VISION BASED ADVANCED DRIVER ASSISTANCE SYSTEM USING DEEP LEARNING

Bhadra Varma[1]
Robotics and Automation
College of Engineering
Trivandrum
bhadravarma@cet.ac.in

Sibin Sam[2]
Robotics and Automation
College of Engineering
Trivandrum
sibinsam@cet.ac.in

Linu Shine[3]
Faculty, Dept of ECE
College of Engineering
Trivandrum
linushine@cet.ac.in

***Abstract*— India is having one of the highest road casualty rates of almost nearly 1.5 lakh a year. The rate of road related deaths come down by a good margin if Advanced Driver Assistance System(ADAS) becomes mandatory for the vehicles. Since Indian car manufacturers focus on cost cutting, only Anti-lock Braking System(ABS) and airbags are offered as standard. With the advent of deep learning methods and computer vision, the use of an on-board camera and real time processing of captured images of road scenes makes it possible to achieve a wide range of ADAS features.**

In this paper a deep learning and computer vision based approach is used to develop a multi-functional ADAS system. The proposed system combines various objectives such as Vehicle Detection, Pedestrian Detection, Traffic Sign Board Detection, Traffic Light Detection and Blind-spot Vehicle Detection. The detection model is implemented using Tensorflow and various models are trained and tested using different detection algorithms and deep neural network(DNN) architectures. Networks such as SSD Inception, SSD Mobilenet, Faster RCNN Inception are trained on the same hardware set-up in order to perform a comparative study. The optimal model is selected based on a trade-off between detection time and accuracy. The model developed can detect 35 classes of objects that are commonly seen while driving a vehicle. The optimal model can be implemented on a suitable hardware and converted to a low-cost portable vehicle accessory.

Keywords - Advanced Driver Assistance System, Artificial Intelligence, Neural Networks, Object Detection

I. INTRODUCTION

India is currently home to world's fourth largest vehicle market. Of the vehicles sold in India passenger vehicles have an annual sale of 33,93,705 units. Since the competition in the vehicle market is high, the manufactures always try to cut the cost of the cars by offering the basic ADAS features like ABS and airbags. Since each advanced ADAS feature comes at a premium, they are offered on high end cars like Volvo, Mercedes, BMW etc. But majority of the customers prefer to buy cars within the range of 10 lakhs, ADAS systems are still considered as a luxury. With Ministry of Road Transport and Highways, India planning to make ADAS for all the passenger cars mandatory from 2022, it is obvious that the price of cars might increase. With the improvement in Machine Learning and the introduction of Deep Learning to solve real world problems, multiple features of ADAS can be achieved using a single pipeline model which runs on hardware which is capable of Graphics Processing Unit(GPU) acceleration.

This paper provide an in-depth description about an ADAS software model which is capable of detecting the vehicles, pedestrian, road signs, traffic lights using a single deep learning model. The model is trained using a large annotated dataset which includes all the objects of interest, so that the model learns to detect these objects on its own, from live video of road scene captured using an on-board camera. Tensorflow is used as the backbone of this work as it provides end to end open source deep learning platform with various day to day applications. Here Tensorflow Object Detection API is used to train and evaluate the model.

This research work also includes a detailed comparative study of various combinations of object detection architectures and image classifier networks. The object detection architectures studied are Faster Region Convolutional Neural Networks (F-RCNN) and Single Shot Dectector(SSD). Image recognition and classification networks studied are Inception v2 and Mobilenet v2.

II. RELATED WORK

Each objective of ADAS has been individually studied and developed by various researchers around the world over the past few years mainly based on machine learning. However works which bring several of the objectives together as a single ADAS module are comparatively less. The integration of several ADAS objectives using deep learning is the main objective of this work.

One of the earliest work on pedestrian detection in static images was done by Navaneet Dalal and Bill Triggs [1] dating back to 2005. Later on the work was expanded to include human detection in videos. Several works have been done in the area of pedestrian detection to improve the detection efficiency by introducing new or improved versions of current methods in different stages of detection, ie. feature extraction and classification. Hemmati et al. [2] introduced parallel and deep pipelined architecture of the HOG feature extractor and SVM classifier which has led to high throughput while lowering the memory utilization for the intermediate result. Tome. et al.[3] proposed a method for pedestrian detection based on Convolutional Neural Network which performs better than alternative approaches based on handcrafted and learned features at a reasonable computational complexity. In this method, general purpose CNNs like AlexNet and GoogLeNet are further fine tuned to detect pedestrians. This model is tested on Nvidia Jetson TX1 development board which is believed to be the brain of self-driving cars.

Tsai et al.[4] presents a new traffic sign content identification algorithm, which describes the detected road sign using centroid-to-contour (CtC) distances of the extracted traffic sign content. The first part of this algorithm performs a traffic sign content extraction procedure which extracts content of the detected traffic sign, and the latter employs traffic sign content description and classification algorithms which identifies the type of the extracted traffic sign content.

Jian-Gang Wang, Yu Pan et al.[5] in Appearance-Based Brake-Lights Recognition Using Deep Learning and Vehicle Detection proposes the detection and recognition of tail light signal which is important to prevent an autonomous vehicle from rear-end collisions or accidents. The approach is to detect the high mounted third brake light along with the tail light to positively predict whether the vehicle has applied brake or not. This eliminates the false detections even if the tail lights are turned on. Yeong-Kang Lai, Chu-Ying Ho et al.[6] in "Intelligent Vehicle Collision-Avoidance System with Deep Learning" proposed a vehicle detection using neural network MobileNet embedded to a mobile platform. The model is trained with images from MS COCO dataset. The model is ported to NVIDIA Jetson mobile platform for the real time detection. The model predicts the presence of cars and traffic lights with an average frame rate of 12 to 14 frames per second.

Ravi Kumar Satzoda, Sean Lee et al.[7] in Snap-DAS: A Vision-based Driver Assistance System on a Snapdragon Embedded Platform propose the use of mobile platforms to be used as a computational resource. A snapdragon 660 based embedded board is used to realise the ADAS system using a monocular camera. Lane Analysis using Selective Region-LASeR algorithm is employed for the detection and classification of ego lane and thereby providing drift warning.

The motivation behind the idea is the ever increasing rate of deaths on the roads due to accidents. Reports says that over 1 lakh people die in Indian roads every year. The major cause of these types of accidents are mainly due to negligent driving. A driver assistance system will provide a helping hand to drivers to monitor roads and alert them in case of an impending incident.

The use of deep learning for the system in this project is the novelty proposed in this paper. The detection models are implemented by deep learning methods using tensorflow. The detection architectures are based on Single Shot MultiBox Detectors(SSD) and Faster Region Convolution Neural Network(FRCNN). Deep Neural Network(DNN) detection models such as SSD Inception, SSD Mobilenet, Faster RCNN Inception are trained on the same hardware set-up in order to perform a comparative study.

III. Methodology

Software algorithms use the input from sensors to synthesize the environment surrounding a vehicle in real time. The proposed model utilizes video feed from the dashboard camera for analyzing the environment around ego vehicle. It issues suitable warnings to the driver about potential hazardous conditions from the environment. For example, a pedestrian is approaching dangerously in front of the vehicle then the system should warn the driver to apply brakes. The speed limits captured from the road signs are displayed on the external display of the ADAS.

A. System Description

The proposed ADAS consists of :

1) Vehicle Detection in the proposed ADAS system is capable of detecting vehicles present on the road such as cars, autorickshaws, motor bikes, trucks, buses etc. The model is trained with the images from all orientations of the vehicle so that the model is viewpoint invariant.
2) Pedestrian Detection is also an important feature of the proposed ADAS. The model is capable of detecting pedestrians from a distance thus alerting driver. Viewpoint variation is also not a problem for the pedestrian detection.
3) Traffic Sign Board Detection assists the driver by providing information about the traffic signs kept on the roads such as speed limits, turns and curves, other mandatory and cautionary road signs. The model detects and classifies 24 road signs which are commonly seen on Indian roads.
4) Traffic Light Detection alerts the driver from a distance about the presence of traffic signals and their condition, whether it is red, yellow or green so that the driver can adjust his speed accordingly. This helps in maintaining safe distance between vehicles and also prevents accidents.
5) Blind-spot Vehicle Detection gives the driver a view of the vehicles present in the blind-spot of ego-vehicle which are often not visible in the rear view mirrors. This can be achieved with the help of cameras suitably placed in the rear view mirror compartments.

B. Selection of Deep learning model

Deep Learning is a powerful machine learning tool that showed outstanding performance in many fields. One of the greatest success of Deep Learning is large scale object recognition with Convolutional Neural Networks (CNNs). CNNs' advantage lies in the fact that it is able to learn data representations directly from data using a hierarchical layer based structure. Due to the recent development in the field of deep learning various researchers come up with their own models for the classification and detection tasks. For ADAS the major criteria is the speed and accuracy of the model to detect an object and classify it accordingly. Since training of these deep neural networks from scratch is a cumbersome process which require the support of high end processors, GPU's and days of continuous training, the networks pre-trained on MS COCO dataset and KITTI vehicle detection dataset is incorporated in this work[8]. The pre-trained network retains all the features from the dataset it is initially trained on. In order to make the network suited for this particular application, network is again retrained with the locally obtained dataset so that the weights get automatically adjusted. This method is known as transfer learning and it takes comparatively lesser time to train a model in this way.

C. Dataset Generation

Deep learning based algorithms can learn features from a huge collection of data. The COCO pretrained models does not cover all the classes of interest needed for this work. To make the network suited for this application, it needs to learn the features of the classes that are not covered in COCO dataset. The locally obtained dataset for the retraining purpose is divided as follows.

- 21331 images were manually annotated to obtain a total of 36684 labels. These images are frames from the video footage taken in an ego car driven on Indian roads.
- Some classes from The Laboratory for Intelligent and Safe Automobiles(LISA) Traffic Sign Dataset [9] which are similar to the Indian road signs were included.
- 15000 vehicle and pedestrian images and their corresponding annotations from Udacity vehicle detection dataset [10].
- Selected 400/900 images from German Traffic Sign Detection Benchmark (GTSDB) dataset [11] which contain signs similar to Indian road signs.

The final dataset contains 1,34,826 labels split into train and test dataset in 80:20 percentage ratio.

D. Training the model

The video dataset collected from the roads using a camera is first split into frames. These frames are then annotated by manually drawing rectangular bounding boxes over the objects of interest using LabelImg software. The resulting extensible markup language(xml) files are first converted to comma separated value(csv) format. This is further converted to Tensorflow records(tf record) which is the accepted input format for training. The tensors are obtained from the labels and their corresponding annotated images. The final dataset given for training consists of a train tf record and a test tf record split in the ratio of 80:20 as per general dataset division criteria.

(a) Speed limit 30 and No Overtaking Road Signs

(b) Pedestrian and ZigZag Road Sign

(c) School Ahead, Car and Autorickshaw

Fig. 1: Annotation of Images using LabelImg Software

The hardware and software set-up on which the network is trained is given below:

- Processor : Intel® Xeon® W-2145 CPU frequency-3.70GHz
- RAM : 32 GB, DDR4, frequency-2666 MHz
- GPU : Nvidia Quadro P4000, Driver Version: 418.39
- CUDA : CUDA is NVIDIA's parallel computing architecture which increases the computing performance by harnessing the power of the GPU. NVIDIA® Cuda compiler driver, release 10.0, V10.0.130
- The NVIDIA® CUDA Deep Neural Network library (cuDNN) is a GPU-accelerated library of primitives for deep neural networks. cuDNN v7.4.2 for CUDA 10.0 is used.
- TensorFlow offers multiple levels of abstraction to build and train models by using the high-level Keras API. Tensorflow v1.13.1 and Tensorflow-gpu v1.13.1 is used for the purpose of training and evaluation. Tensorboard v1.13.1 is used to continuously monitor and visualise training and detection performance metrics.
- All the training and evaluation codes are executed with Python 2.7.12.

The models are trained to detect a total of 35 classes. The classes are primarily divided into vehicles, traffic signs, pedestrian and traffic lights. The detailed division of classes is given below:

- Speed Limit Signs : 20kmph, 30kmph, 40kmph, 50kmph, 60kmph, 70kmph, 80kmph
- Prohibitory signs : Stop, No-Overtaking, Keep-Left, No-U Turn, No-Left Turn, No-Right Turn
- Pedestrian
- Warning signs : Curve Right, Curve Left, Pedestrian Crossing, Signal Ahead, Turn Left, Turn Right, Junction Ahead, Curve Ahead
- Other road signs : Zig Zag Road, Bump Ahead, School, Gap In Median
- Vehicles : Autorickshaw, Bike, Bus, Car, Truck
- Traffic Lights : Traffic light, Traffic light-Green, Traffic light-Yellow, Traffic light-Red

The different models are individually trained for a total of 2 lakhs iterations for a comparative study of their performance metrics. The training is done for both the pretrained model as such and after performing hyper parameter tuning on the model such as scheduling of learning rate over the total iterations, changing score converter, using moving averages etc. After the completion of training the model is exported from the last checkpoint created.

IV. Experimental Results

With the dataset generated for 35 classes, several networks are trained for a comparative study of performance in order to choose the best model for the proposed ADAS. The following pre-trained models are chosen as the base check point to initialise the training process.

1) COCO SSD-Mobilenet V2[12][13]
2) COCO SSD-Inception V2[14]
3) COCO Faster RCNN-Inception V2[15][16]

The training of model is done with images having resolution of 704x480 px, 640x480 px, 1024x522 px, 1980x800 px and 1280x720 px. The training code normalises all the annotated images for training. Thus the final output model is independent of the resolution of input images. Python codes are developed for real-time capture of the road scene and for detection of objects of interest from a video sequence. The model is evaluated using COCO API tools for assessing the performance measures. Tensorflow Object Detection API uses mAP as an evaluation protocol to measure and compare the accuracy of object detection models. According to PASCAL metrics, mAP is calculated as the mean of average precision (AP) of all the object classes. Intersection over Union(IoU) is a useful measure which assists in evaluating the performance of object detectors. IoU signifies the similarity between the predicted region (bounding box) and ground truth. In other words, IoU indicates how good our prediction is as compared to the ground truth. The mAP values are evaluated for the validation dataset at 0.5 IoU.

A. Vehicle Detection

The proposed ADAS system is capable of detecting vehicles of five different classes. The vehicles detected are motorbikes, car, bus, trucks and autorickshaws. The model is trained with the images of the vehicle from all orientations. Thus the model can detect vehicles of the above mentioned classes from all directions. One of the problem faced while testing the model is that the model struggles to identify newer model of cars that are not in the training data.

B. Pedestrian Detection

One of the important factors to be considered while driving is to ensure the safety of pedestrians. The model can detect the presence of pedestrians in roads and it is trained in such a way that the prediction is not affected by scale variations. The pose of the pedestrian is also not a constraint for detection.

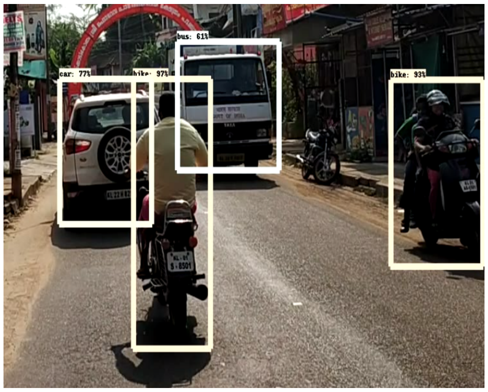

Fig. 2: Detection of vehicles using COCO SSD Mobilenet V2

Fig. 3: Detection of pedestrians using COCO FRCNN Inception V2

C. Traffic Sign Board Detection

Traffic sign boards gives information about road conditions so that the driver can take necessary precaution while driving. The proposed model is capable of detecting 32 Indian Traffic Road Signs placed on road. While running the model with test videos it is observed that Faster RCNN based networks are found out to be more effective in detecting road signs rather than SSD based networks. SSD based networks detects traffic sign boards when they are close to the camera frame. But in region proposal methods the sign boards are detected from a much farther distance.

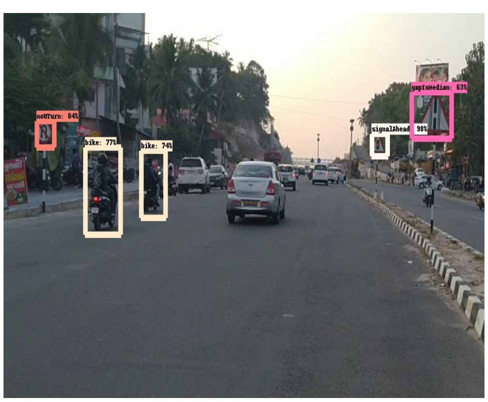

Fig. 4: Detected Traffic Sign Boards No U Turn, Gap in Median and Signal Ahead using COCO SSD Inception V2

D. Traffic Light Detection

Traffic lights are integral part in controlling the flow of traffic. The model proposed here is capable of detecting traffic lights from a distance so that the person who is driving can take necessary actions accordingly. Currently the model can detect traffic light Green, Yellow and Red. The model can also be modified to detect directional traffic lights such as Green Straight, Green Left, Red Right etc.

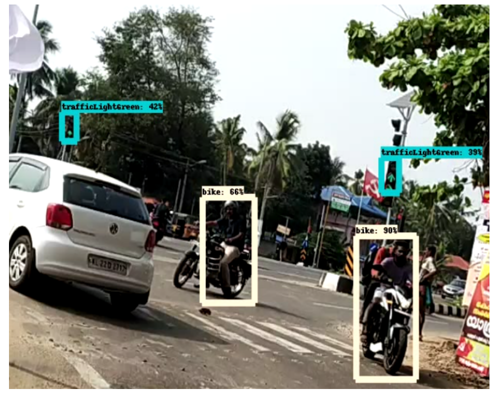

Fig. 5: Detection of traffic lights using COCO FRCNN Inception V2

E. Blind-spot Vehicle Detection

Vehicles present in the blind-spot of the ego vehicle are a major cause of accidents, since the driver is unaware of their presence. Small vehicles such as cars, two wheelers, autorickshaws present in the blind-spot can be detected from the model created. The driver is given an alert, thus the driver becomes aware of the presence of the vehicles mentioned above. This objective is achieved by placing a camera in the side view mirror fixtures. The image sequence from the camera is given to the detection program which is running along with the main program.

Fig. 6: Detection of vehicles in blindspot using COCO FRCNN Inception V2

F. Comparison of models trained

For the comparative study of the performance measures three detection networks are trained for two lakhs steps on the same hardware setup. The parameters compared are Classification Loss, Localisation Loss, Mean Average Precision(mAP) and Detection Time(ms). The comparison is given in Table I.

TABLE I: Comparison of models trained for 2 lakhs iteration on same hardware setup

Model	SSD InceptionV2 COCO	SSD MobilenetV2 COCO	FRCNN InceptionV2 COCO
Classification Loss	7.307	6.552	0.206
Localization Loss	0.958	1.032	0.097
Mean Average Precision(mAP) (%)	51.77	49.46	39.68
Detection Time (ms)	100-115	60-80	700-800

V. Conclusions and Future Scope

A. Summary

Deep learning based methods are the future of ADAS technologies. With the development of newer networks the accuracy and speed of detection improved compared to the earlier ones such as AlexNet, GoogleNet etc. In this work study of recent neural networks and their capability to function as a backbone of ADAS are tested. The major factor to be analysed while selecting a network for ADAS is the accuracy and time taken for detection. So a trade-off between the speed and accuracy is considered for the selection of the deep neural networks. Networks like SSD Moblilenet and SSD Inception have detection time of almost 60ms to 120ms but are prone to many erroneous detections. FRCNN based networks are more efficient in detection and they can detect and classify an object from a much farther distance than SSD based networks. Although false predictions are there in a minor manner, the major setback with these networks is the time taken for detection. FRCNN based networks takes about 700ms to 800ms for the detection.

The main limitation while using deep learning for implementing real life applications is the amount of data required for training the network. The collection of data and annotation is a tedious and time consuming process. Also the networks sometimes tend to predict the objects of interest erroneously. This may lead to confusion in drivers. While implementing the system for autonomous driving this erroneous prediction may lead to the generation of signals that create wrong decisions.

Fig. 7: Error in detection of traffic signs

B. Future Scope

The scope of ADAS is not limited to alerting the driver. With the world moving towards autonomous cars this work can be upgraded for use in self driving cars. The model can be trained for the detection of more than thirty five classes thus many vehicle classes and traffic signs present in the roads can be detected. With adequate dataset the above mentioned networks can also be trained from scratch for better performance. Development of a detection network entirely for ADAS application is an option so that optimised time and accuracy figures can be obtained. The inclusion of lane departure warning and the use of Light Detection and Ranging (LIDARs) for distance measurements can also be incorporated in this work.

References

[1] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1. IEEE, 2005, pp. 886–893.

[2] M. Hemmati, M. Biglari-Abhari, S. Niar, and S. Berber, "Real-time multi-scale pedestrian detection for driver assistance systems," in *2017 54th ACM/EDAC/IEEE Design Automation Conference (DAC)*. IEEE, 2017, pp. 1–6.

[3] D. Tomè, F. Monti, L. Baroffio, L. Bondi, M. Tagliasacchi, and S. Tubaro, "Deep convolutional neural networks for pedestrian detection," *Signal processing: image communication*, vol. 47, pp. 482–489, 2016.

[4] C.-Y. Tsai, H.-C. Liao, and K.-J. Hsu, "Real-time embedded implementation of robust speed-limit sign recognition using a novel centroid-to-contour description method," *IET Computer Vision*, vol. 11, no. 6, pp. 407–414, 2017.

[5] J.-G. Wang, L. Zhou, Y. Pan, S. Lee, Z. Song, B. S. Han, and V. B. Saputra, "Appearance-based brake-lights recognition using deep learning and vehicle detection," in *Intelligent Vehicles Symposium (IV), 2016 IEEE*. IEEE, 2016, pp. 815–820.

[6] Y.-K. Lai, C.-Y. Ho, Y.-H. Huang, C.-W. Huang, Y.-X. Kuo, and Y.-C. Chung, "Intelligent vehicle collision-avoidance system with deep learning," in *2018 IEEE Asia Pacific Conference on Circuits and Systems (APCCAS)*. IEEE, 2018, pp. 123–126.

[7] R. K. Satzoda, S. Lee, F. Lu, and M. M. Trivedi, "Snap-das: A vision-based driver assistance system on a snapdragontm embedded platform." in *Intelligent Vehicles Symposium*, 2015, pp. 660–665.

[8] L. Shine, A. Edison, and J. C. V. , "A comparative study of faster r-cnn models for anomaly detection in 2019 ai city challenge," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2019.

[9] A. Mogelmose, M. M. Trivedi, and T. B. Moeslund, "Vision-based traffic sign detection and analysis for intelligent driver assistance systems: Perspectives and survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 4, pp. 1484–1497, Dec 2012.

[10] A. Dominguez-Sanchez, M. Cazorla, and S. Orts-Escolano, "A new dataset and performance evaluation of a region-based cnn for urban object detection," *Electronics*, vol. 7, no. 11, p. 301, 2018.

[11] S. Houben, J. Stallkamp, J. Salmen, M. Schlipsing, and C. Igel, "Detection of traffic signs in real-world images: The German Traffic Sign Detection Benchmark," in *International Joint Conference on Neural Networks*, no. 1288, 2013.

[12] L. Sifre and S. Mallat, "Rigid-motion scattering for image classification," Ph.D. dissertation, Citeseer, 2014.

[13] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *European conference on computer vision*. Springer, 2016, pp. 21–37.

[14] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[15] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.

[16] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, 2015, pp. 91–99.