Vehicle Detection for Smart Traffic Control System

Özlem ALTINAY
504181224

Fatma Betül YAZICI
504191517

***Abstract*— Growing population and increased vehicles lead to the overcrowded to highways. Vehicle detection is the important technology in traffic management. Computer Vision based techniques are more suitable because these systems do not disturb traffic while installation and they are easy to modify. In this paper we use computer vision based system for vehicle detection and counting. The system is implemented using OpenCV image development kits and experimental results are demonstrated from real-time image taken from single camera. This traffic counting process has been developed by image filtering, image binary and simple thresholding methods. After detecting vehicles with image processing with a Convolutional Neural Network is trained and accuracy of system is calculated.**

Keywords— Traffic jam, image processing, congestion avoidance, traffic density, classification, smart cities

I. INTRODUCTION

Recently, traffic congestion has become a universal problem for most big cities. The main reasons for this traffic congestion are increasing vehicle numbers, economic development and population growth. Traffic light signals that have pre-programmed are not a good solution to the increase in population and number of vehicles in traffic. Traffic light signals that have fixed timings does not pay attention the real-time road conditions and resulting in the traffic congestion.

For the solution of traffic jam, track the traffic changes and respond appropriately in the real time is very significant .There are a number of technologies that are being used to detect the traffic congestion. One such approach is vehicle detection techniques based on image processing. We have in this paper identified the image processing technique and based on the density of the vehicles traffic signal is controlled.

II. RELATED WORK

There have been different studies conducted on traffic management for a smart city. Most of them have been implemented by Image Processing techniques.

Dakshayani Ijeri propose a system that calculates the traffic density with current time density comparing the reference image against captured image[1]. This project aims to prevent heavy traffic in highways. Moreover, for implementing this project following steps must be considered: 1) image acquisition 2) Image enrichment 3) RGB to grayscale transformation and 4) morphological operations. At first, video of highway is captured by a web camera has been installed in highway. Then, the film comes in the form of consecutive frames and each frame is compared with the first frame. After the end, the number of cars in highways is specified.

Vikramaditya Dangi, Amol Parab, Kshitij Pawar & S.S Rathod propose the practice to implement an intelligent traffic system using real time image processing[2]. The image frames from a camera are analyzed using various edge detection and object counting methods to obtain the most efficient technique. Then, the number of vehicles at the intersection is counted and traffic is efficiently managed. The paper also proposes to implement a real-time emergency vehicle detection system. In case an emergency vehicle is detected, the lane is given priority over all the others. The key point of this paper is the technique which is used for edge detection. The authors have given the comparison of various edge detection techniques and conclude that canny edge detection is the best method for edge detection.

Prashant Jadhav, developed a method for estimating the traffic using image processing[3]. This is done by using the camera images captured from the highway and videos taken are converted to the image sequences. Each image is processed separately and the number of cars has been counted. If the number of cars exceeds a specific threshold, warning of heavy traffic will be shown automatically. At first, film of a lane is captured by a camera and images are taken. Then these images are efficiently processed to know the traffic density. According to the processed data from MATLAB, the controller will send the command to the traffic LEDs to show particular time on the signal to manage traffic.

In section III proposed method will be explained, in section IV enhancement of proposed method will be presented and section V is conclusion.

III. PROPOSED METHOD

A. OVERVIEW

In this work, video is used as input, it is converted to individual image frames which is called frame pre-processed. After pre-processing, background subtraction is used to detect moving objects and with morphological operations image is filtered out from noise. With contour detection, white places on black background is detected. In presented work, OpenCV library is used for image signal processing. Block diagrams of proposed method and morphological process are shown in Fig. 1 and Fig. 2, respectively. After vehicle detection process, traffic capacity of the road is calculated.

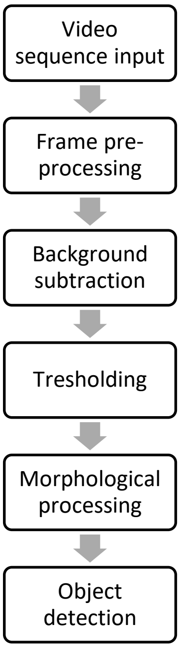


Fig. 1. Overview of the object detection.

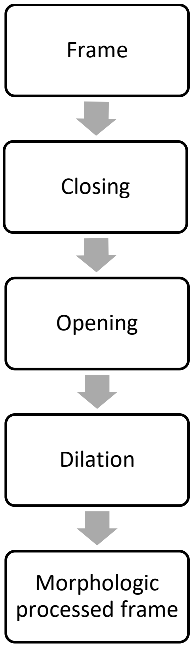


Fig. 2. Overview of the morphological processing

B. BACKGROUND SUBTRACTION and TRESHOLDING

Conventionally, assuming that the background is stationary, then the moving object can be determined by taking the difference between the background image and the input image. Background subtraction finds moving objects information by subtracting background model.

Background modeling consists of two main steps:

1. Background Initialization;
2. Background Update.

In the first step, an initial model of the background is computed, while in the second step that model is updated in order to adapt to possible changes in the scene. Background subtraction process diagram is shown in Fig. 3. In presented work, 500 steps are taken to train background subtraction and as threshold value 240 gray level is used. Original frame, background subtracted frame and tresholded frame can be seen in Fig 4. (a), (b) and (c), respectively.

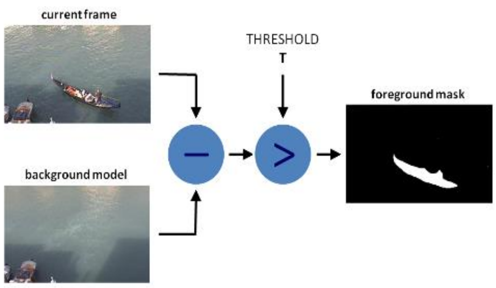
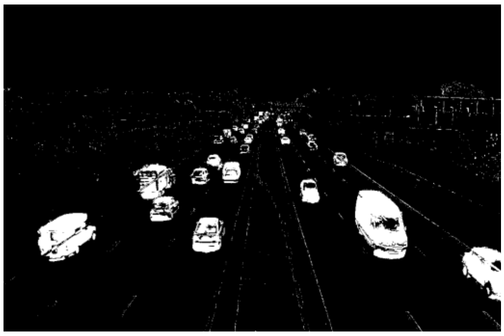


Fig. 3 Background subtraction process

a.Original frame

b.Background subtracted frame



c.Tresholding

Fig 4. Background subtraction from original frame

C. MORPHOLOGICAL PROCESSING

A traditional way to remove the noise regions is using the morphological operations to filter out smaller regions. It is normally performed on binary images. It needs two inputs,

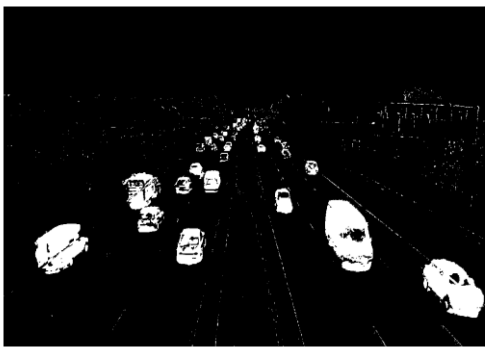


Fig 5.a. Closing

one is our original image, second one is called kernel which decides the nature of operation.

The opening operation is effective for eliminating the background noise and the close operation is effective for removing noise within the object region itself.

Dilation increases the white region in the image or size of foreground object increases. It is useful in joining broken parts of an object. Closing, opening and dilation processes can be seen in Fig. 5 (a), (b) and (c), respectively.

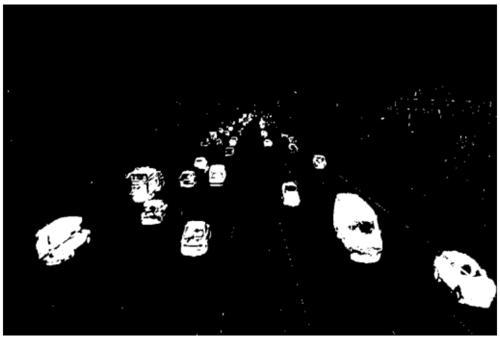


Fig. 5.b.Opening

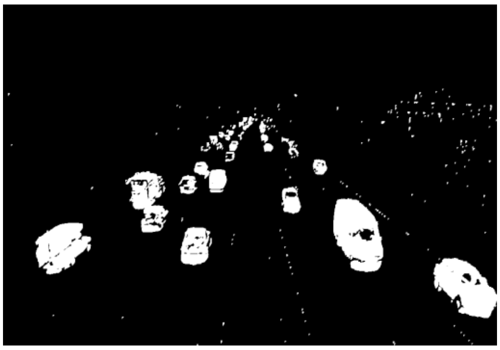


Fig. 5.c. Dilation

D. *VEHICLE DETECTION*

Contours can be explained simply as a curve joining all the continuous points (along the boundary), having same color or intensity. The contours are a useful tool for shape analysis and object detection and recognition. With contouring, corner coordinates of and centroid of an object can be obtained.

Objects are counted according to their contours and vehicle count is printed top of the frame. If contour area is bigger than 100 pixel^2, this object counted as vehicle. Detected vehicles are shown in Fig. 6.

Fig.6 Object detection

IV. ENHANCEMENT OF WORK -TRAFFIC CAPACITY COUNTING

A system is proposed in which calculates the traffic density with current time density by comparing the reference mask against captured image. The allocated signal period can be controlled according to the percentage of matching. It is done by comparing the white point counts of matched image and white point counts are depicted. Due to its high performance, canny edge detector operator is preferred. Percentage matching is done for various sample images and traffic time allocation. The algorithm of capacity counting is shown in Fig. 7.

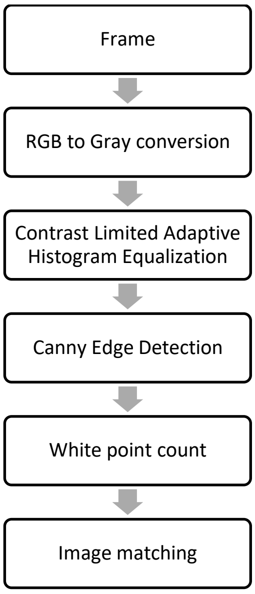


Fig. 7. Capacity counting overview

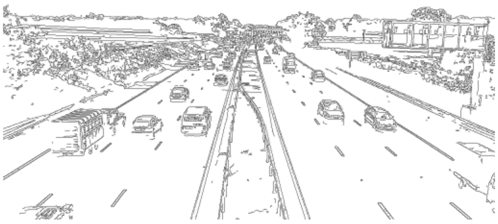
Contrast Limited Adaptive Histogram Equalization is a well-known block-based processing, and it can overcome the over amplification of noise problem in the homogeneous region of image with standard histogram equalization. CLAHE algorithm differs from standard HE in the respect that CLAHE operates on small regions in the image, called tiles, and computes several histograms, each corresponding to a distinct section of the image and use them to redistribute the lightness values of the image[4-6]. CLAHE is used in this work for noise reduction at night time.

The Canny operator is a sort of new edge detection operator. It has good performance of detecting edge, which has a wide application. The Canny operator edge detection is to search for the partial maximum value of image gradient. The gradient is counted by the derivative of Gauss filter. The Canny operator uses two thresholds to detect strong edge and weak edge respectively. And only when strong edge is connected with weak edge, weak edge will be contained in the output value.

Image frame of highway is captured from video stream is shown in Fig 8.a. RGB to gray conversion and CLAHE is applied to original frame is shown in Fig. 8.b. After CLAHE Canny edge detection is applied and frame is blurred. Using tresholding technique binary road image is created and

a.	Original frame

b.	CLAHE applied frame



c.	Canny edge detection

d.	Blurred image

e.	Tresholded binary image

f.	Masked image

Fig 8. Capacity calculation process

masked with predefined mask. This mask chosen for only left lane of the road. Calculating white points on this image percentage of capacity is calculated. All figures of this process can be seen in Fig.8 (c), (d), (e) and (f), respectively. Capacity calculated frame can be seen in Fig. 9.

Fig. 9. Capacity calculation result of the road

A. *CHALLENGES OF IMAGE PROCESSING AS CLASSIC METHOD*

To detect vehicles, pixel based filtered applied to image. But this is trial and error method. In farest points of road, vehicles getting smaller in image and may not be bigger than threshold pixel area.

Moving shadows and glares that are caused by different positions of sun and hence, the shadow of the vehicle and the vehicle count is taken or a shadow may count as vehicle.

Due to segmentation and morphological processes, two or more vehicle can be counted as one vehicle or one vehicle can be counted as two or more. All presented error situations are shown in Fig. 10.

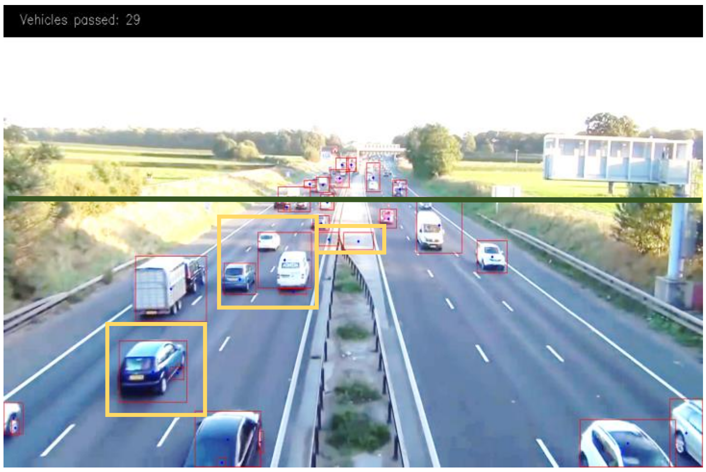


Fig. 10. Errors of classic method

The green line on the image represents the Region of Interest. Until this line some statistics are presented in Table I.

TABLE I. ERROR TYPES AND COUNTS

Error types	**Count**
2 vehicles counted as one vehicle	1
One vehicle and its parts are counted as vehicle	4
Shadows as vehicle	2
True detected	12
Total Vehicle	13

With all these possible disadvantages of image processing technique, to detect vehicles more precisely Convolutional Neural Networks are used.

V. ENHANCEMENT OF WORK – CONVOLUTIONAL NEURAL NETWORS

A Convolutional Neural Network has an input layer, output layer and hidden layers. The hidden layers consists of convolutional layer, RELU layer, pooling layer and fully connected layer. Figure 11 show how to build a simple convolutional neural network and train model KITTI dataset then test with detected images of vehicle and not vehicle.

Convolution layer is the layer that performs the main operation of CNN. This layer is used to extracting features from the input image. The input image applied a filter pixel block by pixel block .In RELU layer, applied an activation function onto feature maps to increase non-linearity in the network and removes negative values from an activation map by setting them zero. In pooling layer, progressively reduces the size of the input representation and it makes possible to detect objects in an image. In flattening layer, flatten the pooled feature map into a sequential column of numbers. In fully connected layer, the convolution and pooling layer merging with a fully connected layer in order to wrap up the complete CNN architecture. The purpose of fully connected layer is to use high-level features from the input image in order to classify classes based on train data [7].

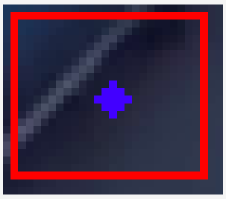
In this work, system is evaluated with a convolutional neural network model for classification of vehicle. CNN model is implemented using Python with Keras library. 1531 data were used to train the model. This data is the data set of KITTI dataset. Table II shows descriptive properties of the training data. Table III shows example detected vehicle and not vehicle image.

Detected images in the proposed system were used as test data. The output is a binary classification (vehicle / not vehicle).

TABLE II. DESCRIPTIVE PROPERTIES OF THE TRAINING DATA

Vehicle	Not Vehicle
262	1269

TABLE III. EXAMPLE DETECTED VEHICLE AND NOT VEHICLE IMAGE

Example Detected Vehicle Image	Example Detected Not Vehicle Image
	

Convolutional neural networks model type that is used is a sequential model. Sequential is the easiest way to build a model in Keras. It allows you to build a model layer by layer. 2D convolutional layer is added to process the 2D input images. The first argument passed to the Conv2D layer function is the number of output channels – in this case we have 8 output channels. The next input is the kernel size, which in this case we have chosen to be a 3×3 moving window. This model take input images of size 50x50x3 and the volume has dimensions 50x50x3 (width, height, RGB color scale). Then we apply the rectified linear unit activation function. Max Pooling is a pooling operation that calculates the maximum, or largest, value in each patch of each feature

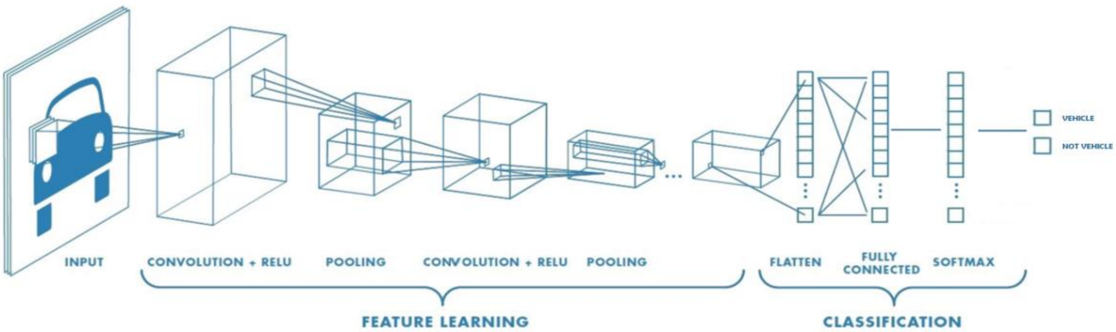


Figure 11. Convolutional Neural Network model for vehicle classification

map. Then, dense is the layer type we use in for our output layer. 128 nodes was specified, each activated by a ReLU function. The second is our soft-max classification, which is the size of the number of our classes (vehicle or not).

Compiling the model takes three parameters: optimizer, loss and metrics. The optimizer controls the learning rate. We use 'Adam' as our optimizer. Adam optimizer adjusts the learning rate throughout training. After 10 epochs, we have 99.35% train accuracy and 72.22% accuracy on our test set. Table IV shows every epoch accuracy result.

TABLE IV. ACCURACY RESULTS

Epoch	Accuracy
1	0.8537
2	0.9589
3	0.9634
4	0.9680
5	0.9791
6	0.9804
7	0.9909
8	0.9941
9	0.9935
10	0.9935

VI. CONCLUSION

In this work, first image processing technique is evaluated to detect vehicles on the road. To detect accuracy of classic approach a CNN is trained with KITTI dataset to detect vehicles. The success of the vehicles detected according to the model is 72.22%.

REFERENCES

[1] Ijeri, Dakshayani & Gulaganji, Amar & Mandewali, Abhishek & Bannad, Pavitra & Aiholli, Jyoti "A Survey on Intelligent Traffic Control System Using Image Processing", 2019 International Conference on Signal Processing and Communication

[2] Ms.Pallavi Choudekar,Ajay Kumar Garg Engineering College, Department of electrical and elec-tronics Ghaziabad,UP ,India,Ms.SAYANTI BANERJEE,Ajay Kumar Garg Engineering College, Department of electrical and electronics,Ghaziabad,UP ,India,Prof.M.K.MUJU,Ajay Kumar Garg Engineering College, Department of Mechanical,Ghaziabad,UP ,India, ‖ Real Time Traffic Light Control Using Image Processing ‖ , Indian Journal of Computer Science and Engineering (IJCSE), ISSN : 0976-5166 Vol. 2 No. 1

[3] Prof. Prashant Jadhav, Pratiksha Kelkar, Kunal Patil, Snehal Thorat, "Smart Traffic Control System Using Image Processing" 2016, Volume 03, Issue 03, Mar2016, International Research Journal of Engineering and Technology

[4] Zuiderveld, K. Contrast Limited Adaptive Histogram Equalization. Academic Press Inc.,1994.

[5] Pisano, E.; Zong, S.;Hemminger, B.; DeLuca,M.; Johnston, R; Muller, K.; Braeuning, M.; Pizer, S. Contrast Limited Adaptive Histogram Equalization Image Processing to Improve the Detection of Simulated

[6] Jintasuttisak, T.; Intajag, S. Color Retinex Image Enhancement by Rayleigh Contrast Limited Histogram Equalization. International Conference on Control, Automation and Systems.2014,10,692-697.

[7] Kurniawan, J., Syahra, S. G., & Dewa, C. K. "Traffic Congestion Detection: Learning from CCTV Monitoring Images using Convolutional Neural Network" 2018, Procedia computer science, 144, 291-297.