IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)
An International Open Access, Peer-reviewed, Refereed Journal

VEHICLE DETECTION AND CLASSIFICATION SYSTEM USING MOBILENET

Ms.R.Vaishnavi[1,*], Dr.A.Peter Soosai Anandaraj[2], Mr.K.Muthu Ramu[3], Dr.B.Guru Prakash[4] and Ms.R.Gowthami[5]

[1] Assistant Professor, Department of Computer Science and Engineering, MNSK College of Engineering, Pudukkotttai, 622305, India
[2]Associate Professor, Department of Computer Science and Engineering, Veltech Rangarajan Dr.Sagunthala R&D Institute of Science and Technology, Chennai, 600062, India
[3] HOD, Department of Computer Science and Engineering, Karaikudi Institute of Technology, Karaikudi, 630307, India
[4] Professor, Department of Computer Science and Engineering, Sethu Institute of Technology, Viruthunagar, 626115, India
[5]Operation Manager, IT Operations, Karunya Info Tech Solutions, Madurai-625016, India
[*]Corresponding Author: Ms,R.Vaishnavi. Email: vaishudharan24@gmail.com

Abstract — Video-based vehicle detection and classification is one of the important components for the intelligent transport systems by using deep learning. The objective of this work is to detect and recognize the vehicle with respect to its color, speed and count. The existing method includes video detection and video recognition and counting using convention method. The proposed method uses a special kind of deep learning architecture named as Mobilenet for vehicle detection and classification whereas the existing system used the conventional method for detection and recognition with deep feature learning. This model is pretrained on Common Object in Context (COCO) dataset for detection and classification of vehicles. In this work three main category of vehicles such as bus, car and truck has been taken for processing. The SSD framework how significant potential in the field of traffic estimation. The experimental results demonstrate a good performance. The overall detection Accuracy for vehicle classification using Mobilenet based Single Shot Detection (SSD) is given as 97.7%.

Keywords — Mobilenet, Single Shot Detection, COCO,Vehicle,Classification.

I. Introduction

Vehicle identification [1] and type play important roles in traffic tracking and control. Vehicle detection is the monitoring of an object in traffic video using CCTV, which is typically used to identify objects from images. The implementation of detection and classification of cars is extensive. To identify and classify them, a range of item detection methods are possible. Vehicle recognition may employ a variety of models, including those based on object detection, SVM class, and picture segmentation. There are numerous types of algorithms that are used for the primary goal of car detection and categorization. We primarily employ object recognition and categorization methods that are based on probabilistic, non-probabilistic, or square distance. Support Vector Machine (SVM), co-forest, k-nearest neighbor, and other categorization techniques are used. A neural network can also be used to learn and resolve complex non-linear patterns. By differentiating between the object to be noticed and other items, item detection via CCTV is achieved. The detection processes used by HOG and Viola Jones both make use of a current database. Both positive and negative data are present in this database. A positive database is a collection of information that contains the item to be detected, whereas a negative database does not.

II. Related Works

A tracking system that makes use of classification label data from a deep learning detection approach is employed to associate the various things. [2] Together with the position and look of the item. Due to limited perspectives, traditional methods of automobile identification and classification usually provide coarse-grained findings. Vehicle recognition and counting in transit security recordings are crucial for applications, as demonstrated by Deep Learning's most recent successes [3]. [4] Spanning from detection of traffic jams to incident and car identity. MobileNets [5] are built on a streamlined design that builds light weight deep neural networks using depth wise separable convolutions. We present two basic global hyper parameters for successfully balancing latency and precision. Vehicle classification and counting [6] in road traffic has numerous uses in transportation planning. Train state-of-the-art Convolutional Neural Network (CNN)-based object detection models for various vehicle classifications.

III.System Design

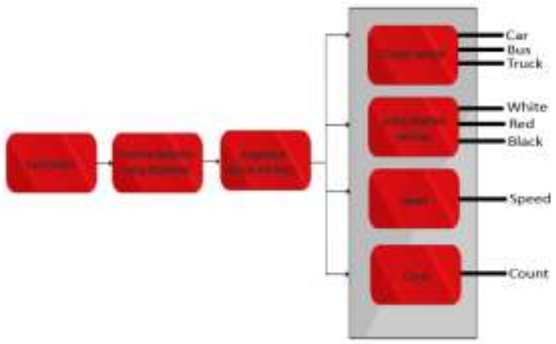

Fig.1.Detection and classification vehicle using mobilenet

The diagram above depicts the general suggested structure. It is made up of two main components: car detection and vehicle classification. The car classification is further defined by its hue, speed, and the number of vehicles counted with Mobilenet SSD.

A.Vehicle Detection Based On Mobilenet

The Single Shot Detector (SSD) [3] algorithm is used to recognise the cars, which is a commonly used approach for recognising objects using a single deep neural network [3]. Traditional car identification is divided into two parts: detection and categorization of movements.

Fig.2. Vehicle detection

The SSD algorithm integrates motion detection and categorization tasks into a unified structure and is learned by directly extracting vehicle features from vehicle data sets. It should be noted that features are useful in object identification and categorization tasks.

B.Mobilenet Architecture

Mobilenet [4] is built on fundamental levels of depth-independent filters. Following the structure of the mobilenet network, descriptions of the two model shrinking hyperparameters width multiplier and resolution multiplier are provided.i) Depth Wise Separable Convolution

Each filter that can be applied to each input channel is referred to as a depth wise convolutional filter. It is possible to write it as

$$G_{k,l,m} = \sum_{i,j,m} K_{i,j,m,n} \cdot F_{k+i-1,l+j-1,m} \quad (1)$$

Standard convolutions have the computation cost

$$D_k . D_k . M . N . D_F . D_F \quad (2)$$

M number of input channels, N the number of output channels. The kernel size D_k x D_k and the feature map size D_F x D_F.

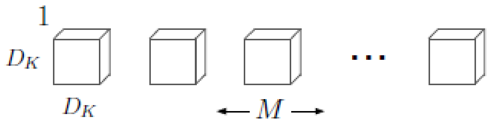

Fig.3.Depth wise convolutional filters

ii) Point Wise Convolution Filter

The result of the depth wise layer is then combined using point wise convolution, a basic 1x1convolution.

$$\widehat{G}_{k,l,m} = \sum_{i,j,m} K_{i,j,m,n} \cdot F_{k+i-1,l+j-1,m} \qquad (3)$$

Where K is a depth-wise convolutional kernel with dimensionsD_k x D_k x M, and the m^{th} filter in K is applied to the m^{th} channel in F to generate the m^{th} channel of the filtered output feature map$\widehat{G}$.The processing expense of point convolution is

$$\widehat{D}_k . D_k . M . N . D_F . D_F \qquad (4)$$

There are M incoming channels and N outgoing channels. The kernel is $\widehat{D}_k$ x D_k, and the feature map is D_F x D_F.

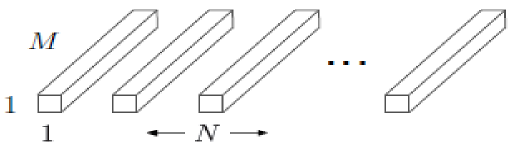

Fig.4.Depth wise convolutional filters

$$D_k . D_k . M . N . D_F . D_F + M . N . D_F . D_F \qquad (5)$$

Which is the sum of the depth and 1x1 point convolutions. When we describe convolution as a two-step procedure of filtering and merging, we get a computation decrease of:

$$\frac{D_k . D_k . M . N . D_F . D_F + M . N . D_F . D_F}{D_k . D_k . M . N . D_F . D_F} = \frac{1}{N} + \frac{1}{D^2{}_K} \qquad (6)$$

CNetwork Structure And Training

There are 22 levels in mobilenet. The region produced by the first convolutional layer could be [224x224x12].

In the ConvNet architecture, the max pooling layer is introduced at regular intervals between consecutive Convolution layers.

The Pooling layer applies the MAX function to each depth segment of the input and resizes it spatially. The picture size is reduced to 112x112x128 after the second convolution layer, followed by a MAX pooling procedure. This is done until the final convolution layer's result is 14x14x512.

As a result, their activations will be calculated using a matrix operation followed by a bias shift.In a multi-class issue, the final layer, "Softmax," gives decimal odds to each class. Softmax is applied just before the output layer via the neural network layer.

Type / Stride	Filter Shape	Input Size
Conv / s2	$3 \times 3 \times 3 \times 32$	$224 \times 224 \times 3$
Conv dw / s1	$3 \times 3 \times 32$ dw	$112 \times 112 \times 32$
Conv / s1	$1 \times 1 \times 32 \times 64$	$112 \times 112 \times 32$
Conv dw / s2	$3 \times 3 \times 64$ dw	$112 \times 112 \times 64$
Conv / s1	$1 \times 1 \times 64 \times 128$	$56 \times 56 \times 64$
Conv dw / s1	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 128$	$56 \times 56 \times 128$
Conv dw / s2	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 256$	$28 \times 28 \times 128$
Conv dw / s1	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 256$	$28 \times 28 \times 256$
Conv dw / s2	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 512$	$14 \times 14 \times 256$
$5\times$ Conv dw / s1	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
$5\times$ Conv / s1	$1 \times 1 \times 512 \times 512$	$14 \times 14 \times 512$
Conv dw / s2	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
Conv / s1	$1 \times 1 \times 512 \times 1024$	$7 \times 7 \times 512$
Conv dw / s2	$3 \times 3 \times 1024$ dw	$7 \times 7 \times 1024$
Conv / s1	$1 \times 1 \times 1024 \times 1024$	$7 \times 7 \times 1024$
Avg Pool / s1	Pool 7×7	$7 \times 7 \times 1024$
FC / s1	1024×1000	$1 \times 1 \times 1024$
Softmax / s1	Classifier	$1 \times 1 \times 3$

Table 1: Mobilenet body architecture

MobileNet is an option because it is more effective than any other network design currently in use. SSD is a cutting-edge object detection framework that employs a deep neural network to forecast numerous bounding boxes for various object classifications. The MobileNet provides a deep learning-based [5] vehicle classification technique that is both quick and effective.

Fig.5. Vehicle classification

C.Vehicle Counting

The ROI line technique is used for vehicle counting [6]. The ROI line is a measure that divides the picture into two sections. When vehicles are near to each other in traffic, there is a danger that two adjacent vehicles will be tallied as one, reducing the accuracy of the answer. It is possible to circumvent this by employing a precise object segmentation algorithm. The count is raised for each car that enters the frame and passes the ROI line. These ROI-based techniques can successfully conduct traffic counting if the following stages are followed:

1. On the original picture, draw a line. (ROI line)

2. To determine whether a car item is within the range of a line, compute the inner product of the vector.

3. Determine an object's prior and present position values with regard to a line.

4. If the prior position and present value of the vehicle object are not the same, raise the corresponding vehicle class counter.

$$\mathrm{Area} = (\mathrm{x}_{\mathrm{end}} - \mathrm{x}_{\mathrm{start}}) * (\mathrm{y}_{\mathrm{obj}} - \mathrm{y}_{\mathrm{start}}) \qquad (7)$$

$\mathrm{S}(\mathrm{x}_{\mathrm{start}}, \mathrm{y}_{\mathrm{start}})$ Starting point of line, at $\mathrm{E}(\mathrm{x}_{\mathrm{end}}, \mathrm{y}_{\mathrm{end}})$

Ending point of line and centroid box $\mathrm{P}_{\mathrm{obj}}(\mathrm{x}_{\mathrm{obj}}, \mathrm{y}_{\mathrm{obj}})$

Fig.6. Vehicle counting

D.Vehicle Color Detection

Color detection [7] is extremely sensitive to light conditions and object direction, which means that various angles of the same object may reflect varying amounts of light. The RGB color space parameters of an item can change significantly depending on light and orientation. In order to correctly extract the hue. A variable is used to determine the size of markers.

$$m_{ji} = \sum_{x,y}(array\ \ (x,y).x^{i}.y^{i}) \qquad (8)$$

So, the area *S* and center coordinates $(\bar{x}, \bar{y})$ are calculated as follows

$$\mathrm{S}= m_{00} \qquad (9)$$

$$\bar{x} = \frac{m_{10}}{m_{00}}\ , \bar{y} = \frac{m_{01}}{m_{00}} \qquad (10)$$

m_{00} Center coordinates of x and y axis, m_{10} the coordinates x axis. m_{01} Represents the coordinates of y axis.

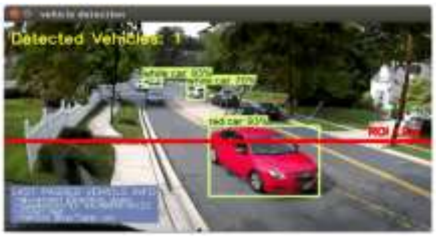

Fig.7. Vehicle color detection

E.Speed Of The Vehicle

In order to estimate the pixel coordinates of the cars in the frame, the speed [8] of the classified vehicles in the movie was calculated from the corner point of the bounding box. A vehicle whose coordinates are recognised in the first frame will have its coordinates in the second frame compared. The calculated values will be utilised to calculate the distance between the two photos using the Euclidean Distance technique.

$$d(p,q) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2} \qquad (11)$$

Where d is the computed distance, x_1= is the detected vehicle's horizontal coordinate on the first frame, y_1= is the detected vehicle's horizontal coordinate on the second frame, and x_2= is the detected vehicle's vertical coordinate on the first frame. y_2= is the vertical coordinate of the witnessed vehicle in the second frame.

The calculate speed can be determined using the equation below after determining the distance using the technique.

$$v = (\frac{d}{t} * 0.036) + x \qquad (12)$$

Where v= estimated speed (km/hr.), d= computed distance (cm), t= duration between the two frames (sec), x= speed of the vehicle containing the camera (km/hr).

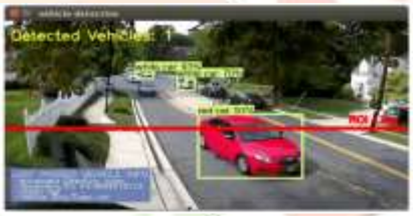

Fig.8. Vehicle speed

IV. Experimental Results And Discussion

This part contains descriptions of various instances. The classification of vehicles can be used to accomplish assessment and research. Some test videos were gathered in order to evaluate the suggested technique. PyCharm is used to carry out the coding.

Expected / Actual	Car	Bus	Truck
Car	196	2	2
Bus	3	195	2
Truck	2	4	194

Table 2: Confusion matrix

The acquired findings are investigated in order to comprehend classification performance. The collected findings validated the effectiveness of Mobilenet-based SSD.

Method	Accuracy (%)	Recall (%)	Precision (%)	F-measure (%)
Proposed	98.3	97.5	97.5	97.5
Existing	96.21	95.14	95.52	94.41

Table 3: The Performance of vehicle classification

Fig.9. Comparison chart of vehicle classification.

According to the obtained vehicle classification findings in Fig 5.1, the suggested method's accuracy, recall, precision, and f-measure were approximately 98.3%, 97.5%, 97.5%, and 97.5%, respectively.

V. Conclusion and Future Work

Following the study of the findings, the created algorithms enable the solution of the examined issue in real-time under various monitoring circumstances. According to Mobilenet, the total detection rate for car classification is 97.7%. Vehicle classification precision will increase in the future. The system's efficiency can be enhanced in the future.

References

[1]Ross Girshick, Jeff Donahue “Region-Based Convolutional Networks for Accurate Object Detection and Segmentation” IEEE Transactions on Pattern Analysis and Machine Intelligence, Tpami.2015.2437384, 2015.

[2]Hui-Lee Ooi,Guillaume-Alexandre Bilodeau, Nicolas Saunier,“Multiple Object Tracking In Urban Traffic Scenes With A Multiclass Object Detector”, Preprint,Arxiv:1809.02073v1,2018.

[3]Minglan Sheng, Chunfang Liu, Qi Zhang, Lu Lou, Yu Zheng ,“ Vehicle Detection And Classification Using Convolutional Neural Networks” In Proceedings On IEEE 7th Data Driven Control And Learning Systems Conference.2018.

[4] Qiaoqian Chen, Na Huang, Jieming Zhou, Zhao Tan, “An Ssd Algorithm Based On Vehicle Counting Method” ,2018.

[5] Andrew G. Howard, Menglong Zhu, Bo Chen Mobilenets: Efficient Convolutional Neural Networks For Mobile Vision Applications”, Arxiv Preprint, Arxiv: 1704.04861v1, 2017.

[6] Mayank Singh Chauhan, Arshdeep Singh Mansi Khemka,“Embedded Cnn Based Vehicle Classification and Counting In Non-Laned Road Traffic”, Arxiv Preprint, Arxiv: 1901.06358v1, 2019

[7] Deep Convolutional Neural Networks For Pedestrian Detection, “Deep Convolutional Neural Networks For Pedestrian Detection” In Proceedings Of Preprint Submitted To Elsevier Journal Of Signal Processing.2016.

[8] Yiren Zhou, Hossein Nejati, Thanh-Toan Do, “Image-Based Vehicle Analysis Using Deep Neural Network: A Systematic Study”, IEEE Transaction On Vol-9, Pp-5090, Issue 16.2016.

[9] Yuan Yuan, “An Incremental Framework for Video-Based Traffic Sign Detection, Tracking, and Recognition”, IEEE Transactions on Intelligent Transportation Systems, Vol. 18, No. 7, 2017.

[10] Keji Zheng, Wei Qi Yan, And Parma Nand, “Video Dynamics Detection Using Deep Neural Network”, IEEE Transactions On Emerging Topics In Computational Intelligence, Vol 17, No8, 2017.