Abstract

Automatic video analysis from traffic surveillance cameras is a fast-emerging field based on computer vision techniques. It is a key technology to public safety, intelligent transport system (ITS) and for efficient management of traffic. In recent years, there has been an increased scope for automatic analysis of traffic activity. We define video analytics as computer-vision-based surveillance algorithms and systems to extract contextual information from video. For this purpose we will use Matlab's computer vision default functions. Detection of moving objects in video streams is the first relevant step of information and for this we have many methods in Matlab. For moving object detection we have Background Subtraction, Gaussian Mixture Model, Kalman Filter and Optical Flow. Then we have to classify these moving objects in car, buses and bikes. For this purpose we can use bag of words or threshold value of area covered by detected objects in a particular box (called bbox). After this we have to collect information about detected objects as company and model of that object. For this we have to prepare a data set that contains various images of particular company cars so that we can extract features and match them with detected object to find out the exact details of detected object. So there is a question arising that which feature should be extracted? In Matlab we have different types of features such as SURF features, MSER features, SIFT features and OCR recognition but which features are to be we store? And at the end we count all the detected object to count total number of vehicles passed through the road but if calculate all the bounding boxes then it will give wrong result so we have to decide ROI (Region of interest). In this particular region our moving object detected so multiple counting of the same object. But remember all the methods that are used to detect or count or classify the vehicles are not hundred percent right. They all depend the pattern of the traffic, weather, position of camera and many other things.

Table of Contents

List of Figures

1. Introduction

As urban road intersections are prone to traffic congestion and traffic accidents, monitoring the crossing of vehicles and predicting the state is needed to reduce traffic congestion and prevent accidents. Conventional technology for traffic measurements, such as sonar or microwave detectors, inductive loops suffer from some serious drawbacks: they are expensive to install, they demand traffic disruption during installation or maintenance, they are not portable and they are unable to detect slow or stationary vehicles. On the contrary, video based systems are easy to install, they can be easily upgraded and they offer the flexibility to redesign the system and its functionality by simply changing the system algorithms. Those systems allow vehicle counting, classification, measurement of vehicle's speed and the identification of traffic incident .A typical video surveillance system has to face many challenges such as:

1. Type of the vehicle
2. Information about detected object such as company and model of that car.

Traffic surveillance system is an active research topic in computer vision that tries to detect, recognize and track vehicles over a sequence of images and it also makes an attempt to understand and describe object behavior, vehicle activity by replacing the aging old traditional method of monitoring cameras by human operators. A computer vision system can monitor both immediate unauthorized behavior and long term suspicious behavior, and hence alerts the human operator for deeper investigation of the event.

A typical surveillance system consists of a traffic camera network, which processes captured traffic video on-site and transmits the extracted parameters in real time. This system is used to detect, recognize and track vehicle from incoming video frames in dynamic scenes then extract the license plate from it.

It has found numerous applications as wide as possible such as: access control in security sensitive areas, securities for communities and important buildings, detection of military target areas, traffic surveillance in cities and highways, detection of anomalies behavior , traffic control management for recognize vehicles that commit traffic violation, such as occupying lanes reserved for public transport, breaking speed limits, crossing red light, entering restricted area without permission; and among many other applications.

The system is designed for real time videos where a camera is used for continuous recording of videos. The view of camera or the area covered by camera is fixed between entry zone and exit zone. Each frame is continuously processed to check the presence of a vehicle. A defined connected component area is taken as threshold; if the detected area is above that threshold value then it will be recognized as a vehicle and will be tracked. A distance is defined between the vehicle and the camera and when the vehicle comes within that range. Counting of vehicle helps to know about how many cars have passed from that road to identify the conditions of traffic at that road and it can also help to measure the economy of that state. It also provides feedback about which company cars are popular among people.

2. Literature Survey

2.1 Detect Moving Objects:-

There are various methods to detect the moving object:-

2.1.1 Background Subtraction :-

Background Subtraction, also known as Foreground Detection, is a technique in the fields of image processing and computer vision wherein an image's foreground is extracted for further processing (object recognition etc.). Generally an image's regions of interest are objects (humans, cars, text etc.) in its foreground. Background subtraction is a widely used approach for detecting moving objects in videos from static cameras.

Background subtraction is mostly done if the image in question is a part of a video stream. Background subtraction provides important cues for numerous applications in computer vision, for example surveillance tracking or human poses estimation.[1]

Background subtraction is simply separating the foreground (object of interest) from the background .This can be carried out in various ways .A video is first converted into separate frames .Then each frame is compared with the previous frame .Since the camera is still, the background would remain constant .Hence change or motion is detected if the variance crosses a certain threshold value. And this changed part is the moving object .Second method is setting a general threshold value for the whole video .The values that lie above it are considered as the foreground and rest all part are rejected .Since a general threshold value is selected there is possibility of presence of other unwanted noise such as line markings on the road etc. To remove these, morphological operations are done on it .Third and the most efficient way of doing the background subtraction is providing a statistical method a set of input training frames .This method will then calculate the mean and variance of the frames .After that it detects the foreground based on the values that cross the threshold value by a specific variance.

After the background subtraction is done on the frames the remaining objects have to pass through a some morphological operations to get cleaner foreground .This step is necessary since there is a possibility of presence of unwanted objects in the foreground along with some additional noise .After a vehicle is detected, it is marked in a certain way such as marking it with a mark on it or bounding it with a rectangle .The next is to calculate the number of the vehicles based on the output and average frequency of the vehicles and non-stationary background regions, such as branches and leaves of trees, a flag waving in the wind or a towed car , should be identified as part of the background.

There are various methods of Background Subtraction [1]:-

1. Background subtraction by Thresholding.

   (A) Advantages of this method:-

   - simple to understand
   - computation method is simpler
   - uses a very less amount of memory

   (B) Disadvantages:-

   - dark-colored cars are rejected along with the background and only light-colored cars are detected
   - Sometimes white colored cars can be counted twice due to the separation by the dark colored glass.

2. Background subtraction by frame differencing:-

   (A) Advantages:-

   Since background is updated each time and not a general threshold value is used this method is more efficient than the previous one.

   (B) Disadvantages:-

   As this method comprises the step of decomposing the video into frames and then compressing them again it involves a lot of memory to store the frames. Besides it involves complex calculations in every step for determining the foreground.

2.1.2 Gaussian Mixture Model :-

A Gaussian Mixture Model (GMM) is a parametric probability density function represented as the sum of Gaussian component densities. GMMs are used as a parametric model. These parameters are estimated from training data using the iterative Expectation-Maximization (EM) algorithm or Maximum A Posteriori (MAP) estimation from a welltrained preceding model. Rather than immediately processing the entire video, the example starts by obtaining an initial video frame in which the moving objects are segmented from the background. This helps to gradually introduce the steps used to process the video. The foreground detector requires a certain number of video frames in order to initialize the Gaussian mixture model. After the training, the detector begins to output more reliable segmentation results. The two figures below show one of the video frames and the foreground mask computed by the detector. It can be through foreground mask computed by the detector. The foreground segmentation process is not perfect and often includes undesirable noise. So we can use morphological opening to remove the noise and to fill gaps in the detected objects. Next, we find bounding boxes of each connected component corresponding to a moving car by using vision.BlobAnalysis object. The object further filters the detected foreground by rejecting blobs which contain fewer than some threshold value. To highlight the detected cars, we draw boxes around them. In the end, we process the remaining video frames. The output video displays the bounding boxes around the moving objects.

(A) Advantages:-

- This method uses a training set to determine the foreground, hence it is more efficient in giving a perfect output.
- Since generalized threshold is not used,it is able to detect white cars as well as the dark colored cars which was not possible using a simple thresholding function
- The number of vehicles in the current frame can be calculated .As a result vehicle frequency can be calculated that is helpful in avoiding traffic congestion.

(B) Disadvantages:-

- Though this method is more efficient than other methods it has other disadvantages such as shadow is not removed and it can confuse the user of the size of the vehicles.

2.1.3 Kalman Filter:-

Within the significant toolbox of mathematical tools that can be used for stochastic estimation from noisy sensor measurements, one of the most well-known and often-used tools is what is known as the *Kalman filter*. The Kalman filter is essentially a set of mathematical equations that implement a predictor-corrector type estimator that is *optimal* in the sense that it minimizes the estimated *error* covariance—when some presumed conditions are met. The Kalman filter has been used extensively for tracking in interactive computer graphics. There are different types of kalman filter [6]:-

1. The Discrete Kalman Filter
2. The Extended Kalman Filter (EKF)

2.1.4 Optical Flow:-

Optical flow method involves calculating the image optical flow field and doing clustering processing according to the optical flow distribution characteristics of image. This method can get the complete movement information of an object and it is useful for detecting the moving object from the background with the 85% accuracy [5], but this method has a few disadvantages including large quantity of calculations, sensitivity to noise, poor anti-noise performance, which make it inappropriate for real-time object detection and tracking. Optical flow estimation tries to assign to each pixel of the current frame, a two-component velocity vector indicating the position of the same pixel in the reference frame. There are several optical flow estimation algorithms known in the literature. According to the taxonomy proposed in, we can cluster algorithms in the following categories: region-based matching, differential (Lucas- Kanade, Horn-Schunk) and energy-based algorithm. Optical flow method generates optical flow field for every pixel in sequential images, in which the velocity and direction of every pixel are obtained. To detect moving regions in an image, optical flow uses flow vectors of the moving objects over time. It is used for motion-based segmentation and tracking applications. It is a dense field of displacement vectors which defines the translation of each pixel region. Optical flow is best suited in presence of camera motion, but however most flow computation methods are computationally complex and sensitive to noise.

2.2 Classify as a Car or Truck :-

There can be two methods for this[6]:-

2.2.1 Area Estimation :-

If the area of blob [4] of the object is between some threshold value then classify it into a car and if the ratio is greater than this then classify it into as a Bus.

2.2.2 Geometric-Based Vehicle Classification Approach :-

There are two basic steps Feature Extraction and Classification [2]: -

(A) Features Extraction: -

The feature extraction expresses the visual content of the images, which should ideally quantify certain significant characteristics of vehicles in the images. Simple dimensional measurement-based features to describe the vehicle are extracted from the images. The extracted features are: width, length and height. The reasons for selecting such features are their low computational cost and storage requirements.

(B) Classification:-

The Support Vector Machine (SVM) algorithm is used for vehicle classification. The extracted geometric features are fed directly into the SVM to classify vehicles to their corresponding classes. SVM is a supervisory classifier originated from the statistical learning theory and attempts to identify a set of support vectors. SVM disparate other learning systems in its decision surface which is an optimal hyperplane in a higher dimensional feature space. This hyperplane minimizes the risk of misclassification and the SVM algorithm identifies a set of support vectors and the final decision of the SVM is based on a "max. voting", in which the category that corresponds to the highest confidence is the winner.

2.2.3 Appearance-Based Classification Approach :-

There are three steps [2]:-

(A) Features Extraction:-

In this study, the two main types of appearance features are extracted from input images: Scale Invariant Feature Transform (SIFT) and Speeded-Up Robust Feature (SURF).Scale Invariant Feature Transform (SIFT), introduced by David Lowe in 1999, is a widely used feature descriptor algorithm. Most state-of-the-art object recognition systems use SIFT to represent images and it has been proven to be the most powerful and successful among local image descriptors. Using SIFT, each input image is represented by a set of relatively invariant local features in which, feature points in the image are detected using Harris- Laplace salient point detector. Then, descriptors for each feature point in the image are computed. Each feature point in the image is represented by a 128-dimensional orientation vector. The second appearance feature used in this study is Speeded- Up Robust Feature (SURF) which introduced by Bayet al. in 2006. SURF is also a well-known descriptor and is similar, in some concepts, to SIFT. In which, both focus on the spatial distribution of gradient information. The SURF descriptors are constructed by extracting square grid around the interest points, then dividing each grid cell into sub-grids.

(B) Bag-of-words Modeling:-

After feature extraction, each input image is represented by a set of feature descriptors (SIFT or SURF). The dimensions of these descriptors are very high (more than thousands).Therefore, it is unreasonable to feed them directly to the classifier, but they have to go through a feature representation. In which, an efficient modeling method is used to transform these high dimensional descriptors to more compact and informative representations. For that reason, the bag-of-words (BoW) paradigm is adapted here. The BoW paradigm was pioneered by Csurka et al. and has received much attention in object recognition. The main idea of the BoW paradigm is to treat image features as words. In which, after feature extraction, each input image is represented by a set of feature descriptors (SIFT or SERF). A visual vocabulary dictionary is constructed by applying a k-means clustering algorithm on the training data images, and each cluster center is considered as a "visual word" in the visual vocabulary dictionary. Then, all feature descriptors extracted from an image are quantized to their closest "visual word". For each image, the number of feature descriptors assigned to each "visual word" is then accounted into a histogram. Therefore, each image is represented as a histogram of frequencies of visual words that are in the image. For more details about using BoW in appearance-based classification approach.

(C) Classification:-

As in the geometric-based approach, the Support Vector Machine (SVM) algorithm is applied for vehicle classification.

2.3 Count the Moving Objects:-

For this purpose first we calculate the total moving objects detected in one frame then count the detected moving objects in next frame. If the difference between the number of detected objects in the current frame and the previous frame is greater than one then add the difference in the total number of moving objects otherwise don't edit the value of total number of moving objects

.

2.4 Gaps :-

There are many gaps:-

- There is still no algorithm that can detect moving objects in real time efficiently.
- The moving object, which we want to detect when comes closer to the camera then the algorithm that is based on detected objects blob area doesn't work correctly as the blob area increases when it comes closer that may show wrong result.
- If we use Appearance based vehicle classification then we need a fine data set and there is no limit of that fine dataset .The accuracy of that algorithm is also not very good.
- There is still no algorithm design that can classify the detected car into their different makes.

3. Problem Formulation

There is still no algorithm design that can detect moving object in real time correctly and there are no efficient algorithms to classify the objects and the algorithms which have been designed or we tried to use for that purpose do not work properly. In case of classification of different makes of the detected car, there is no algorithm till now. There is still no accurate criteria to count the detected vehicles but if we use tracking then counting can be performed.

We are trying to solve these problems:-

- Detect a moving object.
- Classify it into Car or Bus.
- Classify the detected car into different makes.

4. Objectives

The study is aimed at attainment of following objectives:-

- An algorithm to detect moving vehicles in real time.
- Classification of vehicles into Car or Bus
- To devise suitable method for detection of specific make of detected vehicle.

5. <u>Solution Design</u>

These are the steps followed to solve this problem: -

1. Detect moving objects.
2. Classify into Car and Truck.
3. Classify the detected car into their difference makes.

For detecting a moving object we are using optical flow algorithm. In this algorithm we differentiate the moving object by it's velocity, if it is greater than the threshold velocity then we classify it as a moving object and after that we apply blob analysis in which we place a rectangle that is called bbox (it contains 4 parameters as an array of 1x4 matrix that is x co-ordinate, y co-ordinate, width and height, x and y are the co-ordinates of the starting point of the bbox),on the moving object.

To differentiate between Car and Bus we will use ratio concept. In this concept we first calculate the area of the bbox which is over the detected object and then we calculate the blob area of that object. After this we calculate the ratio between the area of the blob and the bbox.

If the ratio is between the threshold values then it is classified as a Car else if the ratio is greater than the threshold value then it is classified as a Truck. If the object comes closer to the camera then the size of the bbox increases because the size of the moving object increases. At the same time blob area of the moving object also increases so the ratio of blob area and the bbox area, remains approximately same. It clears that the result will be same.

Till now we have discussed about differentiating Bus or a Car, now we will discuss how to differentiate different makes of a Car. For that purpose we use **bagOfFeatures**. It contains different types of features [3] as: -

1. *VocabularySize*: - Number of visual words500 (default) | integer scalar

   Number of visual words to include in the bagOfFeatures object, specified as the comma-separated pair consisting of 'VocabularySize' and an integer scalar in the range [2, inf]. TheVocabularySize value corresponds to K in the K-means clustering algorithm used to quantize features into the visual vocabulary.

2. *StrongestFeatures:-* Fraction of strongest features0.8 (default) | [0,1]

Fraction of strongest features, specified as the comma-separated pair consisting of 'StrongestFeatures' and a value in the range [0,1]. The value represents the fraction of strongest features to use from each image set contained in imgSet input.

3. *Verbose:-* Enable progress display to screentrue (default) | false

   Enable progress display to screen, specified as the comma-separated pair consisting of 'Verbose' and the logical true or false.

4. *PointSelection:-* Selection method for picking point locations'Grid' (default) | 'Detector'

   Selection method for picking point locations for SURF feature extraction, specified as the comma-separated pair consisting of 'PointSelection' and the string 'Grid' or 'Detector'. There are two stages for feature extraction. First, you select a method for picking the point locations, (SURF 'Detector' or 'Grid'), with the PointSelection property. The second stage extracts the features. The feature extraction uses a SURF extractor for both point selection methods. When you set PointSelection to 'Detector', the feature points are selected using a speeded up robust feature (SURF) detector. Otherwise, the points are picked on a predefined grid with spacing defined by 'GridStep'. This property applies only when you are not specifying a custom extractor with the CustomExtractor property.

5. *GridStep:-* Grid step size[8 8] (default) | 1-by-2 [x y] vector

   Grid step size in pixels, specified as the comma-separated pair consisting of 'GridStep' and an 1-by-2 [x y] vector. This property applies only when you set PointSelection to 'Grid' and you are not specifying a custom extractor with the CustomExtractor property. The steps in the x and y directions define the spacing of a uniform grid. Intersections of the grid lines define locations for feature extraction.

6. *BlockWidth:-* Patch size to extract upright SURF descriptor[32 64 96 128] (default) | 1-by-N vector

   Patch size to extract upright SURF descriptor, specified as the comma-separated pair consisting of 'BlockWidth' and a 1-by-N vector of N block widths. This property applies only when you are not specifying a custom extractor with the CustomExtractor property. Each element of the vector corresponds to the size of a square block from which the function extracts upright SURF descriptors. Use multiple square sizes to extract multiscale features. All the square specified are used for each extraction points on the grid. This property only applies when you setPointSelection to 'Grid'. The block width corresponds to the scale of the feature. The minimum BlockWidth is 32 pixels.

7. *Upright:-* Orientation of SURF feature vectortrue (default) | logical scalar

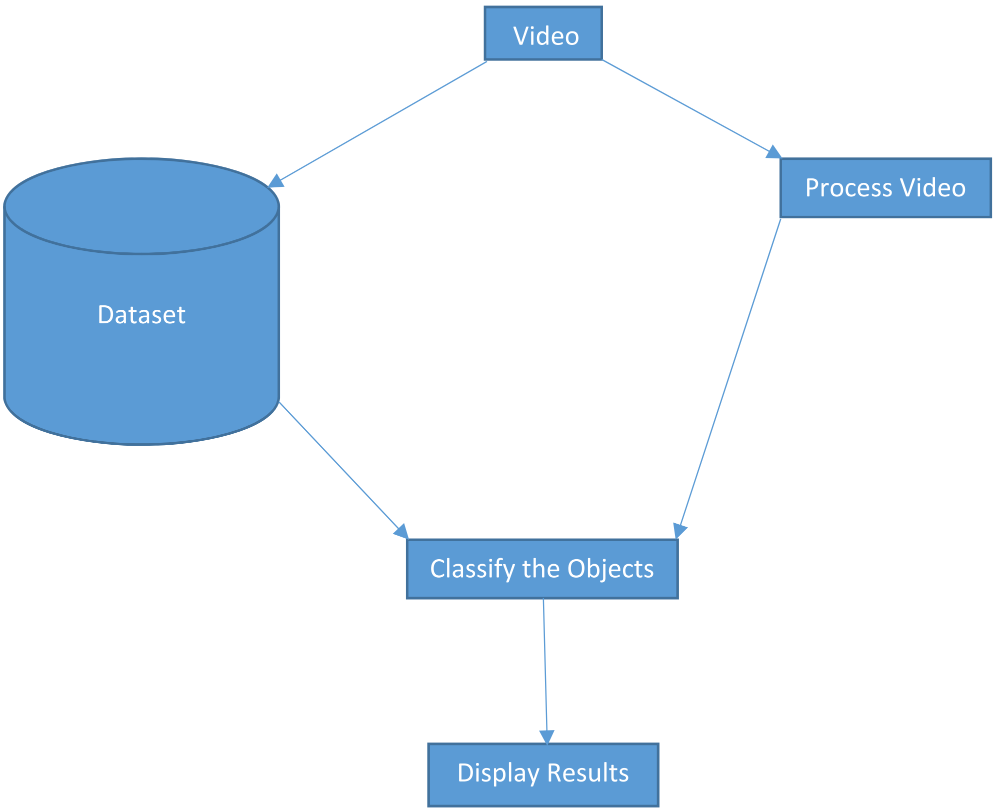
   Orientation of SURF feature vector, specified as the comma-separated pair consisting of 'Upright' and a logical scalar. This property applies only when you are not specifying a custom extractor with the CustomExtractor property. Set this property to true when you do not need to estimate the orientation of the SURF feature vectors. Set it to false when you need the image descriptors to capture rotation information.

We make a dataset that contains different folders (types as Swift, Alto, City) which contain precise images of that particular car. We now extract the features of type of object (folder) from bagOfFeatures() and we store these extracted features. To classify the different makes of the car we crop the bbox that will contain image of that detected car and match it to our previously stored features. The folder whose features matches approximately is the result for that cropped image. To classify into different makes we need a clear cropped image for that purpose we calculate threshold value in which we have a clear cropped image. For accurate result, we have to make a dataset of images which matches the angle of the camera for that object. So we have to make a fine data set. It can contain as much as images. The more the images, more the accuracy.

Then we put an annotation on that detected car which will show the result as Swift, Alto, City
But remember this is not a hundred percent accurate for the accuracy we have to make a dynamic dataset which will contains real time cropped images of detected car so it will increase our dataset. We can also use another approach for this. We can calculate the result in all the frames in which an object is detected and then take that result which will come maximum times. So by this way we find more proper result. It can also depend on the cropped images so the quality of the video should be good so that the cropped image will be good.

Using these approaches we can detect the moving object, classify the object as Car or Bus and can classify the car into different makes.

Solution Design



6. Methodology

6.1 Access the video:-

The **vision.VideoFileReader** function constructs a multimedia reader object that can read video data from a multimedia file.

Then the get method is used to get the general properties and the setting of the video file such as frame rate per second.

6.2 Create the objects:-

- **6.2.1:-** Create optical flow object

  Create Optical flow object for estimating direction and speed of object motion

  Using **vision.OpticalFlow().**

- **6.2.2:-** Create mean object

  Create object for calculate mean over a sequence of inputs using **vision.Mean().**

- **6.2.3:-** Create median filter object

  Create object for median filter to removing speckle noise (multiplicative noise to a image) using **vision.MedianFilter().**

- **6.2.4:-** Create morphological closing object

  Create Morphological closing object for filling holes in blobs using **vision.MorphologicalClose().**

- **6.2.5:-** Create blob analysis object

  Create a blob analysis System object to segment objects in the video using vision.BlobAnalysis().

- **6.2.6:-** Create morphological erode object

  Create Morphological erosion object for removing portions of the road and other unwanted objects using **vision.MorphologicalErode().**

- **6.2.7:-** Explore the video**.**

1. **6.2.7.1:-** Create video player object for original video
   Create one video object to display the original video using **vision.VideoPlayer().**

2. **6.2.7.2:-** Create video player object for resulted video
   Create another video object to display the final video using **vision.VideoPlayer().**

6.3 Develop the algorithm (for the one frame of the video)

- **6.3.1:-** Detect the moving objects.

1. Read the first frame of the video by applying step function onto the video file reader object.
2. Change the frame from rgb2gray to estimate optical flow.
3. Estimate optical flow by applying optical flow object onto the gray frame using step function.
4. Compute the magnitude of optical flow vectors squared which will used for thresholding.
5. Compute the velocity threshold from the matrix of complex velocities.
6. Threshold the image and then filter it to remove speckle noise by applying median filter object onto the previously calculated velocity threshold using step function.
7. Then thin-out the parts of the road and fill holes in the blob by applying Eroding and then Morphological Closing using step function and the result in a variable called segmentedObjects.
8. Then estimate the area and bounding box of the blobs by applying blob analysis object onto the segmentedObject using step function.
9. Then create a ROI (Region of interest). Only in this region, a moving is detected.
10. Then insert a box around the detected object applying any shape inserter on the frame using step function.

- **6.3.2:-** Classify into Car or truck.

1. Calculate the ratio of the area of bbox and blob of the segmented object and store it, into an array.

2. Then decide a threshold value to classify that object into Car or Bus.
3. If the value of the ratio is greater than 0.2 and less than 0.75 then classify it as a Car or if the value of the ratio is greater than 0.75 then classify it as a Bus.
4. This Area based approach is not very accurate because if the moving object detect more than one frame (that will surely happen), but in that case a new bbox and blob will create then it's most likely that the ratio of the area of bbox and blob will lie between the threshold values.

- **6.3.3:-** Classify Car into different makes.

1. To classify car into different makes we have to crop the image of detected object so we have to store the parameters of the bbox so that we can crop that bbox using **imcrop** function in which we will give four parameters of the bbox. But it can also happen that our bbox is empty then we should use **isempty** function to check the image that image is empty or not. So if the image is not empty then it shows the cropped image.
2. After this, in particular frames we store the values of bbox and text corresponding to that frame, in a cell. Then use **strcmp** function to find a text which is come mostly and assign that result to the text. To find out the maximum number we use **max** function.
3. Then insert annotation using **insertObjectAnnotation** show the label upon the detected car.

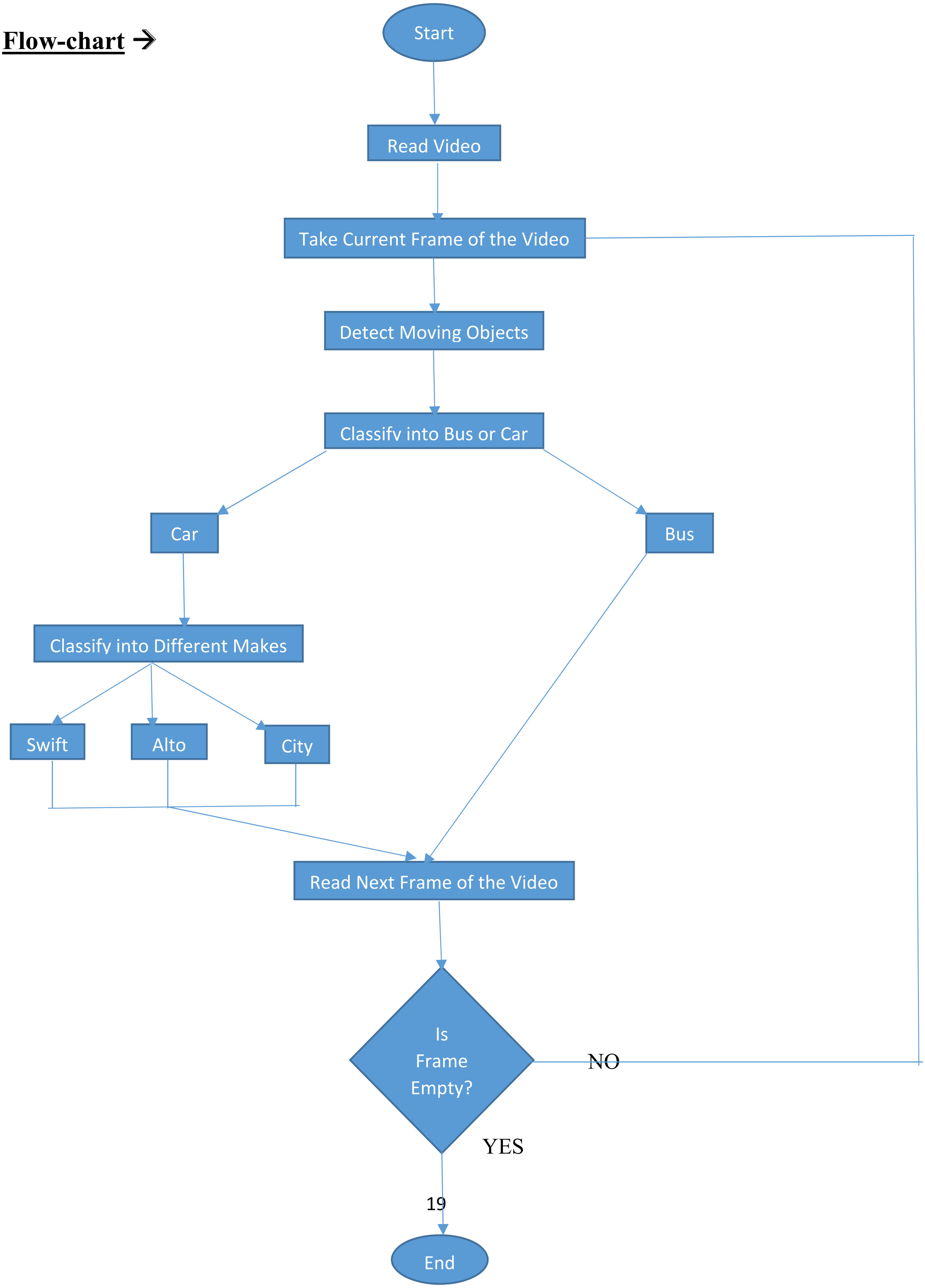
- **6.3.4 :-** Count the bbox

  When bbox is created we increment the counter by the number of bboxes but by this method we count an object multiple times in different frames but if we use a box in which each object is detected only one time then there will be only one bbox corresponding to that object so then if we count bboxes then we automatically count the detected objects.

6.4 Access another frames

To access next frame apply videoPlayerObject on the current original frame or to access the next resultant frame apply another videoPlayerObject on the current resultant frame using step function.

Flow-chart →



Start
Read Video
Take Current Frame of the Video
Detect Moving Objects
Classify into Bus or Car
Car
Bus
Classify into Different Makes
Swift
Alto
City
Read Next Frame of the Video
Is Frame Empty?
NO
YES
End

7. Results

Detect Moving Object:-

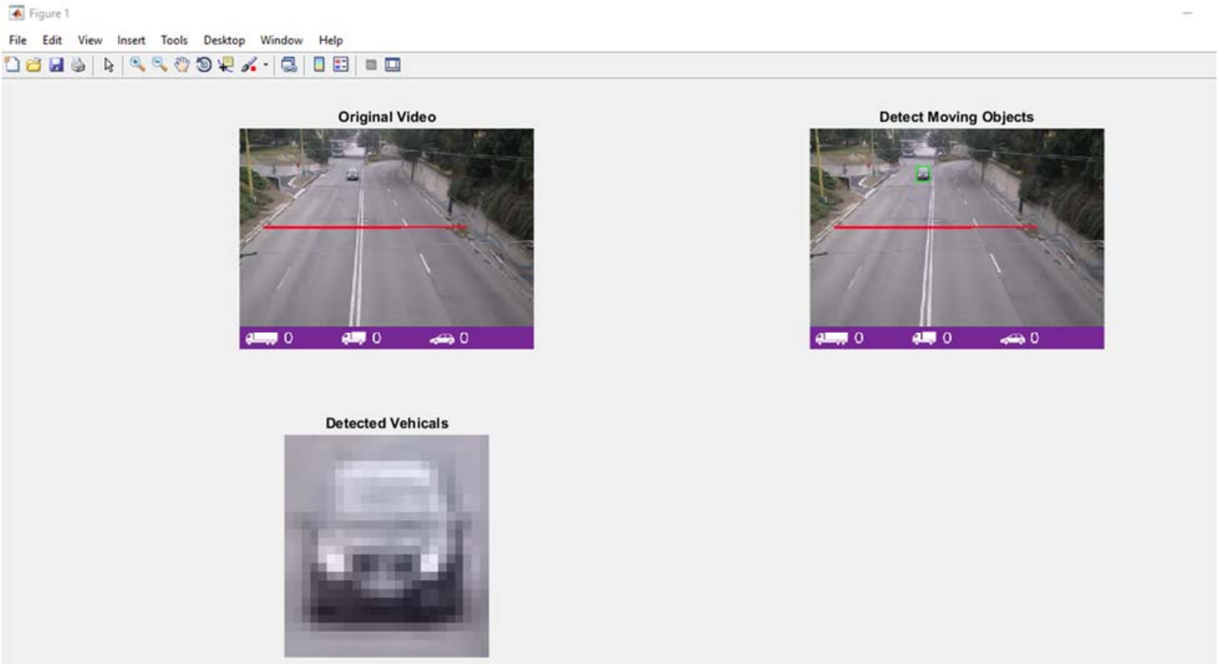


Fig 1. Detect Moving Object

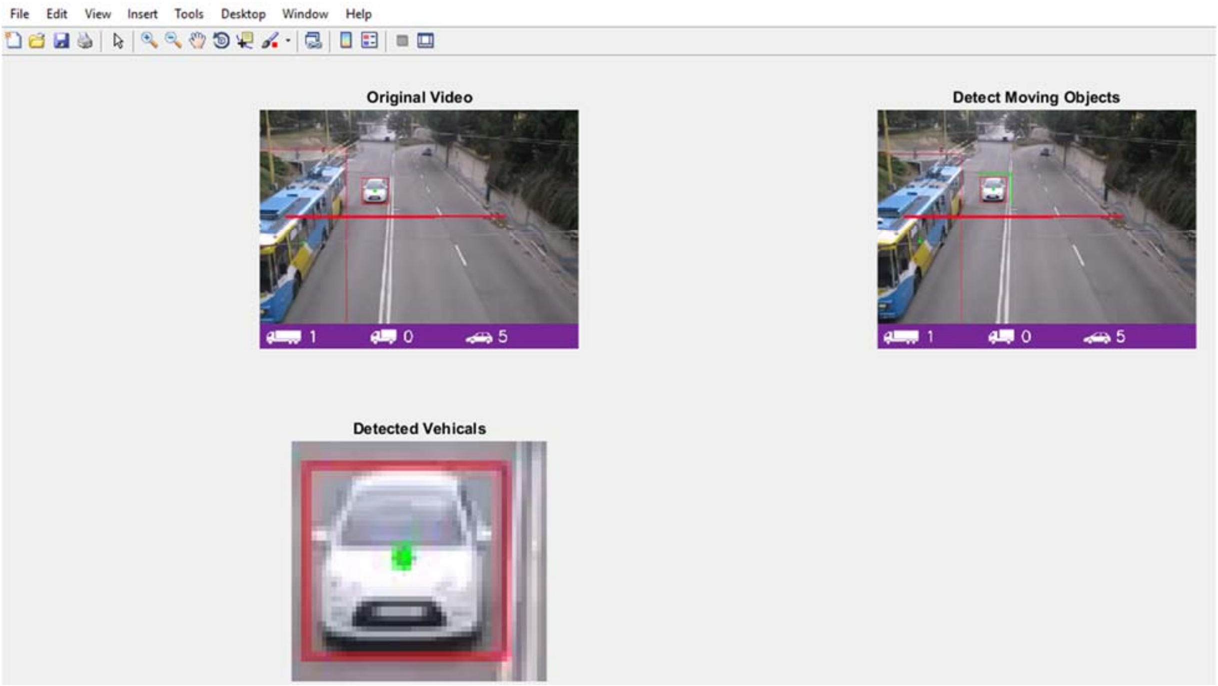


Fig 2. Detect Moving object

Classify into Car or Bus:-

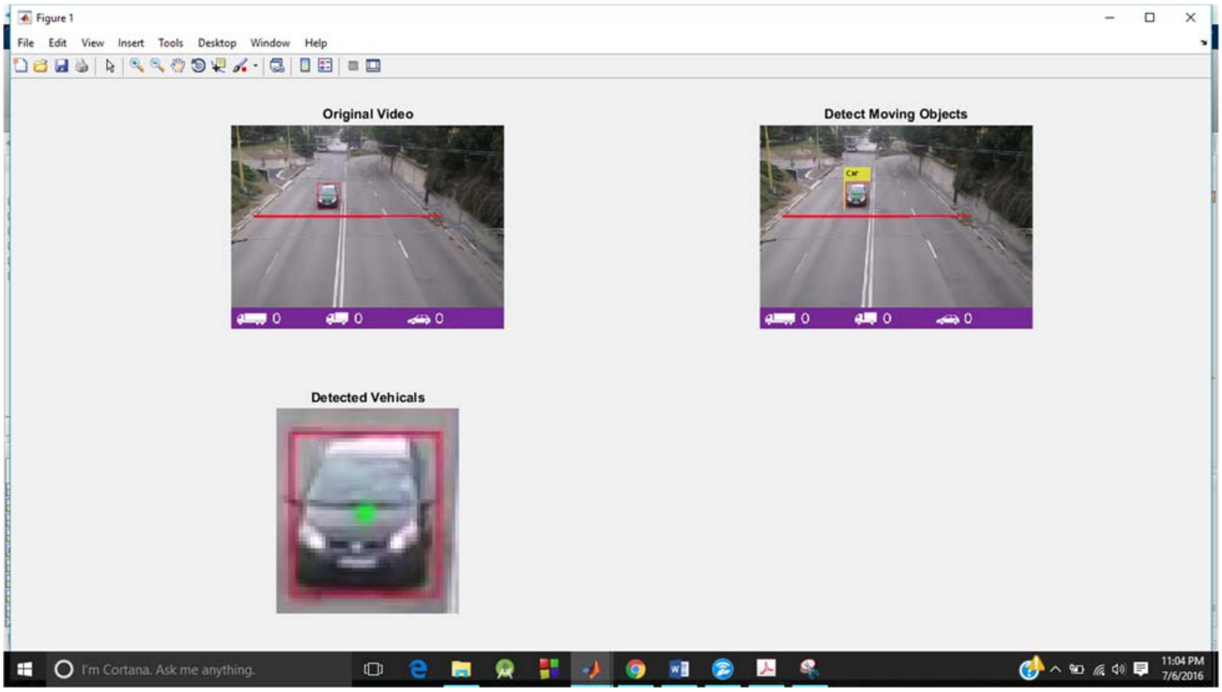


Fig 3. Classify into Bus or Car

Classify Car into Different Makes:-

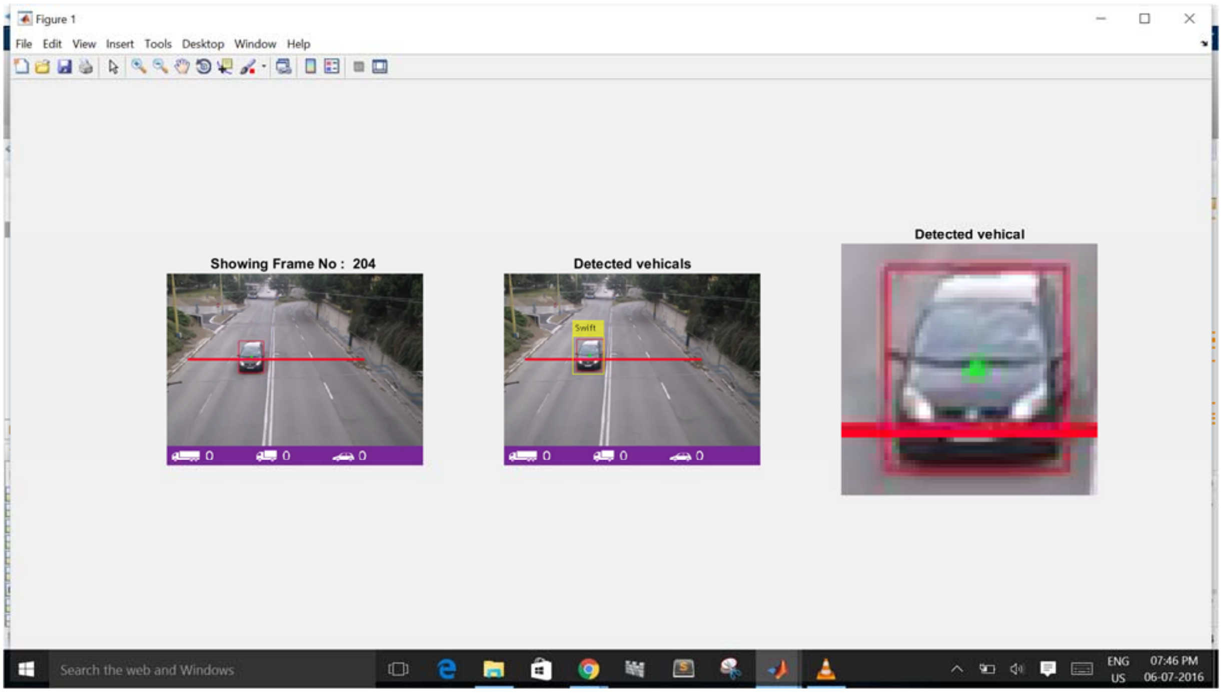


Fig 4. Classify car as Swift

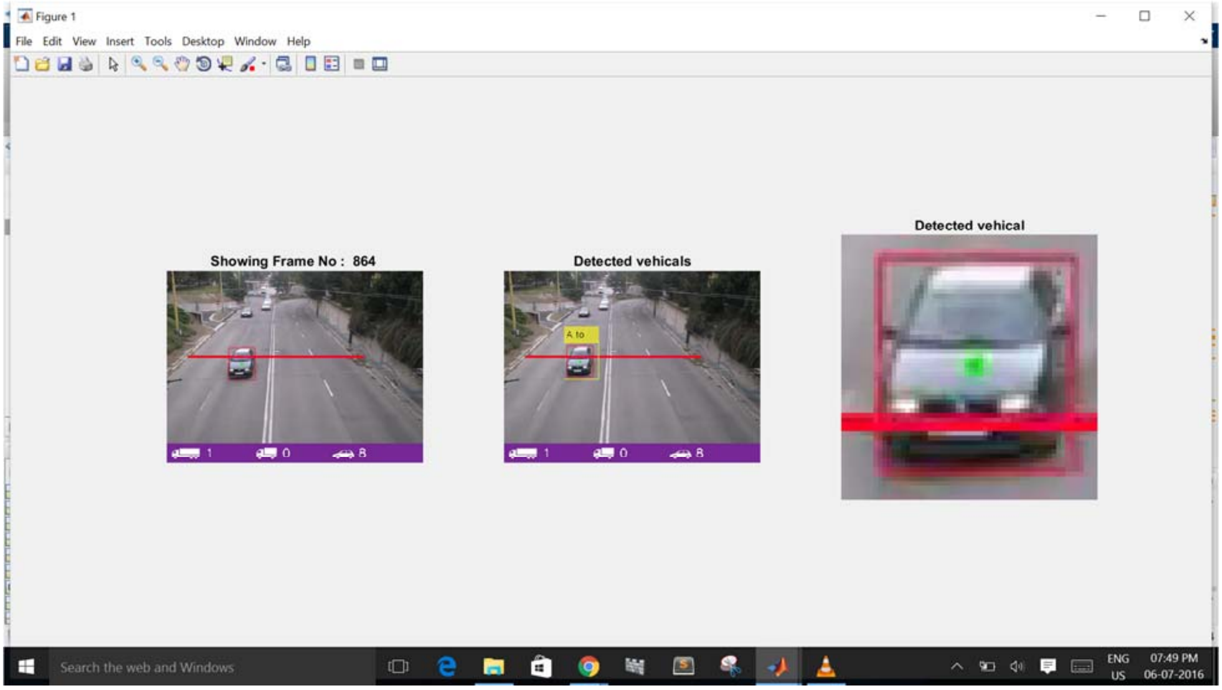


Fig 5. Classify car as Alto

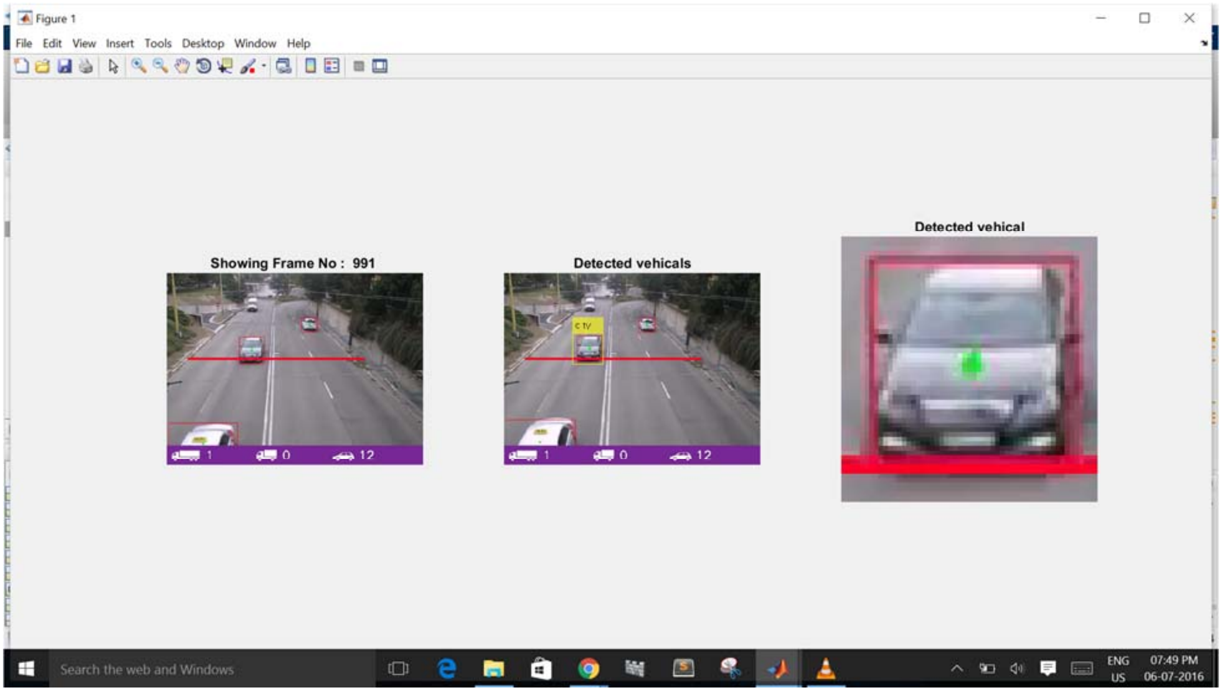


Fig 6. Classify car as City

Count the bbox:-

Fig 7. Count the bboxes (All frmaes)

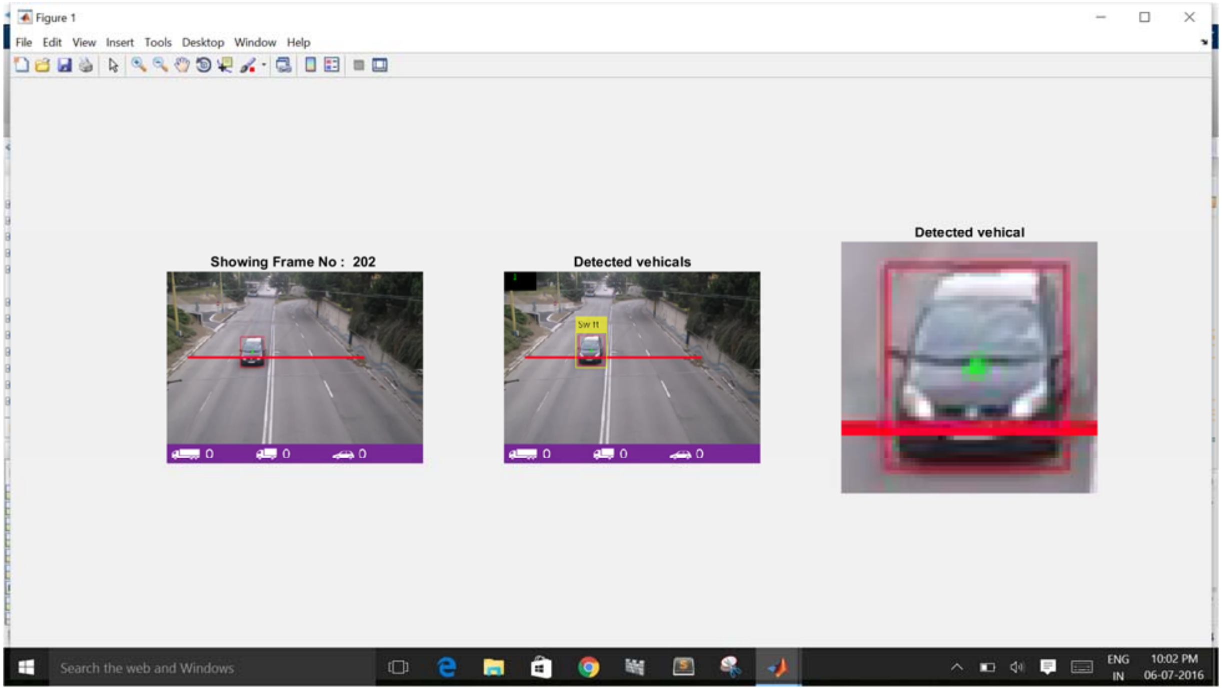


Fig 8. Count the bboxes in a Frame

Cropped images Dataset:-

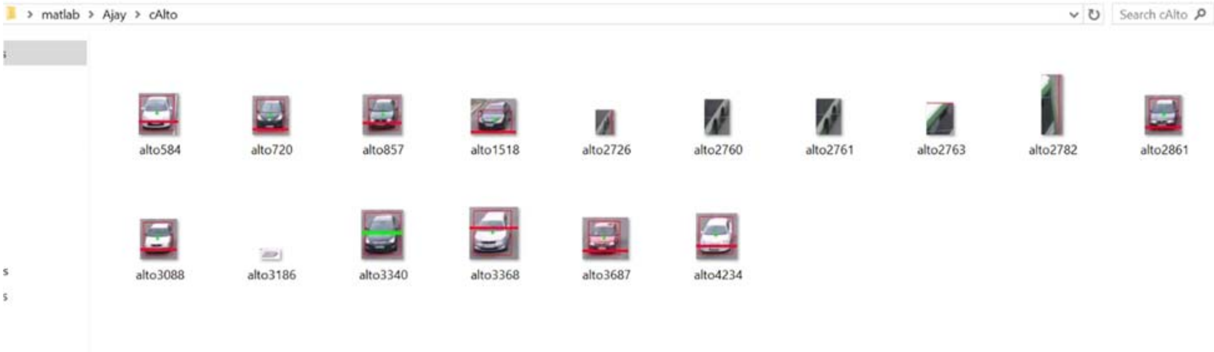


Fig 9. Detected Alto Cars

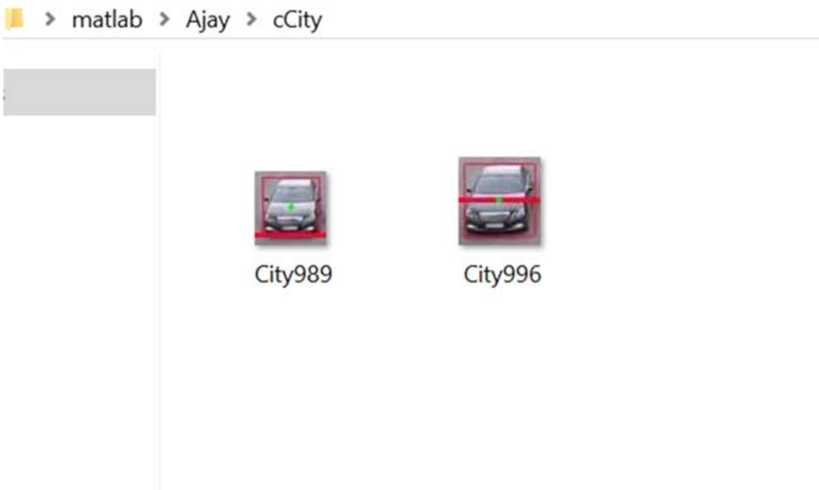


Fig 10. Detected City Cars

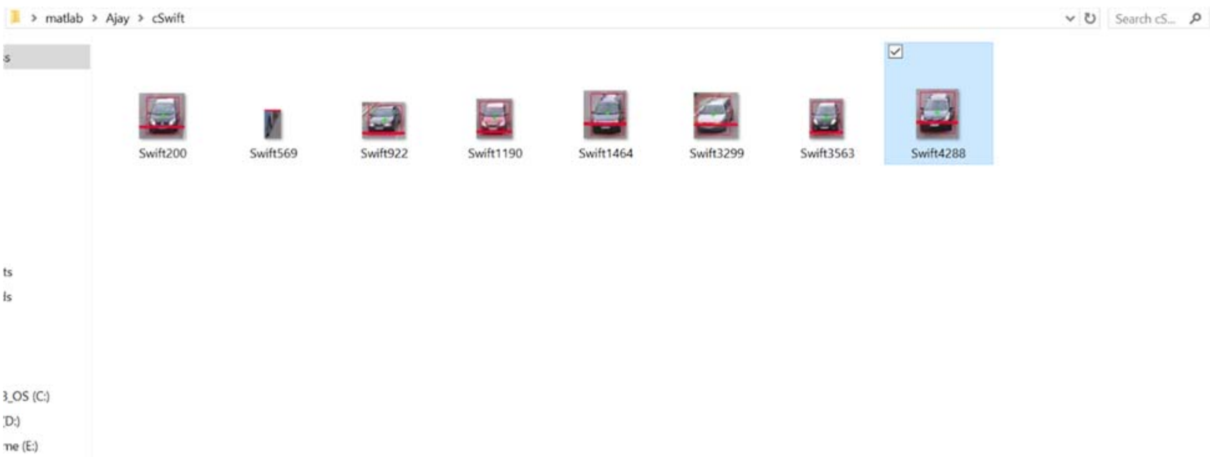


Fig 11. Detected Swift Cars

8. Conclusion and Future Scope

8.1 Conclusion: -

In this project method for detecting moving objects and classify these detected objects as in Car or Bus discussed. Further classify the detected Car into different makes when the position of camera is stationary. This model is implemented in matlab because the library of functions in matlab is huge and it has various methods of image and video processing. To detect the moving objects we use optical flow technique in which we estimate the velocity of the object and detect moving object. It also uses morphological analysis, morphological erode and blob analysis to remove noise and detect the moving objects. It works well and detect moving objects also but it shows the wrong results when there is a shadow of the moving object then it detects that shadow also. To classify objects between Car and Bus we use a threshold ratio value. It also give good results but not hundred percent accurate. It calculate ratio between the area of the blob of the moving objects and the area of the bounding box that is created over the detected object. Further to classify car into the different makes we use bagOfFeature in which we store features of the various makes of car and when we detect any car in video, we draw a bounding box over the object and using imcrop and the parameters of bbox to crop that image, then match the features of that image from previously store features. For which image set the number of feature matching is maximum result is that make of that detected car. So we need a fine data set which contains images of that car, it also depends upon position of the camera.For counting the image we have to use any tracking algorithm. If we count all the bboxes then one object will count multiple times because that object will detect more than one frame. So we need accuracy in these approaches.

8.2 Future Scope: -

In future work the proposed techniques can be extended and the problem arise (that the optical flow detect the shadow) should be correct. Problem in accuracy of classify detected objects as Car or Bus should be increased. Problem arises, when we further classify detected car into different makes, as there is no perfect definition of fine data set and it also depends upon the position of the camera so it also should be improved. Problem in counting that if we use an algorithm of moving object detection then it count the same object multiple times. So this problem should be sort out. But if we use an algorithm for moving object tracking then counting result can be come accurate but we want to develop an algorithm for counting if we use detection algorithm not tracking algorithm. Moreover, it remains a challenge to utilize the capabilities of the proposed algorithm to other kind of machine vision problems, such as security, remote sensing, ship surveillance etc.

References

[1]. Rupali Patra, "Statistical Approach for Detection of Vehicle in Heavy Traffic", Report of National Institute of Technology, Rourkela, June-July 2012.

[2]. Ghada S. Moussa, "Vehicle Type Classification with Geometric and Appearance Attributes", International Journal of Civil, Environmental, Structural, Construction and Architectural Engineering Vol:8, No:3, 2014.

[3]. Matlab Documentation.URL:- http://in.mathworks.com/

[4]. Abhishek R.Lahudkar, Akshay A.Shirsat, Sumant S. Landge, Dr. Mrs. Arati J. Vyavahare , " Traffic Estimation Using Image Processing", International Journal of Engineering Technology, Management and Applied Sciences, April 2015, Volume 3 Issue 4, ISSN 2349-4476.

[5]. Lucky Kodwani, "Automatic Vehicle Detection, Tracking and Recognition of License Plate In Real Time Videos", National Institute of Technology, Rourkela, May-June 2013.

[6]. Greg Welch, Gary Bishop, "An Introduction to the Kalman Filter", University of North Carolina at Chapel Hill, 2001.