Product: Vitruvian

Team: Vee

Abstract

Our project aims to help improve people's posture and health while working from home by monitoring sedentary activity and posture, notifying them via an app or by tactile feedback on a worn device.

Progress has been made into creating a basic setup of the vital components and analysing sensor readings to demonstrate some notion of slouching or sitting. Due to the technician's time constraints, we have been unable to make progress on communication with the app. However, we have created solutions to keep steady progress during these setbacks. Our group has a better understanding into the field due to research into methods for detecting both sedentary activity and slouching, settling on what seems to be suitable models for both. For the hardware, we are investigating adding further complexity to the project by adding additional sensors for the shoulders, but this is at an early stage and not a priority at present. We are also at an early stage of creating a 3D model for the design and talking to the experts about the best material for the casing.

1. Project Plan Update

Task Name	Status
Design Android UI	achieved
Create Android UI	achieved
Research components	achieved
Test sensor	In progress
Check feasibility	In progress
Combine all components	Not achieved
Slouch detection	In progress
sedentary detection	In progress
Notification functionality	Not achieved
Communication between app and hardware	Not achieved
Design model for hardware device casing	In progress

Table 1. Planned goals for up to 2nd Demo

Many tasks have not been achieved according to schedule. This was mainly due to the technician's time constraints that have blocked some key tasks. However, we have tried to be proactive and found solutions and other tasks to do.

Firstly, one of our largest blocked tasks is using the sensor components, due to the virtual setup of SDP. At the moment, we are relying on the technician's to perform essentially all testing for us. Our solution for this has been to create a new iOS application which will record the phone's accelerometer and gyroscope data and export it via email. This will help us to speed up development on the slouching and sedentary detection algorithms.

Another notable issue is the progress in slouch detection being bottlenecked. The technicians have not yet been able to get inputs from both IMUs available, and so we have only been able to do basic back angle measurement, but not the curve detection needed for our full system. This also means we have not implemented our slouch notifications.

Bluetooth connectivity has also not been achieved as the technicians are yet unsure how to implement this. However, our temporary solution will be to create a mock input service, which will provide the Android app with "fake" generated data as if it were receiving it from the hardware. This will allow us to demonstrate functionality of the app and may also benefit the hardware team, as they could use the app to debug the hardware.

We have spoken to the experts about creating a silicon-mold for the casing of the device and are in early stages of creating a 3D model. The design is being blocked though, as there are still some uncertainties due to the changes in the hardware and sensors in progress.

Considering these issues, we have redone our Gantt chart (see Appendix D), and re-prioritized the tasks on our Trello board. We are continuing to handle work organisation at our weekly meeting. The assignment of group members to tasks was:

- **Alasdair**: Working with technicians to get I/O working and writing angle/posture detection system and voice recording for Demo video.
- **Andrew**: Working closely with the technicians and also our own software team to get prototypes built and tested in labs.
- **Mohamad**: Working on a sedentary detection deep learning model and working with other hardware team members to debug initial code for capturing accelerometer inputs.
- **Vincent**: Creating a simulation environment of our device in Webots and ROS.
- **Anelise**: Worked on the Bluetooth connection and other features for the Android App.
- **Morgan**: Medical research into slouching and the body to aid other teams and created Demo video.

- **Jake**: Designing UI for Android app prototype, created iOS application to record sensor values and designed slides for Demo video.
- **Yining**: Researching design and developing 3D model.

2. Technical details

2.1. Simulation

Since the torso of the existing models cannot be bent, we have modified the proto file of the existing proto (Pedestrian PROTO) in Webots. We have used MATLAB to manipulate the data in `PedestrianTorso.proto`, so that the main torso has been divided into three parts (upper torso, middle torso, lower torso). We have added two `HingeJoints` so that the torso can be bent as we desired. The controller for the model is also finished, and can be used to modify the angles of the joints in a very easy way.

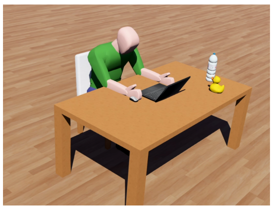


Figure 1. Simulation environment

2.2. Posture Detection

Using a single accelerometer, we implemented a simple algorithm to calculate back angle based on accelerometer input.

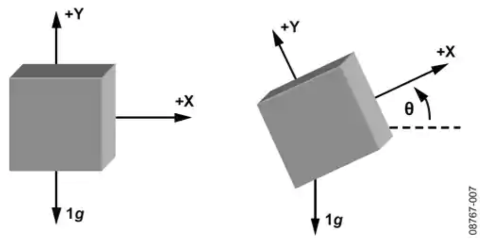


Figure 2. Detecting angle with an accelerometer, via (Digi-Key)

By taking the inverse tan of the ratio of the force of gravity along vector x and the force of gravity along vector y we can calculate the angle between vector x and the gravity vector.

$$\theta = tan^{-1}(\frac{A_{x,OUT}}{A_{y,OUT}})$$

Using this system, we are able to detect when someone is leaning too far forwards or too far back, and so we can carry out a basic form of bad posture detection.
Due to our current inability to read input data from more than one accelerometer, we have not yet implemented the full system using both angle and curve to make nuanced decisions about slouching.

2.3. Slouch Notification

Our slouch detection system is still in progress, but we were able to control our buzzer via software.
The buzzer is capable of playing different pre-programmed outputs (i.e a single sharp click, a long buzz, several buzzes with interludes etc), and so we have confirmed that we will be able to notify the user about exactly what's wrong with more nuance.

2.4. Hardware

Much of the hardware progress since last demo has been testing the devices we have access to in the labs. Here are a list of components being used:

- **Processor** Raspberry Pi Zero W
- **Accelerometers** MPU-9250

  We are using two IMUs to measure both angle and curvature of the back using this library.
- **I2C multiplexer** Adafruit TCA9548A I2C multiplexer (temporary)

  An I2C multiplexer will be required to read data from multiple sensors in our setup.
- **Tactile Feedback** Seeed Studio mini vibration motor on Adafruit 2305 controller board

  Tactile feedback is provided by a small vibration motor controlled by a controller board to connect to the Pi. This has proven easy to control and use, however we have run into issues surrounding the library being for Python 3/circuit Python rather than Python 2.
- **Flex Sensors** Flex Sensor 2.2" or Flex Sensor 4.5"

  The flex sensors have not been delivered yet, so testing to determine suitability has not been conducted.
- **Lightweight Battery**

  Ideally, we will use a lightweight battery. However, for now we are using a power bank for our prototype.

In spite of facing challenges our testing has shown promising results, our hardware seems to be capable of our requirements, and we hope to move to multi-sensor testing and full functionality over the next few weeks.

2.5. Android App

GitHub repository is available here. We have made a start on developing the code to communicate with the device, however this has not been completed due to the reasons already mentioned.

To connect to the device we create a `BluetoothSocket` instance by calling `createRfcommSocketToServiceRecord(UUID)`, the UUID is the same as the one used when it called `listenUsingRfcommWithServiceRecord(String, UUID)`.

Another new feature is that users are now able to watch recommended training videos and tutorials on how to achieve a better posture (implemented using: YouTube Android Player API).

2.6. Generating Data

The process for our team to receive input data from the sensors at the moment has been to ask the technician's to move the sensors in specific ways. Using SSH into DICE, we then connect to the Pi to retrieve the results and view a video of the technician's test.

This makes developing an algorithm for slouch and sedentary detection difficult. As we are only using a single IMU at present, our short-term solution was to build an iOS application to generate data instead.

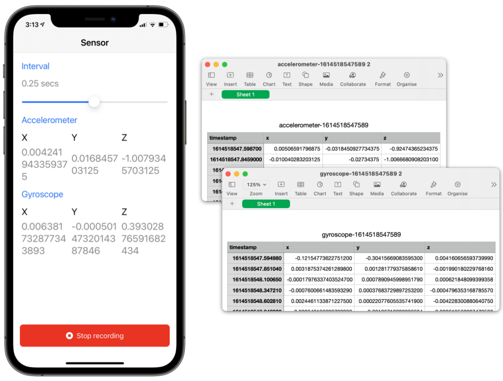


Figure 3. Recording App design and functionality

The app is very simple, it allows the user to click "Start recording" which will start recording **x**, **y** and **z** values of the accelerometer and gyroscope until the user selects "Stop recording". The app will then allow the user to export two CSV files that contains timestamped readings for both sensors. The app also allows the user to set a custom interval in seconds to record readings at.

This application will enable us to generate our own expected data to more easily determine what thresholds are needed and to develop our algorithms from home.

The application was built using Swift programming language and Xcode as an IDE. The code is available on GitHub here. The app can be downloaded via TestFlight here.

2.7. Design

We have settled on an undershirt-based form factor. There are existing products that contain a pocket for carrying a device, used by professional athletes for GPS tracking during training. A typical vest can be found online like here and shown in Figure 4. We have not had any contact with the University Sports department yet, but this is where we will look to source this item from.

Figure 4. Example undershirt design (STATSports)

The design will look like the model in Figure 5. It is a rectangular shape, with a curved side that will mimic the curvature of a human's back. Figure 5 shows a logo along the flat side of the product and a charging port along the shorter edge of the product. The exact size is not yet decided as it will be based on the final component selection. However, we are aiming for it to be no bigger than a palm size.

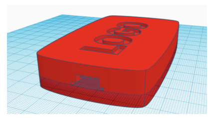


Figure 5. Casing design: with logo and charging port

We plan to use a silicon-rubber-based mold to construct the housing, which will help protect the device. We will keep in mind the need to separate the CPU and battery for heat purposes.

2.8. Software

Posture detection

Our current implementation for this demo uses one IMU and can detect leaning by:

1. Calibrating the accelerometer output within a time interval of 4 seconds and storing them.
2. Measuring the ratio of `y` (up and down) and `z` (forwards and backwards) axis on the accelerometer in the IMU.
3. Taking the inverse `tan` to calculate the angle.
4. Measure the angle difference between the calibrated state and the state when POLL command is used.

In the evaluation section of this report we have performed quantitative analysis on the accuracy of the IMU system.

Sedentary Detection

We have trained two deep neural network architectures and we are currently planning to test them on our own test data. The first one is a **Long Short-Term Memory model (LSTM)** and the latter is a **Convolutional Neural Network (CNN)**. Below we will list the reasons behind our choice of these two neural network architectures.

LSTM is a modified recurrent neural network that has feedback connections. LSTM is capable of processing entire sequences of data which is suitable for voice or video data, or in our case, segments of sensor inputs. LSTM is also known for making accurate classification predictions based on time series data.

CNN has a standard feedforward neural network architecture with the addition of convolutional layers. It is very common with image data processing. Convolutional layers are good at identifying patterns in fixed length segments where the position of a pattern in a segment is irrelevant. Therefore, they will be useful to detect patterns in acceleration values.

The LSTM model reported an accuracy of 83% on test data from (UCI). The CNN model reported an accuracy 91% on test data from (UCI) and an accuracy of 95% on test data from (Kwapisz et al., 2011) (confusion matrix in Appendix E).

The next step is to test these models with our own test data, optimise the model with better accuracy and deploy it to work simultaneously with the buzzer and the Android app.

3. Market Research

Target Audience: Students, adults working from home.
Age: 18-50 years

Problem: Back pain is directly related to poor posture while sitting and standing. Those who spend a prolonged time sitting with poor posture often develop muscular imbalances leading to tight chest muscles and weak back muscles.

Market Competitor: The Upright Pro system is a device which helps provide a sensory reminder to alert users when their posture is poor.

Relevant Study: Columbia University conducted a controller study (Bilal El-Sayed) with patients aged 18-50 years with posture-related lower-back pain. The study determined if patients who used the device above demonstrated improved pain control and self-perception of posture compared to those given standard ergonomic instruction.
The Numeric Pain Rating Scale (NPRS) and the PROMIS Pain Interference Short Form 6b were used to measure. The paper concluded that patients wearing the tech device showed significantly better results.

Our Product: Vitruvian aims to improve this device by adding multiple features such as tracking your progress, providing exercises and monitoring your posture from your computer camera too.

Statistics: According to the Office for National Statistics, 46.6% of people in employment worked from home, 86% of them as a result of the pandemic. Poor posture can lead to back pain and trigger higher levels of stress, anxiety and depression which cause reduced efficiency.

Our Goal: We believe that employers should invest in our product for their employees to maximize productivity and to ensure positive health and well-being in the workplace.

4. Evaluation

We sampled a small collection of input data from our accelerometer held flat and at 90° in order to prove that it is capable of accurate measurement at both angles (critically that it does not suffer from a "division by zero" issue as the force of gravity approaches zero along the measurement vector corresponding to the denominator in the ratio).

As can be seen in Appendix C, the accelerometer suffers less than a degree of deviation when performing angle measurement. We unfortunately found that the surface we were measuring against was not perfectly vertical, and for the next demo we aim to either perform testing in Webots or ask the technicians to create something with known angles that can be used for real-world testing.

5. Budget

Appendix B shows a current estimated total cost to make our product.

Appendix A shows an estimated breakdown of how many technician hours have been used up to this point. It is difficult to quantify our technician time usage due to the format of the labs.

Using TPU to print the casing for the hardware, the material would cost \$87 - \$110 per kg (All3DP, 2019). If we were to use an estimate of 0.5kg of TPU material for multiple prototypes, this would be roughly \$49.25 (£35.37).

If we are to continue with our design which uses an undershirt, we may be able to use existing vests which have a pouch for storing GPS trackers for athletes. A typical vest like this costs £25 (STATSports).

6. Demo Video

Watch here.

Appendix

References

All3DP. How much do 3d printing materials cost?, 2019. URL https://all3dp.com/2/how-much-do-3d-printer-materials-cost/.

Bilal El-Sayed, Noura Farra, Nadine Moacdieh Hazem Hajj. A novel mobile wireless sensing system for real-time monitoring of posture and spine stress. URL http://www.cs.columbia.edu/~noura/posture.pdf.

Digi-Key. Using an accelerometer for inclination sensing. URL https://www.digikey.com/en/articles/using-an-accelerometer-for-inclination-sensing.

Kwapisz, Jennifer R, Weiss, Gary M, and Moore, Samuel A. Activity recognition using cell phone accelerometers. *ACM SigKDD Explorations Newsletter*, 12(2):74–82, 2011.

STATSports. Apex vest. URL https://uk.shop.statsports.com/products/apex-vest?currency=GBP&variant=12353918500962.

UCI. Human activity recognition using smartphones data set. URL https://archive.ics.uci.edu/ml/datasets/human+activity+recognition+using+smartphones.

A. Estimated technician time usage

Week	Technician Time (Hours)
Week 3 (25 - 31 Jan)	1
Week 4 (1 - 7 Feb)	2
Week 5 (8 - 14 Feb)	3
Reading Week	
Week 6 (22 - 28 Feb)	3
Total	9 / 12 hours

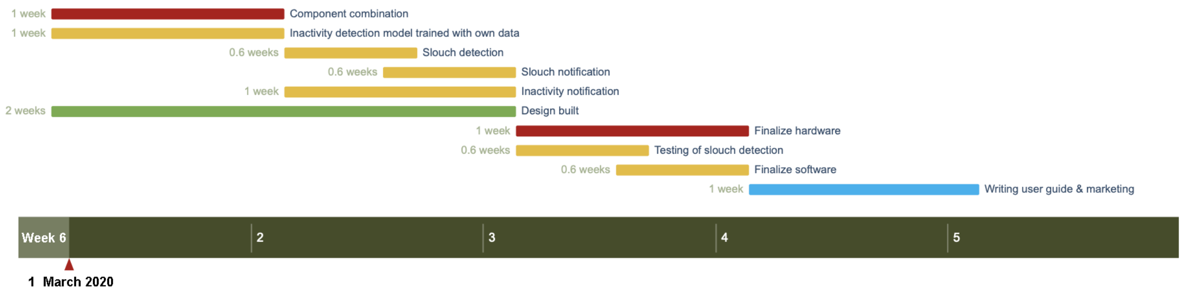
B. Estimated cost for components

Item	Cost
3D printing (TPU)	~ £35
Vest with pouch	~ £25
Raspberry Pi Zero W	£9.30
2x MPU-9250	£11.40
Adafruit TCA9548A I2C multiplexer	£5
Seeed Studio mini vibration motor	£1.09
Adafruit 2305 controller board	£7.28
Battery	£10-30
Total	~ £115

C. Accelerometer samples

(measure degrees to Norm)	Flat	Vertical
	0.07373579037	86.54416318
	0.07651710358	86.77931914
	0.07373579037	86.72014222
	-0.09720396529	86.66024446
	0.005342327662	86.66120284
	0.02242014367	86.54476031
	0.01390673776	86.48564511
	0.08482797792	86.54356482
	-0.03733550879	86.72014222
	0.01105762409	86.66072414
	-0.03444699099	86.60268246
	-0.1541722853	86.48564511
	-0.03444699099	85.901812
	0.02242014367	86.48630194
	0.02524668999	86.48695743
	0.02242014367	86.42712839
	0.02524668999	86.0175592
	0.01105762409	86.25158751
	0.07929289497	86.19212018
	0.04603560032	86.25158751
	0.06815653253	86.36938935
	0.06815653253	86.42712839
	-0.03444699099	86.19403263
	0.01467493978	86.45277568
	0.05957783401	0.2284726638
Mean	0.01467493978	86.45277568
Standard Deviation	0.05957783401	0.2284726638

D. Gantt Chart



E. Confusion matrix for test data from (Kwapisz et al., 2011) on CNN model