# PROJECT – VEHICLE DETECTION

## ENGI 9805 – Advance Topic in Computer Engineering: Computer Vision

Submitted by: Group-12

Disha Bodiwala (201991118)

Rajat Acharya (201990852)

Viren Sagapariya (201991237)

Submitted To:

Ebrahim Karami

Memorial University of Newfoundland

Submitted on 7th April 2020

# TABLE OF CONTENTS

## LIST OF TABLE AND FIGURES

# 1. INTRODUCTION

## 1.1 Introduce the problem

We are dwelling in the era of globalization, where the world has become smaller and faster in unprecedented way. Due to the rapid advancements in science and technology, we can reach to any destination in just a few hours. Since the last decade, we have come to that scenario where every individual owns their own transportation mode to commute between different places. Such travelling soothe has come along with many consequences like traffic congestion, road accidents, parking issues, pedestrian safety and many more. These issues propel us to establish an intelligent traffic management system [1].

## 1.2 Motivation to solve the problem

In order to dwindle the problems created by transportation, a demand for an insightful road management system is originated. For this application, we need basic functionality like vehicle detection and objects identification. Moreover, the vehicle detection application plays a crucial role in civilian purposes like surveillance control and urban traffic planning. These mottos break down to vehicle tracking for investigation purpose, vehicle count for smart traffic signal software, vehicle classification, trace vehicles' speed, vehicle categorizing motive, etc.

## 1.3 Brief discussion about the solution

For last several years, the researchers have scrutinized the work in the Vision-based Intelligent transportation system (ITS) and traffic engineering applications to obtain precise information for traffic analysis and vehicle flow management. Basically, video detection is based on real-time image processing which is efficient for recognizing incidents on roads and tunnels. This system can also be used for vehicle detection after certain modifications at signalized crossroads. This module combines real-time image processing and computerized algorithms for pattern recognition. Furthermore, it uses a vision processor to observe real-time changes in the image [2].

Camera, as video capturing tool, captures video of targeting region and provides video clips as input to vision processor. Video detection is more adaptable to changing conditions which is one of the biggest advantages of image detection. This aids to reduce congestion and improve roadway planning, finding freeways which is very essential to improve emergency response time for hospital staff and police authorities.

The main algorithm here used is cascade classifier class to detect objects in video stream with 'the training' followed by 'Haar cascade' method. It is a machine learning based approach which uses a train of positive and negative images. These images are used to obtain features from it. Each feature is a single value obtained by deducting sum of pixels under the black rectangle. All the possible sizes and locations of each kernel are used to calculate features.

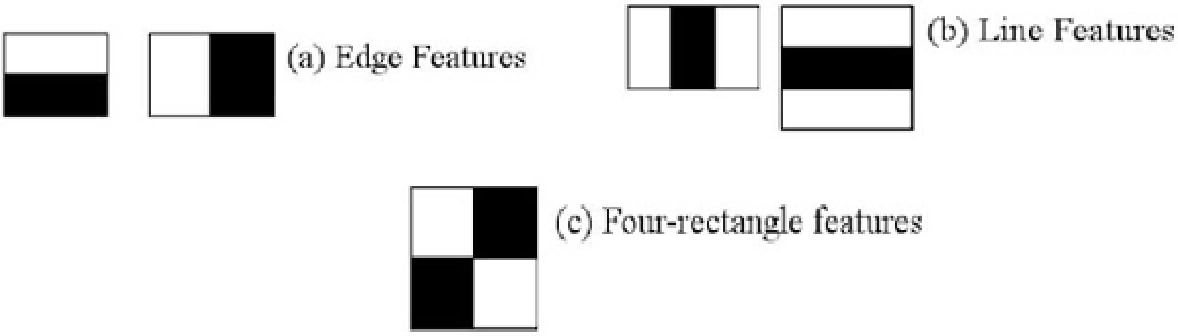

*Figure 1 Positive and negative frames* [3]

After calculating features, frames are collected, and the cars are distinguished by putting rectangle frame on it.

# 2. PROBLEM DEFINITION

With the change of the world into a new era with technology has prompted some increase in amount of vehicles, which results directly in traffic congestion and traffic violations as well as some criminal activities related with traffic, has alarmed a high demand of traffic management in urban areas. Road safety is a major concern nowadays. Lately the road traffic injury is one of the leading causes of deaths, disabilities and hospitalization, with severe socioeconomic prices, across the globe.

Transportation system has become intelligent with time, it uses computer vision technology for refinement in detection and tracking of vehicle for analysis. For some years, vehicle detection is one of the most demanding topics and it plays a crucial role in applications such as defense system as well as handling of management of traffic. Because of the nature of the natural scene setting, moving vehicles can be quickly identified by standard moving vehicle tracking system, by utilizing system discrepancies or context modelling from aerial videos [4].

1)Appearance based and motion based
2) Haar-edge time and center-oriented features

Haar like features are formed using sum and differences of rectangles over an image portion to highlight 'gray level' distribution of adjacent regions. Two to four filters can be used at any point and a scale to extract the features. The output is computed by adding the pixel values for grey and white regions and the difference between two sums are normalized [5].

# 3. PROPOSED SOLUTION

## 3.1 Benefits from the proposed solution

In this section, we will discuss about how we get the solution on how we can detect the vehicle from the traffic or from any area. The reason behind this detection is that we can prevent traffic violation. Also, it will be helpful to solve some road accidents as well as to lessen fatal accidents.

Expanding clog on expressways and issues related with existing finders have brought forth an enthusiasm for new vehicle recognition advances. Existing business picture handling frameworks function admirably in free-streaming traffic, however the frameworks experience issues with clog, shadows and lighting advances [6].

## 3.2 Key feature

These issues originate from vehicles mostly impeding each other and the way that vehicles show up contrastingly under different lighting conditions. We are building up an element based following framework for recognizing vehicles under these difficult conditions. Rather than following the whole vehicle, vehicle highlights are followed to make the framework powerful to fractional impediment. The framework is completely useful under changing lighting conditions due to the fact that the most notable highlights at the given minute are followed.

After the highlights leave the following locale, they are gathered into discrete vehicles utilizing a typical movement limitation. The gatherings speak to singular vehicle directions which can be utilized to quantify customary traffic parameters just as new measurements reasonable for improved robotized reconnaissance.

## 3.3 Employed Technology

Integrated Development Environment: In this project we have used Python 3.6.10 as platform and a proficient IDE for python language, PyCharm 2019.03 Community Version.
Library: In order to make work easier, we need to import certain predefined libraries, using which the predefined functions can ease the task besides save the time. We have used cv2, python3.6, NumPy, cmake and other helping libraries [7].

In this project, we have used Haar classifier which has been initially proposed by Pual Viola and was refined by Rainer lien hart. The classifier is trained with a few samples of a particular object. Two types of samples are captured, positive and negative. Positive are required ones and negatives are arbitrary ones. Identified samples as positive are scaled on same size to perform operation. Once the classifier is trained, it is applied to the area of interest in an image.

Current algorithm of classifier uses following Haar-like features:

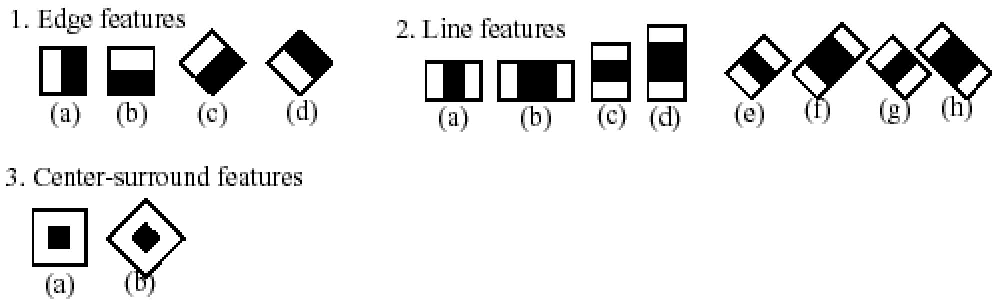

*Figure 2 edge line and centroid features [7]*

## 3.4 Block Diagram

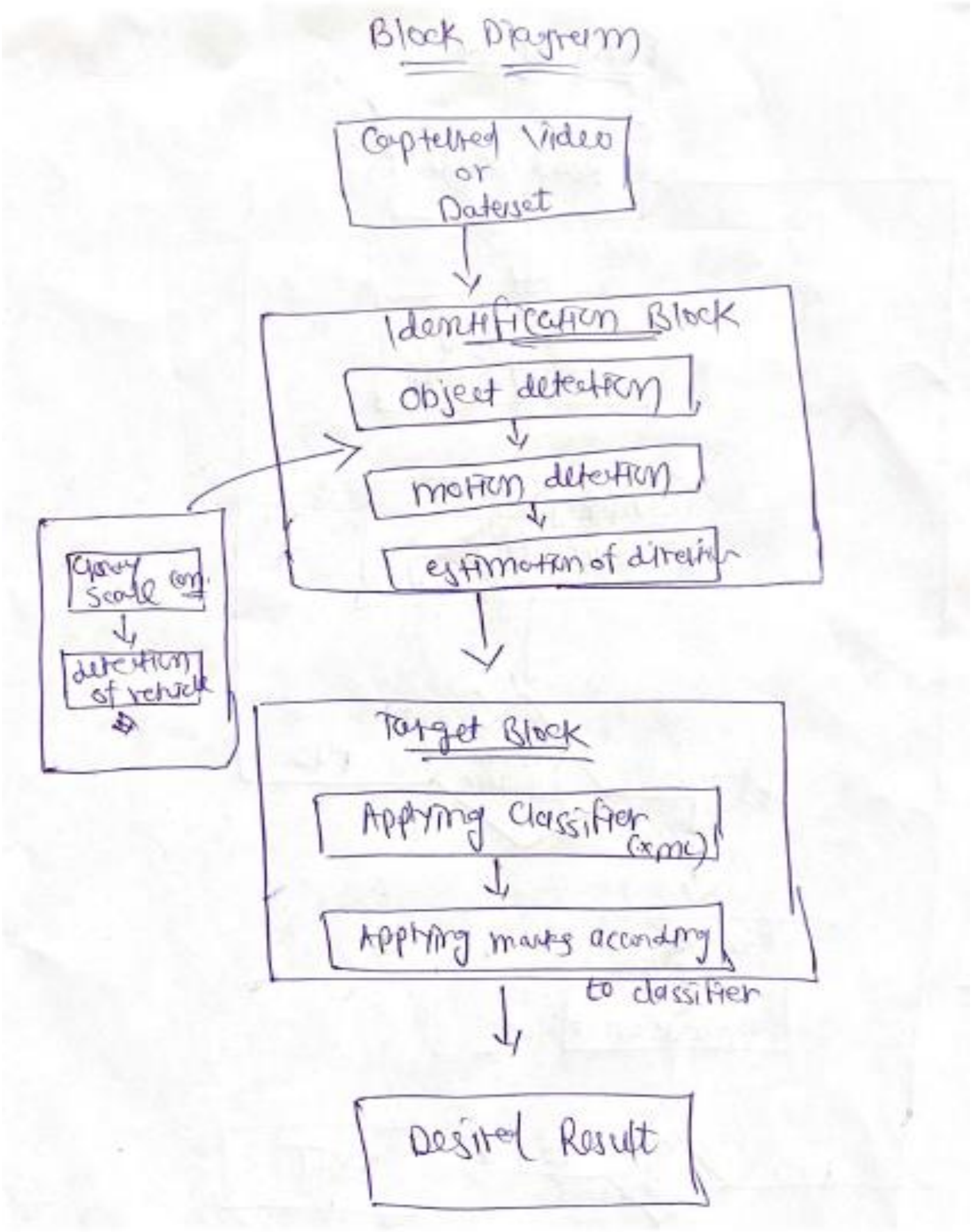

*Figure 3 Block Diagram*

## 3.5 Flow Chart of the method used

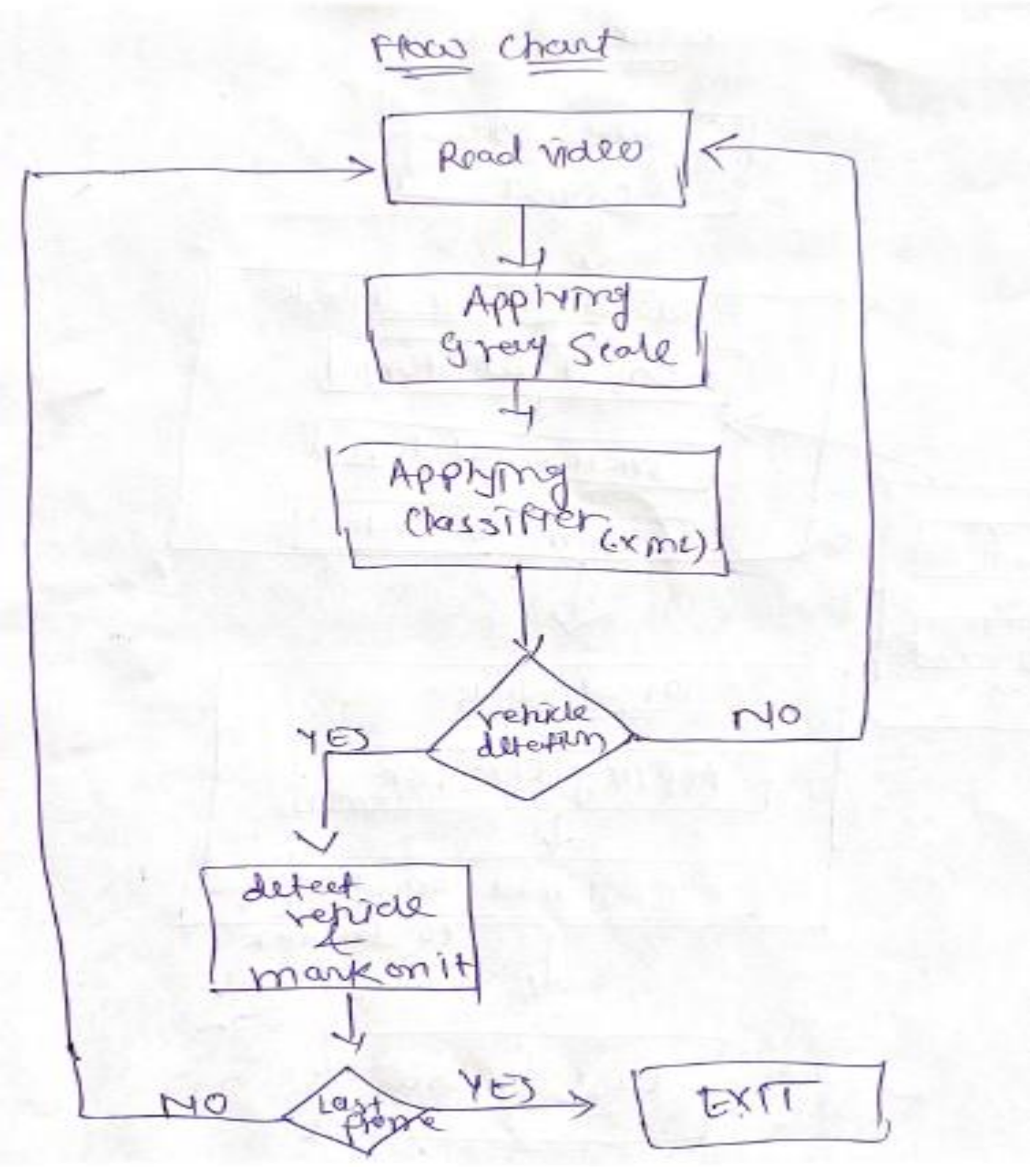

*Figure 4 Flow chart*

## 3.6 Use of the database

As per the solution given above in flow chart and block diagram, we can use datasets or any recorded videos for sample video. We have taken 6 to 7 videos captured at different locations with various angle to check the efficiency of the code and algorithm used.

In identification block there are objection detection, notion detection and estimation of direction. In this part we are focusing on gray scale conversion, and detection of vehicle

The next block is about target block. In this part we are applying the cascade classifier on the object and then trying to detect them and make a mark on them (rectangle). That will be our final output.

## 3.7 Steps followed for algorithm

The breakdown for vehicle detection [8]:

As described earlier, we need to import library cv2 module, which will be followed by creating variables for video and xml file. Now capture the video frames and training the XML classifiers to describe the vehicle objects which are needed to be identified. Create one variable to count the frames.

Here we are using while loop for capturing individual intended frame and will ensure that each frame is captured. It will also look for the vehicles present in the screen, recognize them with rectangle frame. The scaling factor is used in order to improve efficiency of identification [9].

The different sizes of the vehicles are identified and every 10th frame in the video is saved for further processes. In order to make the detection hassle-free, we will resize the frame images with some fixed pixel value (In our code, we have used 340*220 frame size). Once the code is executed as a single unit, it would produce single frame images.

# 4. RESULTS AND DISCUSSIONS

## 4.1 Results:

Here we have used a xml file as classifier so we can apply cascade classifier and detect the vehicle from the video. So, let us understand the HaarCascade Classifier:

This is essentially a machine-based learning approach where a cascade function is trained from both positive as well as negative images. Based on the training it is then used in the other images to identify the objects. So how this works is big .xml files with a lot of sets of features and each xml corresponds to one very specific type of use case.

Here we have attached some screenshots of output, and we can clearly identify the detected vehicles.

**<u>Image before applying algorithm (picture captured from video player):</u>**

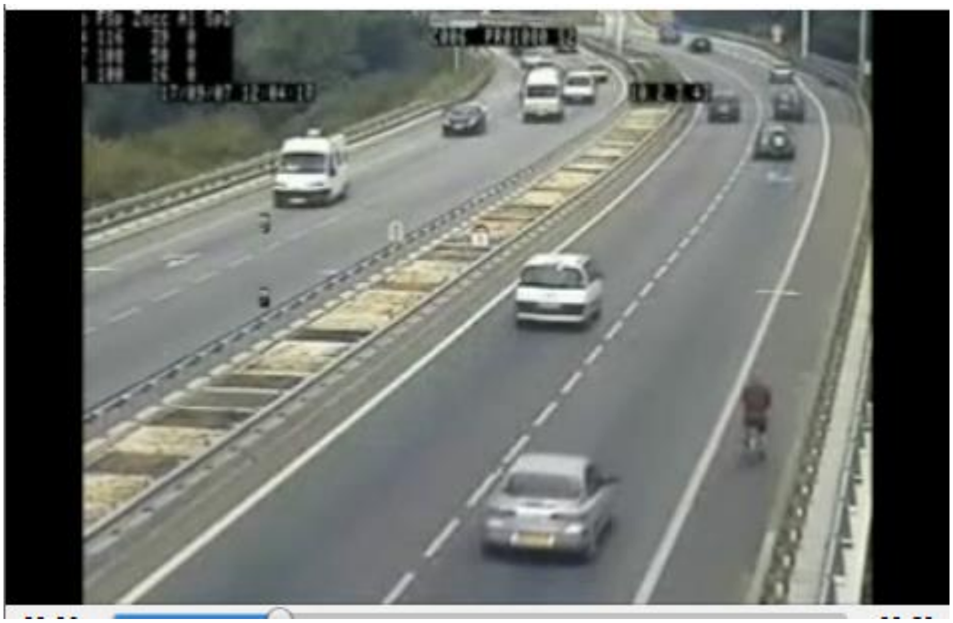

*Figure 5 Input-1 video frame*

As we can easily see the vehicles from the video frame and after applying the algorithm, we can easily detect the vehicles.

**Image after applying algorithm (picture from the output):**

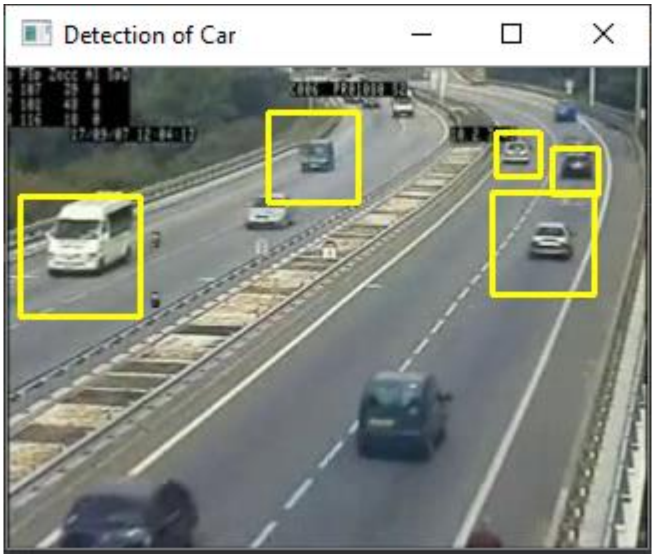

*Figure 6 Output-1 frame-1*

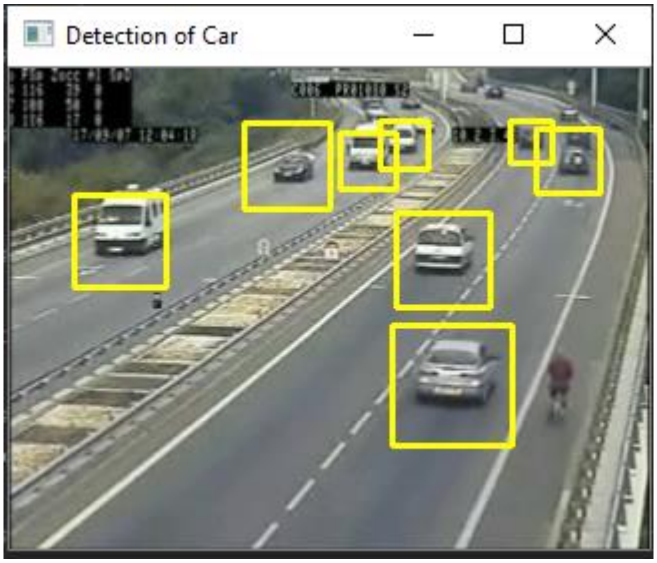

*Figure 7 Output-1 frame-2*

From the pictures as mentioned above we can easily detect the vehicle from the frame.

**Image before applying algorithm (picture captured from video player):**

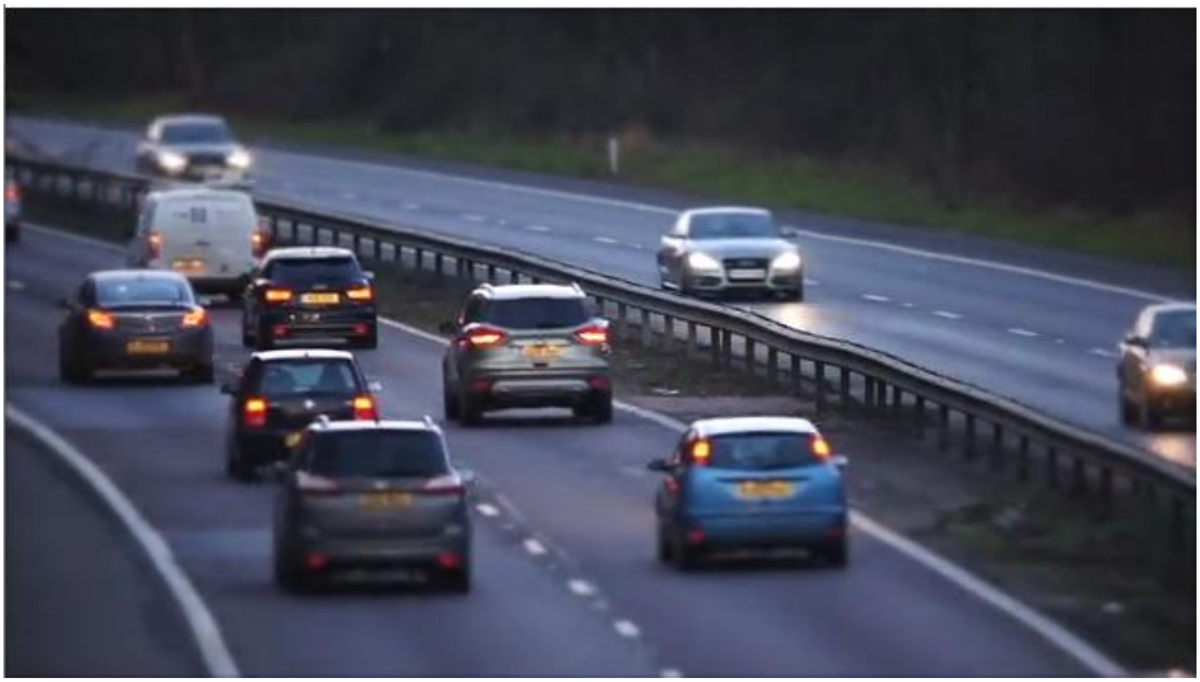

*Figure 8 Input-2 video frame*

**Image-1 after applying algorithm (picture from the output):**

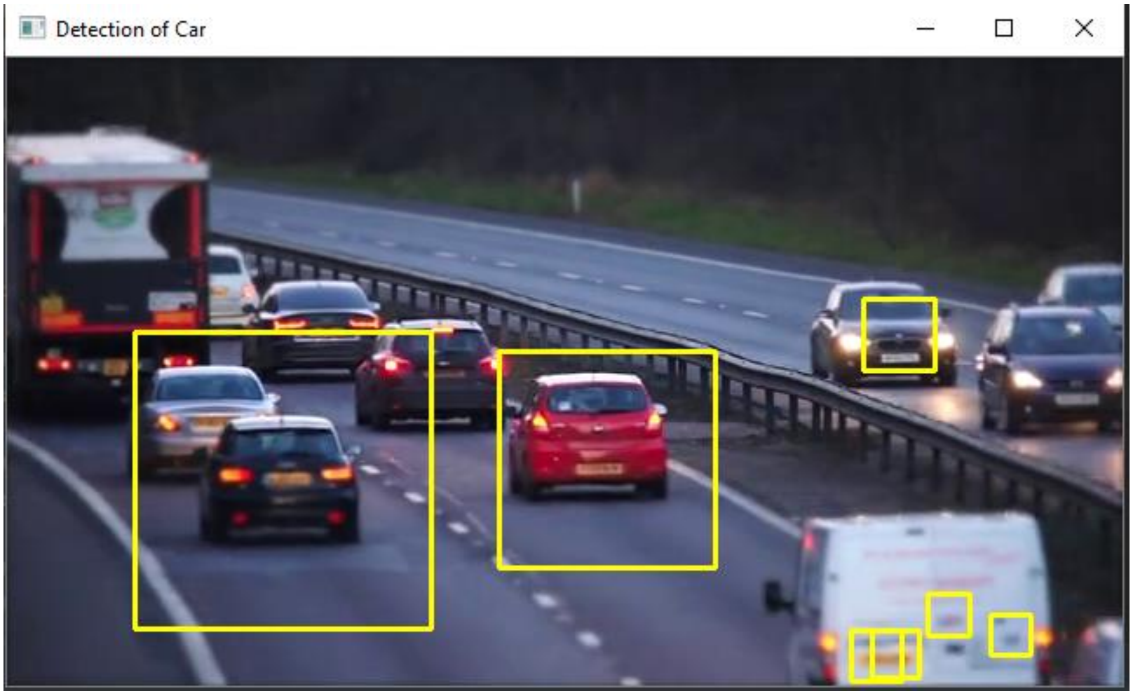

*Figure 9 Output-2 Video frame-1*

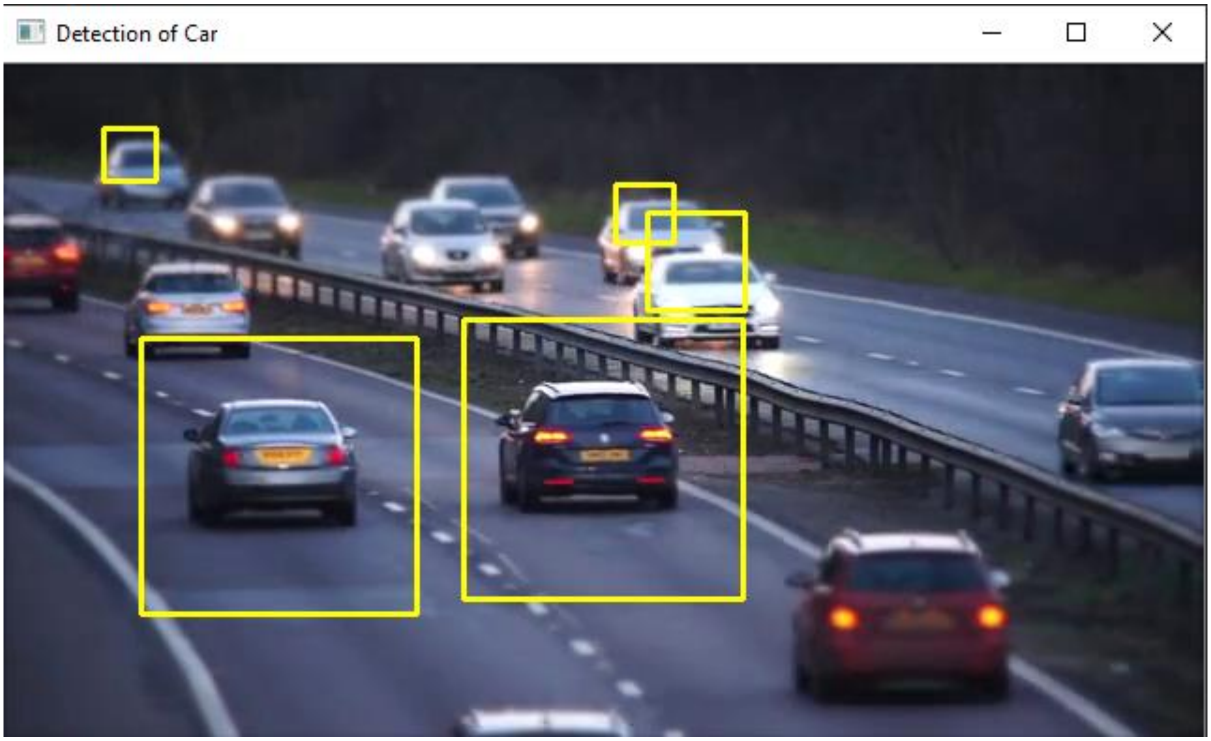

*Figure 10 Output-2 Video frame-2*

As output of code we can detect the vehicles (cars) from the video frame.

**Image before applying algorithm (picture captured from video player):**

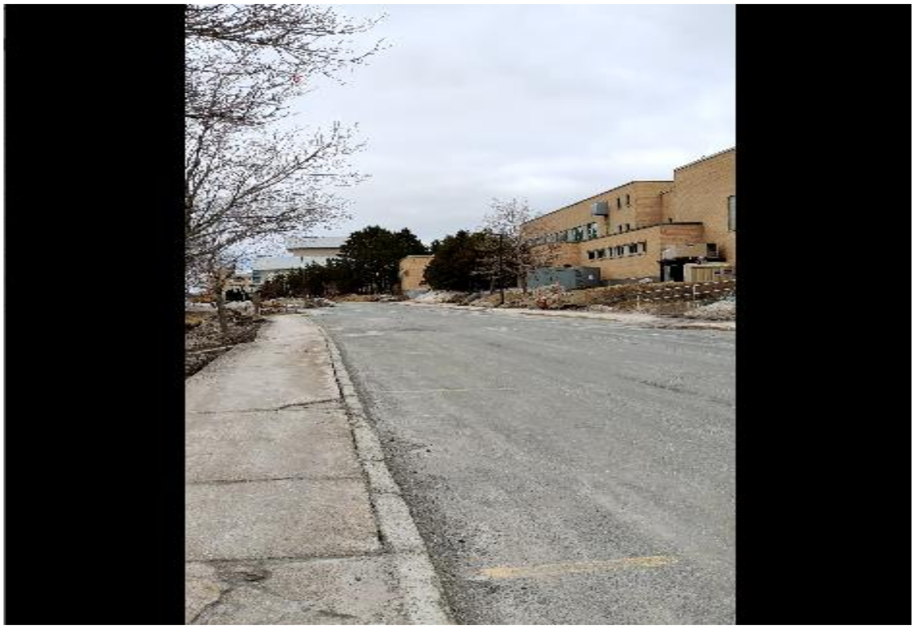

*Figure 11 Input-3 frame (Memorial University Road)*

**Image after applying algorithm (picture from the output):**

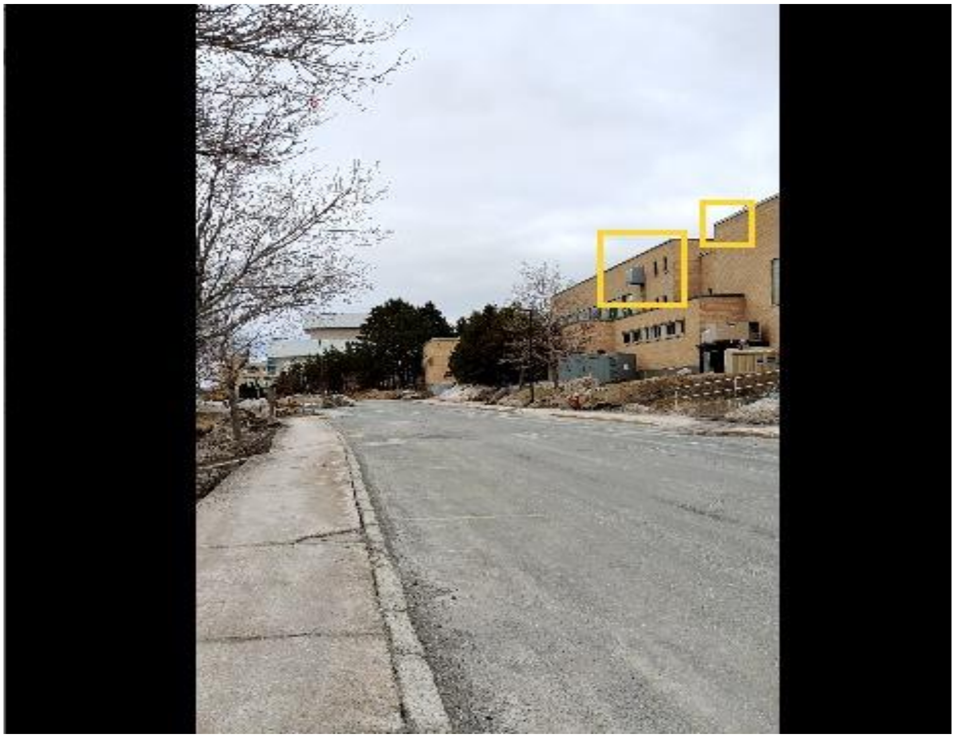

*Figure 12 Output 3 Frame (Memorial University Road) – False Output detected by the algorithm*

The last derived output is falsely identified. The recognized objects are not car-type. They resemble some kind of car having bonnet. The reason behind such bug is the dataset we have used.

**Image before applying algorithm (picture captured from video player):**

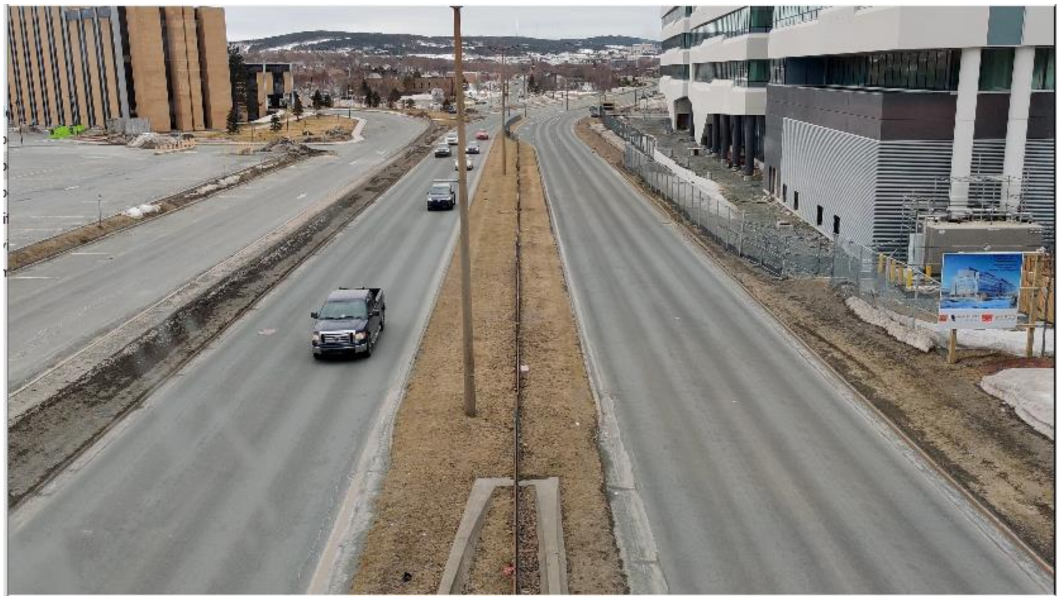

*Figure 13 Input-4 Video Frame(Memorial University-UC-food Court)*

**Image after applying algorithm (picture from the output):**

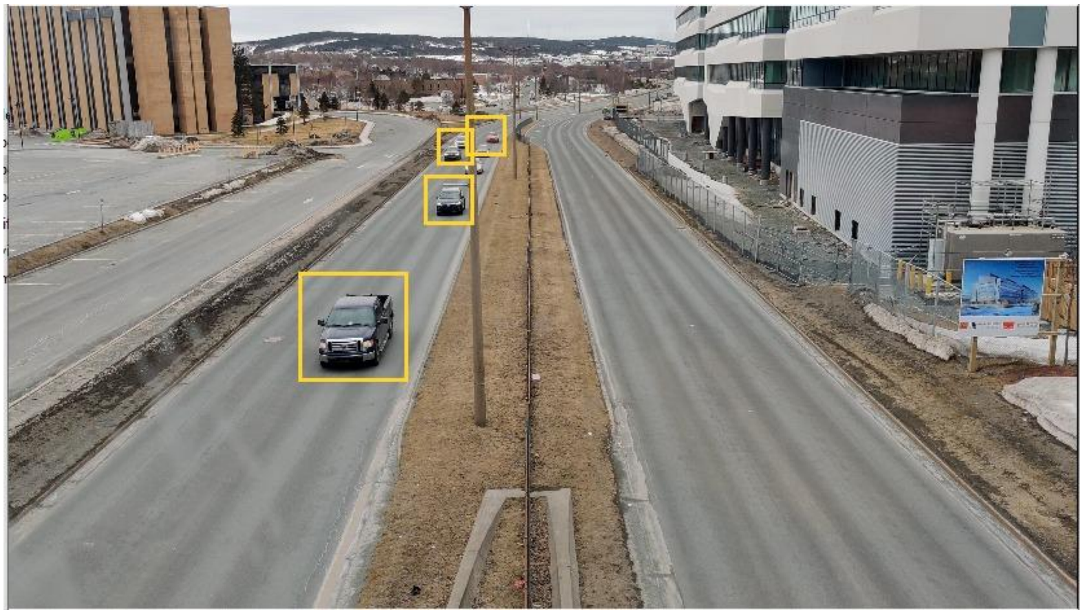

*Figure 14 Output 4 Video Frame (Memorial University-UC-food Court)*

## 4.2 Discussion and Challenges

Haar cascade is a very old method which is based on object detection framework. In order to count the bounded rectangles, we will have to count each frame generated. Since creating a rectangle around moving object is somewhat challenging and thus, the generated rectangle frames are not accurate. Moreover, the count of these frames are not exact since these keep on fluctuating. Thus, for just one car, 4 to 5 boundaries are generated which is incorrect. This is considered as limitation of this algorithm [10].

# 5. CONCLUSION

We presented moving vehicle detection method for various vehicle videos in isle-time and high detection rate. The training algorithm used is Cascade Classifier. We have tried that our classifier can distinguish vehicles exactly from the background and we are getting near success in this regard. The configuration was tested for 7 traffic clips and non-traffic clips with different backgrounds at different locations.

For this system, we get an overall accuracy of 70.52 per cent. Since the rectangular frame around identified object keep on fluctuating, we need to improve our method and terminology used. Accuracy was calculated on the basis of vehicle detection and counting. Even if a few vehicles are detected and some other vehicle, present in the same frame of video stream is not detected, the output was considered wrong, regardless of the vehicle which was properly extracted. Also, the challenges in multi-view vision systems and various approaches in multi-view systems were examined.

# 6. REFERENCES

[1] 2020. [Online]. Available: https://www.researchgate.net/publication/323045672_Vehicle_Classification_using_Haar_Cascade_Classifier_Method_in_Traffic_Surveillance_System. [Accessed: 07- Apr- 2020].

[2] 2020. [Online]. Available: https://www.researchgate.net/publication/267272082_Vehicle_Detection_and_Tracking_Techniques_A_Concise_Review. [Accessed: 07- Apr- 2020].

[3]"OpenCV: Cascade Classifier", *Docs.opencv.org*, 2020. [Online]. Available: https://docs.opencv.org/3.4/db/d28/tutorial_cascade_classifier.html. [Accessed: 07- Apr- 2020].

[4] "Computer Vision — Detecting objects using Haar Cascade Classifier", Medium, 2020. [Online]. Available: https://towardsdatascience.com/computer-vision-detecting-objects-using-haar-cascade-classifier-4585472829a9. [Accessed: 07- Apr- 2020].

[5] B. Momin and T. Mujawar, "Vehicle detection and Attribute based search of vehicles in video surveillance system", 2015 International Conference on Circuit, Power and Computing Technologies, 2020. [Accessed 7 April 2020].

[6] B. Pawar, V. Humbe and L. Kundnani, "Morphology Based Moving Vehicle Detection", 2020. [Accessed 7 April 2020].

[7] "OpenCV: Cascade Classifier", *Docs.opencv.org*, 2020. [Online]. Available: https://docs.opencv.org/3.4/db/d28/tutorial_cascade_classifier.html. [Accessed: 07- Apr- 2020].

[8] Engineeringbigdata.com, 2020. [Online]. Available: https://www.engineeringbigdata.com/vehicle-detection-opencv-python-cv2/. [Accessed: 07- Apr- 2020].

[9] S. S and D. R, "A Review on Video Based Vehicle Detection, Recognition and Tracking", 3rd IEEE International Conference on Computational Systems and Information Technology for Sustainable Solutions 2018, 2018. [Accessed 7 April 2020].

[10] Y. Qu, L. Jiang, and X. Guo, "Moving Vehicle Detection with Convolutional Networks in UAV Videos", 2016 The 2nd International Conference on Control, Automation and Robotics, 2020. [Accessed 7 April 2020].