

Method, Production & Design

Hardware Overview

The structural design of the robot was created in Autodesk Fusion 360, and the PCBs were designed in Autodesk EAGLE. These programs were used since the different file types were compatible with each other, making it easier to visualise a finished robot and simulate the fit of components. Custom made parts, incorporated in the design, were either Laser Cut or 3D Printed. The PCBs were made at a fabrication service.

Chassis Design

The robot is comprised of three separate layers which form the frame of the robot. This creates a robot which is easily accessible for both production, maintenance and repairs. The chassis is made up of three laser cut poly-carbonate plates.

- The bottom plate is designed to hold four DCX19 Maxon Motors, 32 LEDs and Phototransistors (Light Sensors) and a battery holder.
- The middle plate supports the Main PCB and provides additional support to the motors.
- The top plate is used to stop the robot from entering the goal area and protects the electrical components from accidental impacts.

The three plates are connected by lightweight nylon standoffs.

Electrical Components

The robots' electrical system is a combination of 3 printed circuit boards (PCBs). PCBs were used in this robot to reduce the number of external connections, increasing the reliability of connection to different components. The PCBs are connected to each other by Flat Flex Connectors (FFCs) like ones used in many smartphones and tablets in the Tech-Industry. The PCB on the Bottom Plate (Light Sensor PCB) is used to detect the white line. The Light Sensor PCB holds:

- 5-bit Multiplexer
- 32 LEDs
- 32 Phototransistors

The PCB on the Middle Plate (Main PCB) holds all the main circuitry and the bulk of the components. It holds the two microcontrollers which calculates the robot's movement. The Main PCB consists of:

- 1 Teensy 3.5 (Microcontroller)
- 1 OpenMV H7 (Microcontroller)
- 24 TSSPs (IR Sensors)
- 4 VNH5050A (Motor Controllers)
- 1 HC-05 (Bluetooth Module)
- 3.3V and 5V Power Supply for whole robot

Software Overview

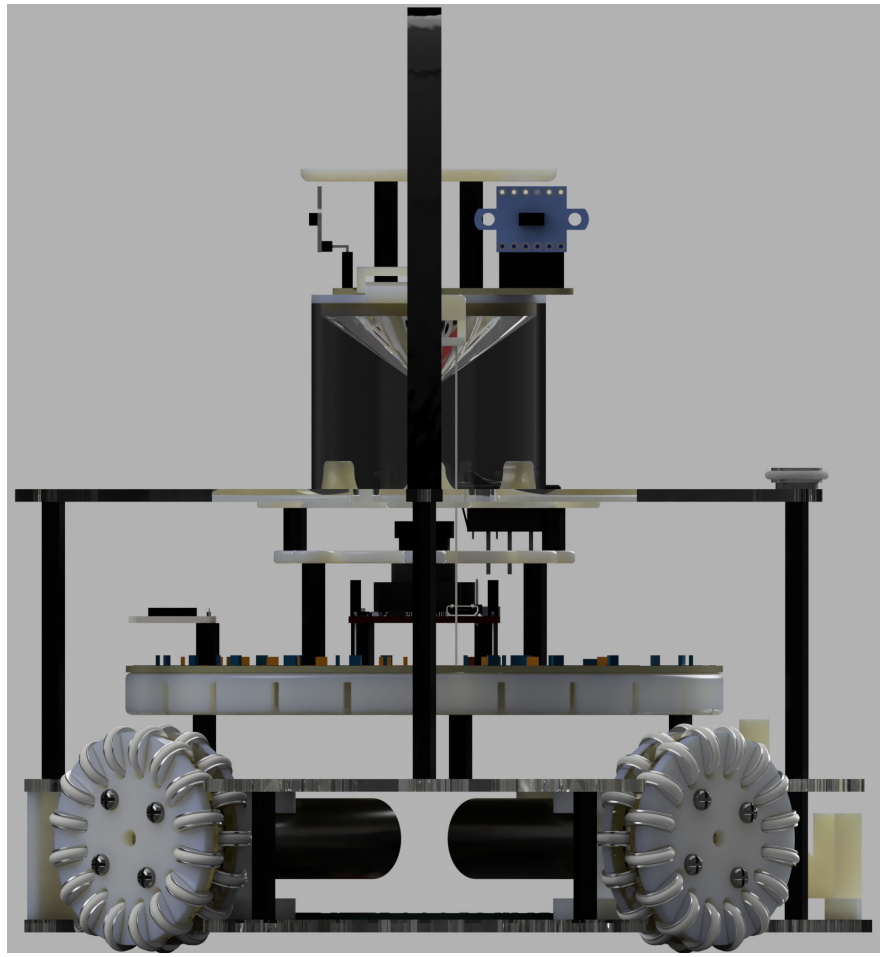
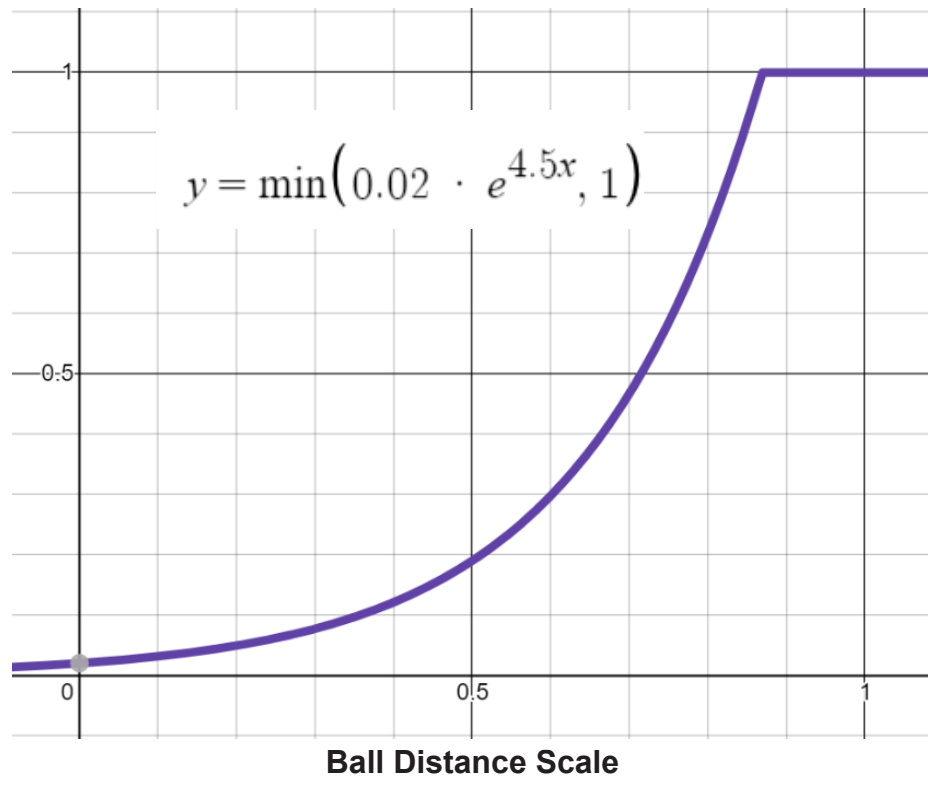
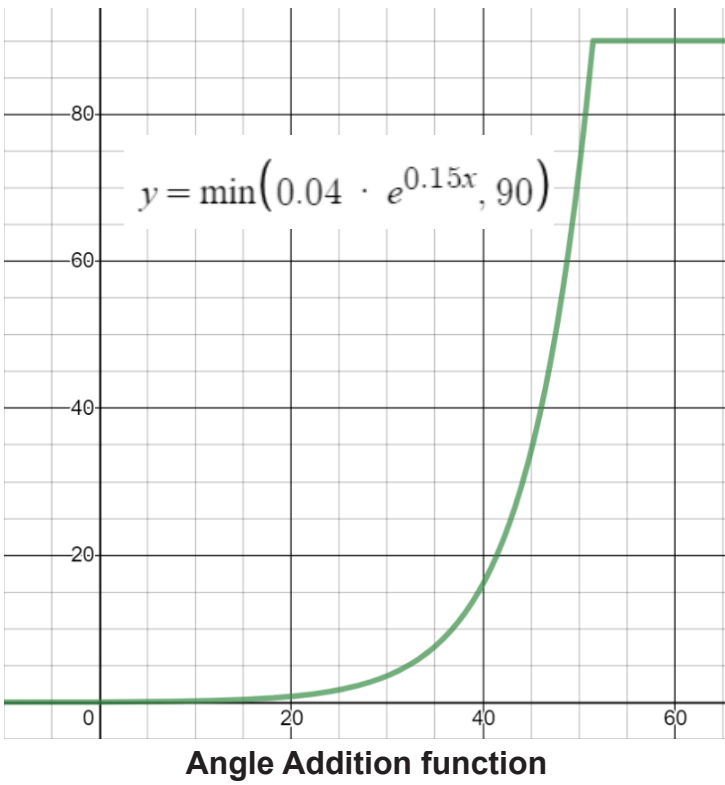
The robots have been programmed using C++ and Python with two microcontrollers on each robot. The Teensy 3.5 does the bulk of the calculations and receives data from all the sensors. The OpenMV H7 is an optical sensor that calculates the goal direction and distance and then sends those values to the Teensy. The Teensy then sends motor values to the motor controllers.

Communication & Game Strategies

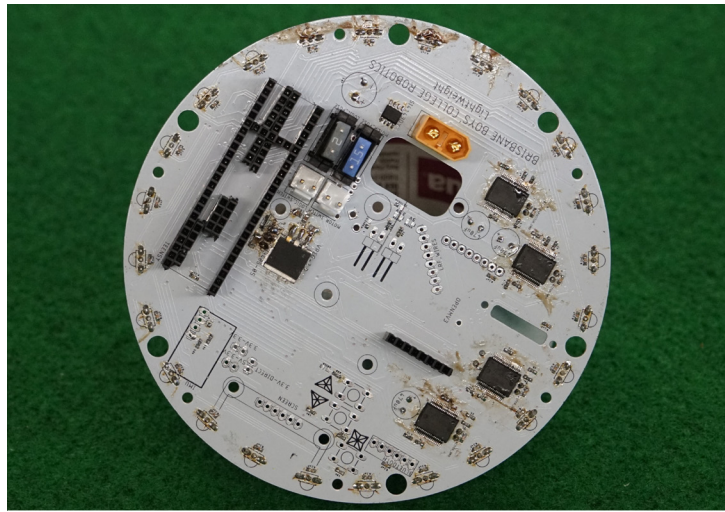
Each Robot has an HC-05 Bluetooth Module which pair with each other. The robots can send their role (Attacker or Defender), the ball direction and the ball strength (distance) using a Serial protocol. The robots can decide whether it should be an attacker or a defender and switch positions accordingly, just like real soccer players. This strategy is implemented in Gameplay and helps reduce the number of Double Defence fouls.

Ball Following

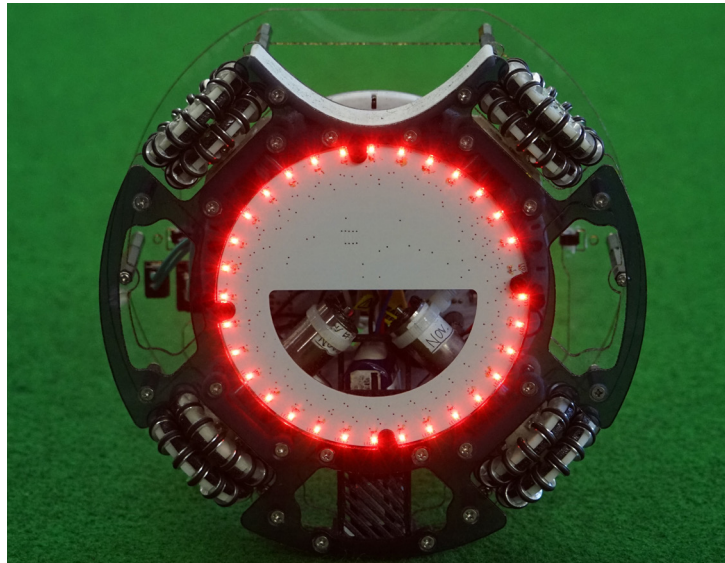
Each robot has 24 digital IR sensors which output a value of 0, when it detects IR light, or 1, when it can't detect IR light. The ball direction is calculated by using Vector Addition. After the ball direction is calculated, two mathematical models are used to calculate the move direction. The first one determines the angle added to the ball direction, so the robot positions itself behind the ball; the second one scales down the angle added based on how far away the ball is.



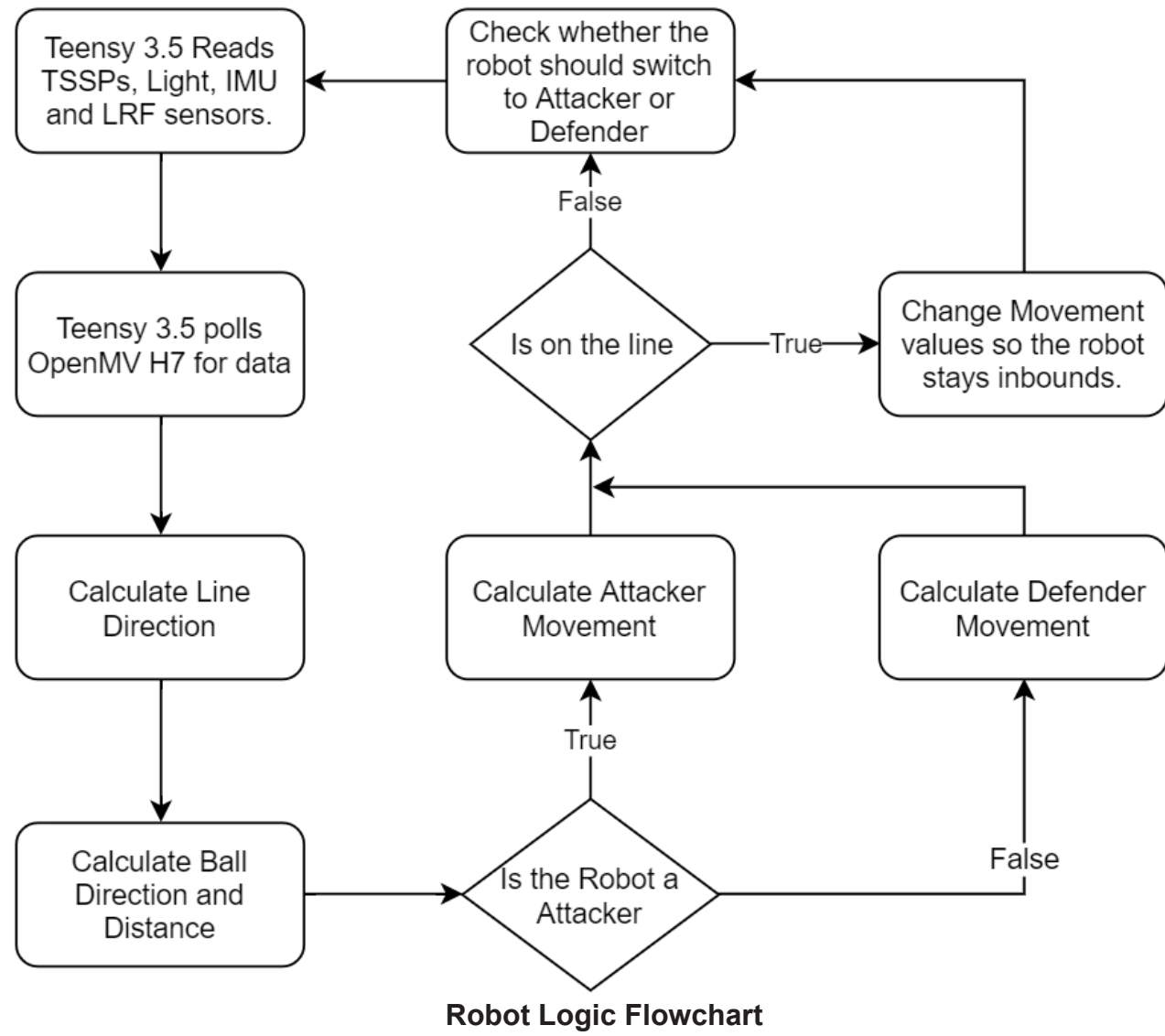
Side on View of the robot



Main PCB



Light Sensor PCB



M&A is an Australian team from Brisbane Boys' College that have made two identical robots to compete in the RoboCup Junior Soccer Lightweight Competition. We consist of 4 team members. Last year we placed 1st at the Queensland State Competition and 2nd at the Australian National Competition. The robots have been developed since September of 2017 with four different design iterations for different RoboCup Competitions. We have worked for over 500 hours and spent approximately AUD\$2500 on parts for the robot.

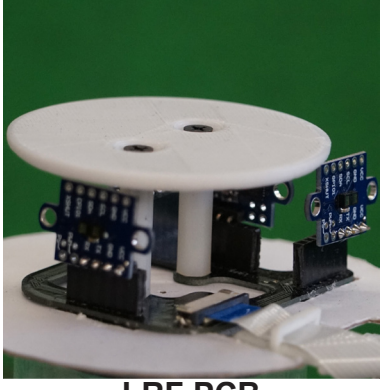
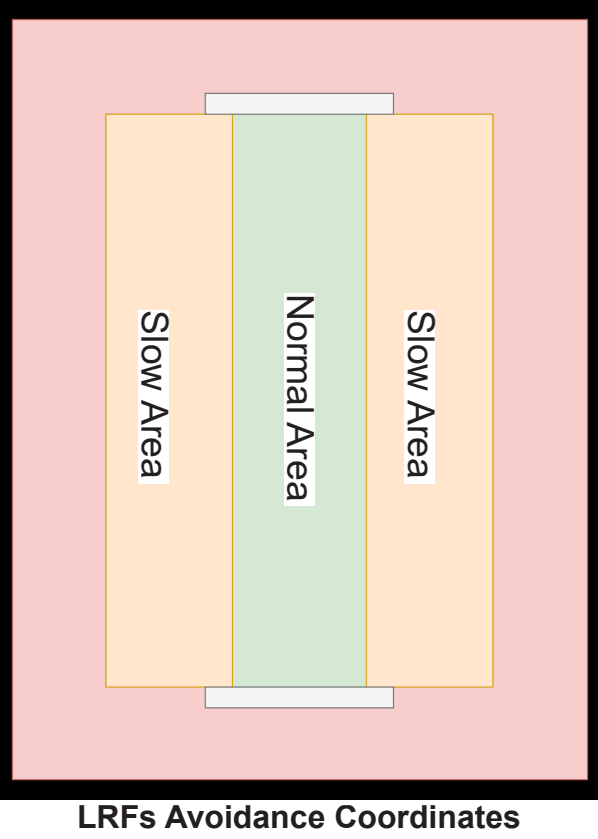
The robots were built using industry technologies and incorporate populated Printed Circuit Boards (PCBs) to increase efficiency and reliability. The robots were programmed using the Arduino Framework. Each robot consists of two microcontrollers, a Teensy 3.5 and an OpenMV H7. The robots communicate via Bluetooth Modules to optimise gameplay performance.

The purpose of this poster is to share our findings on how to best build a robot for this competition so that other teams can improve their robots and to help RoboCup achieve their goal in making robots that can beat the best soccer players in the world by 2050.

M&A

LRFs

The Software implemented on the robot uses a fusion of LRF and Light Sensor values to stay inbounds during gameplay. The LRFs can detect when the robot enters the edge of the playing field and slows down in this area. This is so the robot has a lower chance of going over the line.



LRF PCB

The LRFs send data to the Teensy 3.5 via a Serial protocol. It sends 16-bit values and has millimetre accuracy.

Out Avoidance

The robot incorporates 32 Light Sensors to detect the white line and stay in the field. The light sensors are arranged in a ring, makes it easier to perform vector calculations. The angle of the line is calculated using Vector Addition. This data is sent to the Teensy 3.5 and the robot moves in the opposite direction of the line, so it stays inbounds.

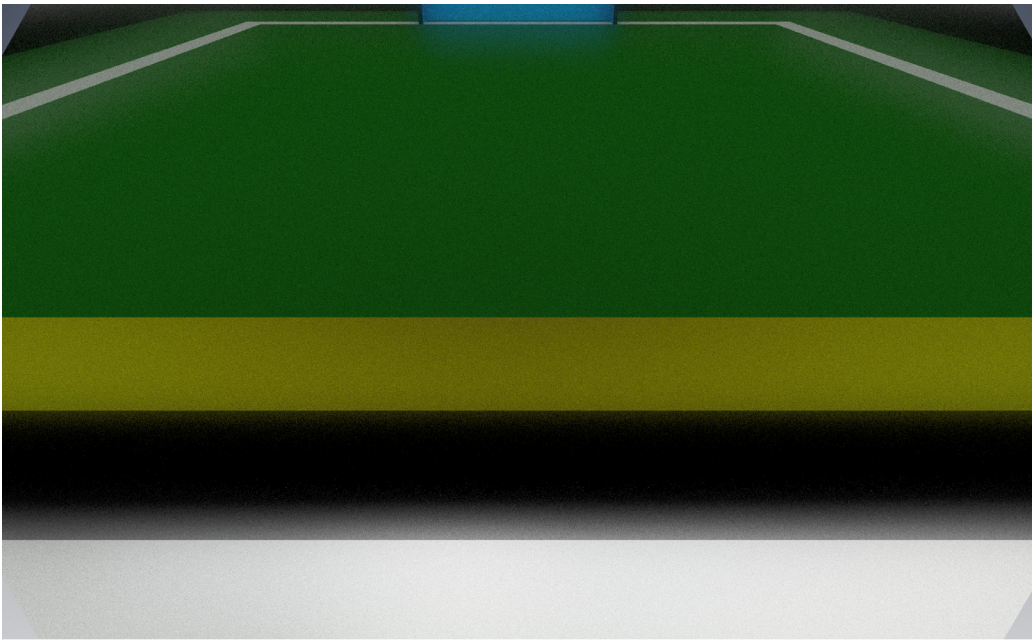
QR CODE

TEAM PICTURE

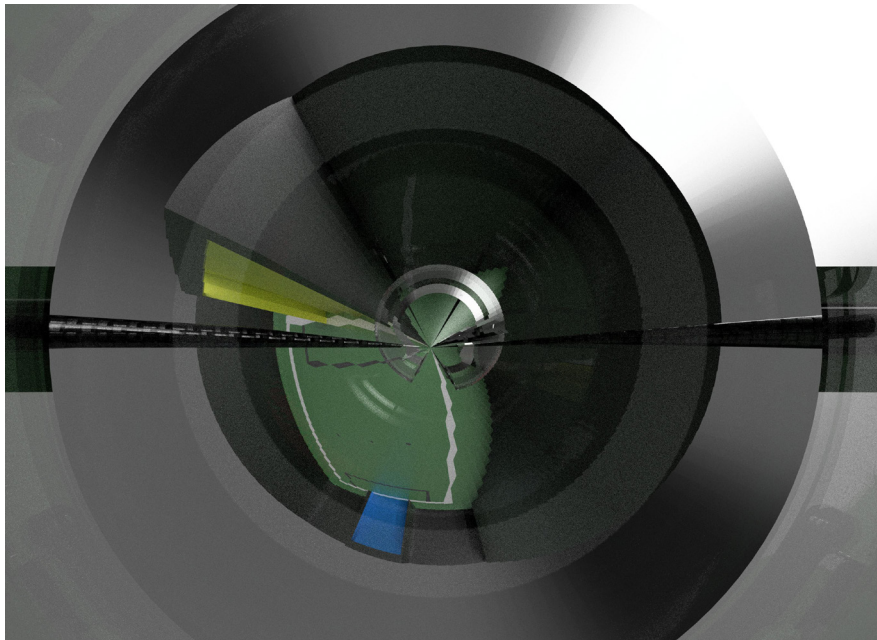
Data, Results & Discussion

Camera

The 360-degree camera vision system is comprised of the Open MV H7, a custom moulded mirror and acrylic tube. The mirror is shaped as a cone and is vacuum formed over an impression to create a detailed mould. The camera is positioned so that it looks vertically at the cone, then through the mirror on the cone can see the entire field. A series of brackets allow the cone to be positioned to the centre of the camera's field of view. A circular piece of card is placed above the cone to protect the camera from glare which would wash out the image.



V Mirror Render



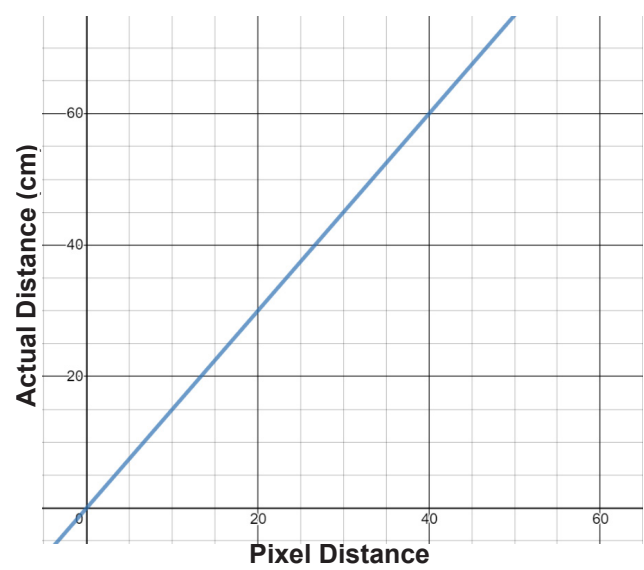
Cone Mirror Render

Pros:

- The mirror could see the goal at great ranges
- Easy to manufacture

Cons:

- Cannot see sideways
- Image Distortion
- **Worse for Gameplay**

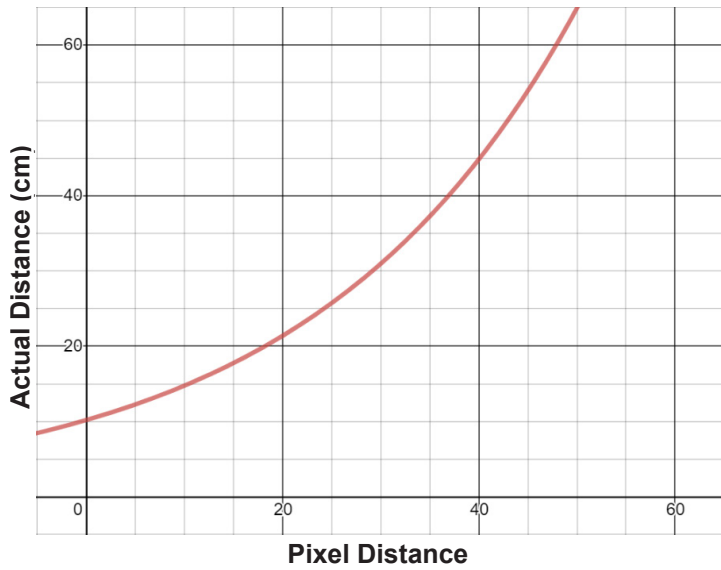


Pros:

- Can see in all directions
- Reduced Image distortion
- **Better for Gameplay**

Cons:

- View distance is reduced
- Harder to manufacture



Graphs of the relationship between the Pixel Distance and the Actual Distance for both mirrors

Vacuum Formed Mirrors

The mirrors on the robot were made by using a mould, vacuum and an oven to heat and mould the mirror material. The mirror material was first heated, then placed on a mould, where the material was vacuumed into the shape of the mould. This provides a smoother mirror surface which made it easier to track the goal.

VACUUM FORMING SETUP

Wheels

An early iteration of this robot included the premade omni-wheels from GTF robotics. These were an easy to use omni-wheel, however they lacked in grip which caused the robot to slide as it attempted to avoid the line. The rubber O-rings included with the wheels also wore down very quickly and we couldn't find any suitable replacement.

A new wheel was designed to fix this issue. It included larger rollers that were produced by a CNC lathe to have more grip on the floor, and the current iteration of this includes two layers to maximise contact with the floor.



Wheel Progression - GTF Wheel, Single Layered Custom Wheel, Double Layered Custom Wheel (Left to Right)