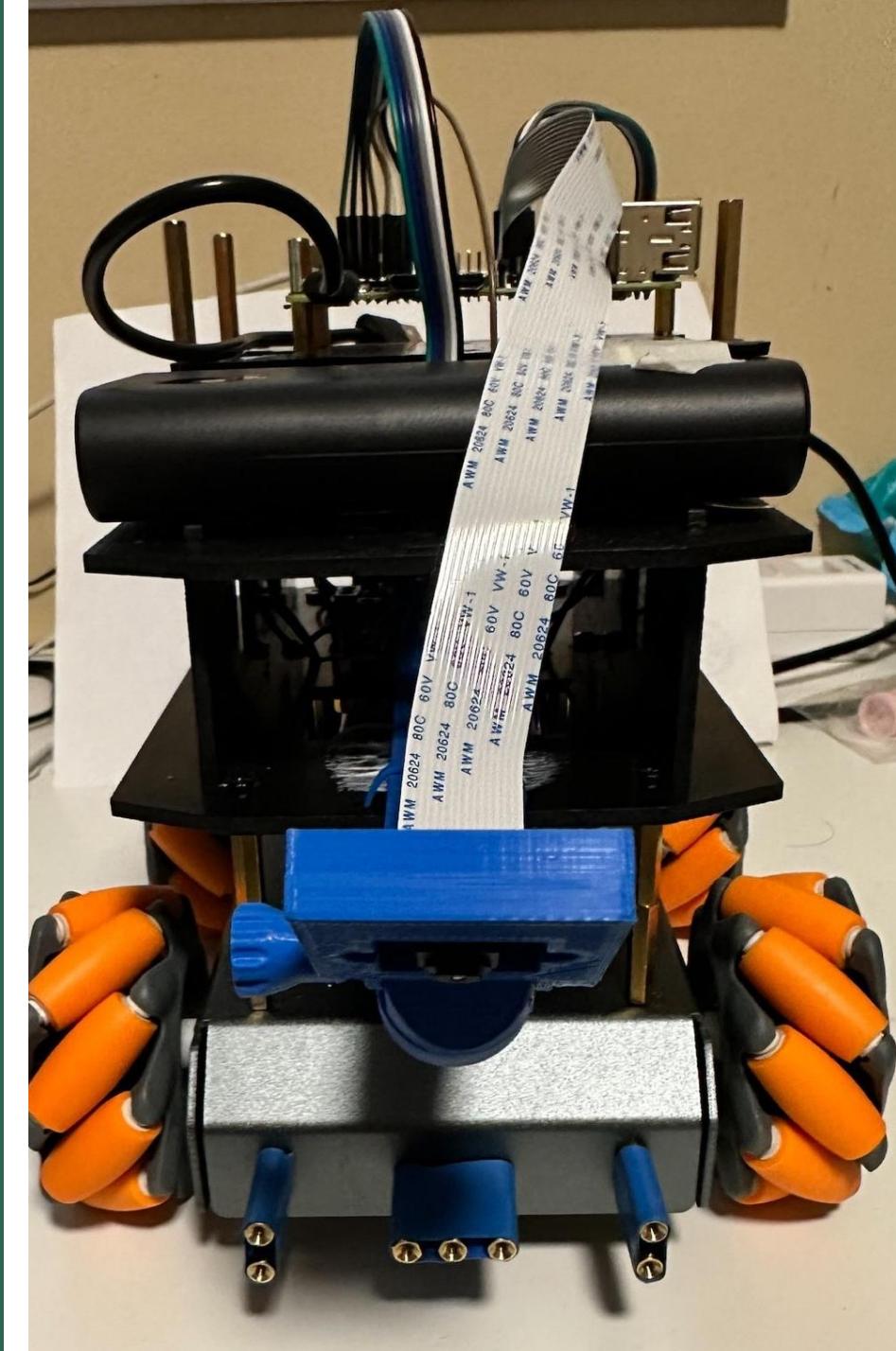


SELF-DRIVING MODEL CAR WITH RASPBERRY PI

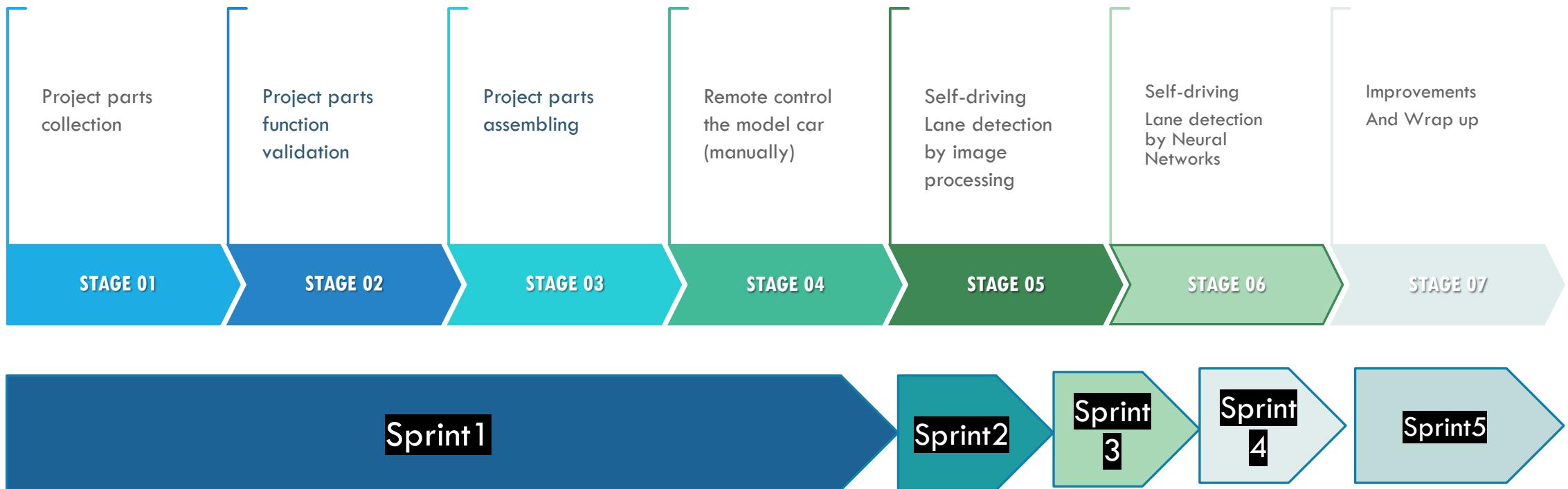
SENG 699 Capstone Project

Student: Zhen Li

Professor: Dr. Mohamed Abusharkh



SELF-DRIVING MODEL CAR PROJECT TIMELINE



sprint	sprint end and project demo	meeting date
Sprint 1	Week 11	Mar-29
Sprint 2	Week 12	April-5
Sprint 3	Week 13	April-12
Sprint 4	Week 14	April-19
Final code submission	Week 16	May-3

SPRINT PLAN

ID	Story	Estimation	Priority
1-01	As a user, I want to control the left-side and right-side motors' turning directions, so that the model car can go forward and backward.	3	1
1-02	As a user, I want to control the left-side and right-side motors' turning speeds, so that the model car can make turns at a curved path.	3	1
1-03	As a user, I want to control the camera, so that it can capture the path image.	8	2
1-04	As a user, I want to use a keyboard to control motors, so that I can control the direction of the model car.	5	1
2-01	As a user, I want the camera can automatically capture the path image in real time.	5	2
2-02	As a user, I want the path images will be processed into histograms.	8	2
3-01	As a user, I want the model car will drive itself with the processed image.	8	3
3-02	As a user, I want to record path videos when I control the car manually.	5	2
3-03	As a user, I want to use recorded videos to train my model.	8	2
4-01	As a user, I want the car to drive itself with the trained model.	13	3
4-02	As a user, I want to improve the model.	13	3
5-01	As a user, I want the model car can drive in more complicated conditions.	21	3

SPRINT 1

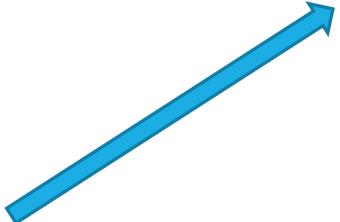
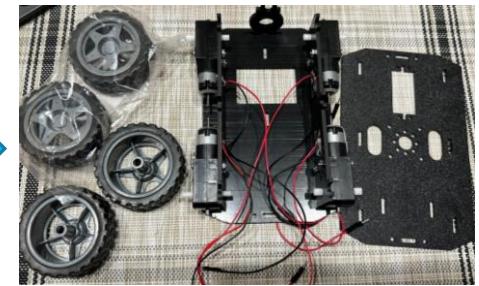
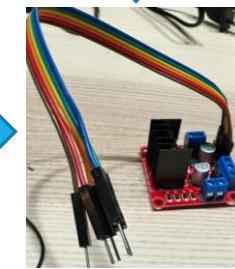
ID	Story	Estimation	Priority
1-01	As a user, I want to control the left-side and right-side motors' turning directions, so that the model car can go forward and backward.	3	1
1-02	As a user, I want to control the left-side and right-side motors' turning speeds, so that the model car can make turns at a curved path.	3	1
1-03	As a user, I want to control the camera, so that it can capture the path image.	8	2
1-04	As a user, I want to use a keyboard to control motors, so that I can control the direction of the model car.	5	1

3 points: know how to do it, take a bit of time

5 points: need some thinking to get the design right //multiple hours of programming

8 points: requires research, various methods maybe various classes to write

HARDWARE



HARDWARE VALIDATION

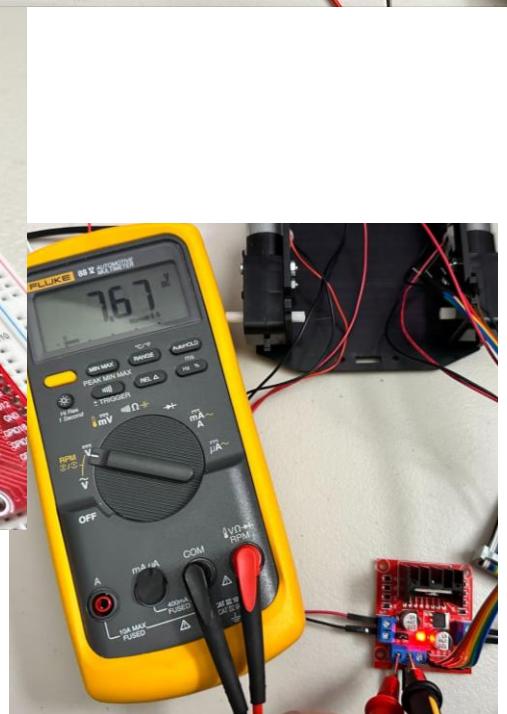
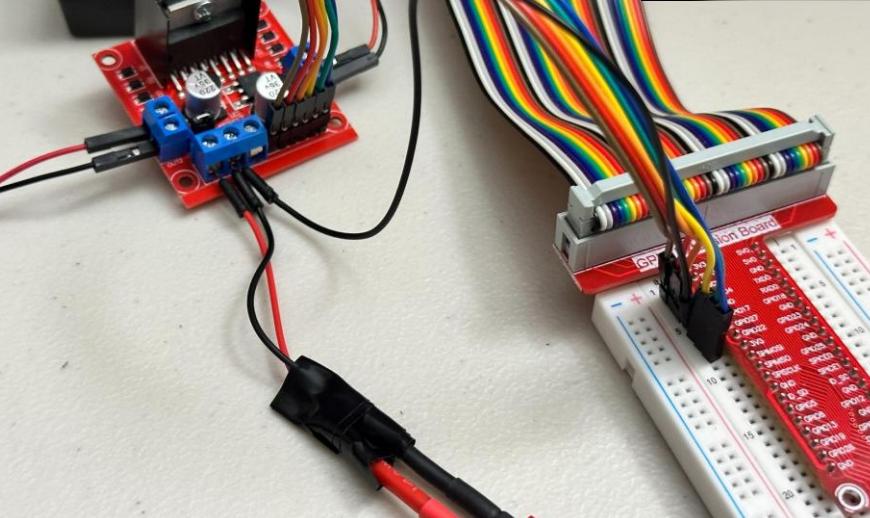
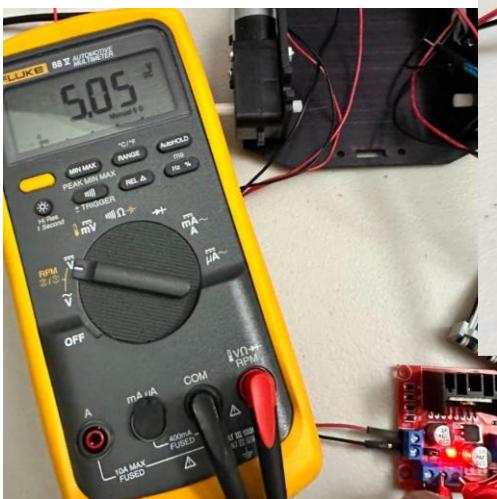
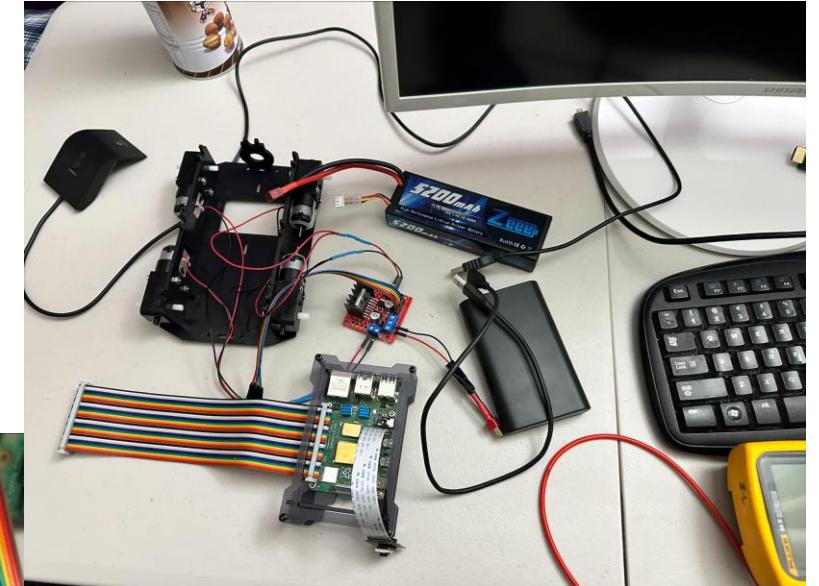
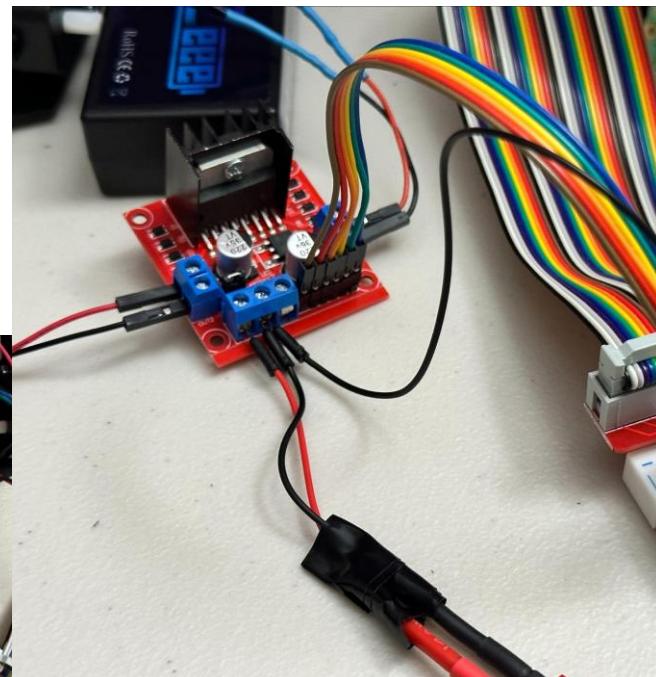
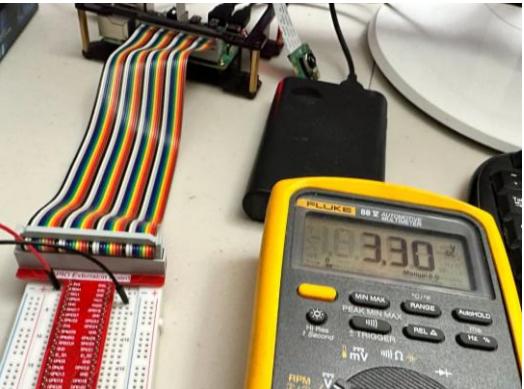
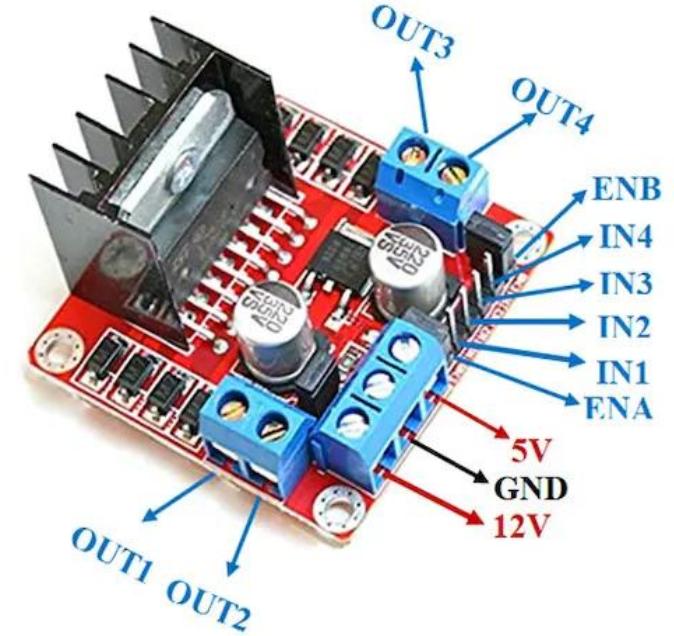


DIAGRAM PINOUT



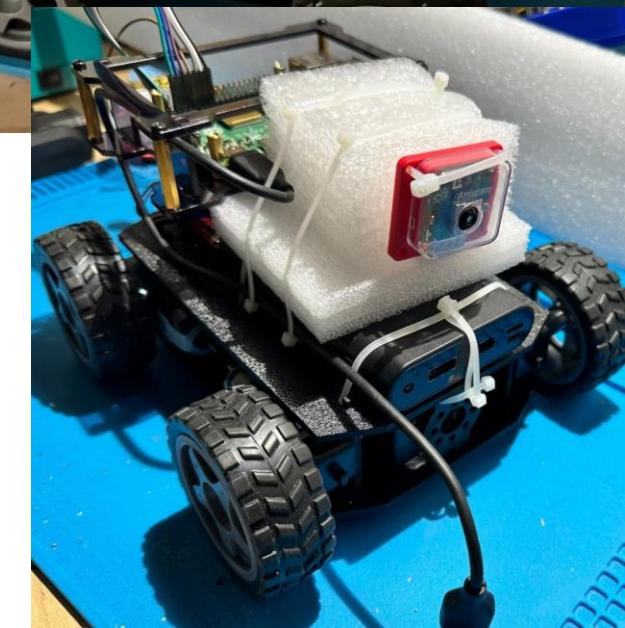
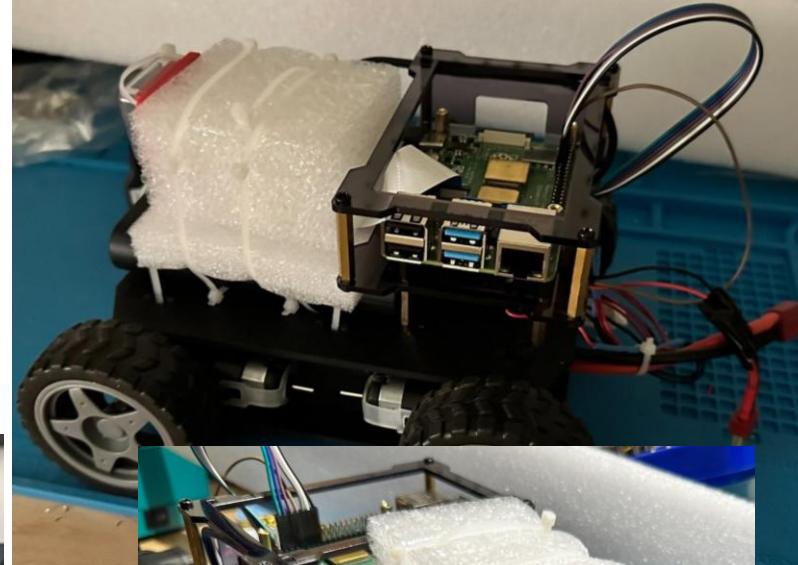
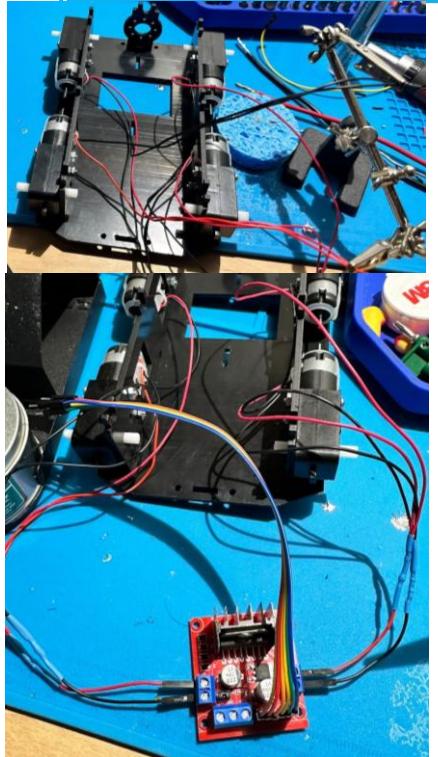
Physical Pins					
Function	BCM	pin#	pin#	BCM	Function
3.3 Volts		1	2		5 Volts
GPIO/SDA1 (I2C)	2	3	4		5 Volts
GPIO/SCL1 (I2C)	3	5	6		GND
GPIO/GCLK	4	7	8	14	TX UART/GPIO
GND		9	10	15	RX UART/GPIO
GPIO	17	11	12	18	GPIO
GPIO	27	13	14		GND
GPIO	22	15	16	23	GPIO
3.3 Volts		17	18	24	GPIO
MOSI (SPI)	10	19	20		GND
MISO(SPI)	9	21	22	25	GPIO
SCLK(SPI)	11	23	24	8	CEO_N (SPI)
GND		25	26	7	CE1_N (SPI)
RESERVED		27	28		RESERVED
GPIO	5	29	30		GND
GPIO	6	31	32	12	GPIO
GPIO	13	33	34		GND
GPIO	19	35	36	16	GPIO
GPIO	26	37	38	20	GPIO
GND		39	40	21	GPIO

<https://toptechboy.com/understanding-raspberry-pi-4-gpio-pinouts/>

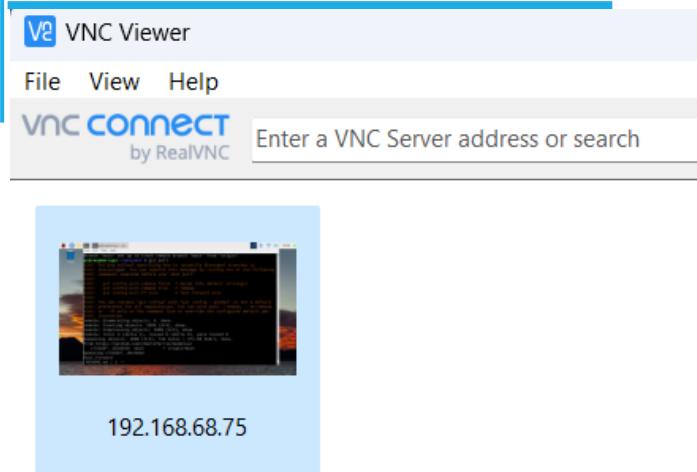


<https://components101.com/modules/i293n-motor-driver-module>

HARDWARE ASSEMBLING



DEVELOPMENT ENVIRONMENT



A screenshot of the Mu 1.0.3 Python IDE. The title bar says "Mu 1.0.3 - ManuallyControl.py". The toolbar has icons for Mode, New, Load, Save, Run, Debug, REPL, Plotter, Zoom-in, Zoom-out, Theme, Check, and Help. There are five tabs open: "LaneModule.py", "ManuallyControl.py" (which is the active tab), "WebcamModule.py", and "MainRobot.py". The code editor contains the following Python script:

```
1 from MotorModule import Motor
2 import KeyPressModule as kp
3
4 # define motor by gpio pinout number
5 motor = Motor(2,3,4,22,17,27)
6 kp.init()
7
8 def main():
9     if kp.getKey('UP'):
10         motor.move(0.6,0,0.1)
11     elif kp.getKey('DOWN'):
12         motor.move(-0.6,0,0.1)
13     elif kp.getKey('LEFT'):
14         motor.move(0.3,0.3,0.1)
15     elif kp.getKey('RIGHT'):
16         motor.move(0.3,-0.3,0.1)
17     else:
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
```

A screenshot of a GitHub repository page for "zhenliferris/modelcar". The top navigation bar shows "Code", "Issues", "Pull requests", "Actions", "Projects", "Wiki", "Security", and "Insights". The main content area shows a list of commits and files. On the right, there's an "About" section with a "No description, website, or topics provided." message, and sections for "Releases" (none) and "Create a new release".

Commit	Message	Date	Actions
zhenliferris remove tensorflow lit model	... (49)	yesterday	49
00_Parameters	re orgnized files	4 days ago	
01_ManuallyC...	adjust parameters for autonomou...	3 days ago	
02_ImageProc...	adjust parameters for autonomou...	3 days ago	
03_NeuralNet...	remove tensorflow lit model	yesterday	
README.md	update readme info	yesterday	
model.h5	run training at PC	2 days ago	

A screenshot of a Visual Studio Code workspace titled "Training.py - Untitled (Workspace)". The Explorer sidebar shows a file tree with "Training.py" selected. The code editor contains a Python script for training a neural network:

```
1 # cv2.waitKey(0)
2
3 # STEP 4 - SPLIT FOR TRAINING AND VALIDATION
4 xTrain, xVal, yTrain, yVal = train_test_split(imagesPath, steerings,
5                                               test_size=0.2, random_s
6 print('Total Training Images: ', len(xTrain))
7 print('Total Validation Images: ', len(xVal))
8
9 # STEP 5 - AUGMENT DATA
10
11 # STEP 6 - PREPROCESS
12
13 # STEP 7 - CREATE MODEL
14 model = createModel()
15
16 # STEP 8 - TRAINING
17 history = model.fit(dataGen(xTrain, yTrain, 100, 1),
18                      steps_per_epoch=100,
19                      epochs=10,
20                      validation_data=dataGen(xVal, yVal, 50, 0),
21                      validation_steps=50)
22
23 # STEP 9 - SAVE THE MODEL
24 model.save('model.h5')
25 print('Model Saved')
26
27 # STEP 10 - PLOT THE RESULTS
28 plt.plot(history.history['loss'])
```

SPRINT 1 COMPLEMENT

As a user, I want to control the left-side and right-side motors' turning directions, so that the model car can go forward and backward.

MotorModule.py

As a user, I want to control the left-side and right-side motors' turning speeds, so that the model car can make turns at a curved path.

MotroModule.py

As a user, I want to control the camera, so that it can capture the path image.

WebcamModule.py

As a user, I want to use a keyboard to control motors, so that I can control the direction of the model car.

keypressModule.py

SOFTWARE IMPLEMENT

<https://github.com/zhenliferris/modelcar.git>

 zhenliferris	Create README.md	dec034d 3 hours ago	 4 commits
 __pycache__	Reorganize the file name,sprint1 complete,manually control deployed	3 hours ago	
 ColorPickerScript.py	first commit	2 days ago	
 KeyPressModule.py	first commit	2 days ago	
 LaneDetectionModule.py	Reorganize the file name,sprint1 complete,manually control deployed	3 hours ago	
 LaneModule.py	add more files	yesterday	
 MainRobot.py	add more files	yesterday	
 ManuallyControl.py	Reorganize the file name,sprint1 complete,manually control deployed	3 hours ago	
 MotorModule.py	add more files	yesterday	
 MotorModule_test.py	add more files	yesterday	
 README.md	Create README.md	3 hours ago	
 WebcamModule.py	Reorganize the file name,sprint1 complete,manually control deployed	3 hours ago	
 utilis.py	add more files	yesterday	
 vid1.mp4	first commit	2 days ago	
 README.md			



ISSUES AND SOLUTIONS 3/27/23

1. limitation from the hardware, use speed different from both sides to control the model car's turning direction. Not accurate to make a sharp turn.

A: Could get better results with swiveled front wheels.

B. Mecanum wheels



<https://www.amazon.com/Mecanum-Wheel-Directional-Accessories-Components/dp/B08TV1HGMG?th=1>

2. When using “VideoCapture.open(0)” from OpenCV, the camera won't stream video, looks like the current, “libcamera” for Bullseye OS has a compatibility issue with OpenCV.

A: Tried to reinstall OpenCV with "pip install opencv-contrib-python" instead of "pip install opencv-python", not fix the problem, next step will try to enable "legacy camera stack", but it is old stack and will reduce the system performance.

ISSUES AND SOLUTIONS 3/29/23

Enable legacy camera stack and replace the camera module 3 (newest version) with module 1 (old version) fixed the OpenCV compatibility issue, but cause a new issue, the VNC viewer “cannot currently show the desktop”

[Raspberry Pi Documentation - The config.txt file](#)

hdmi_group	result
0	Auto-detect from EDID
1	CEA
2	DMT

Fix: force the Force HDMI – Hotplug by modify /boot/config.txt

`hdmi_force_hotplug=1`

`hdmi_group=2`

`hdmi_mode=4`

`hdmi_drive=2`

hdmi_mode	Resolution	Frequency	Screen Aspect
1	VGA (640x480)	60Hz	4:3
2	480p	60Hz	4:3
3	480p	60Hz	16:9
4	720p	60Hz	16:9

SPRINT 2

ID	Story	Estimation	Priority
2-01	As a user, I want to control 2 motor control modules, so that I can control 4 motors at the same time.	3	2
2-02	As a user, I want to control the 4 wheels' tuning speed and direction independently, so that the model car can make precise movements.	5	2
2-03	As a user, I want to control the camera, so that it can stream the road image.	5	2
2-04	As a user, I want the path images will be processed into histograms.	5	1
2-05	As a user, I want the 4 motors to move according to the result of the histograms.	8	1

3 points: know how to do it, take a bit of time

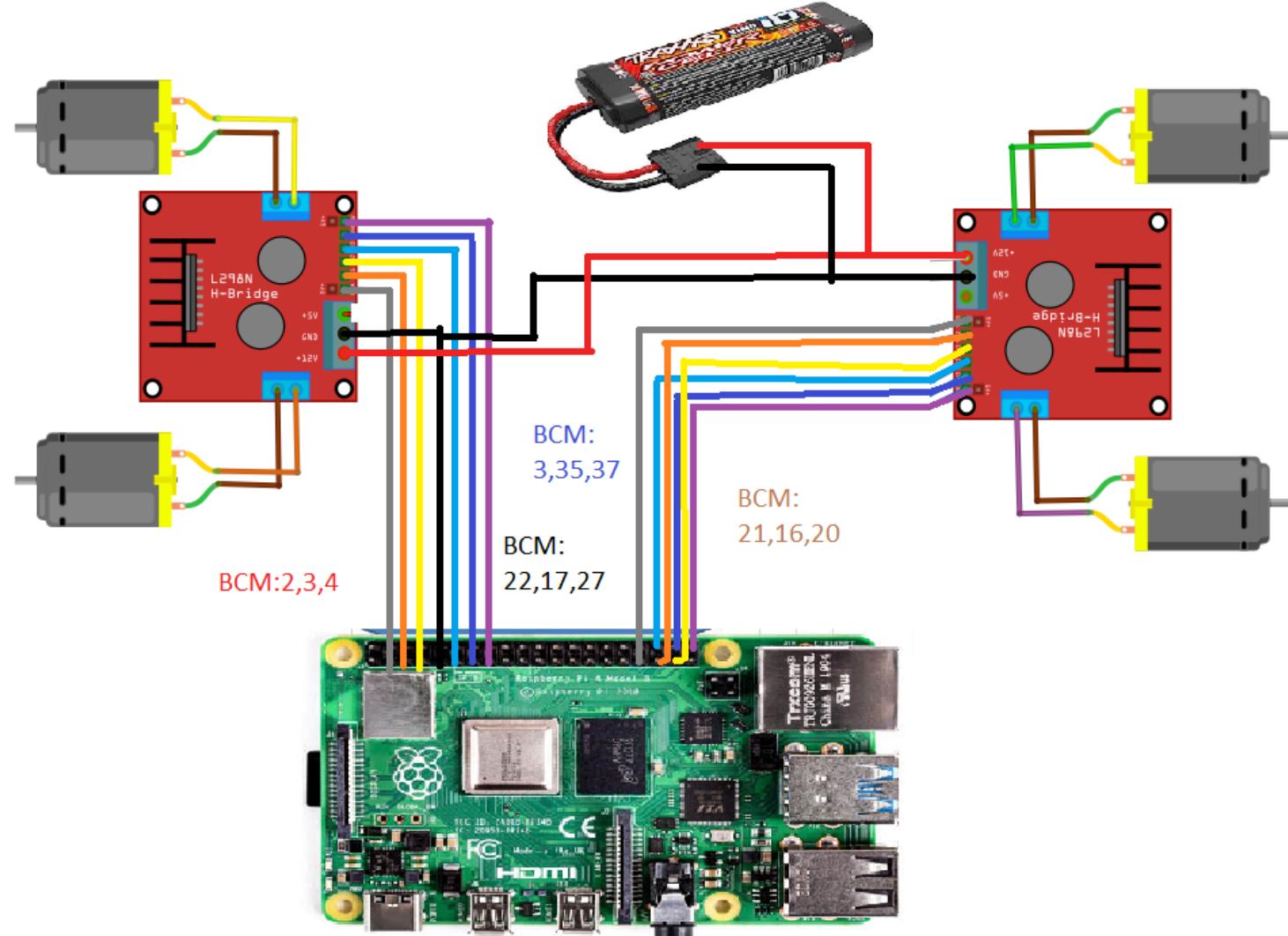
5 points: need some thinking to get the design right //multiple hours of programming

8 points: requires research, various methods maybe various classes to write

UPDATED DIAGRAMS

2L298N & 4 MOTORS

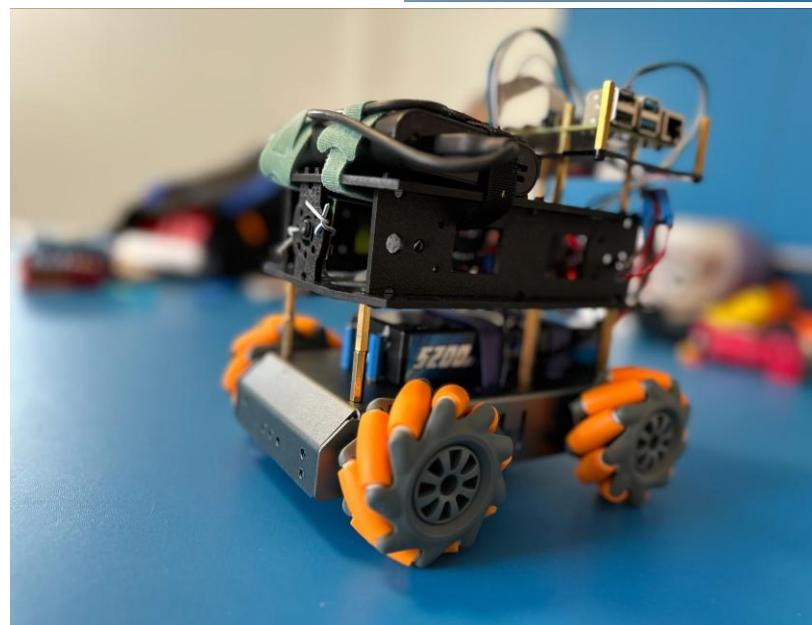
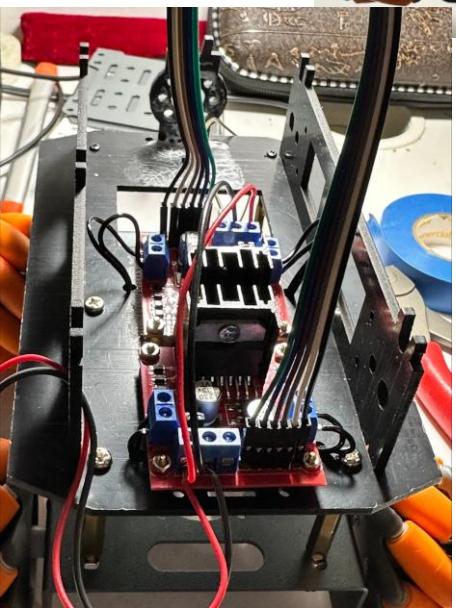
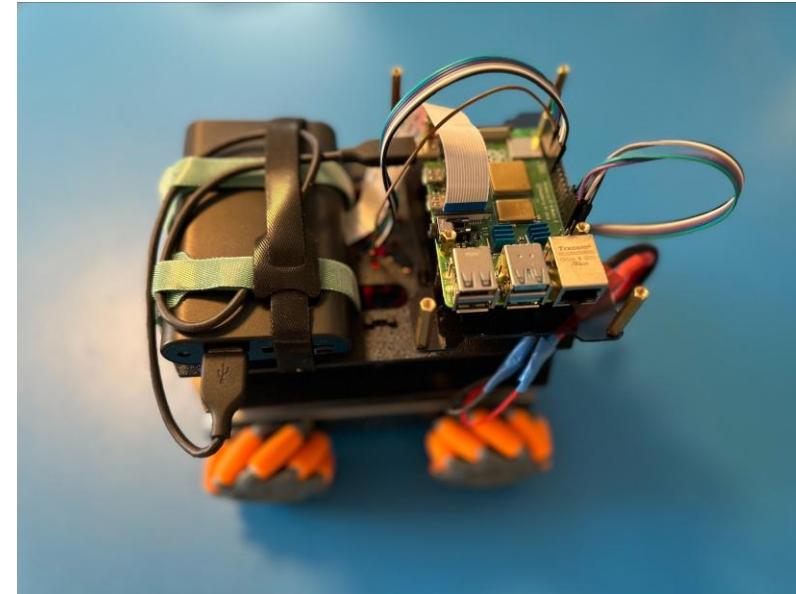
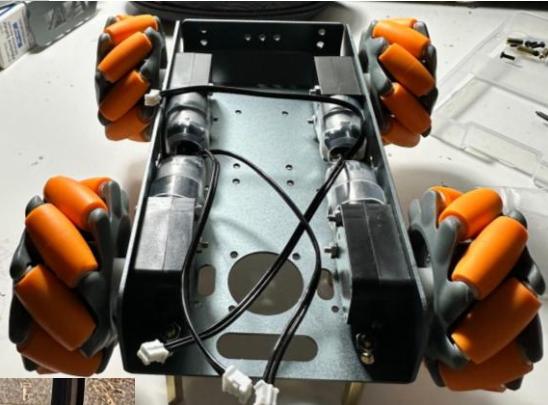
Physical Pins				Function	
Function	BCM	pin#	pin#	BCM	Function
3.3 Volts		1	2		5 Volts
GPIO/SDA1 (I2C)	2	3	4		5 Volts
GPIO/SCL1 (I2C)	3	5	6		GND
GPIO/GCLK	4	7	8	14	TX UART/GPIO
GND		9	10	15	RX UART/GPIO
GPIO	17	11	12	18	GPIO
GPIO	27	13	14		GND
GPIO	22	15	16	23	GPIO
3.3 Volts		17	18	24	GPIO
MOSI (SPI)	10	19	20		GND
MISO(SPI)	9	21	22	25	GPIO
SCLK(SPI)	11	23	24	8	CEO_N (SPI)
GND		25	26	7	CE1_N (SPI)
RESERVED		27	28		RESERVED
GPIO	5	29	30		GND
GPIO	6	31	32	12	GPIO
GPIO	13	33	34		GND
GPIO	19	35	36	16	GPIO
GPIO	26	37	38	20	GPIO
GND		39	40	21	GPIO



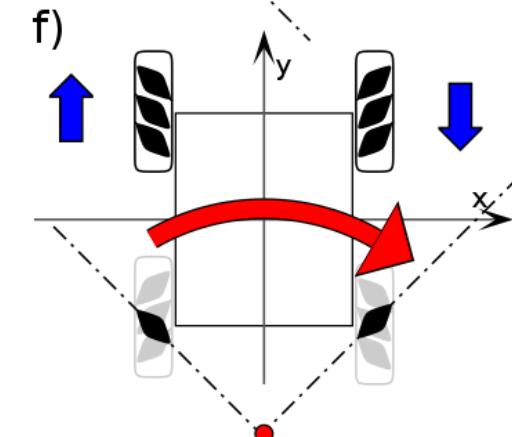
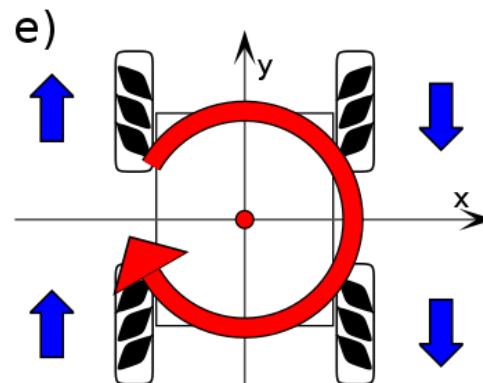
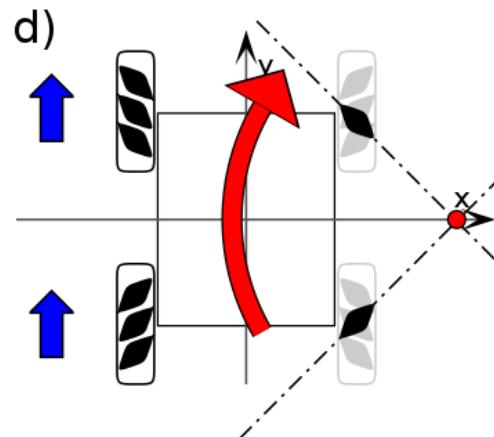
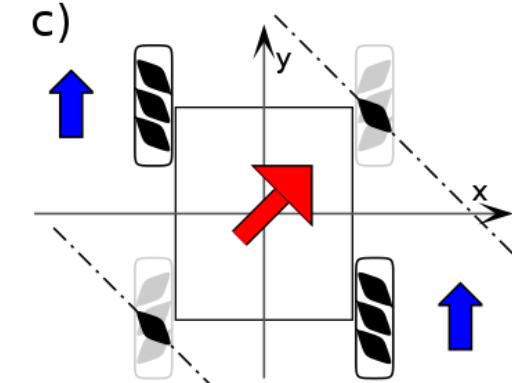
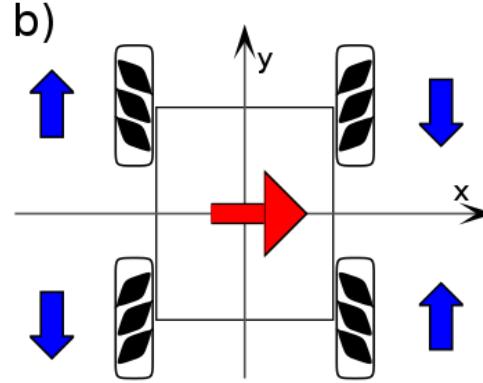
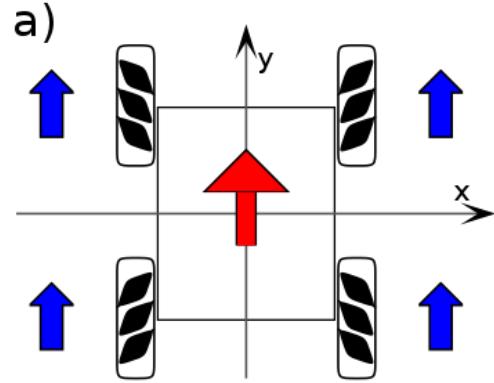
<https://toptechboy.com/understanding-raspberry-pi-4-gpio-pinouts/>

GPIO BCM— The BCM option refers to the pin by “Broadcom SOC Channel. They signify the Broadcom SOC channel designation. The BCM channel changes as the version number changes. The efficiency of L298N is 30%-70%, additional L298N causes total efficiency to be reduced.

HARDWARE UPDATE



MECANUM WHEEL CONTROL PRINCIPLE



https://en.wikipedia.org/wiki/Mecanum_wheel

MANUAL KEYS USED FOR MECANUM WHEELS MOVEMENTS



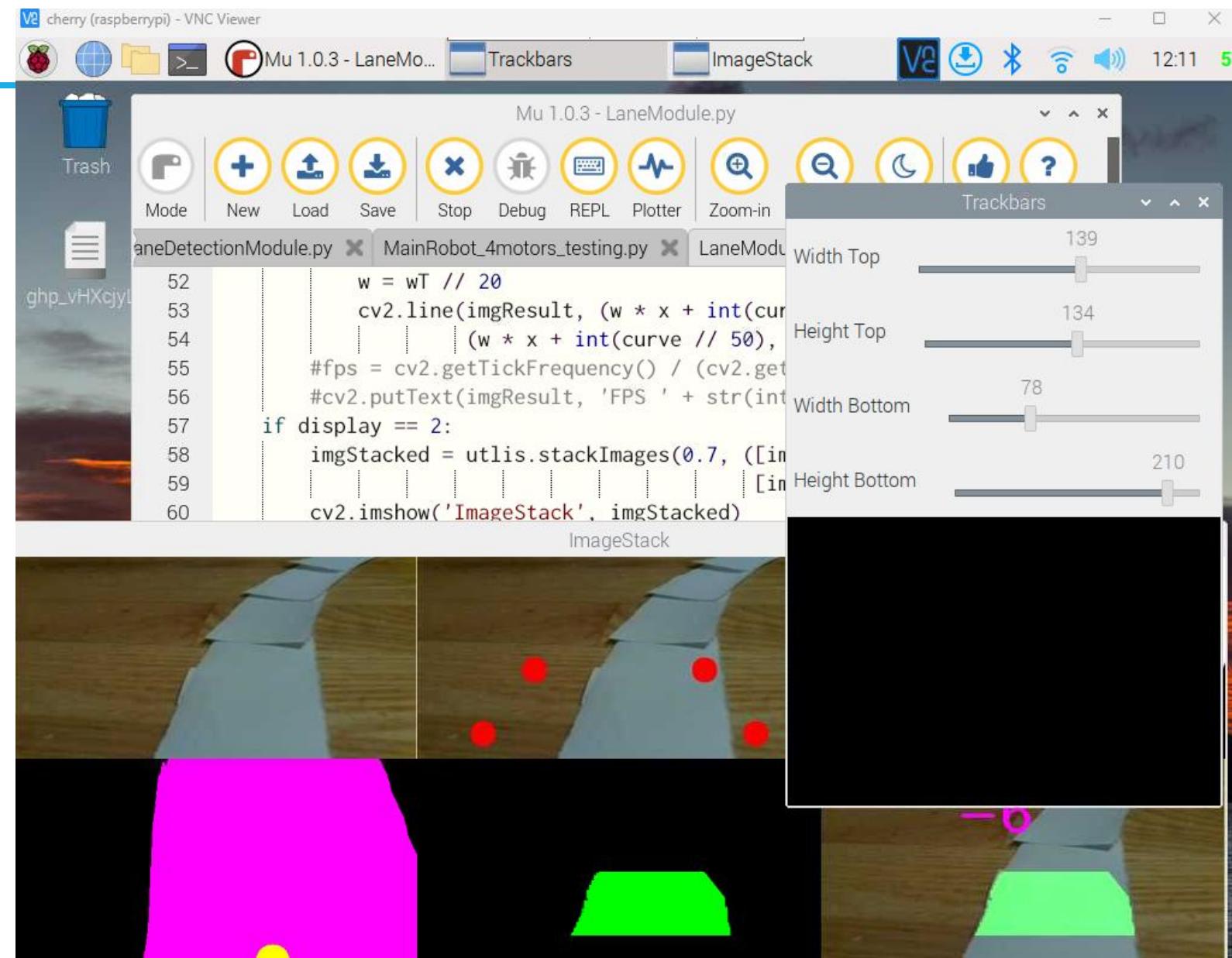
\uparrow $0.6 \uparrow$ $0.6 \uparrow$ $0.6 \uparrow$	D $0.28 \uparrow$ $-0.28 \downarrow$ $-0.28 \downarrow$	E $0.28 \uparrow$ 0 0	X $0.28 \uparrow$ $0.28 \uparrow$ $0.28 \uparrow$
\rightarrow $0.28 \uparrow$ 0	\downarrow $0.6 \downarrow$ $0.6 \downarrow$	A $-0.28 \downarrow$ $0.28 \uparrow$ $0.28 \uparrow$	Q 0 $0.28 \uparrow$
W $-0.28 \downarrow$ $-0.28 \downarrow$	\leftarrow $0.28 \uparrow$ 0	Z $-0.28 \downarrow$ 0 0	C 0 $-0.28 \downarrow$

Keyboard defined for Mecanum wheel movement

IMAGE PROCESSING

Find the path using color detection and then get the curve using the summation of pixels in the histogram.

- a. Mask background
- b. Wrap the image
- c. Find the critique area
- d. Find the middle point
- e. Find the curve



SOFTWARE IMPLEMENT

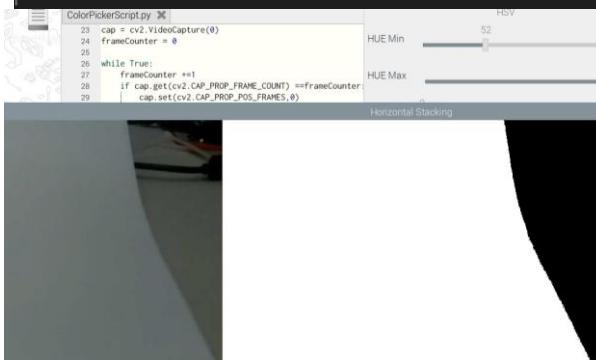
<https://github.com/zhenliferris/modelcar.git>

 zhenliferris	auto following the path with image processing implemated	a25f90e 4 hours ago	 25 commits
 __pycache__	auto following the path with image processing implemated	4 hours ago	
 ColorPickerScript.py	first commit	last week	
 KeyPressModule.py	first commit	last week	
 LaneDetectionModule.py	testing	5 hours ago	
 LaneModule.py	auto following the path with image processing implemated	4 hours ago	
 MainRobot.py	add more files	last week	
 MainRobot_4motors.py	fix confilic	yesterday	
 MainRobot_4motors_testing.py	auto following the path with image processing implemated	4 hours ago	
 ManuallyControl.py	typo fixed	3 days ago	
 ManuallyControl_4motors.py	testing	5 hours ago	
 MotorModule.py	add more files	last week	
 MotorModule_4motors.py	fix confilic	yesterday	
 README.md	Update README.md	4 days ago	
 WebcamModule.py	auto following the path with image processing implemated	4 hours ago	
 imageProcessAuto.py	fix name error for _motor	yesterday	
 utilis.py	add more files	last week	
 vid1.mp4	first commit	last week	

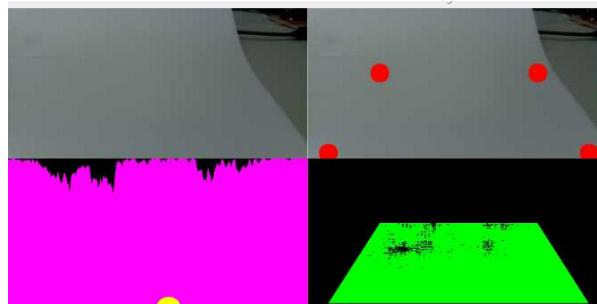


PARAMETERS AFFECTING RESULTS

```
def thresholding(img):
    imgHsv = cv2.cvtColor(img, cv2.COLOR_BGR2HSV)
    lowerWhite = np.array([50, 0, 0])
    upperWhite = np.array([255, 160, 255])
    maskWhite = cv2.inRange(imgHsv, lowerWhite, upperWhite)
    return maskWhite
```



```
initialTrackBarVals = [114, 109, 32, 231]
utlis.initializeTrackbars(initialTrackBarVals)
```



```
sen = 0.9 # SENSITIVITY
maxVal = 0.1 # MAX SPEED
if curveVal > maxVal:
    curveVal = maxVal
if curveVal < -maxVal:
    curveVal = -maxVal
print(curveVal)
if curveVal > 0:
    if curveVal < 0.03:
        curveVal = 0
else:
    if curveVal > -0.03:
        curveVal = 0

turnVal = -curveVal*sen

motor.move(0.20, turnVal, 0.20, turnVal, 0.05)
```

SPRINT 2 COMPLEMENT

As a user, I want to control 2 motor control modules, so that I can control 4 motors at the same time.

MotorModule_4motors.py

As a user, I want to control the 4 wheels' tuning speed and direction independently, so that the model car can make precise movements.

ManuallyControl_4motors.py

As a user, I want to control the camera, so that it can stream the road image.

WebcamModule.py

As a user, I want the path images to be processed into histograms. →small stories

- As a user, I want to mask the background, so that only the paper shows white.
- As a user, I want the wrap the image captured by the camera, so that changing from the front view to the top view
- As a user, I want that the camera sees the middle point of the path edge line.
- As a user, I want to know the curve of the path based on the weight of white.

LaneModule.py

As a user, I want the 4 motors to move according to the result of the histograms.

MainRobot_4motors.py

SPRINT 3

ID	Story	Estimation	Priority
3-01	As a user, I want to control the model car with a joystick.	5	1
3-02	As a user, I want to record path videos when I control the car with a joystick controller.	5	1
3-03	As a user, I want the stored video can be transferred to pictures.	5	1
3-04	As a user, I want to use pictures to train my model.	5	1
3-05	As a user, I want the car to drive itself with the trained model.	13	2

3 points: know how to do it, take a bit of time

5 points: need some thinking to get the design right //multiple hours of programming

8 points: requires research, various methods maybe various classes to write

REARRANGED FOLDER

```
└── modelcar
    ├── 00_Parameters
    ├── 01_ManuallyControl
    └── 02_ImageProcess
    └── 03_NeuralNetworks
        └── 001_DataCollection
            ├── __pycache__
            ├── CamModule.py
            ├── DataCollectionModule.py
            ├── JoyStickModule.py
            ├── ManuallyDrivingDataCollection.py
            ├── MotorModule.py
            └── utlis.py
    ├── 002_Tranining
    └── 003_SelfDring
    └── README.md
```

```
# 00 Parameters finders and Code Testing
# 1. CodeTester
# 2. ColorPickerScript
# 3. vid1.mp4
```

```
# 01 Manually control:
```

```
# 1. JoyStickModule
# 2. KeyPressModule
# 3. MotorModule
# 4. ManuallyKeyboardControl
# 5. ManuallyStickControl
```

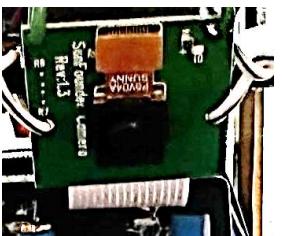
```
# 02 Autonomous Model Car by Image Processing.
```

```
# 1. LaneModule
# 2. MainRobot
# 3. MotorModule
# 4. Utlis
# 5. vid1.mp4
# 6. WebcamModule
```

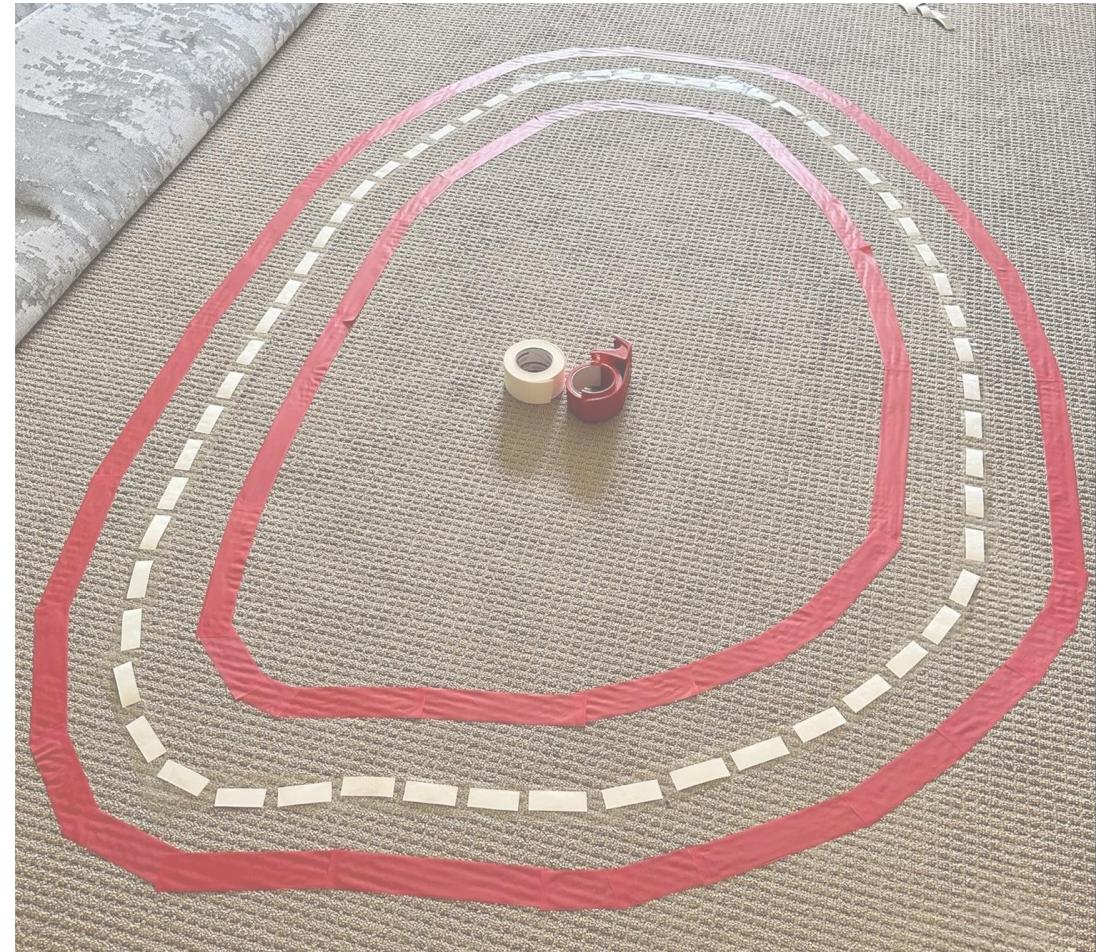
```
# 03 Autonomous Model Car by Neural Networks
```

```
# 1. CamModule
# 2. DataCollectionMain
# 3. DataCollectionModule
# 4. JoyStickModule
# 5. MotorModule
# 6. utlis
# 7. Training
# 8. AutonomousNN
```

DATA COLLECTION



DataCollected	
>	IMG16
>	IMG17
>	IMG18
>	IMG19
>	IMG20
>	IMG21
grid	log_16.csv
grid	log_17.csv
grid	log_18.csv
grid	log_19.csv
grid	log_20.csv
grid	log_21.csv



ISSUES AND SOLUTIONS 4/9/23

```
joyVal = jsM.getJS()
# print(joyVal)
steering = joyVal['axis1']
throttle = joyVal['o']*maxThrottle
if joyVal['t'] == 1:
    if record == 0:
        print('Recording Started ...')
    record += 1
    sleep(0.1)
```



1. When clicking Δ button and starting to collect data, the model car will respond slower and affect the data recorded
2. Don't have multiple racetracks to collect data.

Reducing sleep time will reduce the system response time delay but increase the sample image taken. Adjust to an appropriate value.

When collecting the data, let the model car go clockwise and counterclockwise to expand the sampling range.

ISSUES AND SOLUTIONS 4/9/23

How to install PS4 joystick to Raspberry Pi

```
sudo apt-get -y install jd
```

```
sudo pip3 install ds4drv
```

```
sudo wget https://raw.githubusercontent.com/chrippa/ds4drv/master/udev/50-ds4drv.rules -O /etc/udev/rules.d/50-ds4drv.rules
```

```
sudo udevadm control --reload-rules
```

```
sudo udevadm trigger
```

```
sudo nano /etc/rc.local
```

add after # By default this script does nothing. line, add a new line: /usr/local/bin/ds4drv &

```
sudo reboot
```

“share” & “playstation” to search Bluetooth

SPRINT 3 COMPLEMENT

As a user, I want to control the model car with a joystick.

JoyStickModule.py

As a user, I want to record path images

DatacollectionModule.py

As a user, I want to record steering parameters when I control the car with a joystick.

ManuallyDrivingDatacollection.py

As a user, I want to record the index of images and steering parameters into log files

DatacollectionModule.py

SPRINT 4

ID	Story	Estimation	Priority
4-01	As a user, I want to access to the recorded images and log files.	3	1
4-02	As a user, I want to visualize the recorded images using plots.	5	2
4-03	As a user, I want to prepare the image for the model.	8	1
4-04	As a user, I want to use the processed images and steering parameter to train a model	3	1
4-05	As a user, I want to save the model.	3	2

3 points: know how to do it, take a bit of time

5 points: need some thinking to get the design right //multiple hours of programming

8 points: requires research, various methods maybe various classes to write

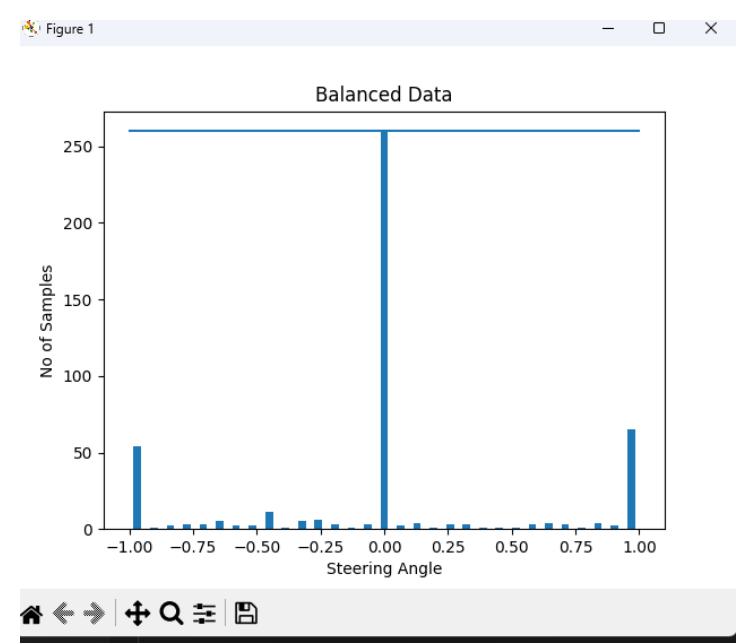
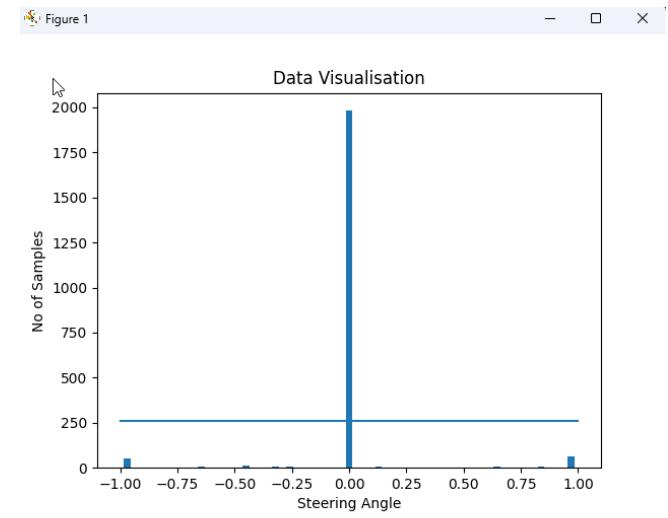
TRAINING PROCESS



File Path	Steering Angle
/home/pi/project/03_NeuralNetworks/001_DataCollection/DataCollected/IMG21/Image_1681245162361836.jpg	0.25
/home/pi/project/03_NeuralNetworks/001_DataCollection/DataCollected/IMG21/Image_168124516258938.jpg	0
/home/pi/project/03_NeuralNetworks/001_DataCollection/DataCollected/IMG21/Image_1681245162808368.jpg	0
/home/pi/project/03_NeuralNetworks/001_DataCollection/DataCollected/IMG21/Image_1681245163025256.jpg	0
/home/pi/project/03_NeuralNetworks/001_DataCollection/DataCollected/IMG21/Image_1681245163243266.jpg	0
/home/pi/project/03_NeuralNetworks/001_DataCollection/DataCollected/IMG21/Image_1681245163461273.jpg	0
/home/pi/project/03_NeuralNetworks/001_DataCollection/DataCollected/IMG21/Image_1681245163678235.jpg	0
/home/pi/project/03_NeuralNetworks/001_DataCollection/DataCollected/IMG21/Image_1681245163897017.jpg	-0.31
/home/pi/project/03_NeuralNetworks/001_DataCollection/DataCollected/IMG21/Image_1681245164114726.jpg	-0.56
/home/pi/project/03_NeuralNetworks/001_DataCollection/DataCollected/IMG21/Image_1681245164332715.jpg	0
/home/pi/project/03_NeuralNetworks/001_DataCollection/DataCollected/IMG21/Image_1681245164550649.jpg	0
/home/pi/project/03_NeuralNetworks/001_DataCollection/DataCollected/IMG21/Image_1681245164769661.jpg	0
/home/pi/project/03_NeuralNetworks/001_DataCollection/DataCollected/IMG21/Image_1681245164987842.jpg	0
/home/pi/project/03_NeuralNetworks/001_DataCollection/DataCollected/IMG21/Image_1681245165205547.jpg	0
/home/pi/project/03_NeuralNetworks/001_DataCollection/DataCollected/IMG21/Image_168124516542348.jpg	-0.27
/home/pi/project/03_NeuralNetworks/001_DataCollection/DataCollected/IMG21/Image_1681245165641638.jpg	0

```
def balanceData(data, display=True):
    nBin = 31
    samplesPerBin = 260
    hist, bins = np.histogram(data['Steering'], nBin)
    if display:
        plt.bar(bins[:-1], hist, width=0.05)
        plt.plot([0], [260], 'r')
        plt.title('Data Visualisation')
        plt.xlabel('Steering Angle')
        plt.ylabel('No of Samples')
        plt.show()
```

- a. INITIALIZE DATA
- b. VISUALIZE AND BALANCE DATA
- c. PREPARE(separate images and steering)
- d. SPLIT SAMPLES FOR TRAINING AND VALIDATION
- e. AUGMENT DATA(translate_percentage, scale, brightness, flip)
- f. PREPROCESS(cvtColor, Blur, resize, normalize)
- g. MODEL CREATION
- h. TRAINING
- i. SAVE THE MODEL



TRAINING RESULT

The screenshot shows the Visual Studio Code interface during training. On the right, a terminal window displays training logs with the command "samplesPerBin = 260". The logs show ETA and loss values decreasing over time. On the left, a code editor window shows the "Training.py" script, which includes a line of code: `plt.plot(history.history['loss'])`. Below the code editor is a status bar indicating "Model Saved". At the bottom, a terminal window shows the total number of images imported (2181) and the count of removed images (1721), leaving 460 remaining images. It also lists the total training and validation images.

```
samplesPerBin = 260

ETA: 3s - loss: 0.1916
ETA: 3s - loss: 0.1914
ETA: 3s - loss: 0.1912
ETA: 2s - loss: 0.1910
ETA: 2s - loss: 0.1907
ETA: 1s - loss: 0.1900
ETA: 1s - loss: 0.1893
ETA: 1s - loss: 0.1899
ETA: 0s - loss: 0.1900
ETA: 0s - loss: 0.1904
ETA: 0s - loss: 0.1899

40s 397ms/step - loss: 0.1899 - val_loss: 0.2223

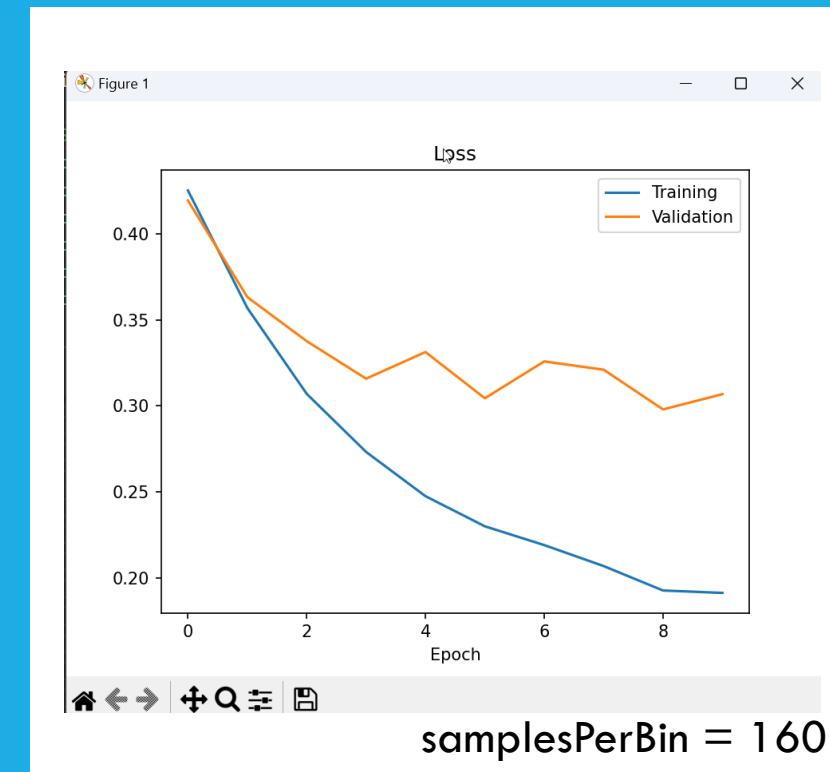
Total Images Imported 2181
Removed Images: 1721
Remaining Images: 460
Total Training Images: 368
Total Validation Images: 92
```

Underfitting

The model is too simple

Training data is not cleaned and contains noise in it

The size of the training data collected is not big enough.



The screenshot shows the Visual Studio Code interface with the "002_Training" folder selected. The code editor displays the "Training.py" script, which includes the line `plt.plot(history.history['loss'])`. The terminal window shows the full execution of the script, including the plotting of the loss history. The output indicates "Model Saved" and provides detailed log entries for each epoch, showing both training and validation loss values.

```
47 # STEP 10 - PLOT THE RESULTS
48 plt.plot(history.history['loss'])
49 plt.plot(history.history['val_loss'])
50 plt.title('Loss')
51 plt.xlabel('Epoch')
52 plt.show()
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

Model Saved

samplesPerBin = 160

SPRINT 4 COMPLEMENT

As a user, I want to access the recorded images and log files.

`training.py`

As a user, I want to visualize the recorded images using plots.

`utils.py`

As a user, I want to prepare the image for the model.

`utils.py training.py`

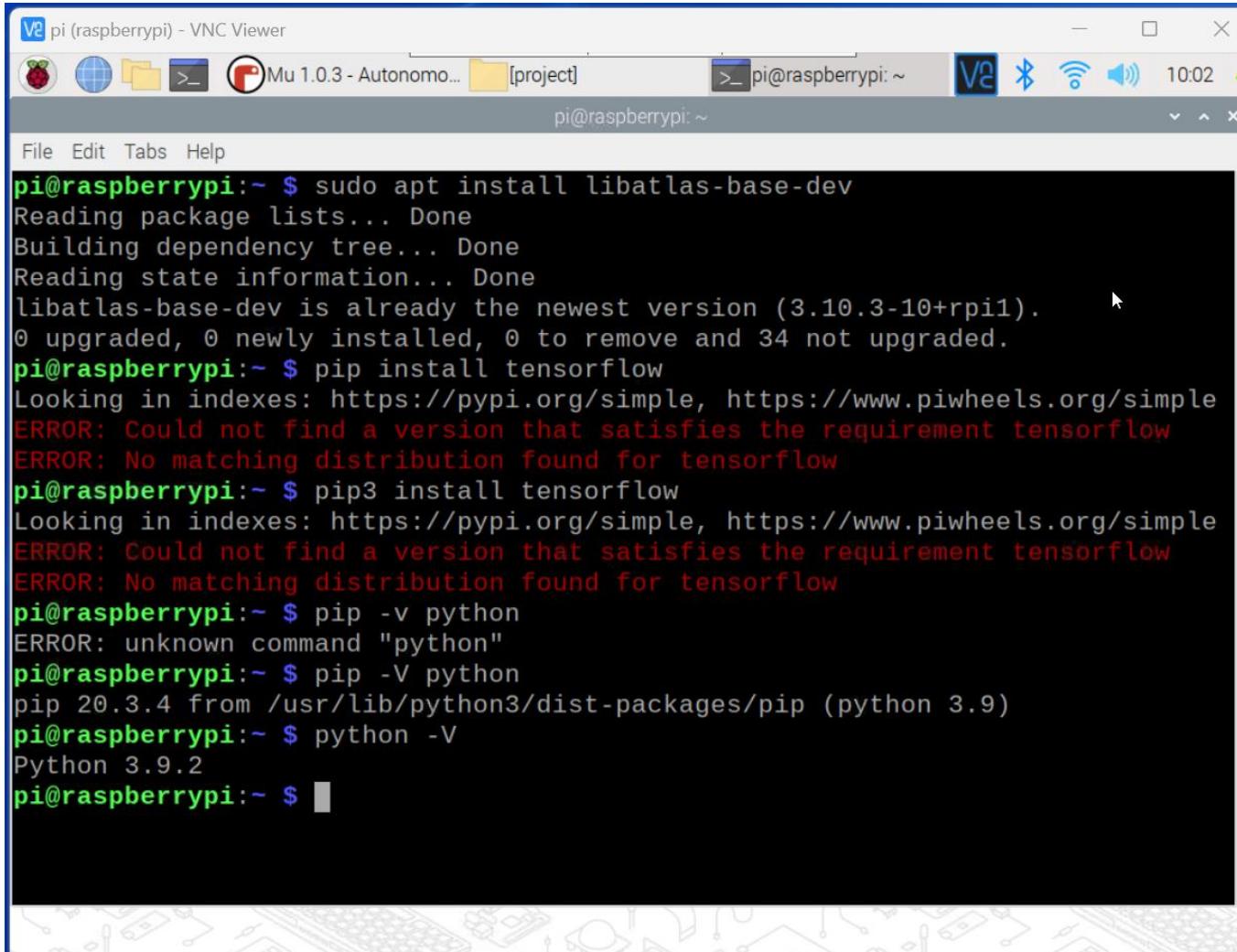
As a user, I want to use the processed images and steering parameter to train a model.

`utils.py training.py`

As a user, I want to save the model.

`training.py`

ISSUES AND SOLUTIONS 4/12/23



```
pi@raspberrypi:~ $ sudo apt install libatlas-base-dev
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
libatlas-base-dev is already the newest version (3.10.3-10+rpi1).
0 upgraded, 0 newly installed, 0 to remove and 34 not upgraded.

pi@raspberrypi:~ $ pip install tensorflow
Looking in indexes: https://pypi.org/simple, https://www.piwheels.org/simple
ERROR: Could not find a version that satisfies the requirement tensorflow
ERROR: No matching distribution found for tensorflow

pi@raspberrypi:~ $ pip3 install tensorflow
Looking in indexes: https://pypi.org/simple, https://www.piwheels.org/simple
ERROR: Could not find a version that satisfies the requirement tensorflow
ERROR: No matching distribution found for tensorflow

pi@raspberrypi:~ $ pip -v python
ERROR: unknown command "python"

pi@raspberrypi:~ $ pip -V python
pip 20.3.4 from /usr/lib/python3/dist-packages/pip (python 3.9)

pi@raspberrypi:~ $ python -V
Python 3.9.2

pi@raspberrypi:~ $
```

ISSUES AND SOLUTIONS 4/15/23



Raspberry Pi OS with desktop

Release date: February 21st 2023

System: 32-bit

Kernel version: 5.15

Debian version: 11 (bullseye)

Size: 924MB

[Show SHA256 file integrity hash:](#)

[Release notes](#)

Raspberry Pi OS with desktop

Release date: February 21st 2023

System: 64-bit

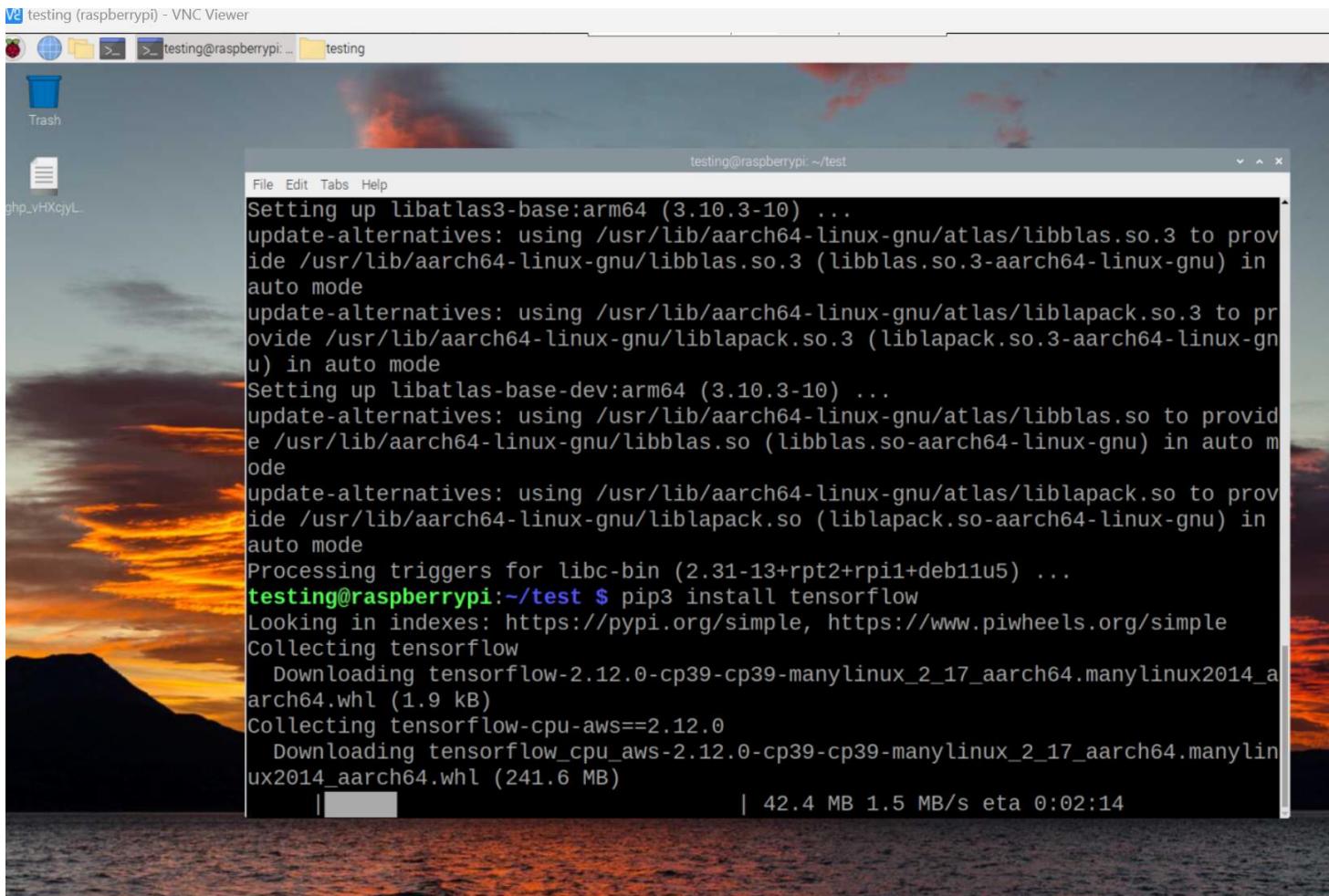
Kernel version: 5.15

Debian version: 11 (bullseye)

Size: 816MB

[Show SHA256 file integrity hash:](#)

[Release notes](#)



SPRINT 5

ID	Story	Estimation	Priority
5-01	As a user, I want to access the saved model.	3	1
5-02	As a user, I want to capture path images.	5	1
5-03	As a user, I want to prepare the image for the model.	5	1
5-04	As a user, I want the model to predict the steering value for each input image.	5	1
5-05	As a user, I want the motors to turn according to the steering values.	5	2

3 points: know how to do it, take a bit of time

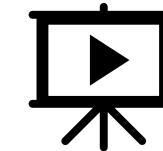
5 points: need some thinking to get the design right //multiple hours of programming

8 points: requires research, various methods maybe various classes to write

SPRINT 5 COMPLEMENT

As a user, I want to access the saved model.

AutonomousNN.py tensorflow → keras → load_model()



As a user, I want to capture path images.

CamModule.py

As a user, I want to prepare the image for the model.

AutonomousNN.py same method as used in training data process

As a user, I want the model to predict the steering value for each input image.

AutonomousNN.py model predicts steering values according to the images fed by the camera.

As a user, I want the motors to turn according to the steering values.

MotorModule.py

UPDATE AND IMPROVEMENT

ID	Story	Estimation	Priority
6-01	I want to add a camera mount; so that I can adjust to a better angle for camera view.	3	1
6-02	I want to collect a bigger set of data to solve the underfitting issue.	3	1
6-03	I want to change the test/validation ratio from 80:20 to 95:05 to see if the model benefit from the change.	3	1
6-04	I want to increase “steps_per_epoch” to see if the model benefit from the change.	3	1

3 points: know how to do it, take a bit of time

5 points: need some thinking to get the design right //multiple hours of programming

8 points: requires research, various methods maybe various classes to write

CAMERA MOUNT 6-01



3D Print file:

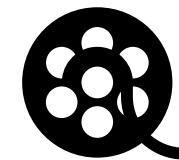
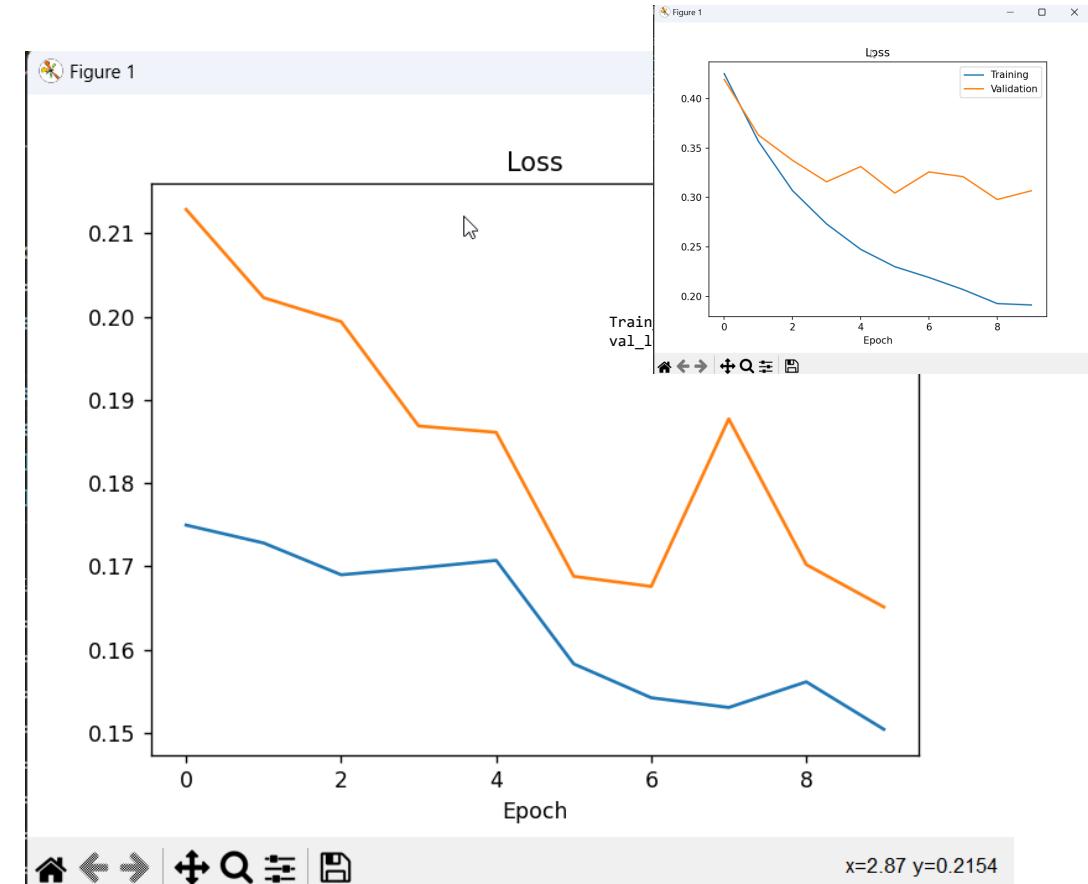
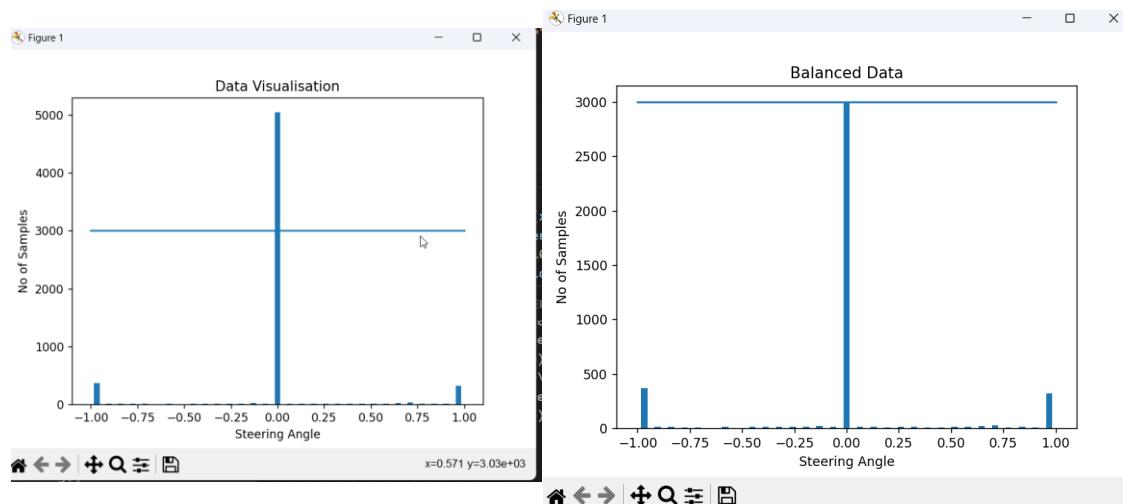
<https://www.thingiverse.com/thing:1579789>

Adjust the angle for the camera mount



INCREASE SAMPLE SIZE (NEW BASELINE) 6-02

```
Total Images Imported 6080
          Center   Steering
0  IMG0\Image_1681926365124344.jpg      0.0
1  IMG0\Image_1681926365717247.jpg      0.0
2  IMG0\Image_1681926365969177.jpg      0.0
3  IMG0\Image_1681926366212663.jpg      0.0
4  IMG0\Image_1681926366446208.jpg      0.0
Removed Images: 2045
Remaining Images: 4035
Total Training Images: 3228
Total Validation Images: 807
```



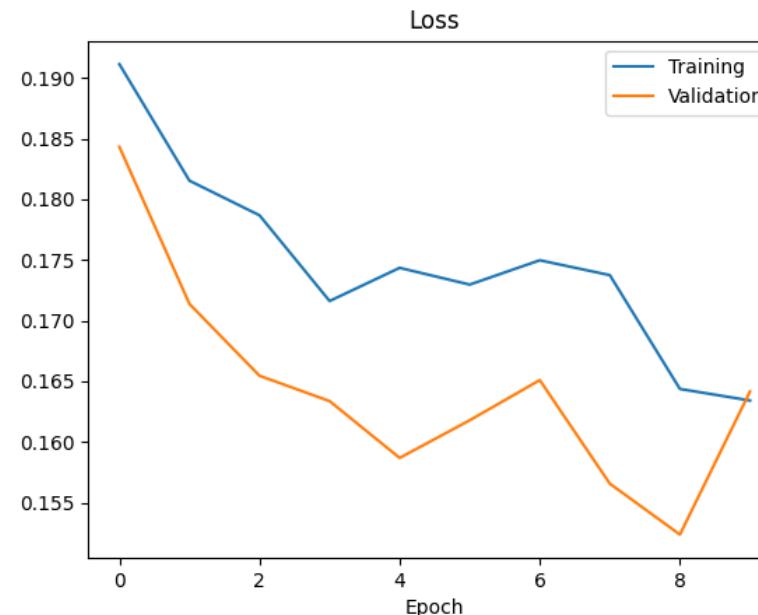
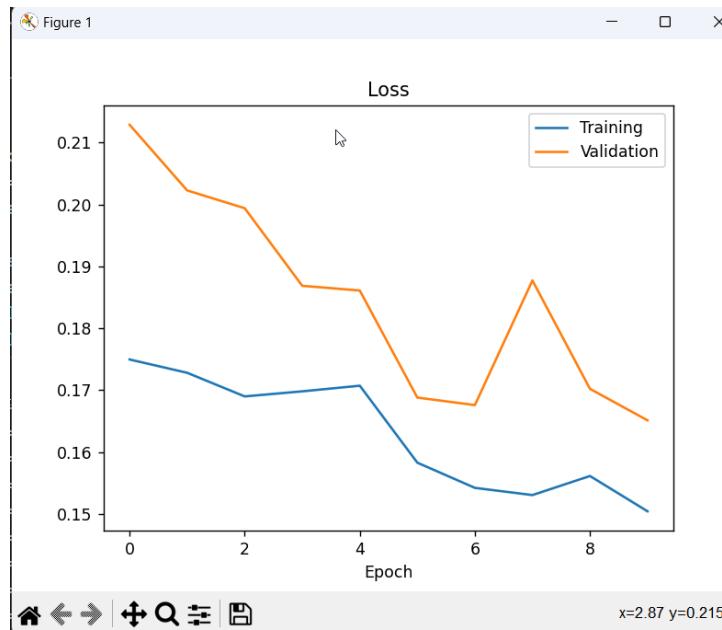
Count clockwise One round of track



Clockwise one round of track

ONLY CHANGE TEST/VALIDATION RATIO 6-03

```
xTrain, xVal, yTrain, yVal = train_test_split(imagesPath, steerings,  
                                              test_size=0.05, random_state=10)
```

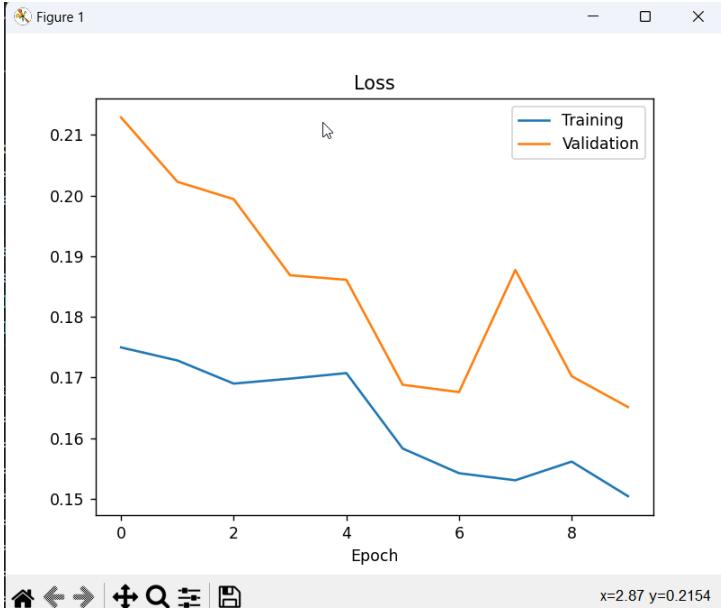


```
Total Images Imported 6080  
Center Steering  
0 IMG0\Image_1681926365124344.jpg 0.0  
1 IMG0\Image_1681926365717247.jpg 0.0  
2 IMG0\Image_1681926365969177.jpg 0.0  
3 IMG0\Image_1681926366212663.jpg 0.0  
4 IMG0\Image_1681926366446208.jpg 0.0  
Removed Images: 2045  
Remaining Images: 4035  
Total Training Images: 3833  
Total Validation Images: 202
```

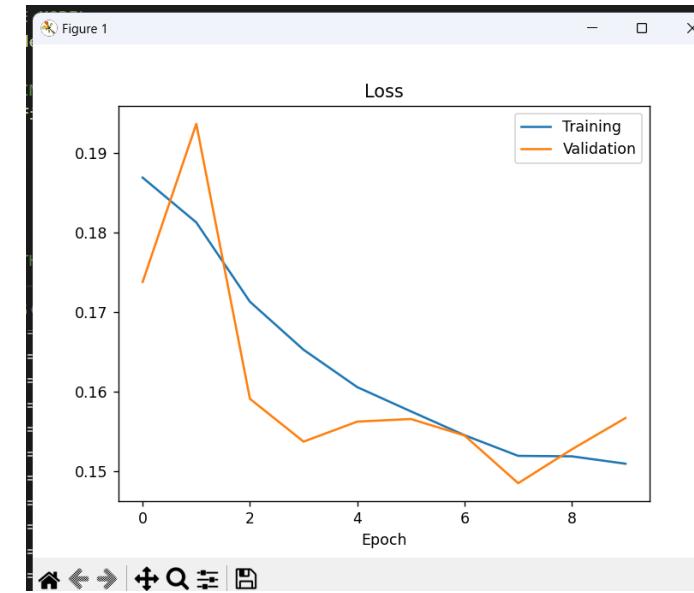
Train/Val = 0.80/0.20
Train_loss: 0.1504
val_loss: 0.1651

Train/Val = 0.95/0.05
Train_loss: 0.1634
val_loss: 0.1642

ONLY INCREASE “STEPS_PER_EPOCH” 6-04

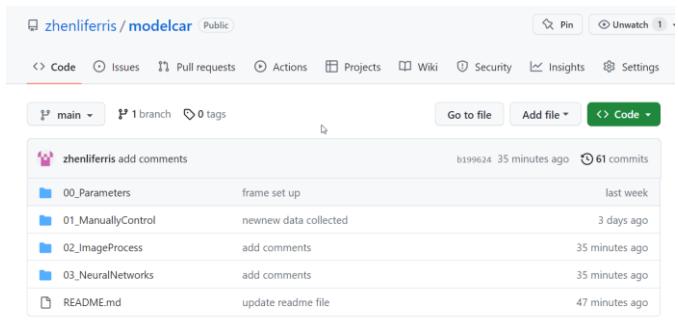


steps_per_epoch=100
Train_loss: 0.1504
Val_loss: 0.1651

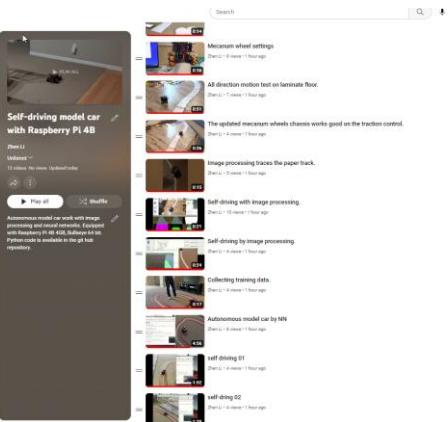


steps_per_epoch=300
Train_loss: 0.1509
Val_loss: 0.1567

Project Repo



Project YouTube playlist



PROJECT LINKS

FUTURE IDEAS

1. add Coral to improve system performance.

Products

Helping you bring local AI to applications from prototype to production



https://lh3.googleusercontent.com/-R0H37d9aKorHo_VYWf8hCfukvbZoIBaW2SHW1uDDn1G411r3MqemjxPZa9f44q8OwlfYlkGxSoj-GQbZGd2j7IxtyzSkIIQVUWvo9r88mn8CzB-rcw=w2000-rw

2. Build more racetracks with different lengths and sharp to collect data and verify model performance.

REFERENCE

<https://www.computervision.zone/courses/self-driving-car-using-raspberry-pi/>

https://www.raspberrypi.com/documentation/computers/camera_software.html

<https://www.pygame.org/docs/ref/joystick.html>

<https://sourceforge.net/p/raspberry-gpio-python/wiki/Home/>

<https://github.com/raspberrypi/libcamera-apps/issues/487>

<https://www.shellhacks.com/raspberry-pi-force-hdmi-hotplug/>

https://www.raspberrypi.com/documentation/computers/config_txt.html#video-options

<https://github.com/murtazahassan/Neural-Networks-Self-Driving-Car-Raspberry-Pi>

<https://www.thecodebuzz.com/python-windows-error-123-filename-directory-name-syntax-is-incorrect/>

<https://pimylifeup.com/raspberry-pi-low-voltage-warning/>

THANK YOU

