MLiS

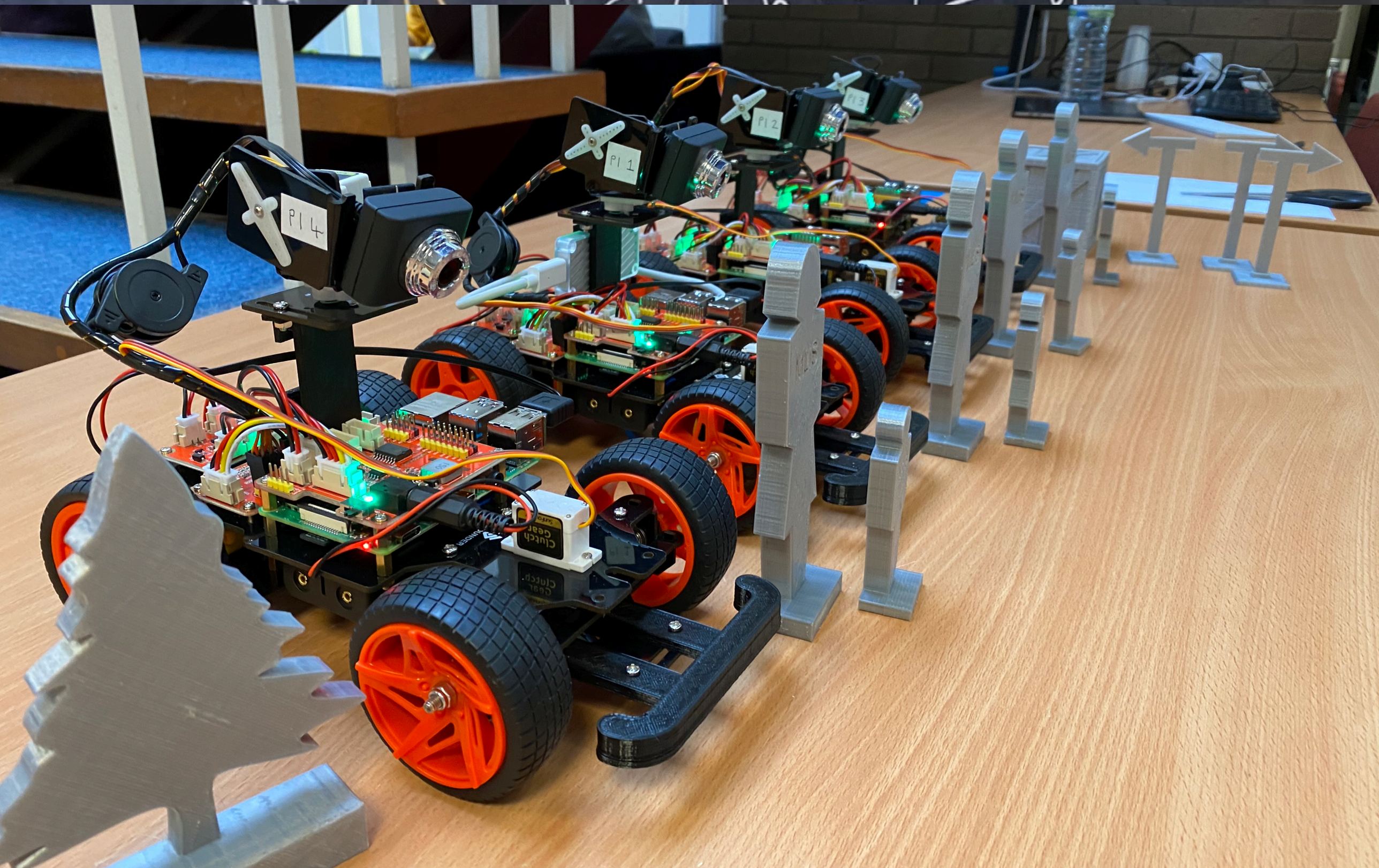
Pi Car Project

School of Physics & Astronomy
and
Faculty of Science

MLiS

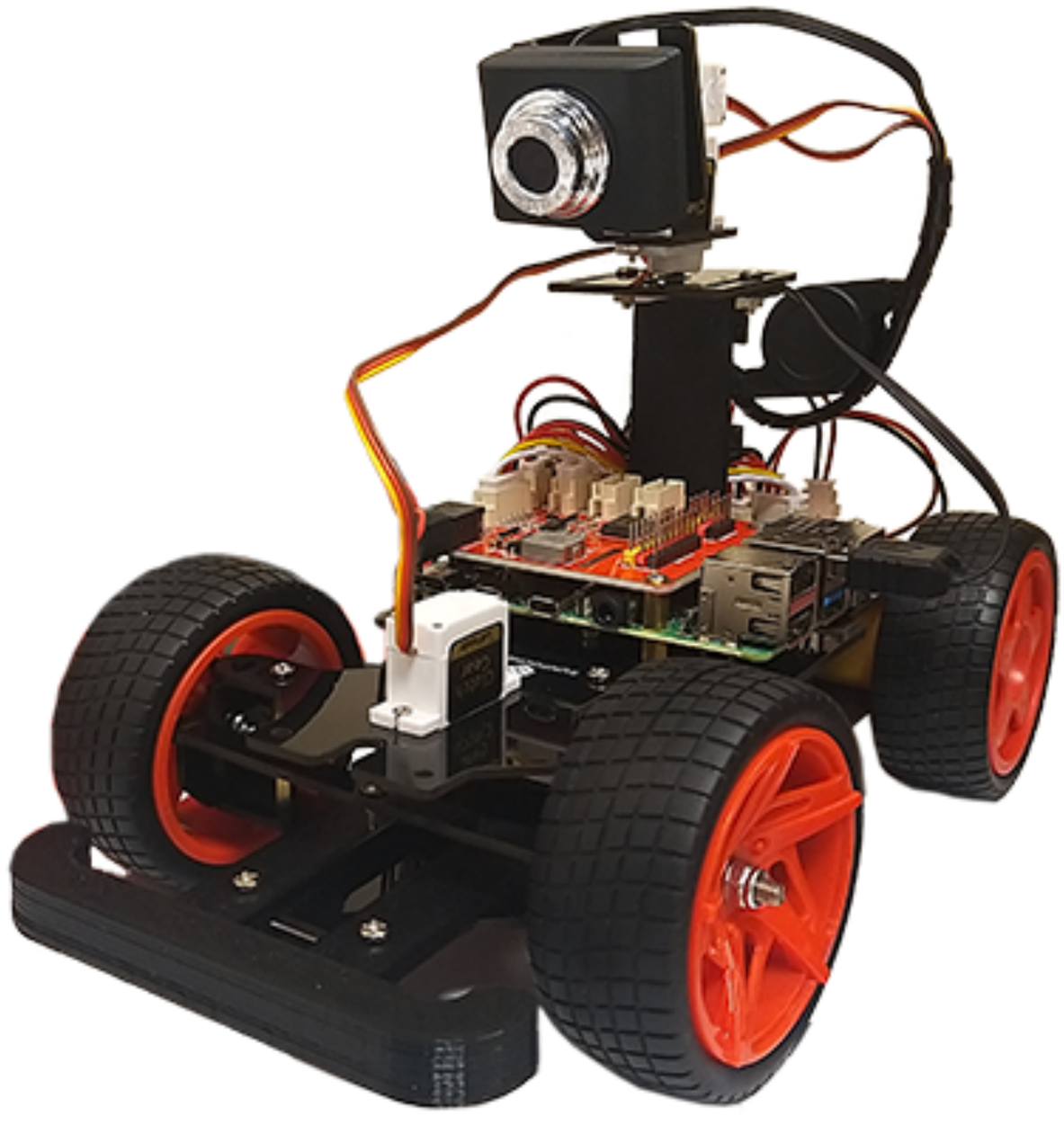
Machine Learning in Science

School of Physics & Astronomy and Faculty of Science



Pi Car

- Raspberry Pi 4
- Onboard Camera
- Coral Edge TPU
- Access to 2080 Ti GPU



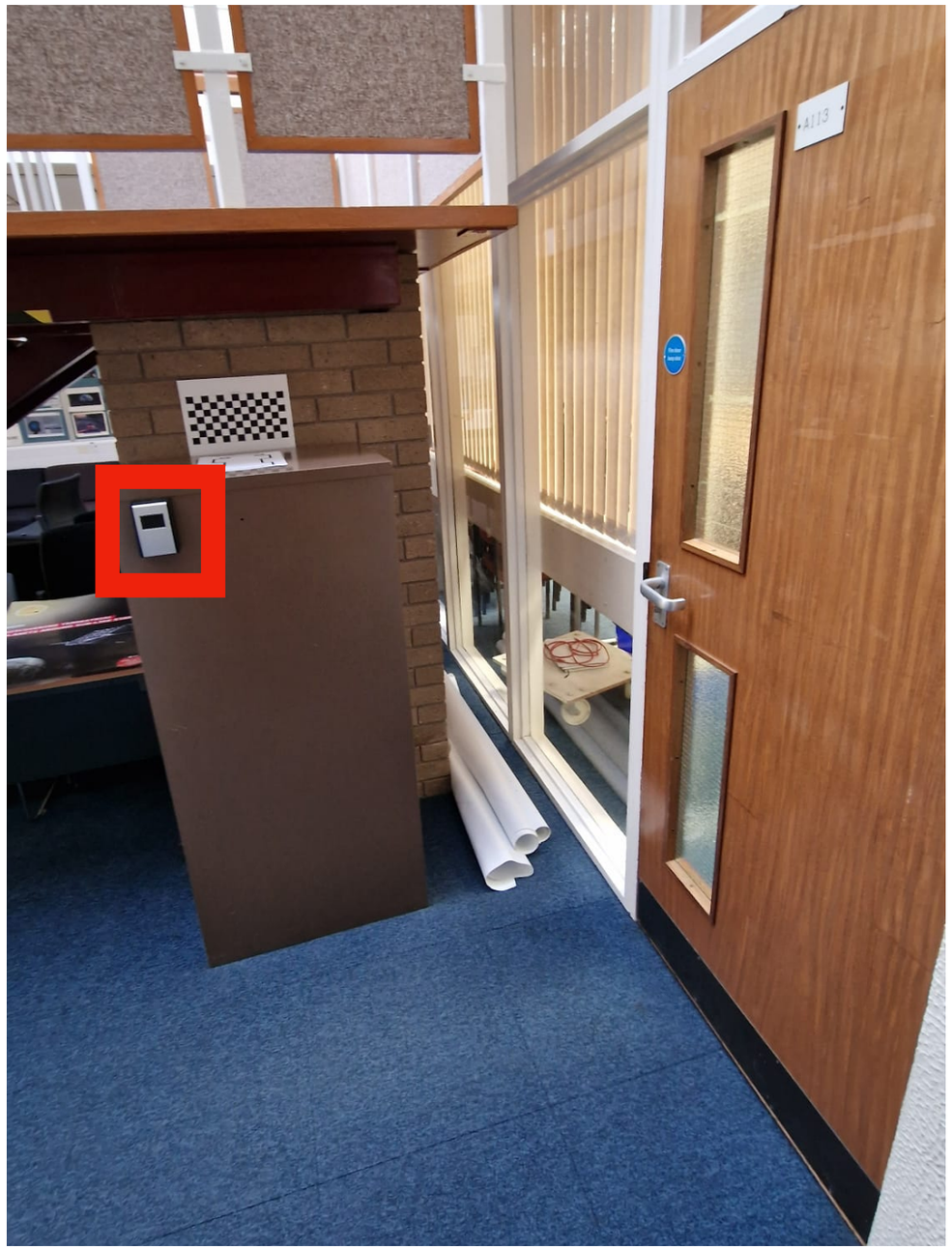
Tasks

- Collect data
- Train Machine Learning algorithm
- Navigate test circuits
- Make manoeuvres

MLiS Machine Learning in Science

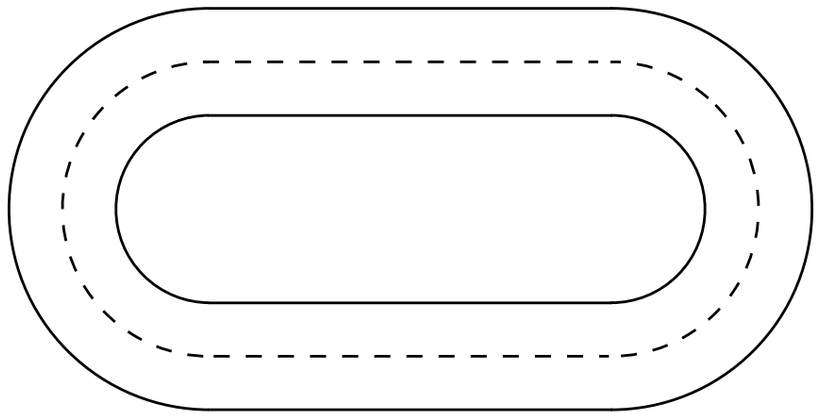
School of Physics & Astronomy and Faculty of Science

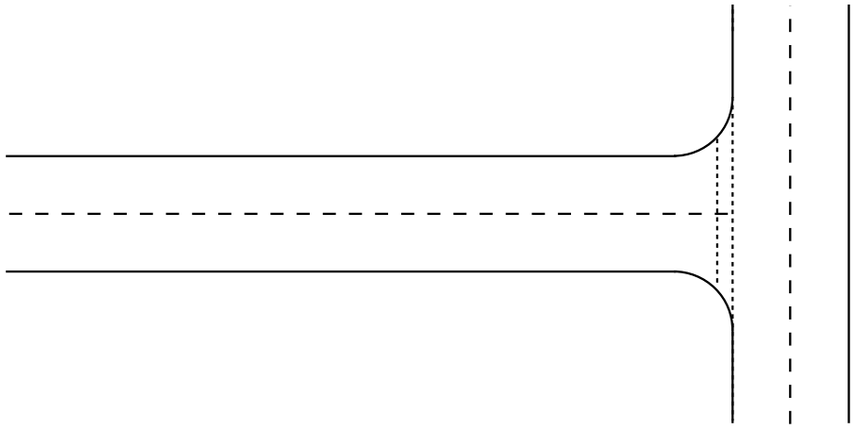
Location ▸ Room A113 CAPT

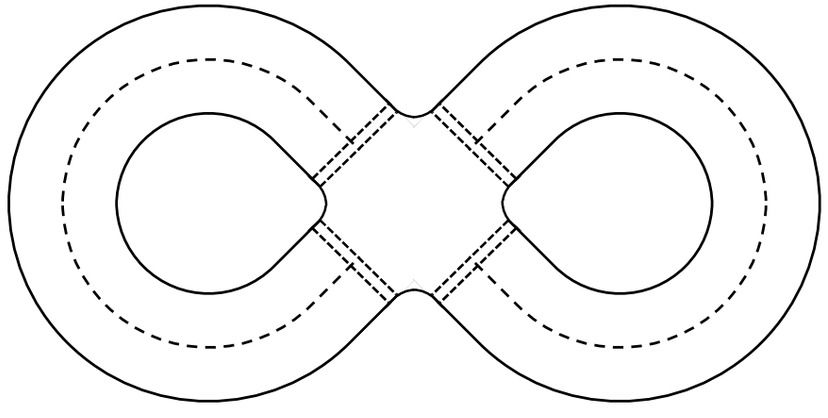


- Passcode: year neural network invented

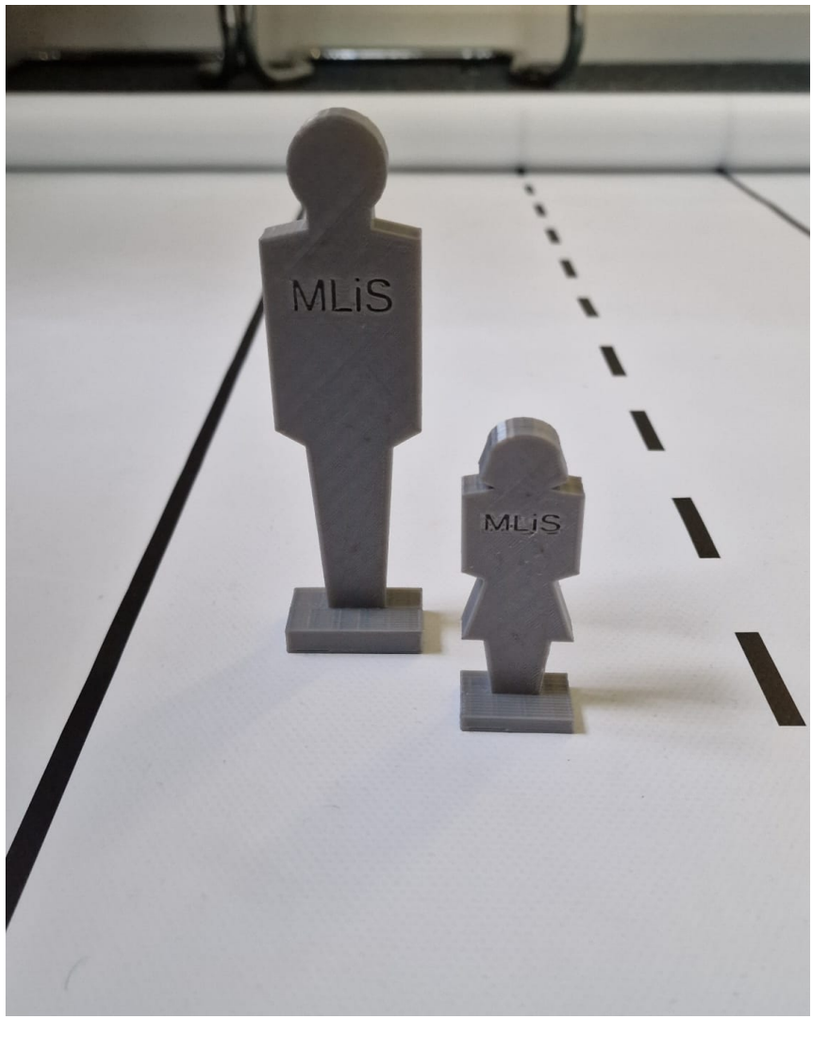
Tracks

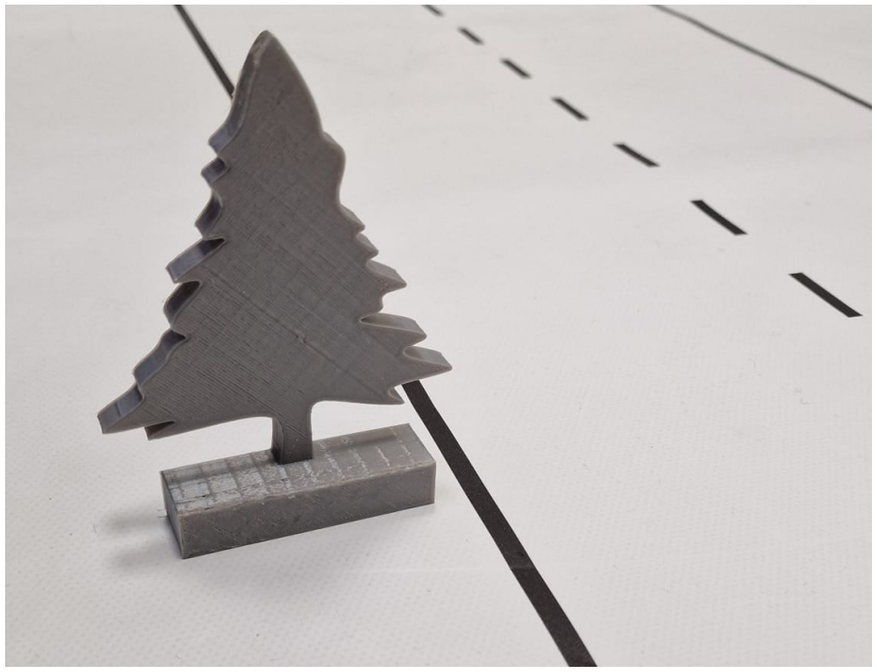


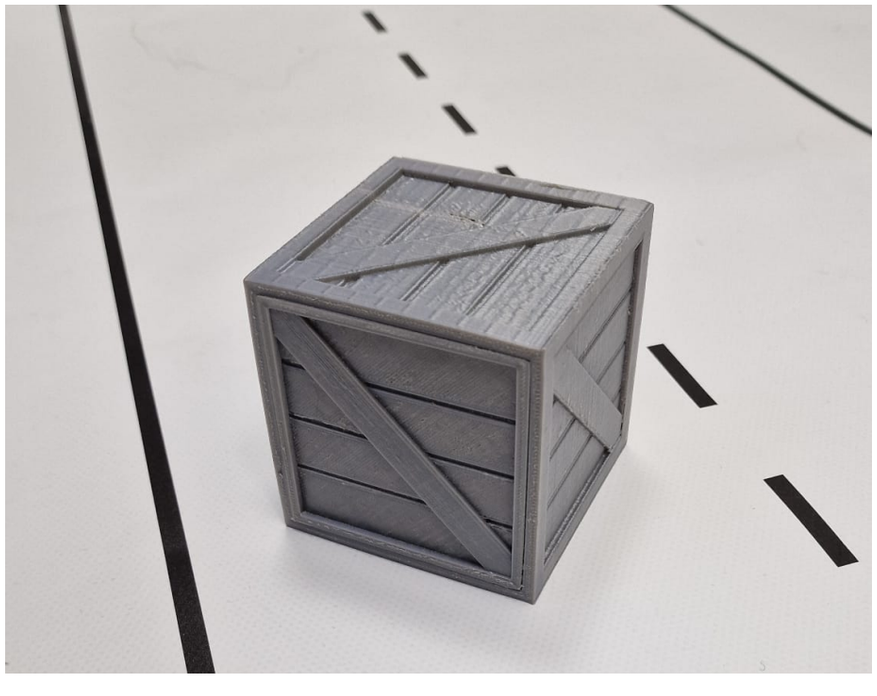




Obstacles

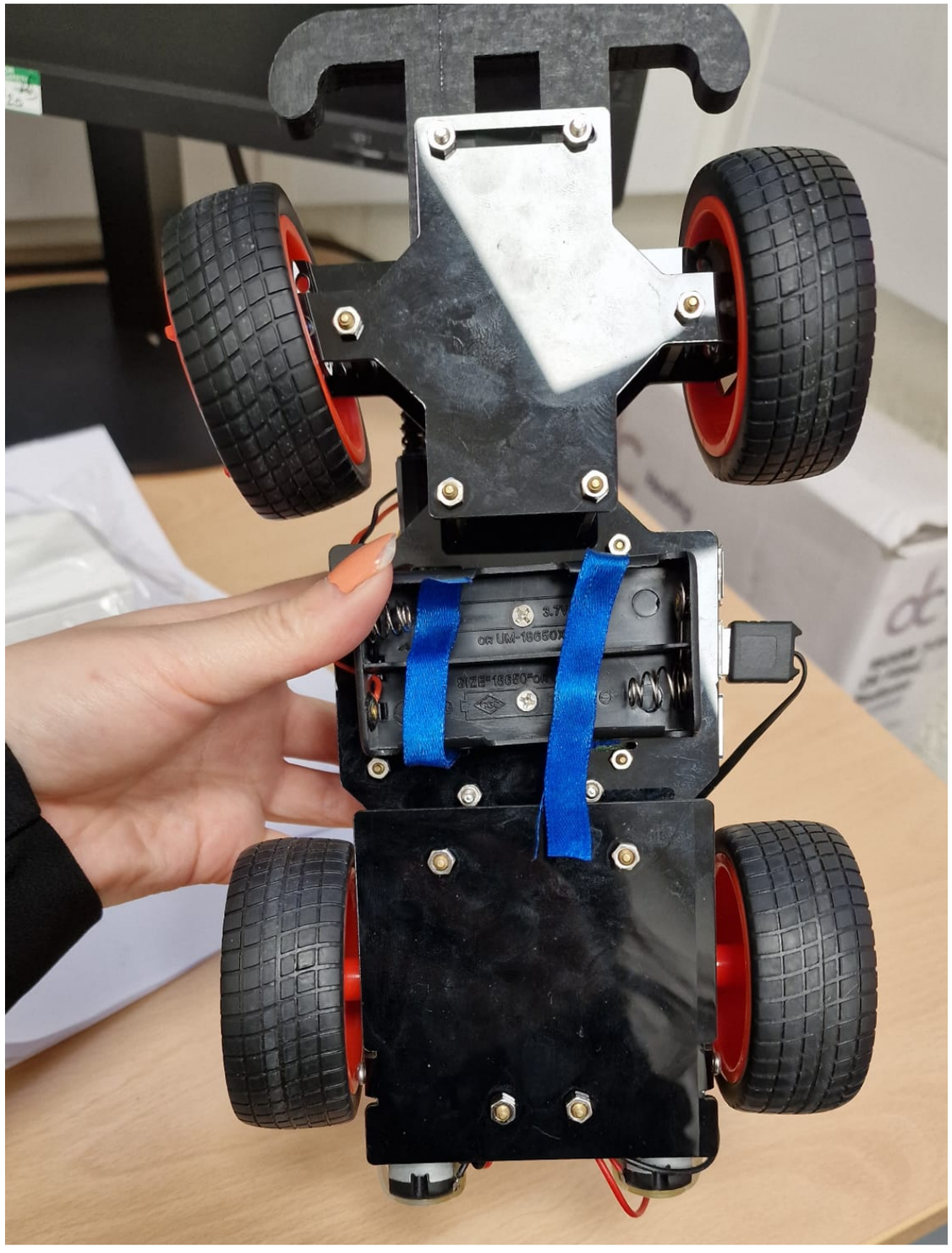






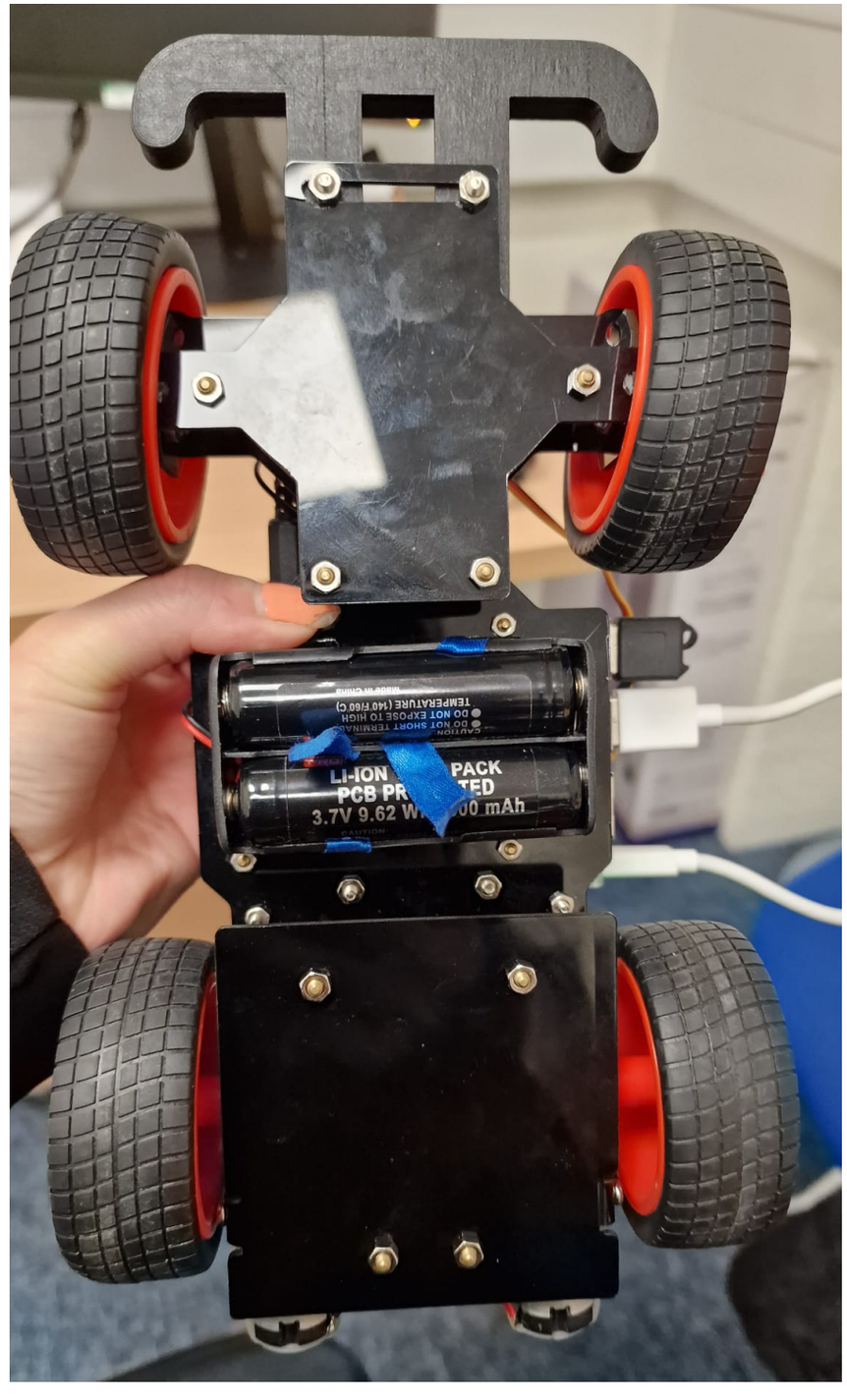
Set up:

- Use ribbons beneath battery



Set up:

- Use ribbons beneath battery
- Helps to remove them easier
- Make sure to charge batteries for next person!



Batteries

- Make sure positive end of battery goes into positive end of battery case and charger

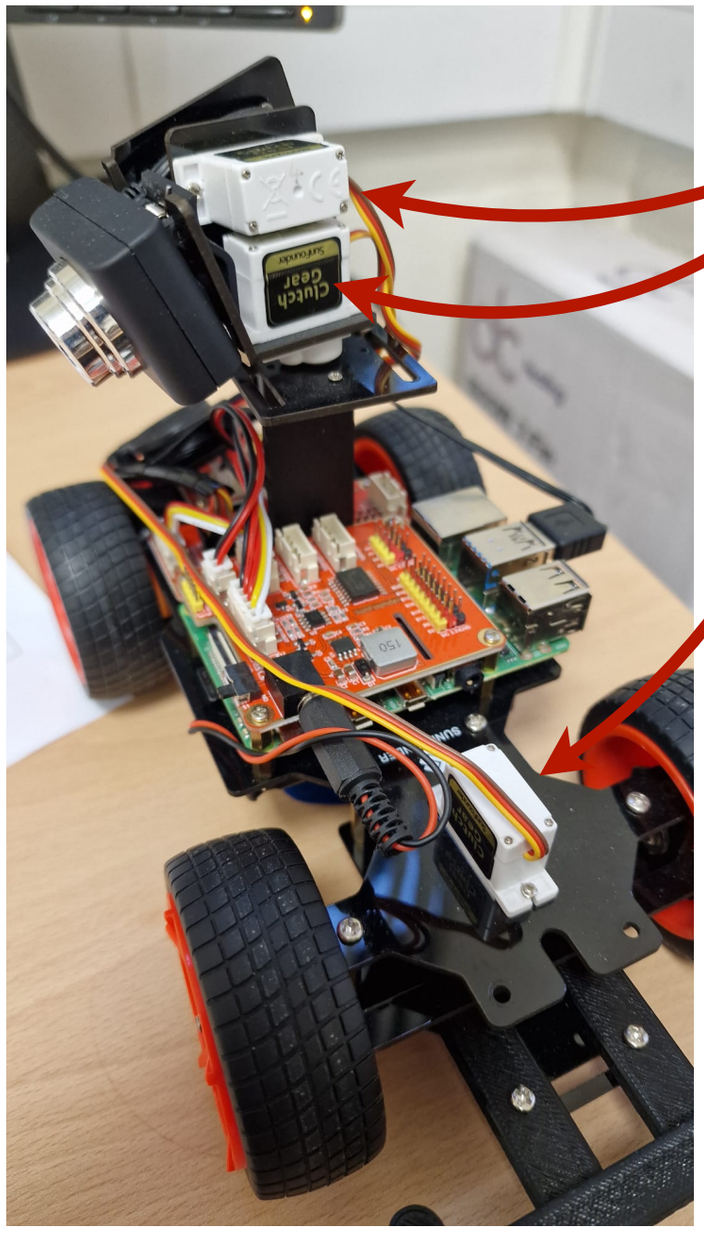
Positive **Negative**

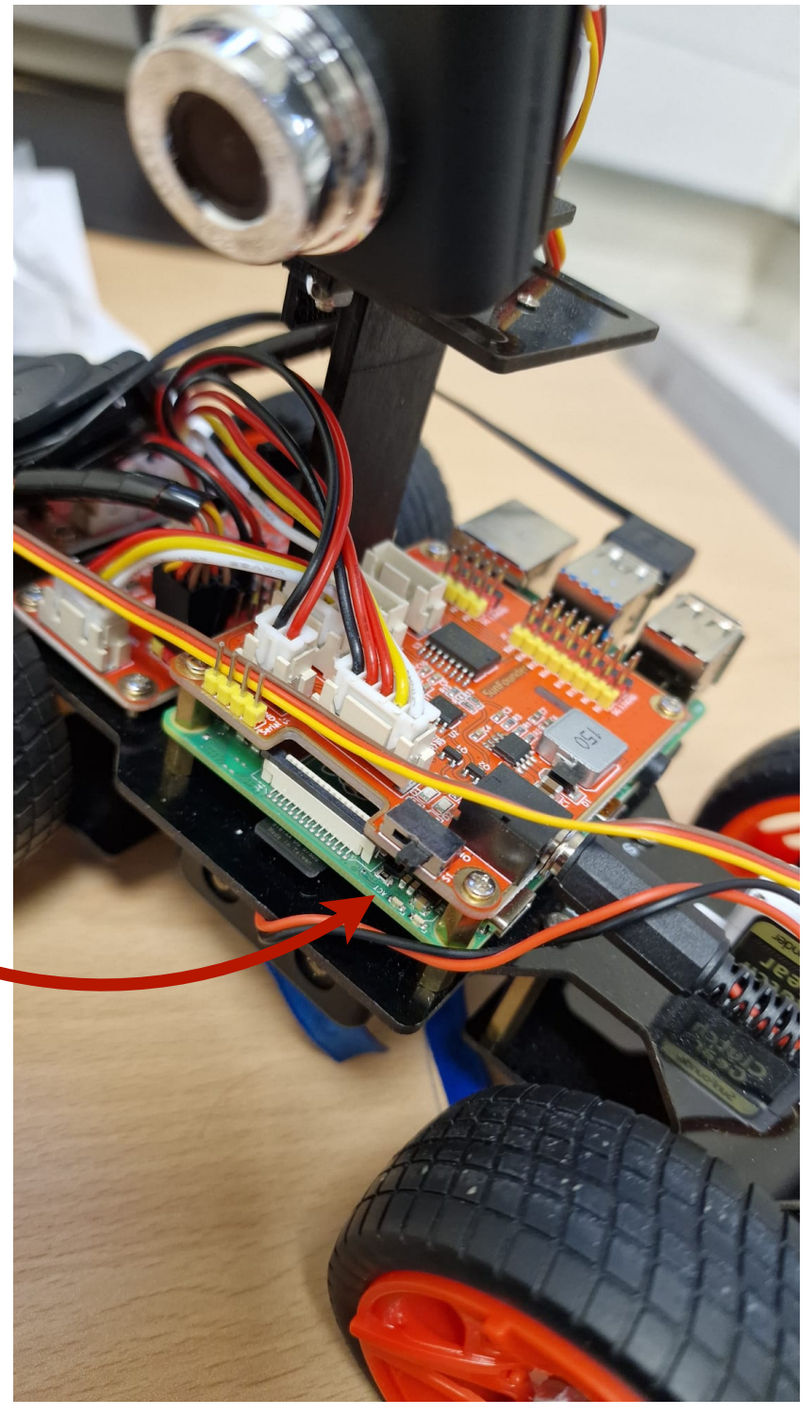
Positive

MLiS Machine Learning in Science

School of Physics & Astronomy and Faculty of Science

Set up:



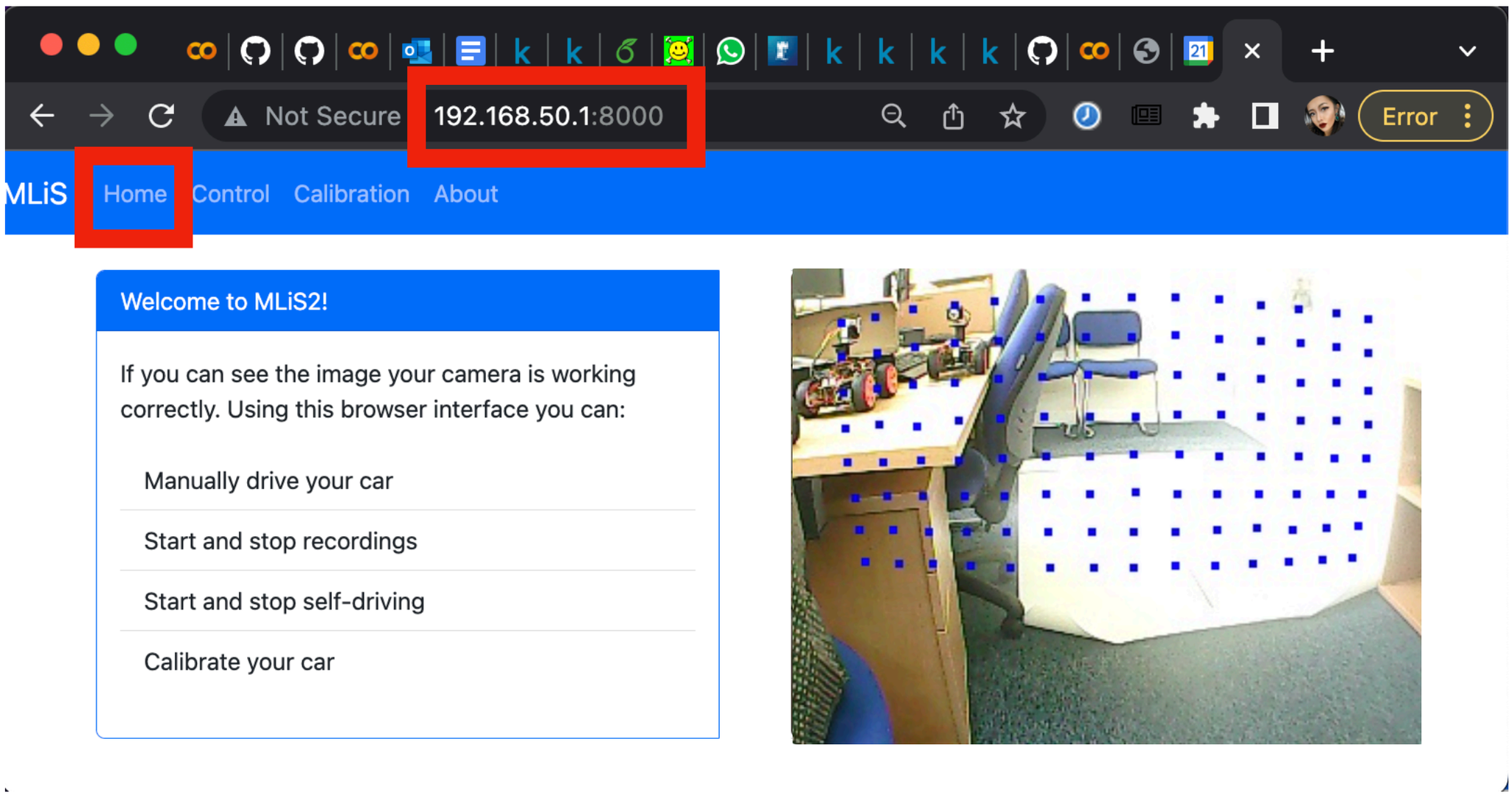


MLiS Machine Learning in Science

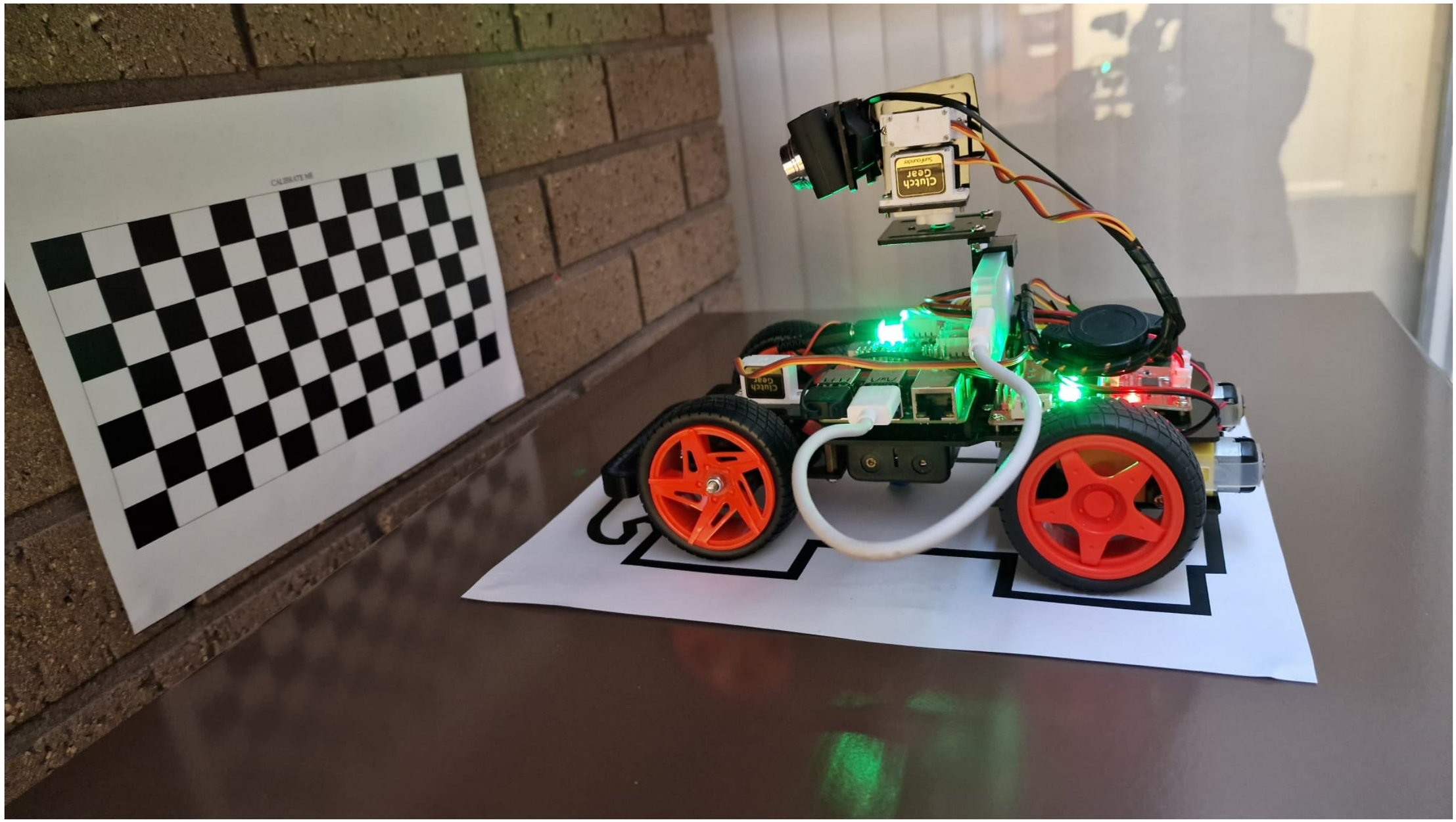
School of Physics & Astronomy and Faculty of Science

Wifi: backprop

Calibrate:


Not Secure
192.168.50.1:8000
Error
MLiS
Home
Control
Calibration
About
Welcome to MLiS2!
If you can see the image your camera is working correctly. Using this browser interface you can:
Manually drive your car
Start and stop recordings
Start and stop self-driving
Calibrate your car

Calibrate:

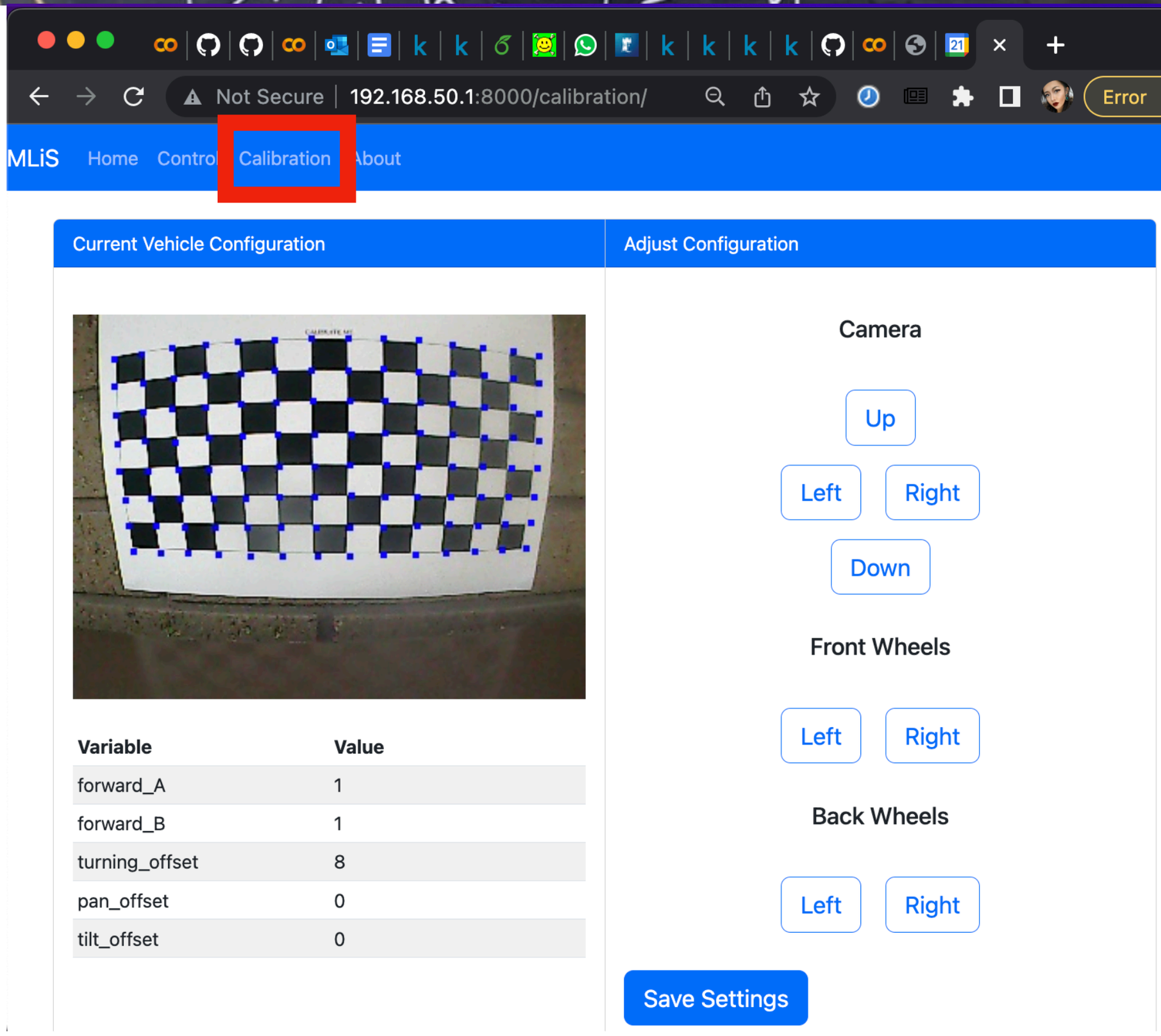


MLiS Machine Learning in Science

School of Physics & Astronomy and Faculty of Science

Calibrate:

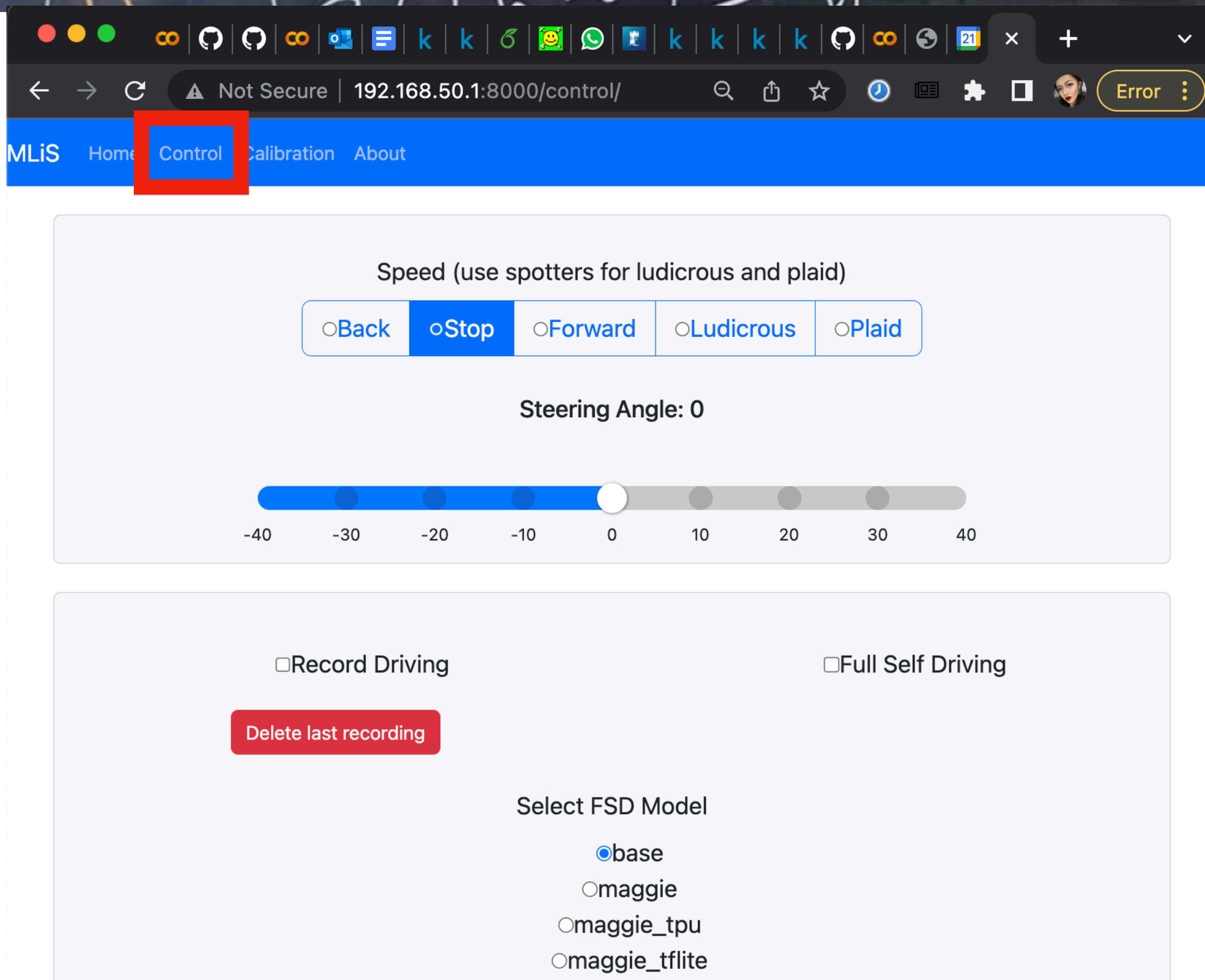
value limits:
-30 - 30



MLiS Machine Learning in Science

School of Physics & Astronomy and Faculty of Science

Drive:



MLiS Machine Learning in Science

School of Physics & Astronomy and Faculty of Science

Terminal Shell Edit View Window Help

ppzml1 — -bash — 74×20

```
> ssh pi@192.168.50.1    #remote connect
```

MLiS Machine Learning in Science

School of Physics & Astronomy and Faculty of Science

Terminal Shell Edit View Window Help

ppzml1 — -bash — 74×20

```
> ssh pi@192.168.50.1    #remote connect
pi@192.168.50.1 password:
> rasp
```

Terminal Shell Edit View Window Help

ppzml1 — -bash — 74×20

```
> ssh pi@192.168.50.1    #remote connect
pi@192.168.50.1 password:
> rasp
> cd SunFounder_PiCar-V/remote_control/capture/    #move to folder
```

Terminal Shell Edit View Window Help

ppzml1 — -bash — 74×20

```
> ssh pi@192.168.50.1    #remote connect
pi@192.168.50.1 password:
> rasp
> cd SunFounder_PiCar-V/remote_control/capture/    #move to folder
> ls    #list files
```

MLiS Machine Learning in Science

School of Physics & Astronomy and Faculty of Science

Terminal Shell Edit View Window Help

ppzml1 — -bash — 74×20

```
> scp -r pi@192.168.50.1:SunFounder_PiCar-V/remote_control/capture/* .
```

MLiS Machine Learning in Science

School of Physics & Astronomy and Faculty of Science

Terminal Shell Edit View Window Help

ppzml1 — -bash — 74×20

```
> scp -r pi@192.168.50.1:SunFounder_PiCar-V/remote_control/capture/* .
> ls
```

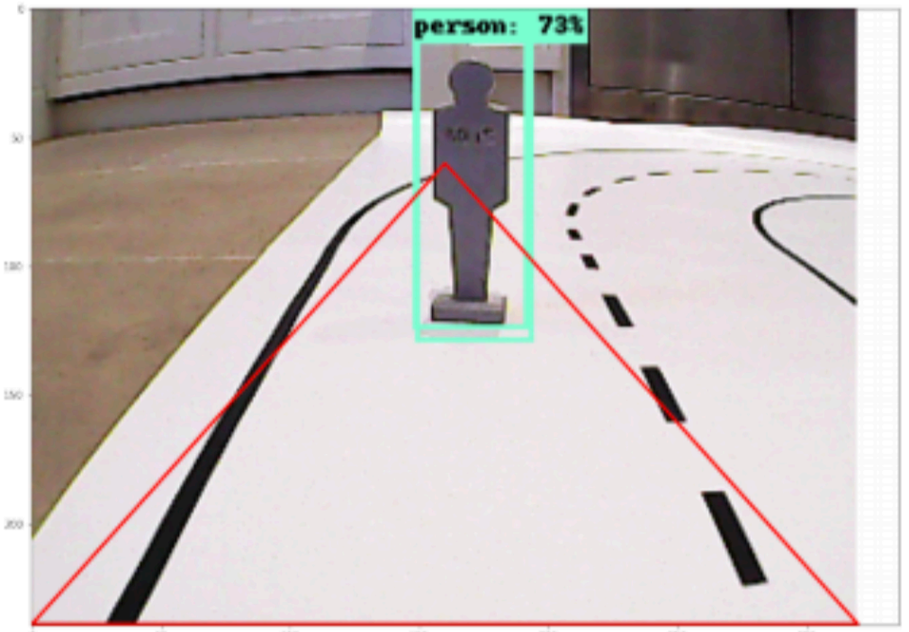
Kaggle

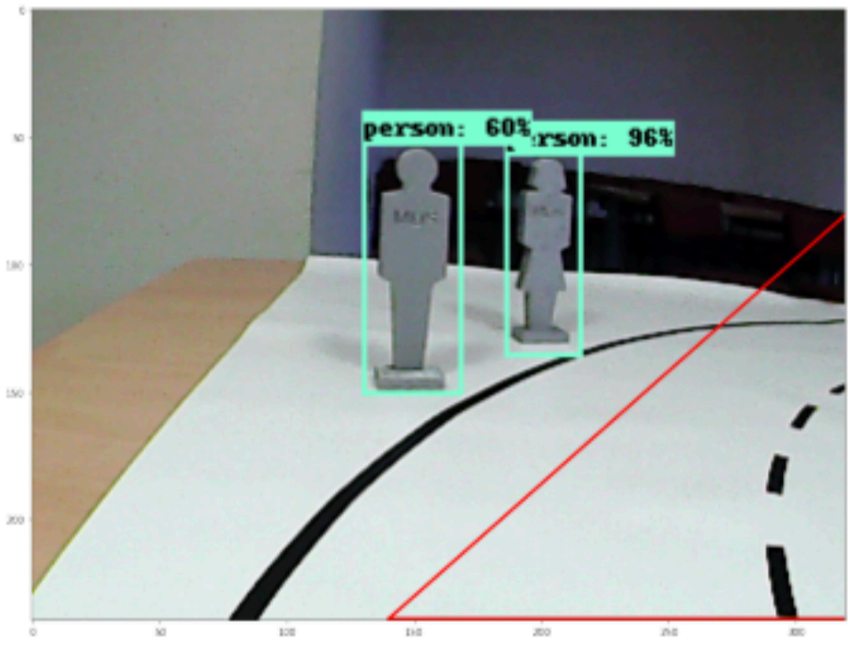
https://www.kaggle.com/competitions/machine-learning-in-science-ii-2025

Deadline: 28/03/2024

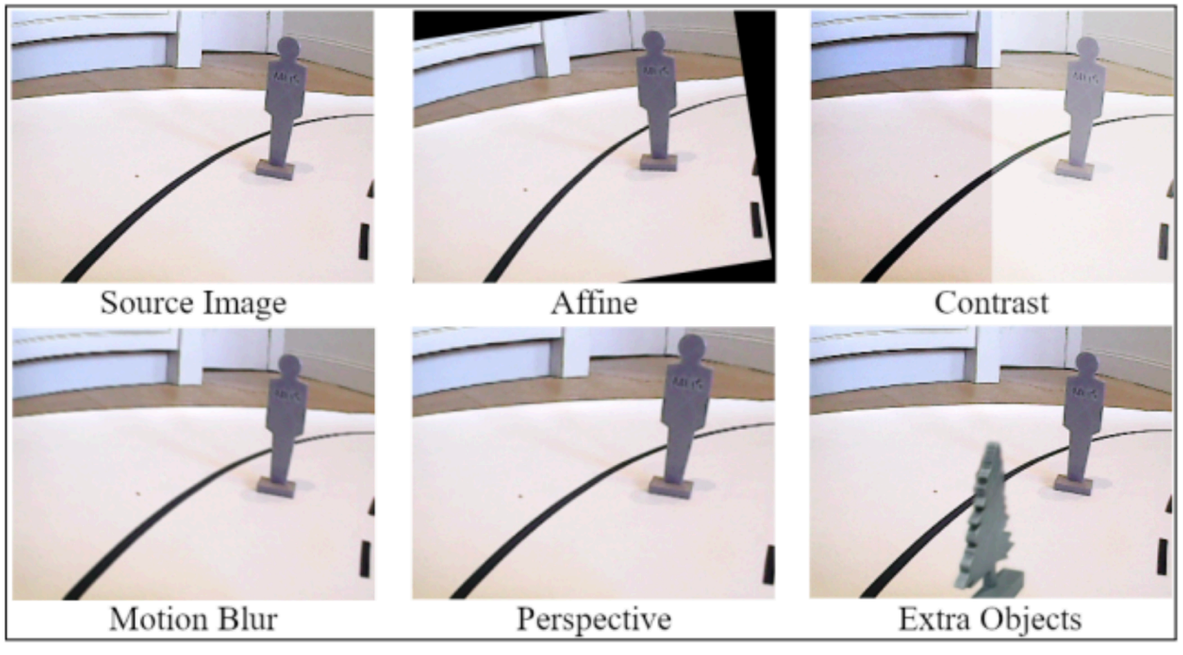
Results

- Apply knowledge
- Live test track
- Presentation & Report





Source: A.Mahmood



Source: R. Marshall-Hawkes, O. Wiech

Getting ready for the live testing

Car models stored in:
> /autopilot/autopilot/models/

Make model directory
> mkdir MyPiCarModel

Example: model.py

Terminal Shell Edit View Window Help

ppzml1 — pi@raspberrypi: ~/autopilot/autopilot/models — ssh pi@192.168.50.1 — 101×30

```
from tensorflow import keras
import numpy as np
import imutils
import cv2
import os


class Model:

    saved_model = 'autopilot.h5'

    def __init__(self):
        self.model = keras.models.load_model(os.path.join(os.path.dirname(os.path.abspath(__file__)),
 self.saved_model))
        self.model.summary()

    def preprocess(self, image):
        image = cv2.cvtColor(image, cv2.COLOR_BGR2YUV) / 255.0
        image = imutils.resize(image, width=80)
        image = image[int(image.shape[0] / 4):, :, :]
        return image

    def predict(self, image):
        image = self.preprocess(image)
        angle, speed = self.model.predict(np.array([image]))[0]
        # Training data was normalised so convert back to car units
        angle = 80 * np.clip(angle, 0, 1) + 50
        speed = 35 * np.clip(speed, 0, 1)
        return angle, speed
(END)
```

Getting ready for the live testing

Run model

> python3 run.py --model base

> python3 run.py --model base --mode drive --duration 60

Teams

> Teams of 2-3 emailed to us by end of Friday