IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

SMART TRAFFIC LIGHT

SPARSH JINDAL [1] NISHANT NEHRA [2] VIBHOR SHARMA [3]

[3] Assistant Professor

[1,2,3] Department of Information Technology

[1,2,3] Maharaja Agrasen Institute of Technology (IPU), Delhi, India

ABSTRACT

Emergency vehicles such as ambulances, fire trucks are delayed in traffic as they confront numerous challenges, and valuable human lives are lost as a result of bad traffic management. In this research, a model is proposed for predicting traffic heaviness on roadways utilising image processing techniques with an emergency vehicle recognition system, as well as a traffic signal management model based on information collected from real time traffic videos.

The majority of traffic signal systems use a timing mechanism that switches the lights after a set amount of time. In some situations, this may be ineffective. The controller receives no feedback from the regulated variable (traffic volume) (traffic control system using lighting). For example, whether the road is crowded or not, the duration of the green light will always be the same. To solve these congestion problems, it is better to build a new control system; a smart and intelligent control system.

The smart traffic light system senses the presence or absence of vehicles and reacts accordingly. The idea behind smart traffic systems is that drivers will not spend unnecessary time waiting for the traffic lights to change. In addition, on detecting emergency vehicles in a lane the traffic signal is turned to green immediately so that a free passage is provided to these vehicles. This would help the emergency vehicles to reach the destination on the time thereby saving time as well as loss of life.

INTRODUCTION

We've been able to measure metrics like traffic flow, vehicle speed, and number of cars by employing image processing techniques to monitor traffic. Image processing is utilised in traffic surveillance for vehicle tracking, licence plate recognition, and identifying road impediments, among other things. Image processing in traffic surveillance can lead to better traffic flow control and detection of indiscreet drivers and illegal speed offenders. Many academics have already proposed numerous methods for automatically detecting automobiles using image processing techniques. The goal of this study is to see how effective our proposed technology is for detecting emergency vehicle and thereby reducing clutter.

Congestion can be defined mathematically as the number of cars passing through a place in a flow. There are a variety of situations that exacerbate traffic congestion, with the majority of them resulting in an increase in the number of volumes necessary for a given number of persons. Rainfall, in terms of traffic congestion, increases traffic capacity and speed, resulting in more congestion.

Our proposed method uses the adaptive background subtraction technique, followed by morphological operations to locate and detect vehicles and to remove irrelevant objects.Our model uses our custom haar cascade for emergency vehicle detection. The proposed method is computationally cost-effective. This paper will explain, review and test the proposed system and includes the performance test and conclusion.

METHODOLOGY

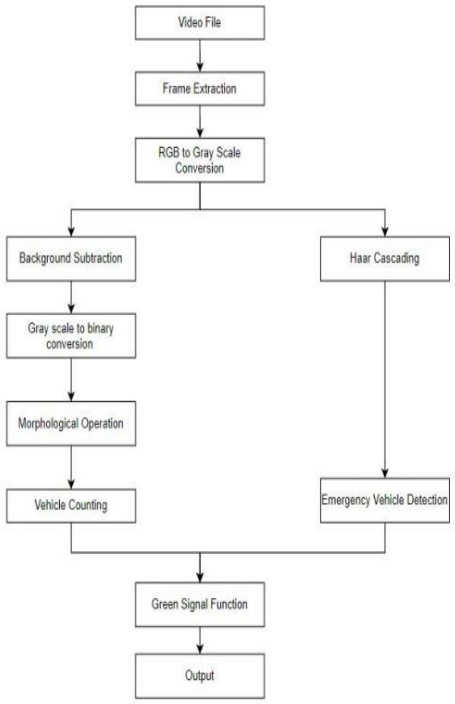

Figure 1. System Design for Smart Traffic Light

A)DETECTION

Object Detection is a computer technology related to computer vision, image processing, and deep learning that deals with detecting instances of objects in images and videos.

Knowing where an object is in an image is called localization in computer vision. Using contour detection, we can detect the borders of objects, and therefore, localize them easily.

Object detection can be applied in solving hard problems in areas like image search and video surveillance. It is used widely in computer vision tasks including face

detection, face recognition, and object tracking. All object types have special attributes that help in classifying them. For instance, all faces are round. Object detection algorithms use these special attributes to identify objects in images and videos.

In our project we have used CHAIN_APPROX_SIMPLE algorithm for contour detection. It removes all redundant points and compresses the contour, thereby saving memory.

In the image we can see that a car is detected and our model has created a green rectangle bounding box surrounding our object(car).

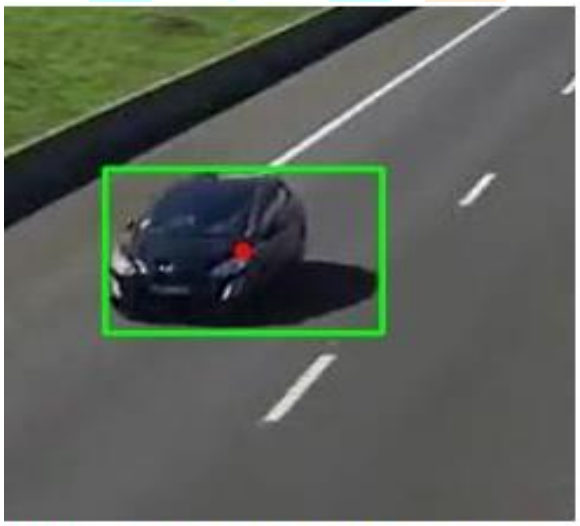

Figure 2. Vehicle Detection

B) BACKGROUND SUBTRACTION

Background or image subtraction is the process of extracting the foreground of an image from its background. If you have a background image like a road without vehicles in it, you can subtract this image from another image of the same road (from the exact same view) which contains vehicles to detect those vehicles. The background pixels would cancel each other out and the objects in the foreground would pop out.

Adaptive background subtraction is one of the techniques in the field of image processing and machine vision. It extracts the information of objects from the current frame, by subtracting the current frame from the background model. After the image pre-processing step (which includes noise removal, etc.), adaptive background subtraction can be used to determine the area of moving objects. The background subtraction technique is used extensively for detection of moving objects in fixed cameras, we can detect the movement of objects by calculating the difference between the current frame and the average total of several previous frames, known as “background frame” or “background model”. Background model usually is the average total of N previous frames.

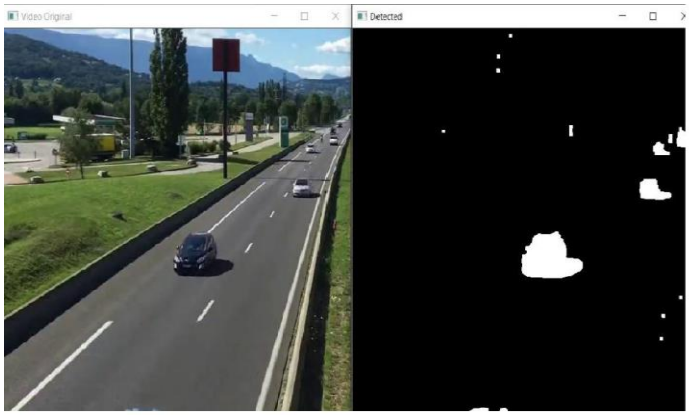

Figure 3. Background Subtraction

In the image on the left hand side is the original video and on the right hand side the video is passed through background subtraction. In background subtracted video only objects which are moving such as cars and grass/trees moving due to wind are shown with white colour all other static background objects are replaced with black colour.

C) COUNTING

After passing the video through background subtraction algorithm we get moving objects in that video.Vehicles count is increased when a vehicle cross a line at an exit point of frame.Using a counting line makes it easier to count vehicles moving in a certain direction.

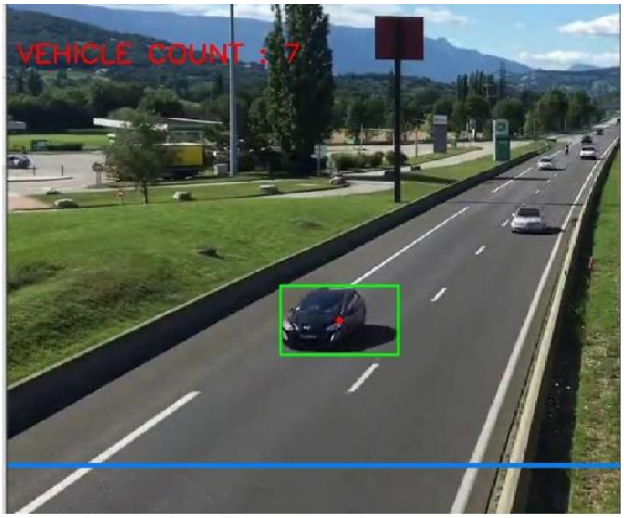

Figure 4. Vehicle Counting

D) HAAR CASCADING

Haar Cascades are object detection models depending on the concept of Haar-like features developed by Paul Viola and Michael Jones. They work with image intensities i.e. the RGB values of the images and help in feature calculation.

It is a machine learning based approach where a **cascade function** is trained from a lot of positive and negative images (where positive images are those where the object to be detected is present, negative are those where it is not). It is then used to detect objects in other images.

The idea of Haar cascade is extracting features from images using a kind of 'filter', similar to the concept of the convolutional kernel These filters are called Haar features .The idea is passing these filters on the image, inspecting one portion (or window) at the time. Then, for each window, all the pixel intensities of, respectively, white and black portions are summed. Finally, the value obtained by subtracting those two

summations is the value of the feature extracted. Ideally, a great value of a feature means it is relevant.

Haar Cascade is generally faster than deep learning models and can run on devices with minimal hardware requirements when compared to the hardware requirements needed to run a deep learning model.

- The haar cascading detection works only on grayscale images. So it is important to convert the color image to grayscale.
- detectMultiScale function is used to detect emergency vehicles. It takes 3 arguments — the input image, scaleFactor and minNeighbours. scaleFactor specifies how much the image size is reduced with each scale. minNeighbours specifies how many neighbors each candidate rectangle should have to retain it.
- Output of detectMultiScale function contains a list of coordinates for the rectangular regions where cars were found. We use these coordinates to draw the rectangles in our image.

Figure 5. Emergency Vehicle Detection Using Haar Cascade

In the image we can see that our model was able to detect an emergency vehicle(Fire Truck) successfully.

RESULT

Traffic being one of the major issues in urban cities, it becomes a necessity to resolve it through creative thinking integrated with technology. We have finally achieved following objectives:

1) Traffic management
2) Traffic signal Importance
3) Priority given to Emergency vehicles on road

The advantages of this system is to calculate the number of vehicles automatically by using a camera. With this system, we will get real time data that can be done for monitoring the density of traffic. In addition, the advantages of the system is that it can perform the classification of vehicles into- emergency and non-emergency vehicles. The weakness of the system dependent on the illumination, with

good illumination this system will have a very good accuracy. However, if the illumination is not good, as at night, the accuracy of this system will decrease significantly.

These goals achieved would have a great impact on society.

CONCLUSION

The idea of the whole project lies in dynamically allocating green traffic light on the basis of the number of vehicles present in the lane of over four lane junctions. Not only this, we have also given priority to emergency vehicles through detection of these kinds of vehicles.

In the project for each lane we have counted the number of cars using background subtraction algorithm and have detected emergency vehicles using haar cascading.

If emergency vehicles are present on any lane then that lane gets green signal otherwise the lane with maximum number of vehicles will get green signal.

REFERENCES

[1] https://docs.opencv.org/3.4/d1/dc5/tutorial_background_subtraction.html

[2] https://www.pyimagesearch.com/2021/04/12/opencv-haar-cascades/

[3] http://ijsrd.com/Article.php?manuscript=IJSRDV8I90154

[4] https://www.ijrte.org/wp-content/uploads/papers/v8i6/F9526038620.pdf

[5] https://core.ac.uk/download/pdf/83533829.pdf

[6] https://www.ijcst.org/Volume6/Issue10/p2_6_10.pdf

[7] https://iopscience.iop.org/article/10.1088/1742-6596/1025/1/012025/pdf

[8] https://www.analyticsvidhya.com/blog/2020/04/vehicle-detection-opencv-python/