

Table of Contents

[Introduction 1](#_Toc67395705)

[Aim & Objectives 2](#_Toc67395706)

[Aim 2](#_Toc67395707)

[Objectives 2](#_Toc67395708)

[Justification 3](#_Toc67395709)

[Solution 4](#_Toc67395710)

[References 5](#_Toc67395711)

# Introduction

Vehicles have changed our life and way we live. Self-Driving vehicles are going to change the mobility and disturb automobile industry. Self-Driving vehicles uses different sensors, radars, cameras, artificial intelligence to sense the environment in order to run without any human participation. [(*Accelerating Autonomous Vehicle Technology* 2021)](#autoIntro) Autonomous vehicle can bring various advantages to society. It makes driving more easier and faster. Different sensors used in autonomous like radar, lidar and camera help to detect obstacles in its way and stop vehicle automatically in case of emergency. This makes autonomous vehicle lot safer. Autonomous vehicle reduces traffic cognitions like stop-and-go traffic. Autonomous vehicle enhance independence for physically disables people, old people and people who cannot drive. They can go wherever they want and live their life as they want with highly automated vehicle.



Figure : Autonomous Vehicles

Our team have developed driverless vehicle with obstacle avoidance system which can drive itself and improve driving experience. This system is built in real four-wheeler vehicle. Camera is used to detect obstacles. This vehicle ensure that it makes safe distance from the obstacles it detects. If the object is near and the vehicle detects it going to collide then it will stop automatically which makes it lot safer to drive. Artificial intelligence to move on the path it has already been through once is also implemented in this vehicle.

# Aim & Objectives

## Aim

Develop a driverless vehicle that can drive itself without human interference and can impersonate human task through machine learning.



Figure : Aims and Objective

## Objectives

* Driverless car that can be controlled through remote connection
* Detects obstacle by using camera
* Impersonate human task
* Detects traffic light, lanes, humans, animals, etc and take decision
* Discover its path to reach the destination and executes task through machine learning

Justification

Vehicle has become primary source of mobility. They should be comfortable, safer and be able to reach in destination at less amount of time. Driving normal vehicle, a person has to face lots of challenges. After a long hard day, everyone wants to reach home faster while relaxing in their vehicle. But driving normal vehicle, a person has to be always aware of the road and cannot take rest. A long ride is even hard to travel. Person cannot relax in the vehicle because they have to keep an eye on the road and always be aware of their surrounding which causes fatigue to drivers. 

Figure : Road accident

## Solution

Driverless vehicle with collision avoidance system is developed. This vehicle run automatically without any human in the steering. Obstacle detection and collision avoidance system is added in this vehicle. This system makes driving more easier, safer and faster.



Figure : Driverless Vehicle

This vehicle provide comfort, safer and easier to drivers. Sensor used in this vehicle detects any obstacle near the vehicle and if the obstacle is close to it, it will automatically stop. Driver can relax while driving this vehicle. Collision avoidance system make it easy, relaxing and safer to drive the vehicle. Artificial intelligence to recognise the path it has already been through once is also implemented in this vehicle. The technology improves the driving experience. Manual control system is also added in case to apply emergency break in case the system malfunction or does not detect obstacle.

# Tools and technologies

## Hardware

### Arduino Mega

Arduino mega is used to control the microservice of the car. The steering rotation trigger, throttle forward and reverse trigger, speed signal and reading of steering potentiometer is processed by Arduino mega in this project.

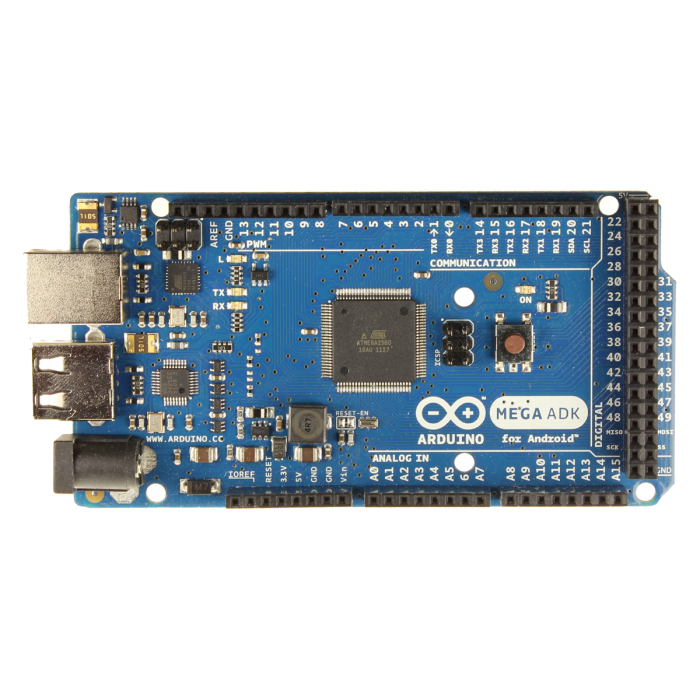


Figure 5:Arduino Mega

Specifications:

1. ATmega2560 microprocessor
2. 5v operating voltage
3. 54 I/O pins (15 PWM)
4. 256KB flash memory
5. 8kb SRAM
6. 16 MHz clock speed

### Single Channel Relay 5v (1pc)

1 single channel relay is used to trigger the throttle forward or backwards.



Figure :Arduino Relay

Specifications:

1. 3.75V-6V
2. Active current 70mA
3. Maximum contact voltage 250VAC
4. Maximum current 10A

### Dual Channel Relay 5v (1pc)

2 Dual channel relays are used to trigger another 12v relay to control the positive or negative current flow of the steering motors.



Figure :Dual channel Arduino Relay

Specifications:

1. Supply voltage 3.75V-6V
2. Trigger current 5mA
3. Current active (both) 140mA
4. Maximum contact voltage 250VAC
5. Maximum current 10A

### 12v 30Ah Two Channel Relay (2 pc)

2 12v relays are used to trigger the positive, negative or neutral current of the 24v motor used for steering.



Figure :12v 30Ah 2 channel Relay

Specifications:

1. Supply voltage 12V
2. Current 24A-40A

### 24v Motor (1pc)

1 24v motor is used to turn the steering attached to the steering wheel with bicycle chain.



Figure :24V 250 Watt DC Motor

Specifications:

1. Maximum supply voltage 24V
2. Speed 400 RPM
3. Wattage 250W (0.33 Horsepower)
4. Torque: 22
5. 9 Tooth Sprocket for 1/2" x 1/8" Pitch Chain

### 48v Rickshaw Motor (1pc)

1 48V rickshaw motor is used to rotate the differential of the car.



Figure :48V 1000Watt DC Motor

Specifications:

1. Maximum supply voltage 48V
2. Power 1000W
3. Speed 2000-6000 RPM
4. BLDC type

### 12v Battery (4pc)

4 12v battery is used to supply power for the differential motor of the car.



Figure :Acidic 12V Battery

Specifications:

1. 12v power supply
2. 80ah current capacity

### 12v Battery (1pc)

1 12v battery is placed independently for the motor control of the steering.



Figure :12V 18Ah Exide

Specifications:

1. EP 18-12 type
2. 12 voltage power supply
3. 18 Ah current capacity

### Potentiometer (1pc)

1 potentiometer is used to calculate the angle of the steering. It is attached to piston holder which rotates circularly.



Figure :Potentiometer

Specifications:

1. 500 Ohm~1M Ohm potentiometer potential
2. 1A at AC/DC 125V rated power
3. 300-degree rotation travel
4. S.P.S.T switch circuit

### Depth Camera D435 (2pc)

2 depth cameras are used on both front and back of the car to detect the objects and collision for up to 2m far.

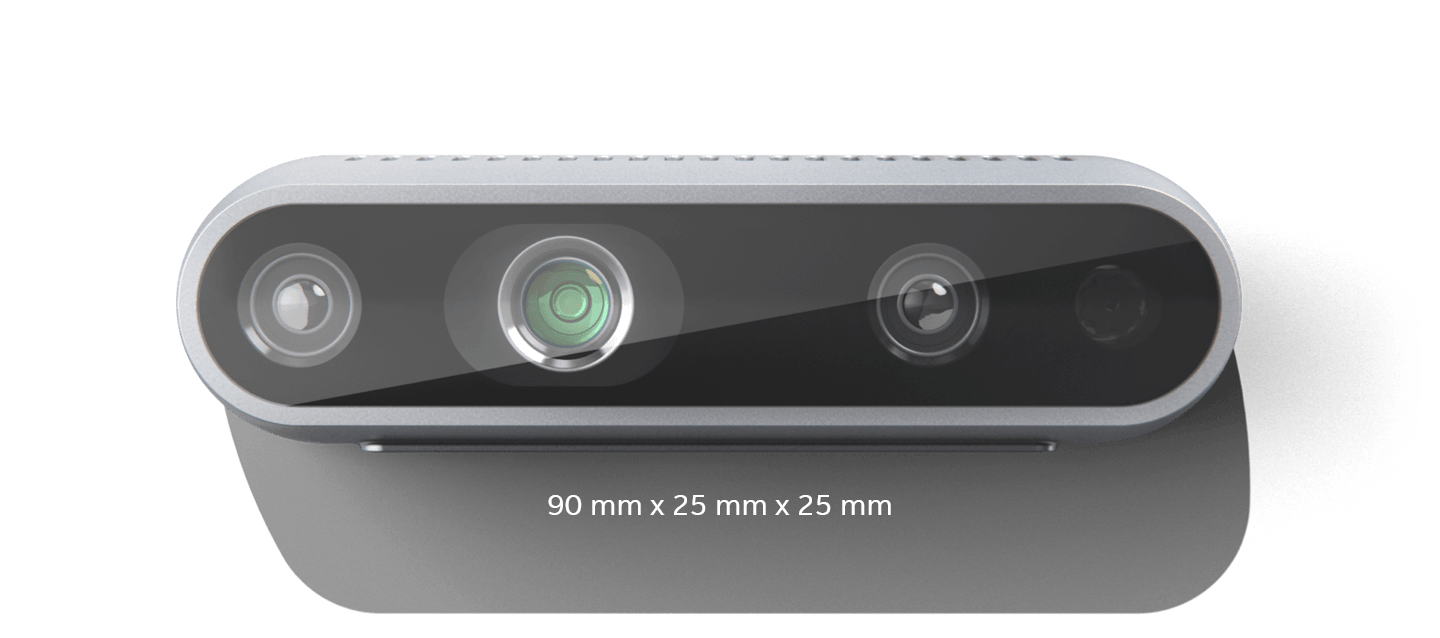


Figure :Depth Sensing Camera

Specification:

1. Global Shutter, 3μm × 3μm pixel size
2. Active IR Stereo
3. 86° × 57° (±3°) Depth FOV
4. 28 cm depth distance
5. 64° × 41° × 77° (± 3°) RGB FOV

### Logitech 1080p camera (1pc)

The Logitech camera is placed at the front of the car, below the bonnet of a car. This expands the viewing scope of the application and is used to detecting the cones and/or lanes in future versions.



Figure :USB HDMI Logitech camera

Specifications:

1. 8.0 MP still-image resolution
2. CMOS image sensor
3. Enabled auto focus

## Software

### Arduino IDE with C

Open-source Arduino IDE is used to push the sketch to Arduino mega. The program is used to program the Arduino device to control the microservice of the car is written in C.



Figure :Arduino Software IDE

### Python

Python is used as the main programming language to link image processing and microcontroller. The communication to Arduino is done with serial communication with pySerial library. The depth camera is processed with intel’s official source code using pyrealsense2.



Figure :Python Language

### Opencv2

OpenCV (Open-Source Computer Vision Library); an open-source computer vision and machine learning software library is used to handle the camera frames of the python software. It processes two depth cameras and one 1080p RGB camera simultaneously. The frames of camera feed are converted into array of colour pixel values, which then is used to apply logic by accessing and augmenting the pixel values of the frame. [(*About - Opencv* 2021)](#opencv)

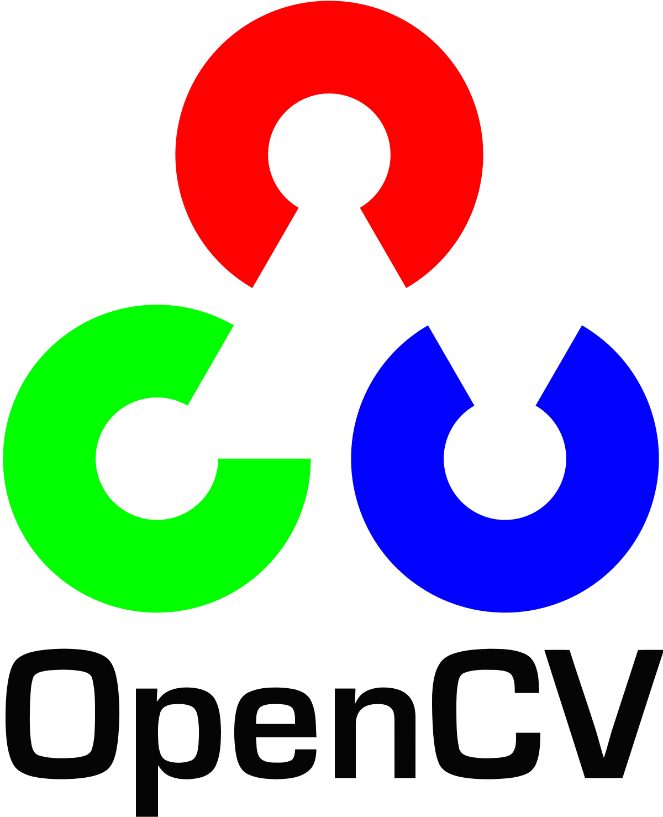


Figure :Python image processing library OpenCV

### TensorFlow

TensorFlow; an end-to-end platform to build and deploy machine learning models. Models are trained with input, multiple layers and output. In this project, it is used for object detection like cones, people, car, bikes, etc.

Figure :Python Object Detection Library TensorFlow

The layers of the model extract the feature of the input with weights and biases according to the features. The model then can predict the object based on the features given into new input. [(*Why Tensorflow* 2021)](#tensorflow)

## Algorithm

### Mobile net SSD

The Mobile Net network architecture is a special class of convolutional neural models that are built using depth-wise separable convolutions and are therefore more lightweight in terms of their parameter count and computational complexity. [(*Object Detection With SSD And Mobilenet* 2021)](#obj)

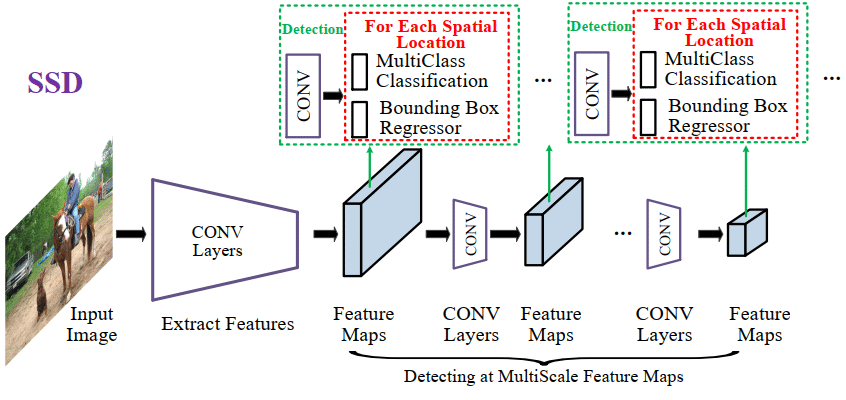


Figure :Mobile Net SSD Algorithm

Single Shot Detector (SSD) is a method for detecting objects in images using a single deep neural network. The SSD approach discretises the output space of bounding boxes into a set of default boxes over different aspect ratios. After discretising, the method scales per feature map location. The Single Shot Detector network combines predictions from multiple feature maps with different resolutions to naturally handle objects of various sizes. SSD excludes the proposal generation and subsequent pixel or feature resampling stages and encapsulates all computation in a single network. It is easy to train and as well as easy to integrate into systems that require a detection component using low resources. [(Choudhury et al. 2021)](#choudh)

The integration of Mobile Net into the SSD framework forms one of the core aspects of mobile and faster object detection platform.

### Custom path finding from cone

Once the cone detecting model is trained and loaded, the custom written algorithm from our team, looks for two cones at a time of process. The region of interest is 450 offsets for each side from the middle of the frame.

If the frame has orange cone inside the ROI, the steering commands for going right is triggered. The angle of steering and speed of the throttle is proportional to the distance of the bounding lines of ROI. And if the frame has green cone inside the ROI the actions are reversed. If 2 same-coloured cones are detected, the steering angle is either hard right or hard left.



Figure :Object detection camera view

If cones are out of the bounding box, the car goes on a path with previous steering angle and throttle.

### Depth-align

Depth aligning is a process of eliminating the camera feeds from the given distance while capturing the inner distanced feed. The frame is normal up to a given distance while static colour is defined after the distance. The process is further upgraded to classify 2 colour, black for outer bound and white for inner bound of the distance. If certain percentage of white colour are present in the frame, there is a collision in front of the camera.

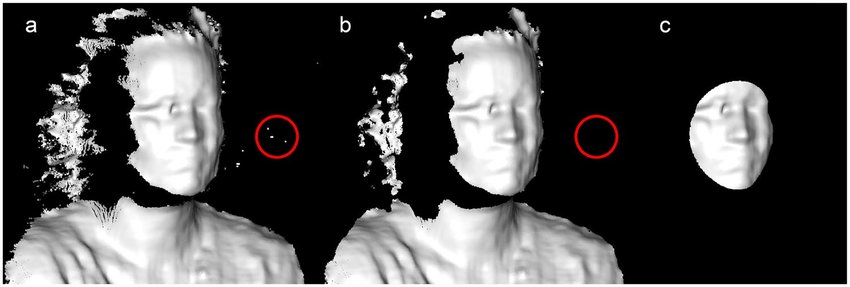


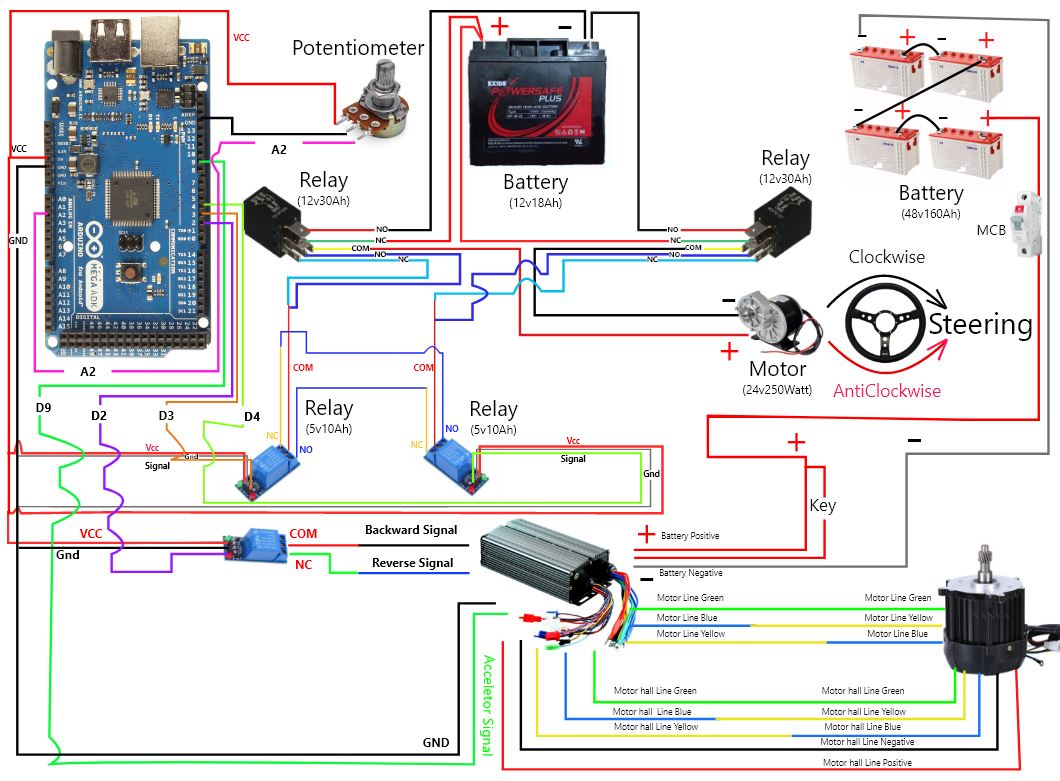
Figure :Depth align

In this project the minimum distance for collision detection is specified to 1.5m and the percentage of collision in white colour is 15%.

# The Build

## Circuit Diagram

RUN .PY SCRIPT



1000Watt

48V Motor

Figure :Circuit Diagram

* MCB switch is turned on to start the connection between motor controller, motor and Battery,
* USB connection is made between laptop and Arduino mega to perform serial communication.
* Python script is executed to test the vehicle.
* Steering movement is calibrated as per the potentiometer value to turn left and right.
* Motor forward and backward movement is calibrated by switching of relay.
* After calibration meets the standard another python script is executed to make it driverless car.

## Working Mechanism

## 

Figure :Flow chart showing the working mechanism of driverless car

MCB is turned on to give power to motor controller and motor. Python Calibration script is executed to calibrate and check the vehicle left, right, backward and forward movement. Arduino-Mega receives the signal from the script and passes to the motor controller and steering motor. Then car starts to move as per the command given from python script. After the calibration is completed another python script is executed where the car can be controlled remotely from anywhere and can mimic the task performed by human.

# Testing

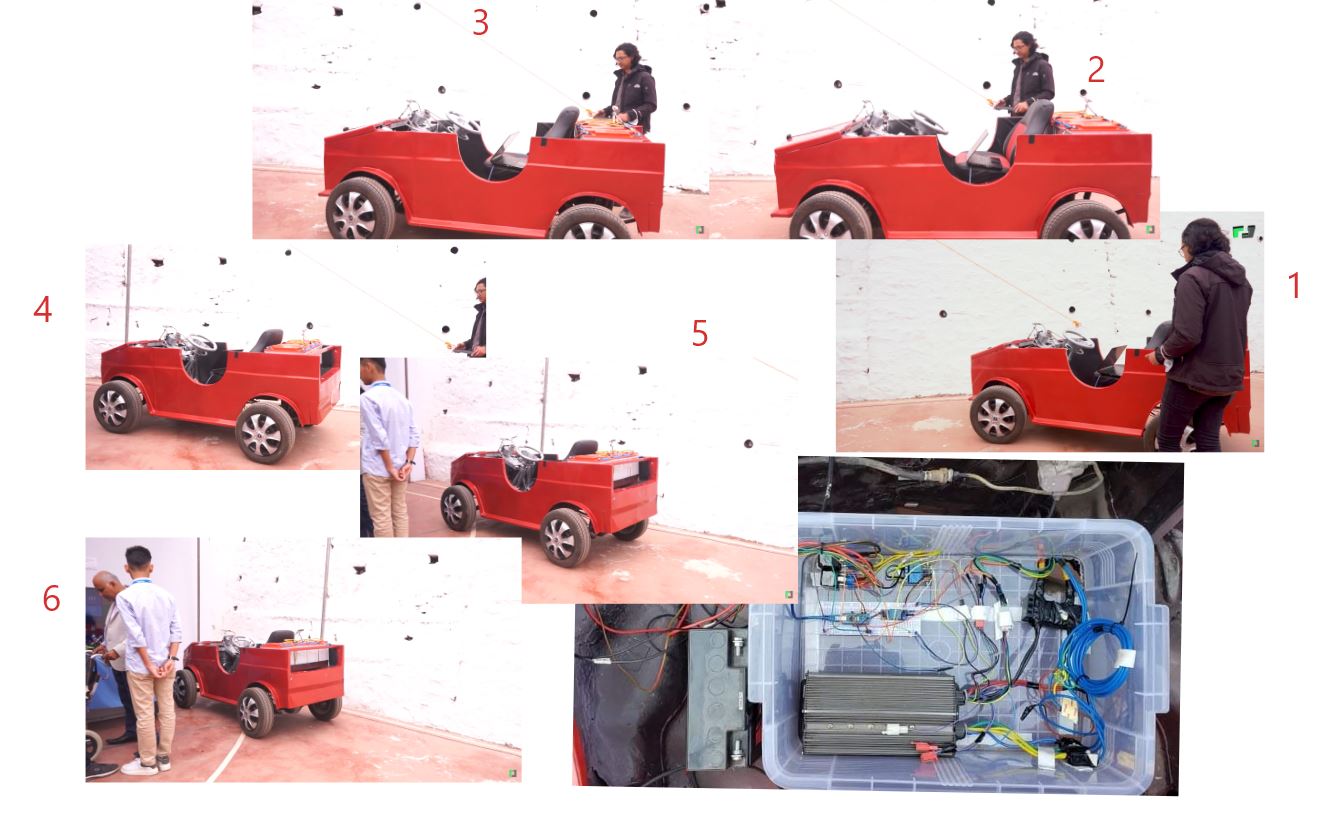


Figure :Testing of vehicle movement to the defined destination

The demonstration and explanation of the project is available on the link below: -

<https://www.youtube.com/watch?v=IBTrBeLULDw&t=1s>

### Solved issues after testing

* Noises were created due to the high ampere wires on same place. It was solved by exiting the wires from different places.
* 12V 10 Ampere dc motor was used for Steering movement but it failed so next 24V DC motor with 250Watt was used to make it work.
* For clockwise and anticlockwise movement 12V relay with 10 Ampere was used but due to the current flow fluctuation it raised up to 18Ampere so, another 12V 30Ampere was used to solve the issue.

### Issues that still remains

* Acceleration signal got fluctuation while transferring its signal value to motor controller so to smooth the signal LM398 was used to make it better but is not the perfect solution and has fluctuation till now.
* While executing python script it sometimes fails to read the value of acceleration

# Privacy and Ethical consideration

Many materials like past research paper, books, journals and project ideas were used to make the project successful. These materials were well acknowledged and cited. Huge amount of data was collected using different sensors. The collected data are used to operate driverless vehicle successfully. The project is for educational purpose so only required data were collected while maintaining data privacy and security.

# References

*Road Traffic Injuries* (2021) available from <https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries> [21 March 2021]

*Road Traffic Injuries And Deaths—A Global Problem* (2021) available from <https://www.cdc.gov/injury/features/global-road-safety/index.html#:~:text=Each%20year%2C%201.35%20million%20people,on%20roadways%20around%20the%20world.&text=Every%20day%2C%20almost%203%2C700%20people,bicycles%2C%20trucks%2C%20or%20pedestrians.> [22 March 2021]

*Accelerating Autonomous Vehicle Technology* (2021) available from <https://spectrum.ieee.org/transportation/self-driving/accelerating-autonomous-vehicle-technology> [22 March 2021]

*10 Advantages Of Autonomous Vehicles | Itsdigest* (2021) available from <https://www.itsdigest.com/10-advantages-autonomous-vehicles> [23 March 2021]

*About - Opencv* (2021) available from <https://opencv.org/about/> [25 March 2021]

Choudhury, A., Choudhury, A. and Choudhury, A. (2021) *Top 8 Algorithms For Object Detection One Must Know* [online] available from <https://analyticsindiamag.com/top-8-algorithms-for-object-detection/> [27 March 2021]

*Why Tensorflow* (2021) available from <https://www.tensorflow.org/about> [27 March 2021]

*Object Detection With SSD And Mobilenet* (2021) available from <https://medium.com/@aditya.kunar\_52859/object-detection-with-ssd-and-mobilenet-aeedc5917ad0> [28 March 2021]

*10 Advantages Of Autonomous Vehicles | Itsdigest* (2021) available from <https://www.itsdigest.com/10-advantages-autonomous-vehicles> [29 March 2021]