

# CGC ASSIGNMENT – 5

Name: RAHUL VARMA

Roll No: S20200010212

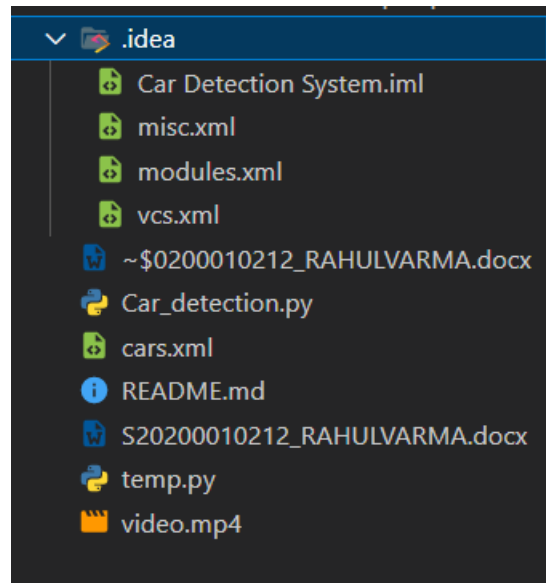
## Car Detection:

Car detection is a computer vision task that involves identifying cars or other vehicles in images or videos. It is an important application of object detection, which is a subfield of computer vision.

In car detection, a machine learning model is trained using labeled data to detect cars in images or videos. The model is typically trained using deep learning techniques such as convolutional neural networks.

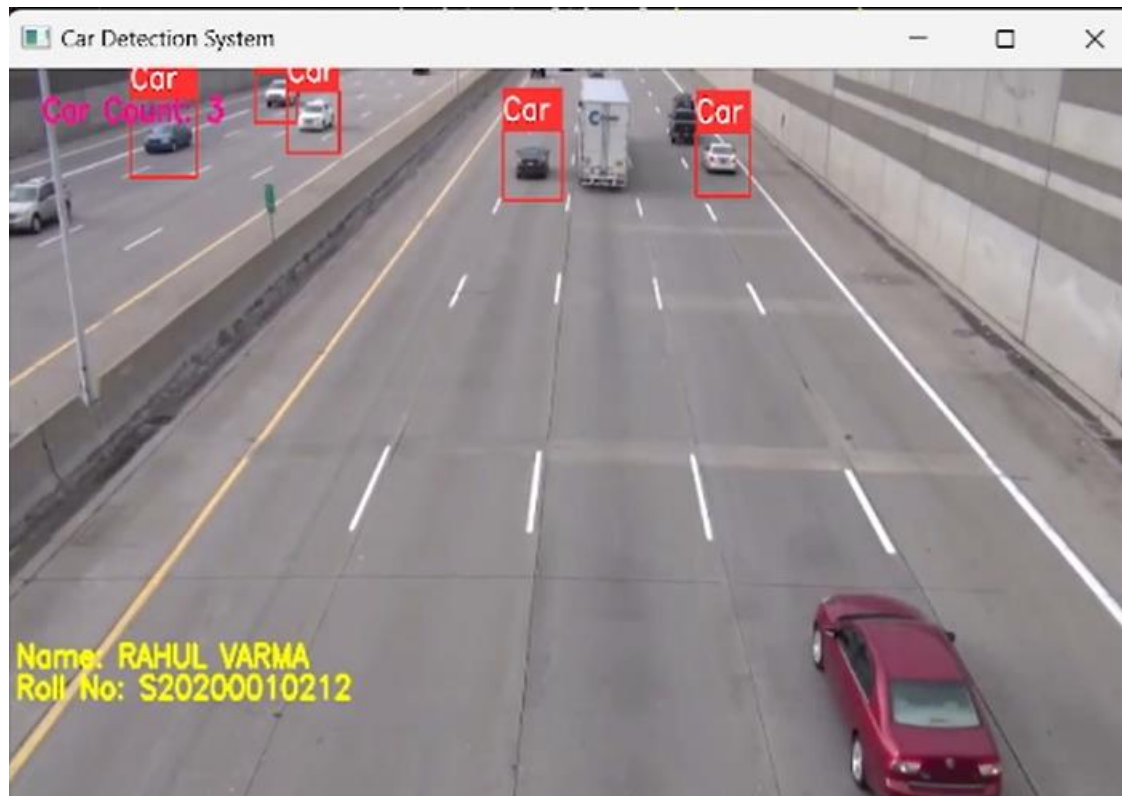
Once the model is trained, it can be used to detect cars in real-time video streams or images. Car detection has numerous applications, including traffic monitoring, surveillance, and self-driving cars.

## Folder Structure:

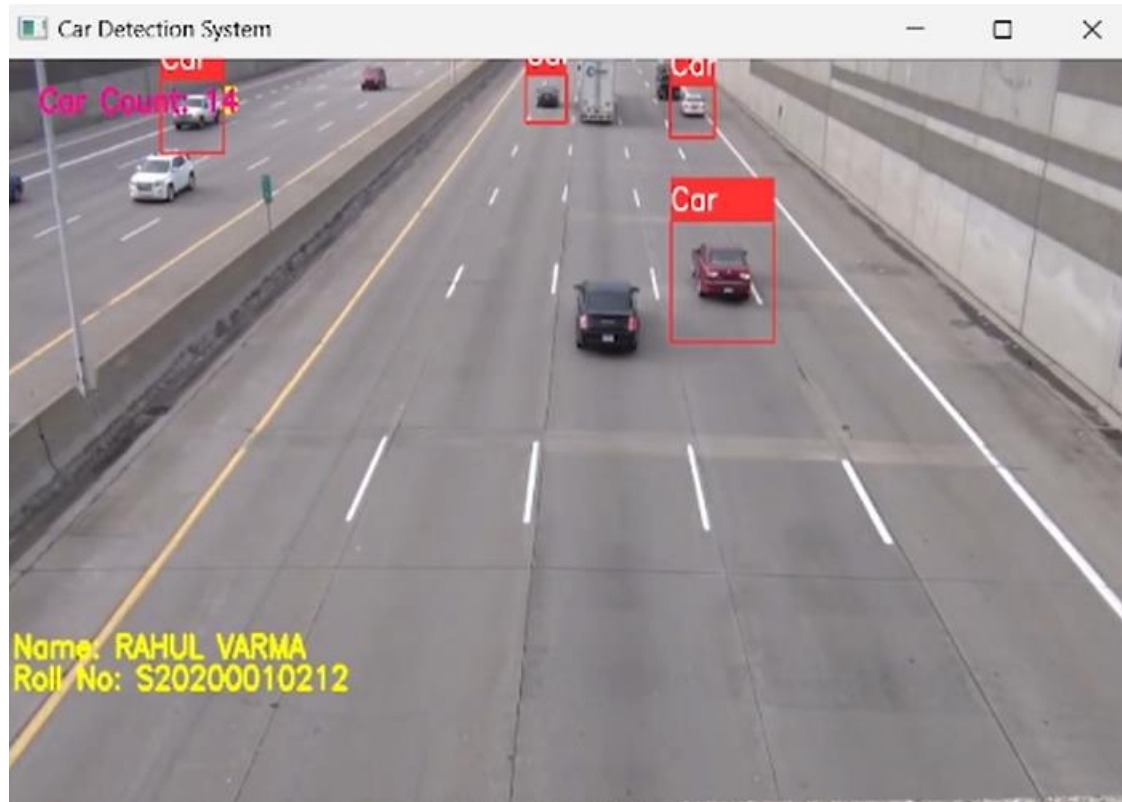


Output:

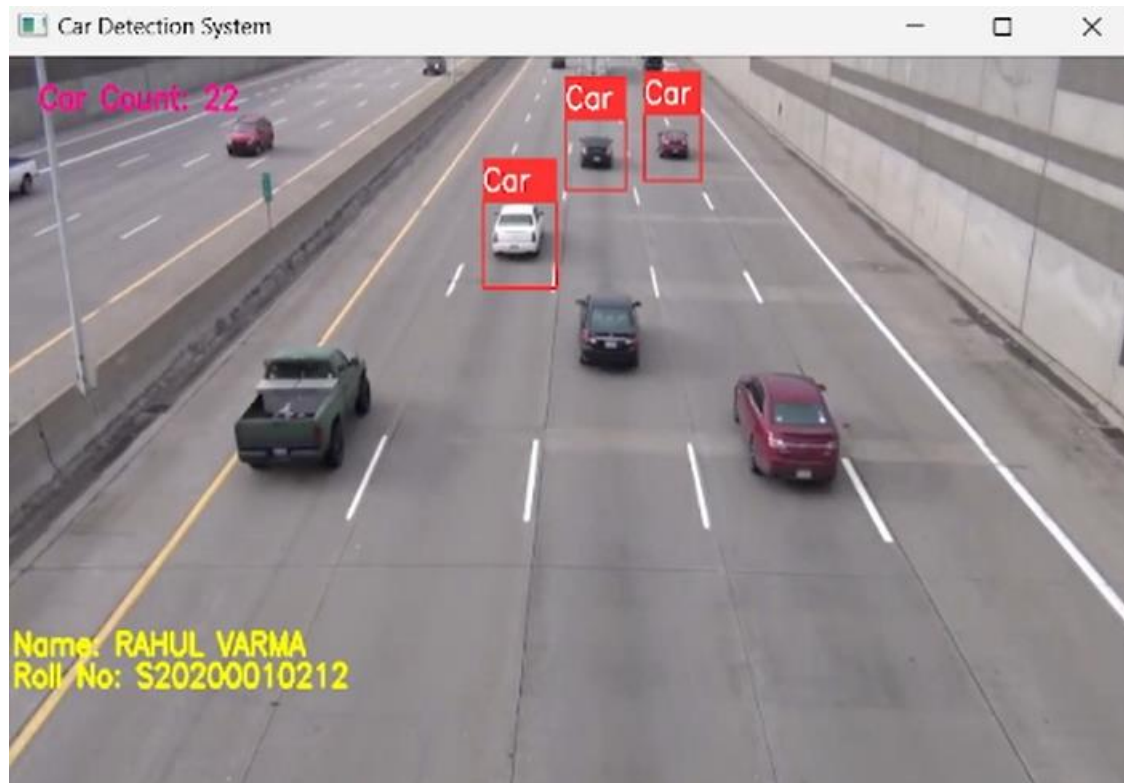
Car count = 3



Car Count = 14:



Car Count = 22



## Code:

```
1 import cv2
2 import numpy as np
3
4 cap = cv2.VideoCapture('video.mp4')
5 car_cascade = cv2.CascadeClassifier('cars.xml')
6
7 car_count = 0
8
9 while True:
10     ret, frames = cap.read()
11     gray = cv2.cvtColor(frames, cv2.COLOR_BGR2GRAY)
12     cars = car_cascade.detectMultiScale(gray, 1.1, 9)
13
14     for (x,y,w,h) in cars:
15         plate = frames[y:y + h, x:x + w]
16         cv2.rectangle(frames,(x,y),(x +w, y +h) ,(51 ,51,255),2)
17         cv2.rectangle(frames, (x, y - 40), (x + w, y), (51,51,255), -2)
18         cv2.putText(frames, 'Car', (x, y - 10), cv2.FONT_HERSHEY_SIMPLEX, 0.7, (255, 255, 255), 2)
19         cv2.imshow('car',plate)
20         car_count += 1
21         lab1 = "Car Count: " + str(car_count // 10)
22         cv2.putText(frames, lab1, (40, 50), cv2.FONT_HERSHEY_SIMPLEX, 1, (0, 255, 255), 3)
23
24         cv2.putText(frames, "Name: {}".format('RAHUL VARMA'), (10, 570), cv2.FONT_HERSHEY_SIMPLEX, 1, (0, 255, 255), 3)
25         cv2.putText(frames, "Roll No: {}".format('S20200010212'), (10, 600), cv2.FONT_HERSHEY_SIMPLEX, 1, (0, 255, 255), 3)
26
27         cv2.putText(frames, lab1, (40, 50), cv2.FONT_HERSHEY_SIMPLEX, 1, (147, 20, 255), 3)
28     frames = cv2.resize(frames,(600,400))
29     cv2.imshow('Car Detection System', frames)
30     if cv2.waitKey(25) & 0xFF == ord('q'):
31         break
32 cap.release()
33 cv2.destroyAllWindows()
```

```

import cv2
import numpy as np

cap = cv2.VideoCapture('video.mp4')
car_cascade = cv2.CascadeClassifier('cars.xml')

car_count = 0

while True:
    ret, frames = cap.read()
    gray = cv2.cvtColor(frames, cv2.COLOR_BGR2GRAY)
    cars = car_cascade.detectMultiScale(gray, 1.1, 9)

    for (x,y,w,h) in cars:
        plate = frames[y:y + h, x:x + w]
        cv2.rectangle(frames,(x,y),(x +w, y +h) ,(51 ,51,255),2)
        cv2.rectangle(frames, (x, y - 40), (x + w, y), (51,51,255), -2)
        cv2.putText(frames, 'Car', (x, y - 10), cv2.FONT_HERSHEY_SIMPLEX, 0.7, (255, 255, 255), 2)
        cv2.imshow('car',plate)
        car_count += 1
        lab1 = "Car Count: " + str(car_count // 10)
        cv2.putText(frames, lab1, (40, 50), cv2.FONT_HERSHEY_SIMPLEX, 1, (0, 255, 255), 3)

        cv2.putText(frames, "Name: {}".format('RAHUL VARMA'), (10, 570), cv2.FONT_HERSHEY_SIMPLEX, 1, (0, 255, 255), 3)
        cv2.putText(frames, "Roll No: {}".format('S20200010212'), (10, 600), cv2.FONT_HERSHEY_SIMPLEX, 1, (0, 255, 255), 3)

        cv2.putText(frames, lab1, (40, 50), cv2.FONT_HERSHEY_SIMPLEX, 1, (147, 20, 255), 3)
    frames = cv2.resize(frames,(600,400))

```

```
cv2.imshow('Car Detection System', frames)
if cv2.waitKey(25) & 0xFF == ord('q'):
    break
cap.release()
cv2.destroyAllWindows()
```

## Detailed Explanation:

- Python code that uses OpenCV library to detect cars in a video file. The code reads a video file using the `cv2.VideoCapture()` method and initializes a cascade classifier to detect cars using `cv2.CascadeClassifier()` method.
- It then enters into an infinite loop where it reads the video frames using `cap.read()` method and converts the color of the frame to grayscale using `cv2.cvtColor()` method. The cascade classifier then detects cars in the grayscale frame using `car_cascade.detectMultiScale()` method.
- If the classifier detects a car, the code draws a rectangle around the car using `cv2.rectangle()` method and puts a text 'Car' on top of it using `cv2.putText()` method. It also increases the `car_count` variable by 1 for each detected car.



- Finally, the code displays the original frame with car detection bounding boxes and car count text using `cv2.imshow()` method. It waits for 25 milliseconds for a keyboard event to occur using `cv2.waitKey()` method. If the key pressed is 'q', it breaks out of the infinite loop and exits the program using `cap.release()` and `cv2.destroyAllWindows()` methods.