
Shri B.V.V. Sangha's

# BASAVESHWAR ENGINEERING COLLEGE (AUTONOMOUS), BAGALKOT-587102

**2018-2019**

## DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

**A PROJECT REPORT ON**

# "OPEN SOURCE BASED SELF DRIVING CAR"

PROJECT GUIDE

Dr. B G SHEEPARAMATTI

HOD

Dr. P N KULKARNI

PROJECT ASSOCIATES

| | USN |
|---|---|
| 1) Mr. RUTWIK R PATIL | 2BA15EC065 |
| 2) Mr. SAGAR PAWAR | 2BA15EC068 |
| 3) Mr. BASAVARAJ S S | 2BA16EC405 |
| 4) Mr. SANTOSH ROTTI | 2BA15EC075 |

# CONTENTS

# INTRODUCTION

- An autonomous vehicle is a system capable of sensing its environment and navigate without the need of human action.
- It includes various sensors like LIDAR, radar, camera inertial sensors and GPS.
- The car will have to take logical decisions for the real life based on its map of the zone and the obstacles it detects, while following the fastest of the possible paths that will lead it to its desired destination.
- Self-driving cars might become a cheaper and better, representing a shift on the way cities are planned.
- In a smart city with most of its vehicles being connected and autonomous, traffic optimization

# OBJECTIVES

- To build a self-driving car based on IoT and RF using entirely open source software (OpenCV, TensorFlow, Python) and hardware (Arduino) technology.
- To reduce the cost of system, use CNN (Convolution Neural Network) for learning and control and also provide virtual remote control of car by human if necessary.
- To use a single camera as input for navigation and obstacle avoidance.

# THEORITICAL BACKGROUND

Software Components

1. Python programming language
2. OpenCV
3. TensorFlow
4. NVIDIA CUDA
5. Convolution Neural Network(CNN)

# THEORITICAL BACKGROUND contd.

Additional Software Functions

1. pySerial
2. NumPy
3. Houghline Transform
4. Canny Edge detection
5. Gaussian Blur
6. TensorFlow Object detection API
7. AlexNet CNN

# THEORITICAL BACKGROUND contd.

Hardware Components

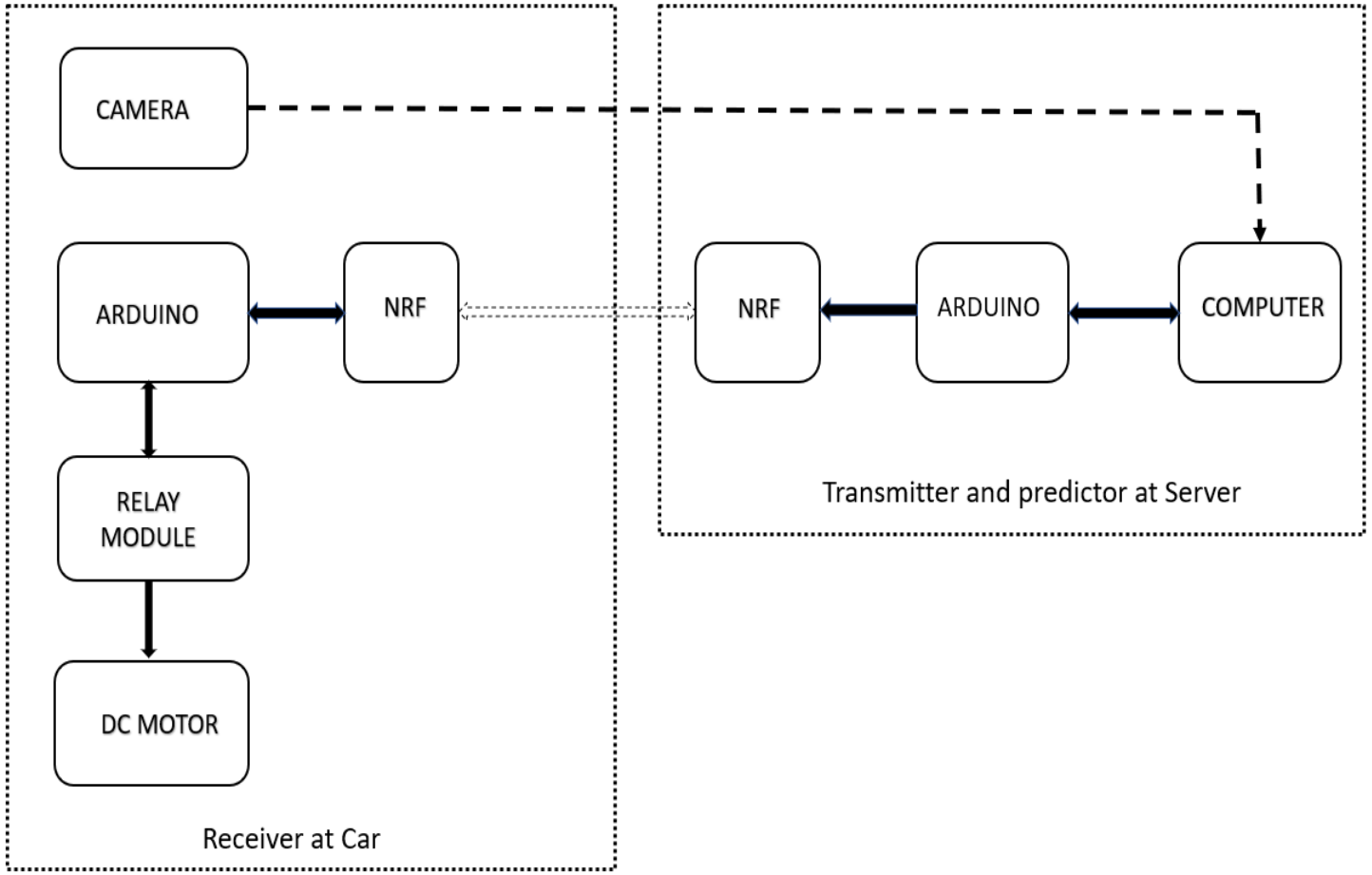
1. Arduino
2. NRF24L01
3. DC Motors
4. Relay Module
5. Server Computer

# PROPOSED WORK

The work is carried out in two steps

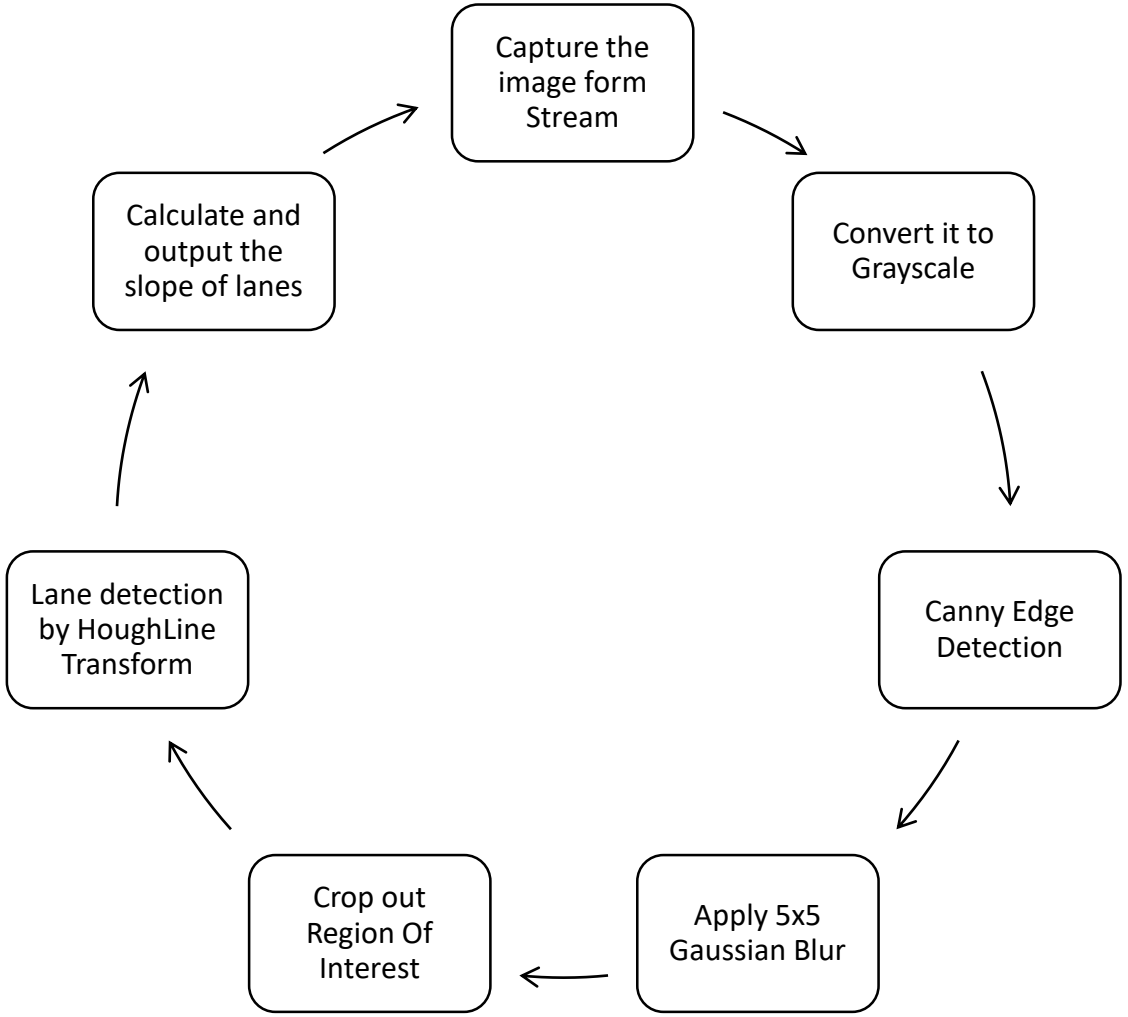
1. In the first part Houghline Transform is used to find the lanes and send steering commands accordingly
2. Second part discusses the use of a Convolution Neural Network (CNN) AlexNet to train a model and use it to make steering predictions.

# BLOCK DIAGRAM OF SETUP



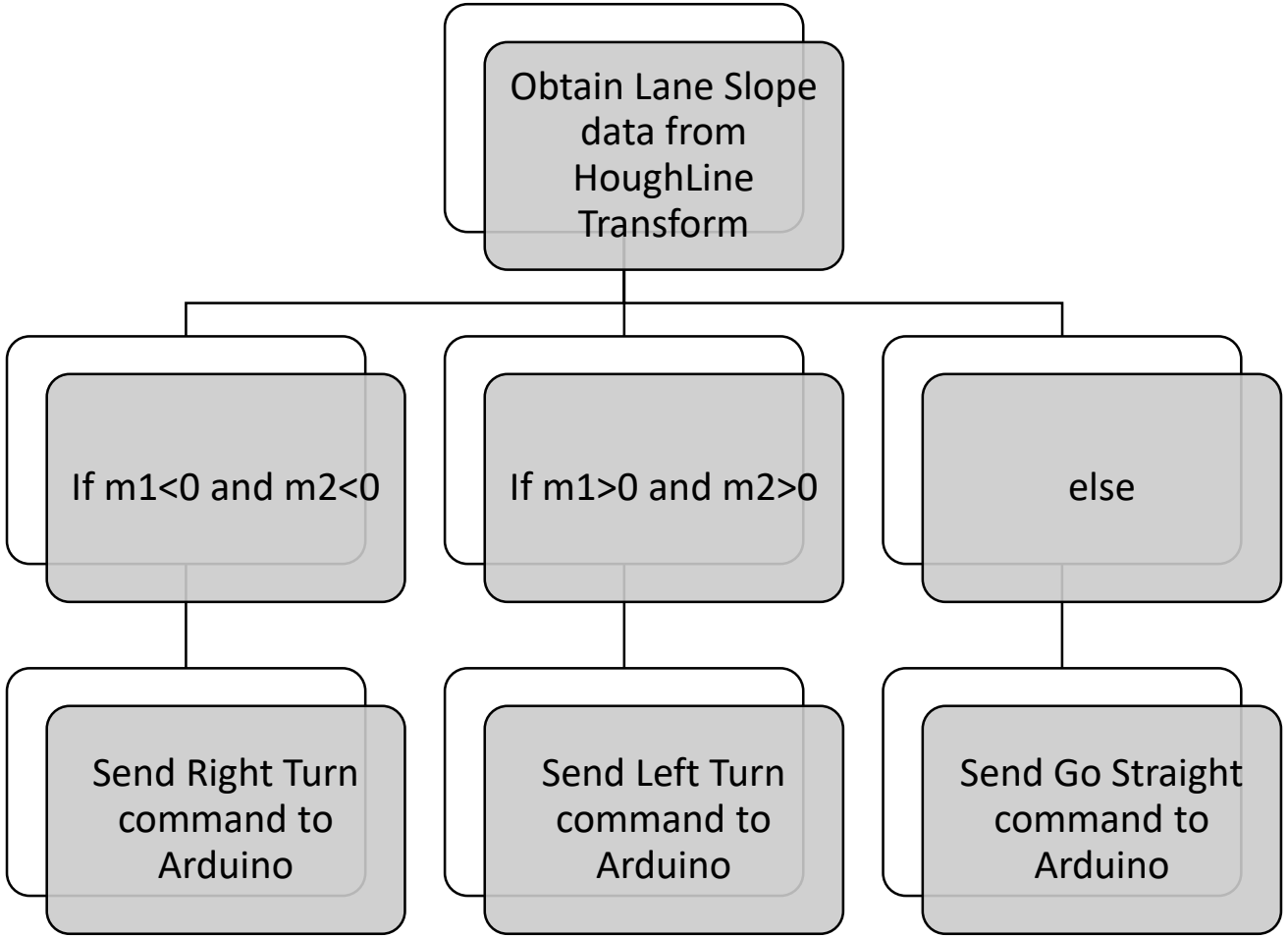
# Part 1 Lane Detection and Steering

# 1.1 Lane detection using Houghline Transform

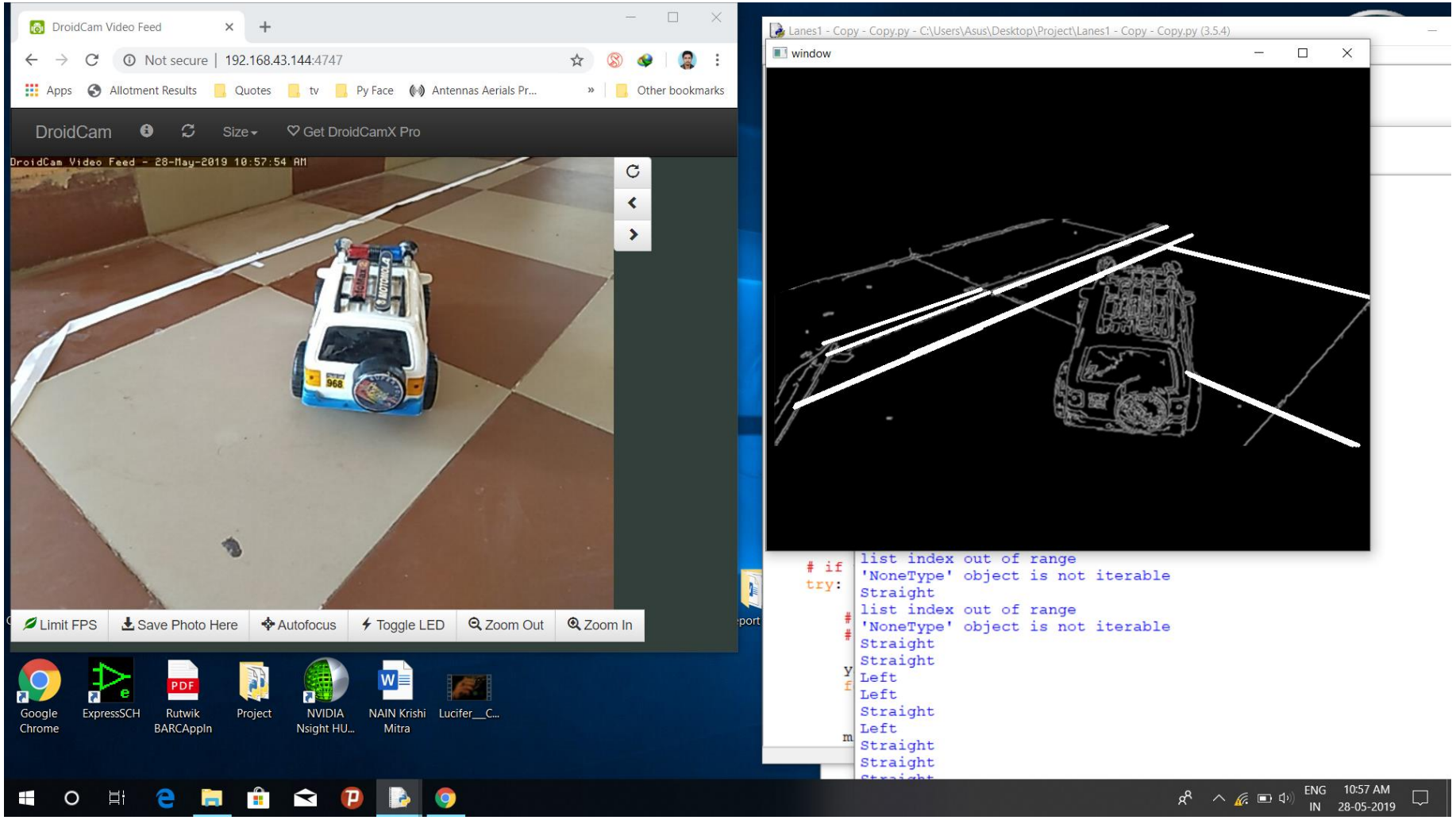



Flow Chart for Lane Detection Algorithm

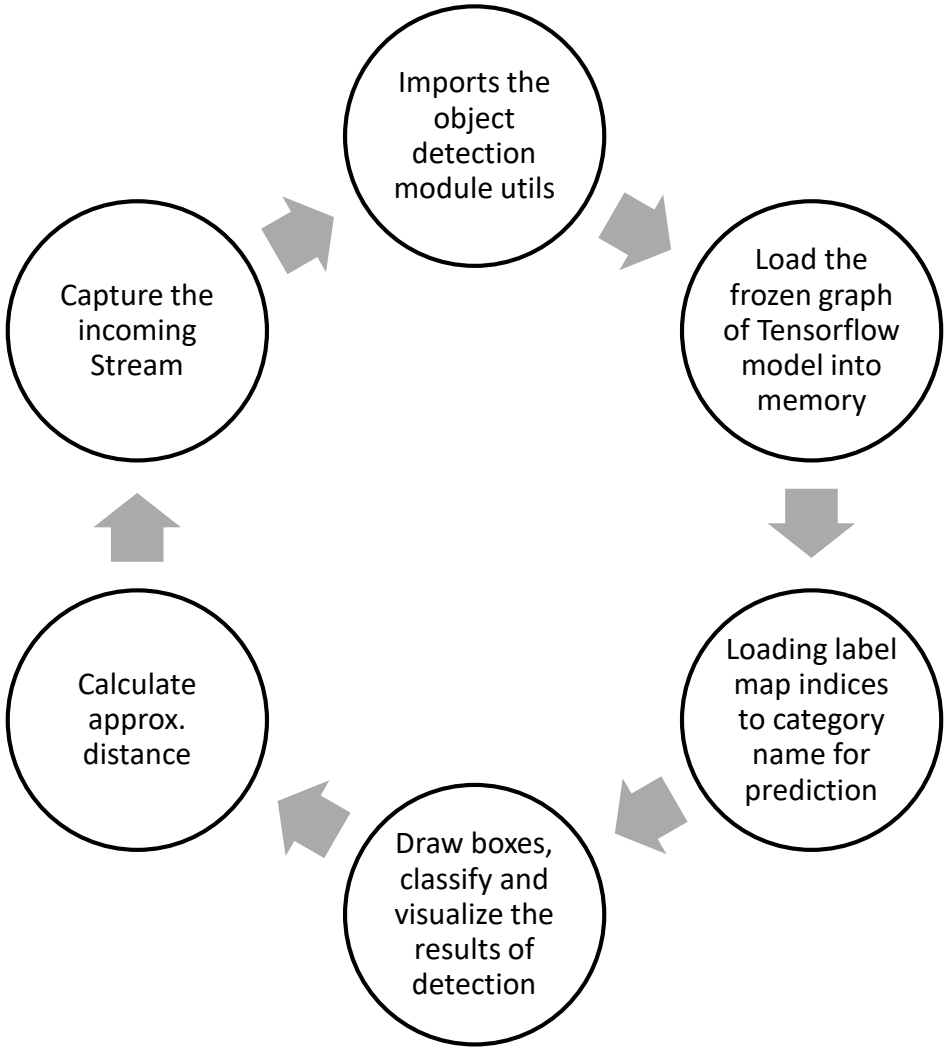
# 1.2 Send Drive Command to Arduino via Serial Interface



# Screenshot of Lane detection and sending drive command to Arduino

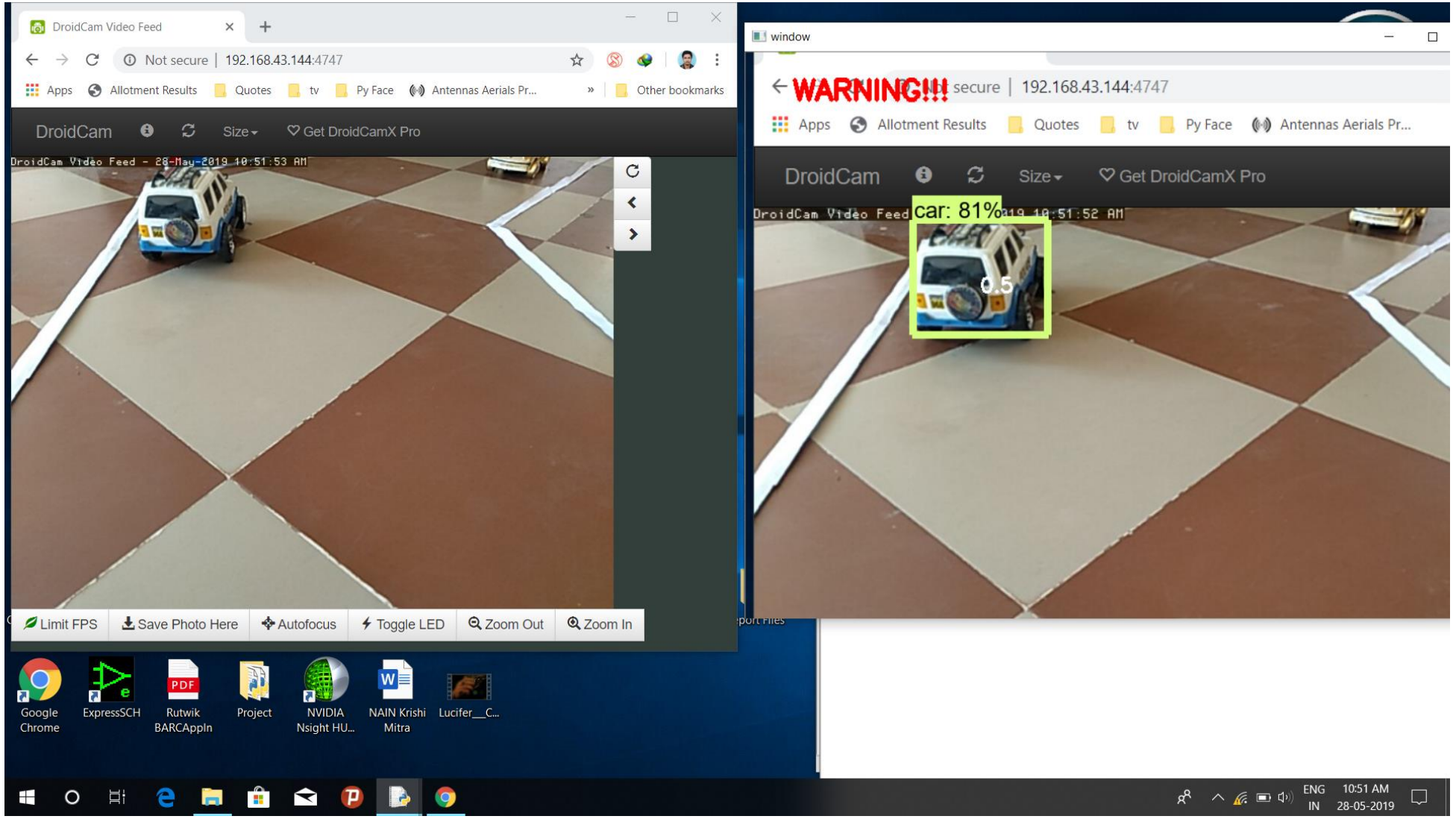


# 1.3 Implementation of the TensorFlow Object Detection



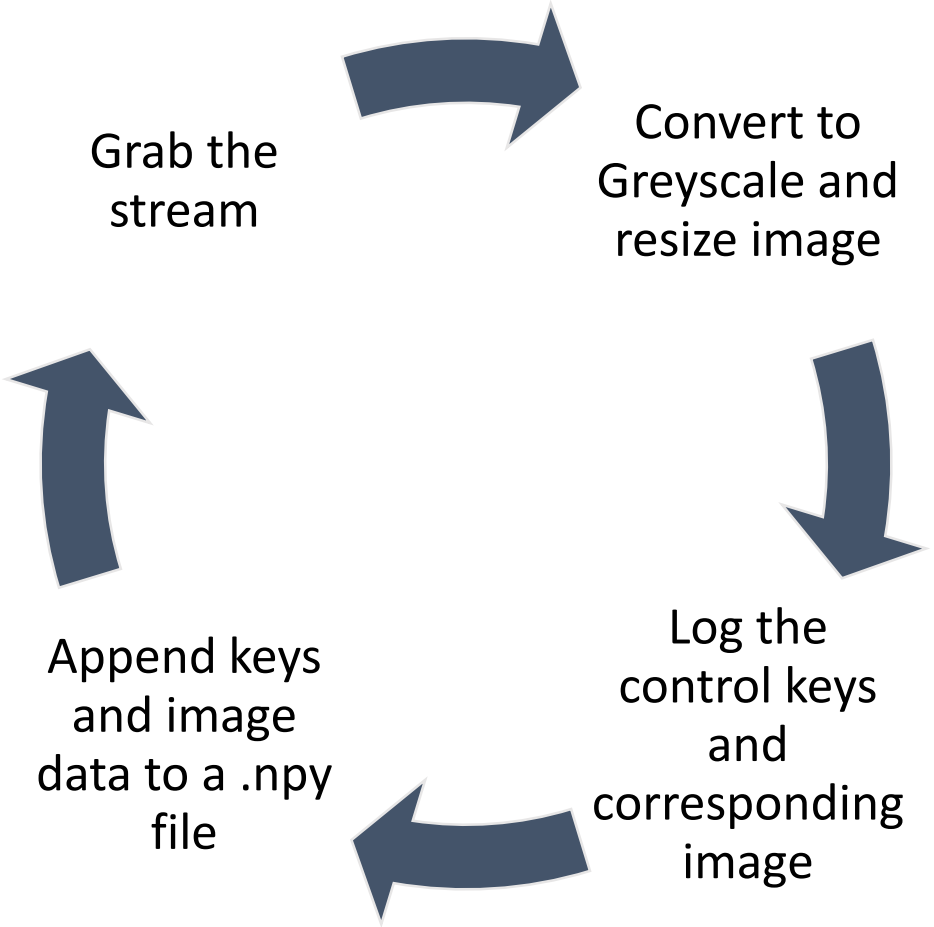

Object detection using Tensorflow on GPU

# Screenshot of Object detection in action

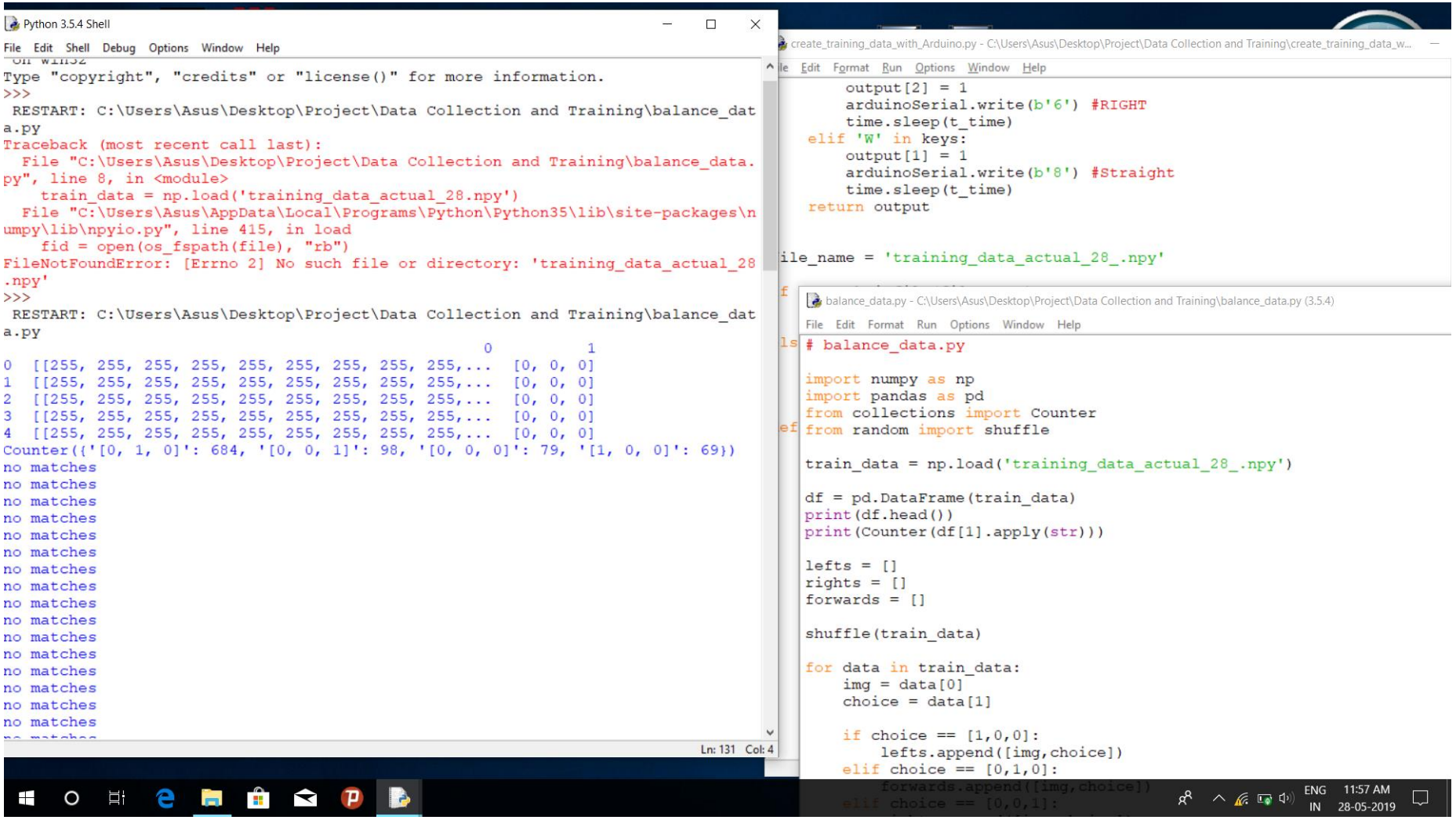


# Part 2 Training a Neural Network to make driving predictions

# 2.1 Algorithm for collecting the data

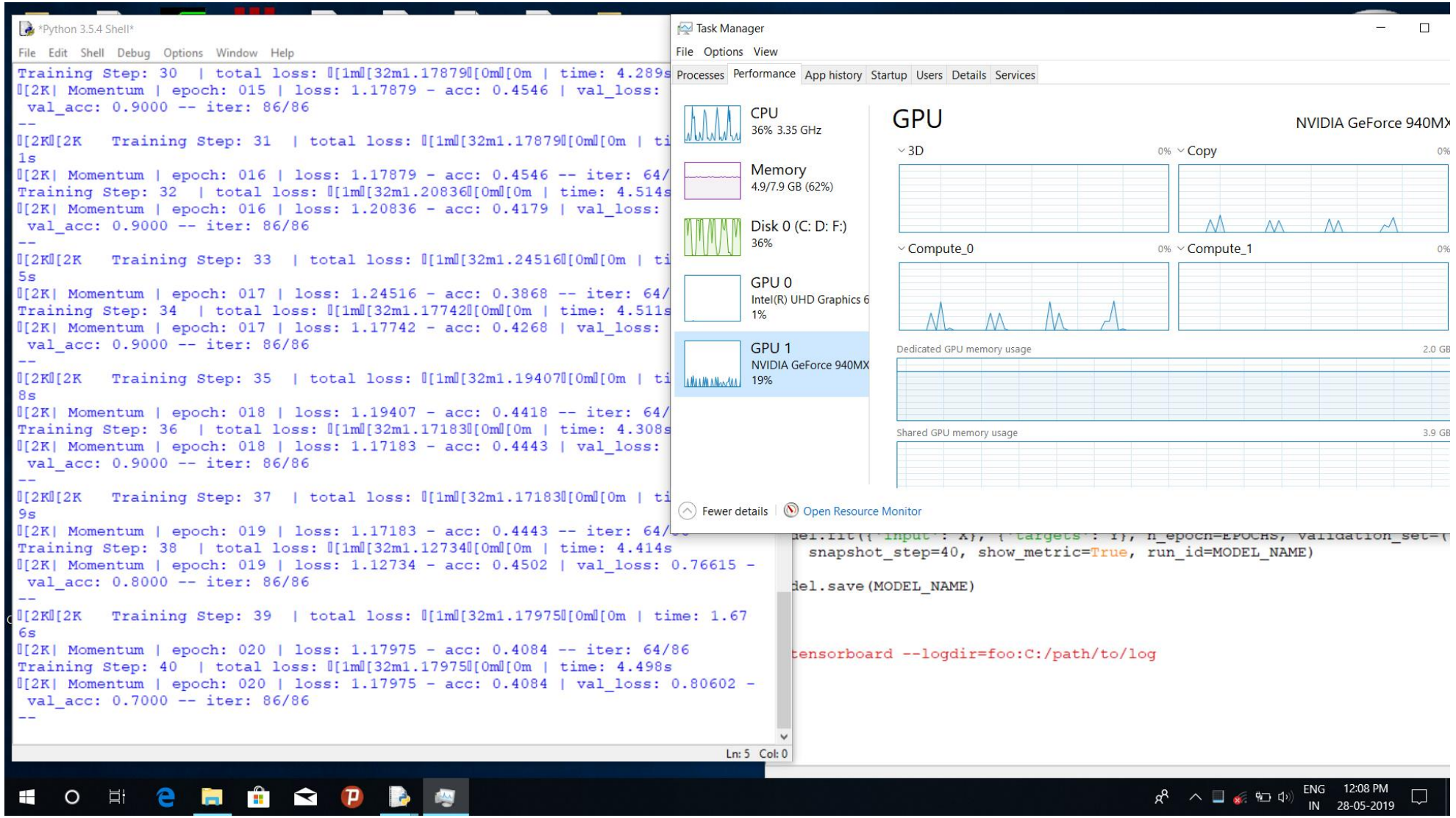


# Screenshot of acquired data and balancing it

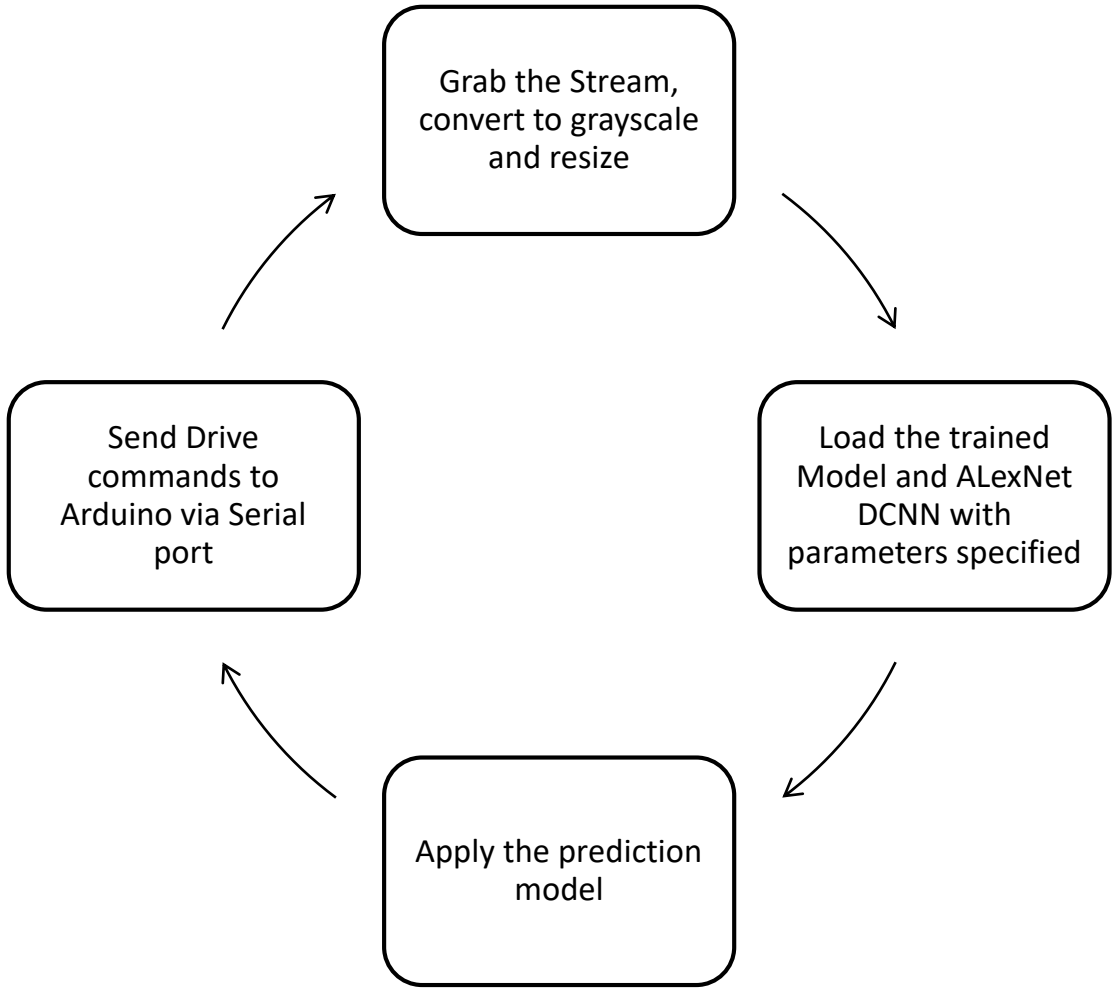


# 2.2 Training dataset using TensorFlow AlexNet on GPU

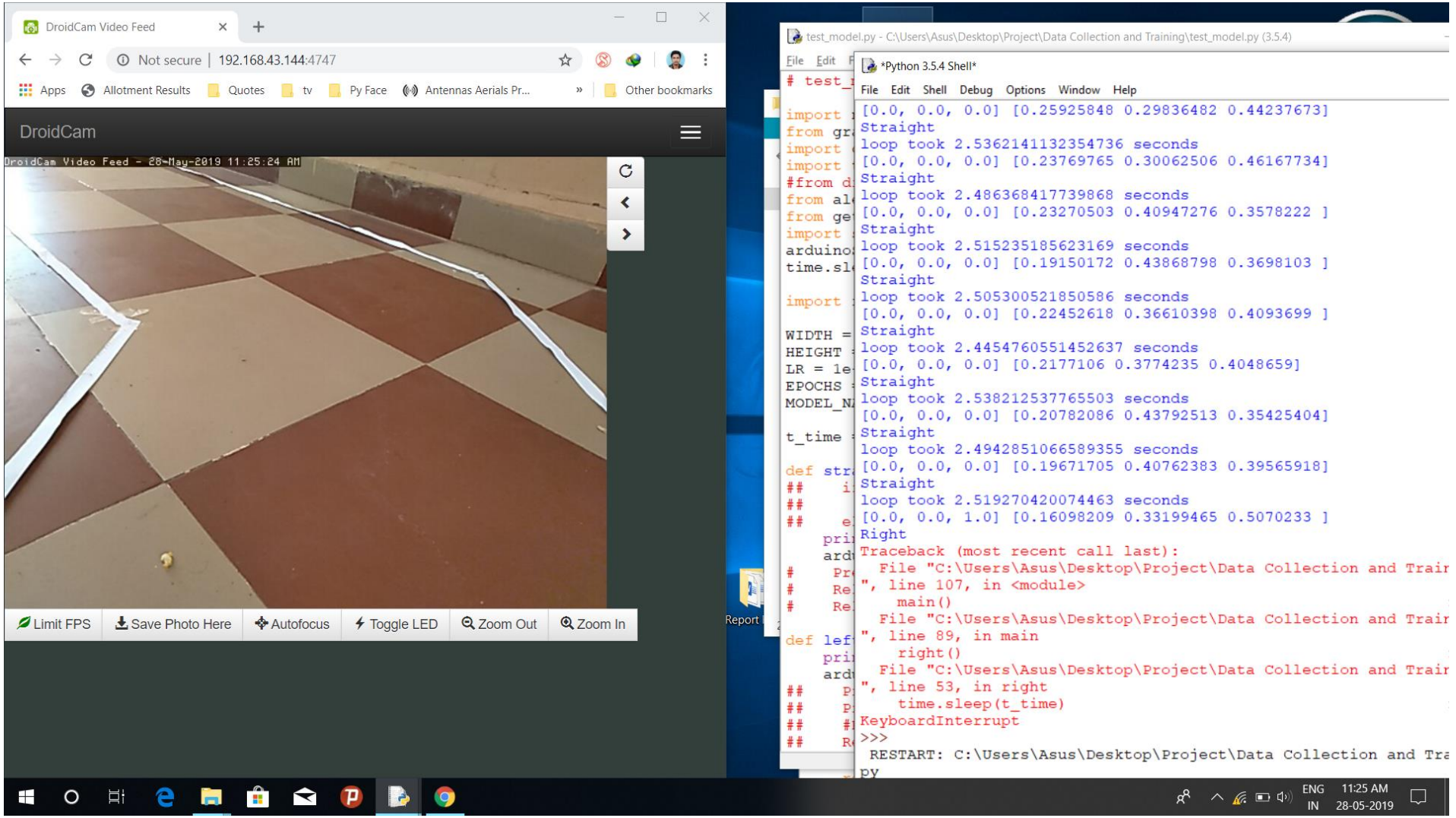
# Screenshot of the model in training



# 2.3 Flowchart for using trained model to drive the car



# Screenshot of car in action



# RESULTS AND CONCLUSIONS

- In this project, a self-driving car based on IoT and RF is successfully built. This uses entirely open source software (OpenCV, TensorFlow, Python) and hardware (Arduino) technology.
- By using open source technologies, cost of the system has been reduced.
- Successfully used the CNN (Convolution Neural Network) for learning and controlling of car is done and virtual remote control by human is provided if necessary.
- Single camera is used for both navigation and obstacle avoidance.