Volume 10, Issue 1
February 2021

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

Heavy-Vehicle Classification and Detection using Deep-Learning models featuring Transfer-Learning Techniques

[1] V. Sowmya and [2] Dr. R. Radha

[1]Research Scholar, Research Dept. of Computer Science, SDNBV College for Women, University of Madras, Chennai, India
[2]Associate Professor, Research Dept. of Computer Science, SDNBV College for Women, Chennai, India
E-mail: [1]v.sowmy81@yahoo.in, [2]radhasundar1993@gmail.com

ABSTRACT

Vehicle Classification and Detection is one of the principal challenging tasks for autonomous vehicles as computational intelligence requires perpetually pioneering technology for the development of most advanced systems. The traditional vehicle classification and detection methods are inaccurate, time-intensive, and render low classification rates for diverse target models. With the focus on the aforementioned-shortcomings for developing an effective model utilizing the benefits of cutting-edge techniques of the deep learning models, this paper proposes an efficacious framework for real-time vehicle classification and detection based on Inceptionv3 and YOLOv3 algorithm. In this work, we have considered two classes of heavy vehicles such as buses, trucks for classification, and detection. Initially, a pre-trained inceptionv3 architecture is employed as a base network for vehicle classification. In the subsequent layers, Transfer learning and Data augmentation techniques are incorporated to avoid over-fitting besides augmenting the training speed. A classifier is built to identify the vehicle. Finally, a fine-tuning YOLOv3 algorithm is implemented for the detection of the vehicle. Experimental results on our custom dataset and CIFAR dataset demonstrate that the proposed algorithm has achieved greater detection accuracy of 95.14% mAP in comparison to other state-of-approaches.

Keywords: *Deep-Learning, Data Augmentation, Inceptionv3, Transfer-Learning, Vehicle Classification, Vehicle Detection, YOLOv3.*

1. INTRODUCTION

Over recent years, scholars in a growing trend commenced progressively research efforts for vehicle testing and the evolution of driving support technology. Vehicle detection using the machine vision framework is a focus area in the computer vision domain. Presently, many scholars have employed image processing, machine learning, and pattern recognition for vehicle detection and accomplished good results contributing to R&D and engineering efforts. [1] proposes a method for detecting obstacles in vehicle detection by application of vision and lidar point cloud fusion. In the following process, it is mapped to get a separate region of interest (ROI). Further, the YOLO algorithm is applied to ROI to detect vehicles. [7] uses a back-propagation algorithm in vehicle detection to enhance the various parts of vehicle characteristics based on a deep dual-vehicle deformable part model. [3] proposes an improved YOLOv1 network for detecting objects. In this work, margin style is replaced with proportion style and append the spatial pyramid pooling (SPP) layer. Then, the inceptionv3 module is added with a 64x1x1 kernel to reduce the weight parameters. [4] uses a YOLOv3 algorithm to detect objects and the results achieve 98.14 mAP which is higher than other algorithms [5] uses a YOLOv3 algorithm for multi-label classification to detect tiny objects and also proves that it is faster than other algorithms.[6] utilize Faster RCNN algorithm for detecting the vehicle. Two public datasets such as MIT and Caltech vehicle databases were used. The results show that the deep neural network method achieves high efficiency and effectiveness. [2] employ a distant-infrared image vehicle detection algorithm during night time based on deep learning. Initially, the non-vehicle pixels are removed with visual saliency computation. Then, a vehicle candidate is generated based on prior data such as vehicle size and camera parameters. Finally, a classifier is trained with deep neural networks to detect vehicles at night-time.

In this paper, real-time vehicle classification and detection are proposed based on deep-learning models. A pre-trained inceptionv3 model and fine-tuned YOLOv3 algorithm are employed for vehicle classification and detecting heavy vehicles. An OpenCV algorithm is applied for testing a model in a real-time environment. The experiment results demonstrate that the proposed method achieves significant accuracy.

The rest of the paper is formatted as follows – Section 2 explains the overall framework of the model. Section 3 details the proposed methodology. Section 4 discusses the experimental analysis and results, and Section 5 concludes the paper.

2. ARCHITECTURE MODEL

The overall framework of the model is shown in Figure 1. In this work, we have considered two types of vehicles such as bus and truck, which contain 1500 images each. A pre-trained inceptionv3 model is implemented as a base network for vehicle classification. In the subsequent layers, Transfer learning and Data augmentation techniques are incorporated to avoid over-fitting besides augmenting the training speed. Categorical cross-entropy is used as an optimizer for identifying a vehicle. Finally, a fine-tuned YOLOv3 algorithm is applied for detecting the vehicle. An OpenCV algorithm is used for testing a model in the form of images and videos in a real-time environment.

ITEE Journal
Information Technology & Electrical Engineering
ISSN: - 2306-708X

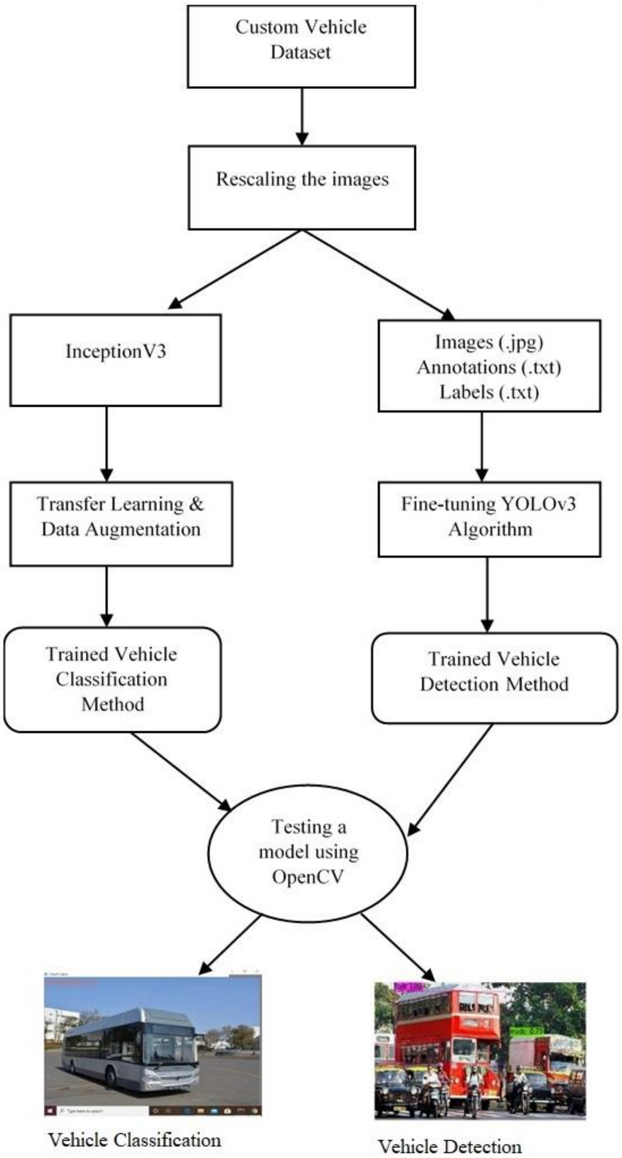


Figure 1. Heavy Vehicle Classification and Detection Framework

3. PROPOSED METHODOLOGY

3.1 InceptionV3 Model

The conception and implementation of InceptionV3 architecture are by Szegedy et al. in the year 2015. The original paper "Rethinking the InceptionV3 architecture for Computer Vision" laid the foundation for the advancement of this model. Inception v3 consists of 48 deep Convolution layers. This helps to load a pre-trained version of the network with more than a million images from the image net database [9]. The pre-trained network contains 1000 classes of images. Thus, the network has a significant characteristic representation. The input size image for the network is 299 by 299. In this paper, Inceptionv3 pre-trained model is implemented. Inceptionv3 pre-trained model is used as a base model and also to identify the vehicle.

3.2 Transfer Learning and Data Augmentation

Transfer learning and Data augmentation techniques are incorporated to avoid over-fitting besides augmenting the training speed in this proposed model. The application of Transfer Learning is the customization of the pre-trained model by optimizing the fine-tuning parameters while Image data augmentation regenerates picture files into preprocessed tensors to train the proposed model by using various parameters such as shifting, flipping, and rotating images.

3.3. YOLOv3 Model

The YOLOv3 model is the fastest and precise object detection method. It accurately distinguishes the objects by applying logistic classifiers compared to the softmax approach used by YOLOv2. This makes us capable of multi-label classifications. The YOLOv3 uses Darknet53 as a feature extractor, which has 53 convolutional layers, each followed by a batch normalization layer and Leaky ReLU activation. The downsample of the feature maps is performed by the convolutional layer along with stride two without the pooling techniques. This leads to the prevention of loss of low-level features, more precise than the Darknet19 used by v2, and achieving greater accuracy. Also, a small object detector detects objects of small size present in the image, thereby overcoming the limitation of detection in the v1 model. The following formula describes how the network output is transformed to obtain bounding box predictions:

$$b_{x1} = \sigma(t_{x1}) + c_{x1}$$
$$b_{y1} = \sigma(t_{y1}) + c_{y1}$$
$$b_{w1} = p_{w1} e^{t_{w1}}$$
$$b_{h1} = p_{h1} e^{t_{h1}}$$
$$\Pr(object) * IOU(b, object) = \sigma(t_1) \tag{1}$$

Where,

$t_{x1}, t_{y1}, t_{w1}, t_{h1}$ are predictions of YOLO

c_{x1}, c_{y1} denotes top-left coordinates of the grid cell

p_{w1}, p_{h1} denotes width and height of the anchor

$b_{x1}, b_{y1}, b_{w1}, b_{h1}$ are the forecasted boundary box

$\sigma(t_1)$ denotes box confidence score

By using the above network function, the custom dataset is trained, and the average loss is 0.0297. The proposed algorithm is detailed as follows:

Step 1: Collect the Data from various sources such as real-time video, icrawlers, and the internet.

Step 2: Pre-process the Data:

- Cropping and Resizing the images using the Computer Vision technique into standard width and height (600x600).
- Create Annotation file (.xml to .txt) for each image.

Step 3: Train the custom Dataset using Deep Learning models separately:

ITEE Journal
Information Technology & Electrical Engineering
ISSN: - 2306-708X

- Heavy-vehicle classification using fine-tuned inception V3 model Parameters with the learning rate of 0.0001 and 0.001.
- Heavy-vehicle detection using fine-tuned YOLOv3 parameters with the learning rate of 0.0001 and 0.001.

Step 4: Data Augmentation features and appropriate filter weights are staged for training the dataset and network.
Step 5: The dataset is trained separately for both classification and Detection of Heavy-vehicle.
Step 6: Test the proposed model with CIFAR Dataset.
Step 7: Finally, Test the algorithm in a real-time environment to visualize the test-images using Computer-Vision Technique.

4. EXPERIMENTAL ANALYSIS AND RESULTS

The vehicle classification experiment is tested in Anaconda3, which uses Keras as the development environment. The vehicle detection model is trained on Google Colab which is equipped with a Tesla K80 GPU for faster and efficient training of the network. The system environment for simulation is as follows: CPU - Intel Core i5-8265 1.80 GHz, SSD-512, RAM - 8 Gb.

4.1 Dataset

The custom vehicle dataset contains 3050 images of buses and trucks. The images of the vehicles are populated from icrawlers. Some of the images are taken from real-time traffic video which includes all kinds of vehicles such as a car, bus, motorbike, etc., and also downloaded from the internet. Further, rescale the images with width and height as 600x600. This dataset is classified into a training set and a testing set. The Vehicle training dataset is organized as two separate entities with a ratio of 7:3 where the numbers of images in the training and test datasets are 2135 and 915, respectively.

4.2 Training a Dataset for Heavy-Vehicle Classification

The pre-trained Inception V3 is used as a base network for vehicle classification. The image attribute is 224x224x3 for input size, and the final feature map attribute for InceptionV3 is 2048x5x5. For each model, the layers and feature maps are interconnected. Then, two sophisticatedly connected classifiers with a dropout of 0.5 on top of these layers, and categorical cross-entropy acts as an optimizer. Finally, train the custom vehicle training set from scratch, and the optimal model parameters of the model are obtained as the learning rate is 0.0001, the number of epochs is 20, the batch size is 20. The Feature Map is used as input in the Fully-Connection layer to get the classification results. To avoid over-fitting images, Data Augmentation and Transfer learning techniques are applied in this model. For Fine-tuning Inception V3, mixed8, and mixed9 layers are chosen among many layers that are used for feature extraction. Further, these layers are trained with two fully densely connected classifier layers with a dropout of 0.5%, and categorical cross-entropy is used as an optimizer. The training and validation accuracy of Fine-tuning Inceptionv3 is 99.33% and 98.87% respectively. Thus, the performance is improved, and overfitting is reduced as shown in Figure 2.

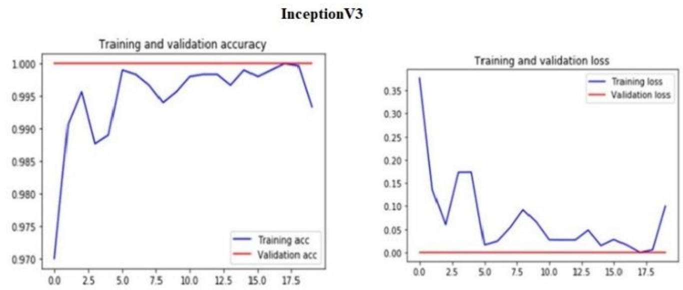
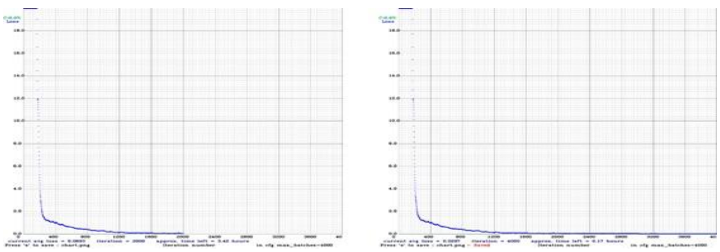


Figure 2. Training and Validation loss of the custom vehicle dataset

4.3 Training a Dataset for Heavy-Vehicle Detection

After creating an annotation file for each image, the images and their annotation files along with the label file are bound together. The enhancement of configuration is achieved by training the three YOLO layers using yolo.cfg. In a conventional procedure to training, each object is trained to a minimum of 2000 iterations. Hence, the dataset will be executed for 4000 iterations since there are two classes, i.e 2*2000=4000. For optimized speed of training, the attributes of batch and subdivisions are tuned to 64, and 8 respectively. For optimized speed of training, the attributes of batch and subdivisions are tuned to 64, and 8 respectively. The height and width attributes were tuned to 416x416 each for maximum speed and better accuracy of detection. The number of filters used in the convolution layer is set to 21 as the value relies on the total number of classes as filters = (classes + 5) *3. The total network training time with the above-mentioned configurations was approximately 5-6 hours. The algorithm uses a pre-trained network darknet53. conv.74. Finally, tune the YOLOv3 algorithm's network hyper-parameters. The batch normalization layer is frozen during the training process. Initially, the iterations are set to 2000 times with a learning rate of 0.0001. All parameters are frozen except the last 3 convolutional layers, and the weights are saved as shown in Figure 3(a). Again, trained the network for the next 2000 times with an initial learning rate is 0.001. Therefore, the final weights are saved and used for testing the model and also to evaluate the performance metrics. The average loss of YOLOv3 is 0.0297 as shown in Figure 3(b).



(a) (b)

Figure 3. Average loss of the custom vehicle dataset

Volume 10, Issue 1
February 2021

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

4.4 Performance Metrics

The parameters employed for testing the efficacy of the proposed model are mAP, IoU, Classification Report, and Confusion Matrix as shown in Figure 4 and Table 1 and 2. A confusion matrix is an interpretation of prediction results on a classification problem. The Average Precision of 99% and 91% of buses and trucks are achieved as shown in Figure 4. 95.14% mAP (Mean Average Precision) with 81.84% IoU (Intersection over Union) at 35 FPS (Frame per Second) speed with 1 second detection time as shown in Table 2. Evaluation analysis of the proposed model on Precision, Recall, and F1 score is briefly summarized in Table 1.

Table 1. Classification Report of the Proposed Model

Type	Precision	Recall	F1 Score
Vehicle Classification	0.98	0.96	0.97
Vehicle Detection	0.92	0.86	0.89

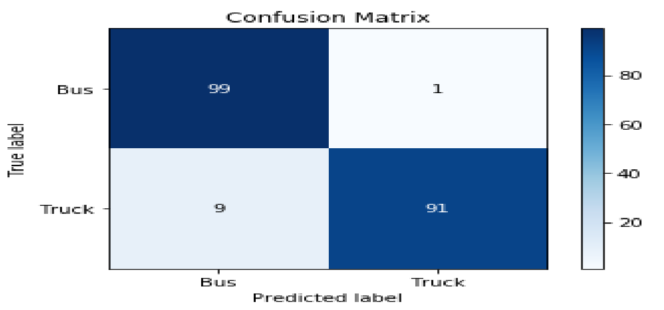


Figure 4. Confusion Matrix of the Proposed Model

Table 2. Testing Results of the Proposed Model

Indicator	mAP	IoU	Inference Time	Avg_FPS
Value	95.14%	81.84%	1s	35

4.4 Testing a model with CIFAR Dataset

CIFAR dataset is a public dataset for evaluating algorithms. It contains 600 images per class, which include annotation files. In this paper, 600 images are used for training the proposed network. This paper proposes a system that uses the Inceptionv3 and YOLOv3 that are trained on a custom dataset and transfers them to be trained on a CIFAR dataset. With relation to the initial training, Transfer Learning enables us to execute the learned features on the ImageNet dataset and modification these features. Table 3 shows the comparison of detection results using the CIFAR dataset.

Table 3. Comparison of results using CIFAR dataset

Method	Bus	Truck
Alexnet-CNN [10]	84.00%	95.90%
CNN [11]	71.72%	65.27%
Our Model	98.25%	97.62%

4.5 Comparative Testing of the Proposed model with Existing Methods

The proposed model is experimented with to ensure the optimal performance of the algorithm, and compared with the execution efficiency of other advanced methods is shown in Table 4. The classifier uses the Computer Vision technique for the testing cycle. The tracking feature is performed to further optimize the performance of the system. The classifier was tested on various traffic intersections in Chennai and Pune, as shown in Figure 5. The vehicle detection system is tested in real-time traffic densities in Chennai at various places.

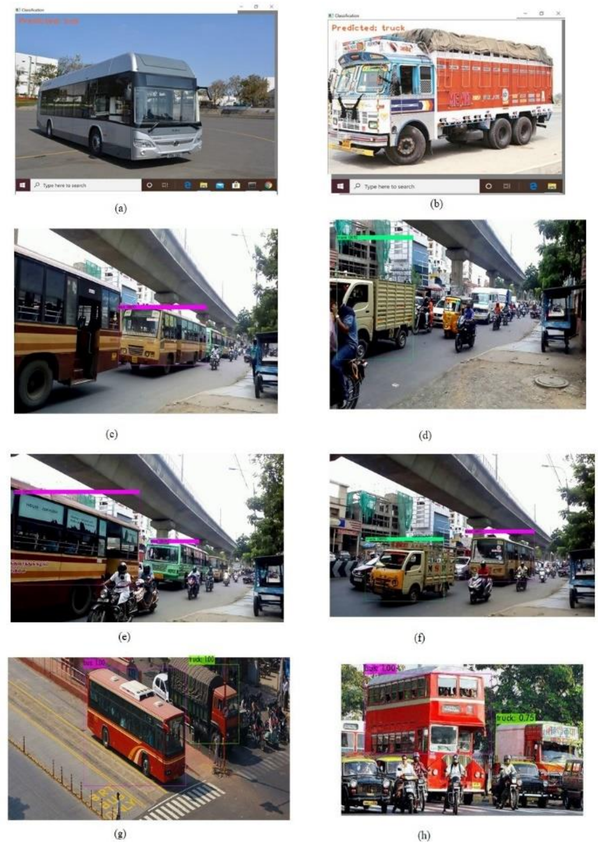


Figure 5. (a), (b) Heavy-Vehicle Classification using Inceptionv3; (c), (d), (e), (f) Heavy-Vehicle Detection using Yolov3 at various places in Chennai and (g), (h) in Pune.

Table 4. Overall Vehicle Detection results comparison

Models	Overall(mAP)
YOLO [1]	70.58%
Back-Propagation [7]	94.7%
VGG16 [6]	84.4%
DBN method [2]	92.3%
CNN [12]	93.8%
YOLOv3 [14]	84.96%
Our proposed model	95.14%

5. CONCLUSION

In this study, an optimized Deep-Learning framework for rapid and precise vehicle classification and detection is proposed. The effectiveness of the detection is achieved by the optimal utilization of the core parameters of the system's framework – the pre-trained Inceptionv3 architecture as a base network for identify the vehicle and integrating with the YOLOv3 algorithm for detecting the vehicles, besides enabling the Transfer-learning feature and Data Augmentation techniques to improve speed, accuracy, and reduction of over-fitting with a focus on fine-tuning the inefficiencies to secure optimal performance. In comparison with other models, the proposed model shows better results in both classification and detection. The findings demonstrate that the proposed network is simplistic and efficient. In the future, this paper will proceed with emerging algorithms for further enhancements to improve the classification and detection results of heavy vehicles.

REFERENCES

[1] Wang, Hai & Lou, Xinyu & Cai, Yingfeng & Li, Yicheng & Chen, Long.” Real-Time Vehicle Detection Algorithm Based on Vision and Lidar Point Cloud Fusion”. Journal of Sensors. 2019. 1-9.

[2] Yingfeng Cai, Xiaoqiang Sun, Hai Wang, Long Chen, Haobin Jiang, "Night-Time Vehicle Detection Algorithm Based on Visual Saliency and Deep Learning", Journal of Sensors, vol. 2016, Article ID 8046529, 7 pages, 2016.

[3] Tanvir Ahmad, Yinglong Ma, Muhammad Yahya, Belal Ahmad, Shah Nazir, Amin ul Haq, "Object Detection through Modified YOLO Neural Network", Scientific Programming, vol. 2020, Article ID 8403262, 10 pages, 2020.

[4] Omkar Masurekar, Omkar Jadhav, Prateek Kulkarni, Shubham Patil, “Real-Time Object Detection Using YOLOv3”, International Research Journal of Engineering and Technology, Volume. 07, 03, Mar 2020 .3764-3768.

[5] A. Vidyavani, K. Dheeraj, M. Rama Mohan Reddy, KH. Naveen Kumar, “Object Detection Method Based on YOLOv3 using Deep Learning Networks”, International Journal of Innovative Technology and Exploring Engineering, ISSN: 2278-3075, Volume-9 Issue-1, November 2019.

[6] Li Suhao, Lin Jinzhao, Li Guoquan, Bai Tong, Wang Huiqian, Pang Yu,” Vehicle type detection based on deep learning in traffic scene, Procedia Computer Science”, Volume 131,2018, Pages 564-572, ISSN 1877-0509.

[7] Yingfeng Cai, Ze Liu, Xiaoqiang Sun, Long Chen, Hai Wang, Yong Zhang, "Vehicle Detection Based on Deep Dual-Vehicle Deformable Part Models”, Journal of Sensors, Article ID 5627281, 10 pages, 2017.

[8] V.Sowmya, R. Radha. “Efficiency - Optimized Approach - Vehicle Classification Features Transfer Learning and Data Augmentation Utilizing Deep Convolutional Neural Networks”. International Journal of Applied Engineering Research, ISSN 0973-4562, Volume 15 (4), 2020. pp. 372-376

[9] ImageNet. http://www.imagenet.org.

[10] Neha Sharma, Vibhor Jain, Anju Mishra, “An Analysis Of Convolutional Neural Networks For Image Classification”, International Conference on Computational Intelligence and Data sciences”, Procedia Computer Science 132. 2018.pp 377–384.

[11] Mahbub Hussain, Jordan J. Bird, and Diego R. Faria. “A Study on CNN transfer learning for image classification”. UK Workshop on Computational Intelligence systems - Nottingham, United Kingdom. Volume 840, Springer, September 2018. pp 191-202.

[12] Wong Z.J., Goh V.T., Yap T.T.V., Ng H. (2020) Vehicle Classification using Convolutional Neural Network for Electronic Toll Collection. In: Alfred R., Lim Y., Haviluddin H., On C. (eds) Computational Science and Technology. Lecture Notes in Electrical Engineering, vol 603. Springer, Singapore.

[13] W. Maungmai and C. Nuthong, "Vehicle Classification with Deep Learning," IEEE 4th International Conference on Computer and Communication Systems (ICCCS), Singapore, 2019, pp. 294-298.

[14] K. Kim, P. Kim, Y. Chung, and D. Choi, "Performance Enhancement of YOLOv3 by Adding Prediction Layers with Spatial Pyramid Pooling for Vehicle Detection," 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), Auckland, New Zealand, 2018, pp. 1-6.

AUTHORS PROFILE

Ms. V. Sowmya is pursuing her Doctor of Philosophy in Computer Science at University of Madras, India. Her research interests include image processing, deep learning, machine learning and object detection.

Dr. R. Radha is an Associate Professor of Computer Science at SDNBVaishnav College for women affiliated to University of Madras. She holds a Ph.D. in Computer Science in the area of data mining. She has successfully guided 5 Masters’ dissertations and is supervising 6 Ph.D. students. Her areas of interest include machine learning, cloud computing, pattern recognition, data privacy.