-18	25.5
FInd motion tracking camera	10-04-18	10-31-18	19.5
Front end UI creation	10-08-18	12-11-18	46.0
Motion data analysis	10-10-18	12-11-18	44.5
Add motion data to web app	11-19-18	12-13-18	18.5
Phase 2	12-27-18	04-08-19	129.5
Research pressure sensors	12-27-18	01-24-19	20.5
Research back end	12-27-18	01-24-19	20.5
Construct hardware sensor	01-25-19	03-08-19	30.5
Create back end for web app	01-29-19	03-20-19	36.5
Add sensor data to web app	03-08-19	04-08-19	21.5
Phase 3	03-11-19	04-30-19	80.0
Clean up web app	03-11-19	04-19-19	29.5
Final Testing	03-19-19	04-22-19	24.5
Documentation	03-26-19	04-30-19	26.0
			389.0

Figure 3. Original Gantt Chart

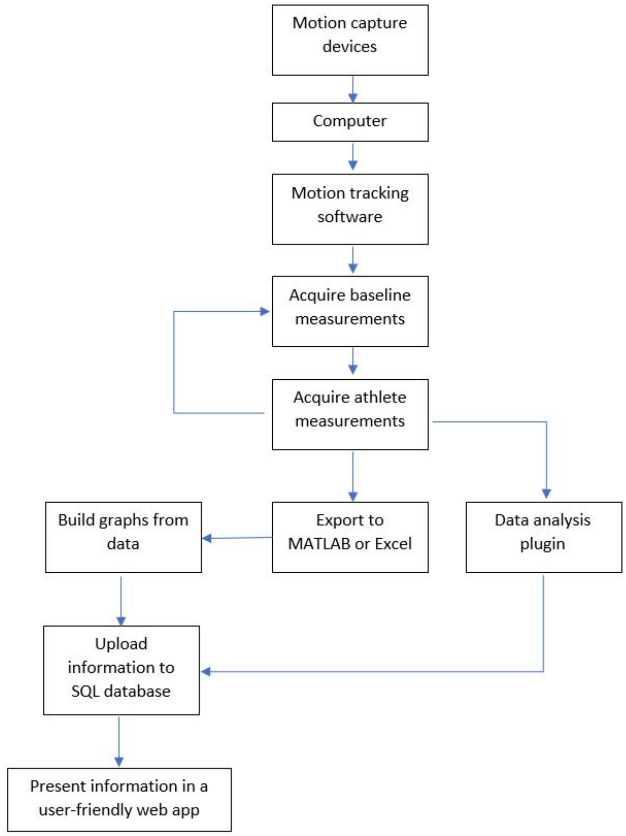


Figure 4. Block Diagram

Figure 5. Microsoft Kinect

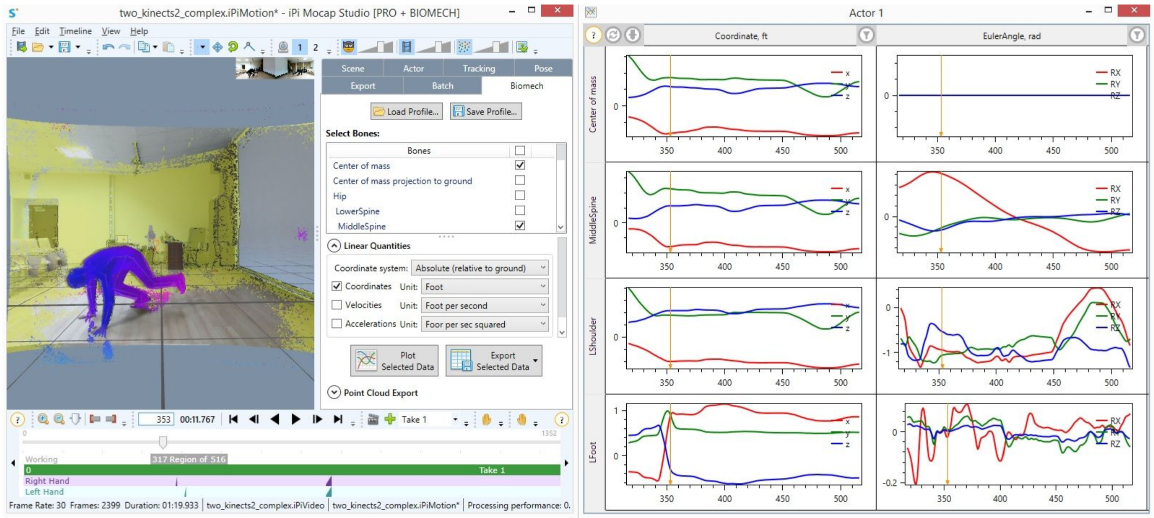


Figure 6. iPi Interface

Figure 7. Pivotal Measurements for Cyclist

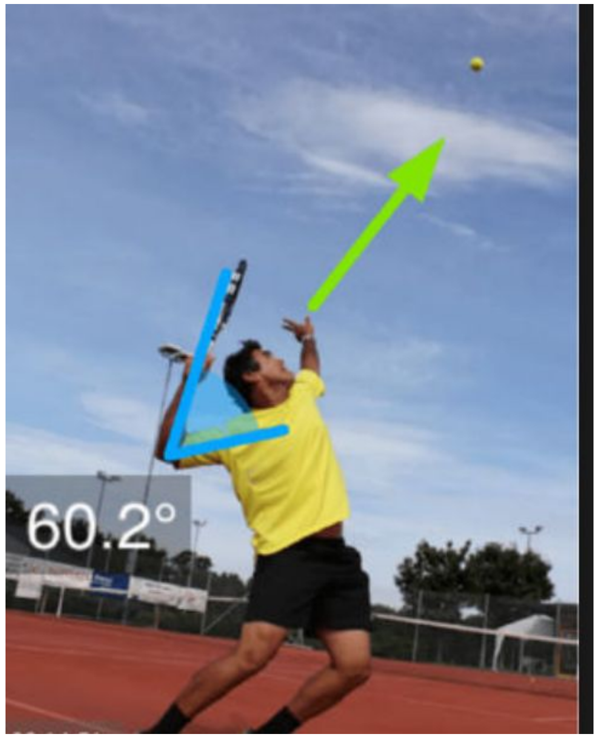


Figure 8. Dartfish GUI

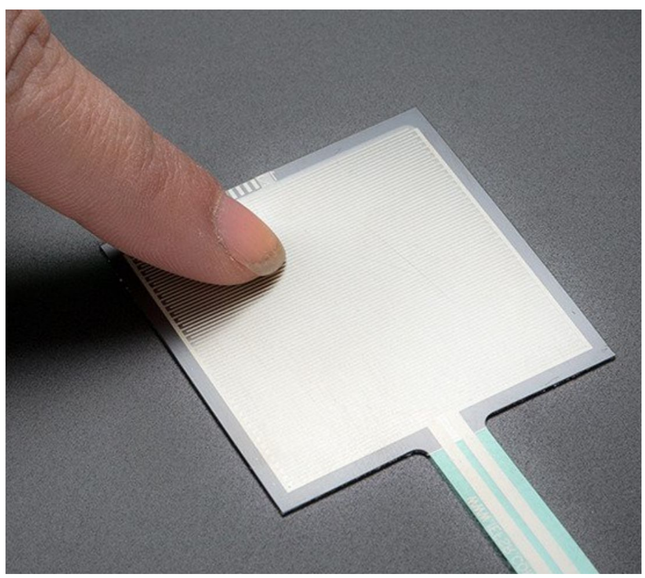


Figure 9. Thin Film Flexible Force Resistor

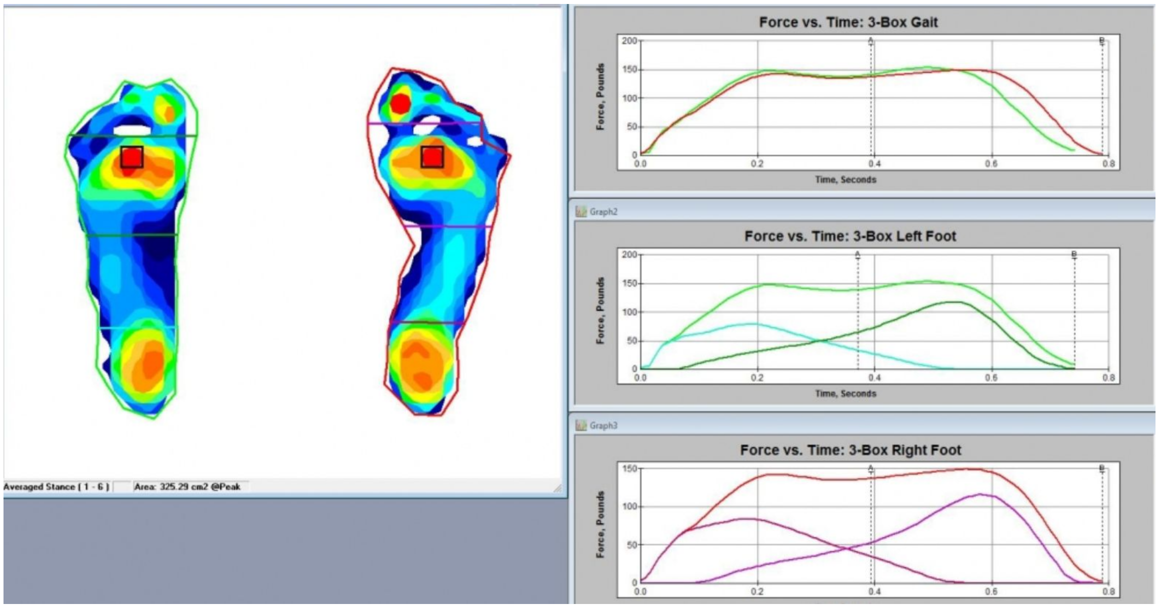


Figure 10. Gebiomized

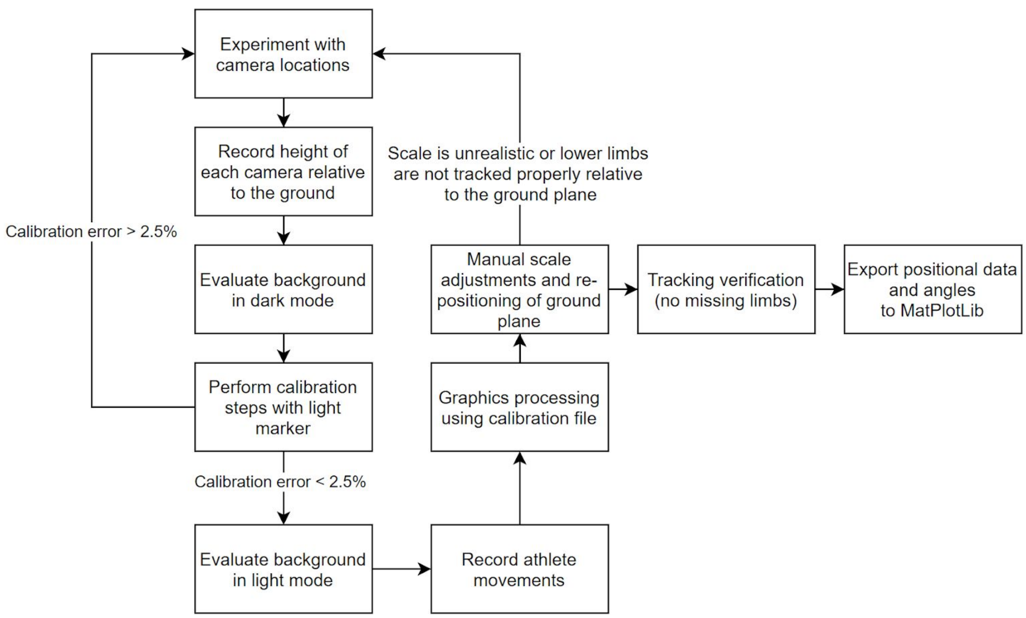


Figure 11. Camera Test Plan Block Diagram