Proceedings of the Second International Conference on Smart Electronics and Communication (ICOSEC).
IEEE Xplore Part Number: CFP21V90-ART; ISBN: 978-1-6654-3368-6

2021 2nd International Conference on Smart Electronics and Communication (ICOSEC) | 978-1-6654-3368-6/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICOSEC51865.2021.9591747

Deep Learning based Object Detection Model for Autonomous Driving Research using CARLA Simulator

Niranjan DR, VinayKarthik B C, Mohana
Electronics and Telecommunication Engineering, RV College of Engineering®
Bengaluru, Karnataka, India.

Abstract- **Autonomous vehicle research has grown exponentially over the years with researchers working on different object detection algorithms to realize safe and competent self-driving systems while legal authorities are simultaneously looking into the ways of mitigating the risks posed by fully autonomous vehicles. These advancements can result in a much safer commuting environment, reduced accidents and also eliminate the necessity for human driving. The creation of data and access to data for autonomous driving research is difficult challenge that research communities are facing. Hence, open source simulators such as the CARLA simulator (CAR Learning to Act) help us train and test models and to gain insights into autonomous driving with ease. This paper proposes the application of object detection algorithm on CARLA simulator to derive useful results for autonomous driving research. Further, the comparison of CARLA simulator with other available simulators, key players in the field of autonomous vehicle technology, state-of-the-art algorithms being used for autonomous driving, real time implementation challenges and future technologies are also discussed.**

Keywords— Single Shot Detector (SSD), CARLA, Convolutional Neural Network (CNN), Object Detection, Artificial Intelligence, Video Surveillance.

I. INTRODUCTION

Self-driving or autonomous driving is a field that will have tremendous implications on our future. It will drastically improve road safety, increase access to mobility, and will also reduce costs of transportation. A self-driving car uses continuous inflow of data from several onboard sensors such as LIDAR, RADAR and camera to detect obstacles and lanes on the road, to make decisions to control the speed, direction and acceleration of the vehicle. LIDAR and RADAR prove to be an accurate technique to detect obstacles. However, camera onboard the vehicle proves to be the most accurate when used with efficient object detection algorithms [37]. Hence, computer vision and image processing play a vital role in autonomous driving. In recent years, R&D in the field of autonomous driving has been democratized by CARLA simulator which is an open-source simulator for autonomous driving research [26]. CARLA has been developed from the ground up to support development, training, and validation of autonomous urban driving systems. In addition to open-source code and protocols, CARLA also provides open digital assets such as urban layouts, buildings, vehicles. The simulation platform supports flexible specification of sensor suites and environmental conditions [29]. The CARLA simulation software allows us to gather image dataset to train proposed object detection models on. It also provides a platform to deploy object detection models for testing through Python APIs. There are several object detection algorithms available for use through TensorFlow API. The choice of the algorithm to be used is done by a tradeoff between Mean Average Precision (mAP) and the speed of the algorithm. The SSD (Single Shot Multibox Algorithm) is one such algorithm which is one of the fastest object detection algorithms and also has an appreciable mAP value [35] [36].

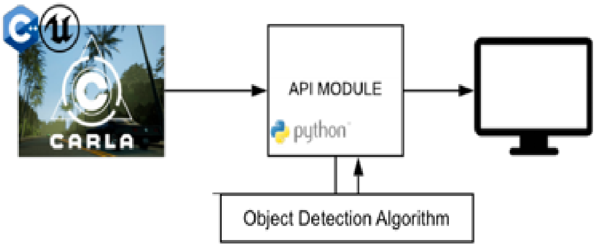

Fig.1. Block diagram of implementation.

Figure 1 shows the block diagram of implementation. In this paper, we used CARLA simulator to first generate dataset, on which an object detection model was trained. The model was later tested upon test images to measure its performance on CARLA. The dataset was created from different scenarios on the CARLA simulator. Five scenarios from different town and weather settings were considered. Images were later labelled for training purposes. The classes of objects considered were: Vehicles (cars and trucks), Bikes, Motorbikes, Traffic Lights and Traffic signs.

II. LITERATURE SURVEY AND PROBLEM ANALYSIS

Rodrigo Gutiérrez [1] describes a modular and extendable waypoint tracking controller for Robot Operating System used in self-driving cars. It describes controller performing accurate interpolation of the waypoints and uses effective methods to make sure proper trajectory tracking at cruising speed in city driving conditions. Vehicle propagation speed is implemented for path classifying using a velocity segregator. Comparison of wide variety of simulators and various test scenario such as Linear Velocity Profiler and Curvature Calculation and

Tracking Error estimation is discussed [32]. Prabhjot Kaur [2] identifies essential qualities of a good simulator, which can be used for comparison of popular simulators. This paper provides a comparison of commonly used simulators using parameters such as the multi-view geometry, Traffic Infrastructure 3D and Virtual Environment. The analysis shows that CARLA and LGSVL simulators are efficient simulators for complete validation of self-driving cars, Along with this survey of present challenges for simulation and testing for building fully autonomous cars. Shivesh Khaitan [3] describes obstacle avoidance technique, a two-stage predictor made up of short-term and long-term pre-diction is deployed in planning layer to make safe trajectories for the self-driving cars. Its efficiency in a Frenet frame planner is demonstrated. An efficient controller is used along with tube Model Predictive Control (MPC) guaranteeing proper implementation of the trajectory in the presence of noise and disturbance. A Gaussian regression model is used for detection of the uncertainty [38]. The safety, effectiveness and real-time performance of framework is demonstrated using CARLA simulator. Ford Mustang is considered for simulation including various situations such as overtaking, intersections and curves.

Henry Pulver [4] aims towards achieving the proper equation with planning quality, safety and efficiency which is a challenge for self-driving cars. The optimizer gives an important layer of online protection from learning failures or deficiency which can compromise on safety of passenger. Simulation is carried out in CARLA and is seen that PILOT has significant reduction in runtime. Junha Roh [5] explains distributed navigation with multiple non- communicating elements at street intersections with or without traffic signs junctions in city. Collision avoidance during these situations depends ability to predict and react quickly. Primarily the geometric structure of junction ahead and the plan to move and avoid collisions .Multiple Topologies Prediction (MTP), is a data-dependent trajectory-planning method that rebuilds trajectory prediction of high-likelihood modes in a busy intersection. MTP is more efficient than benchmark multimodal trajectory prediction baseline (MFP) when compared by prediction accuracy of 78.24% is demonstrated on the CARLA simulator.

Kaleb Ben Naveed [6] proposes a Hierarchical Reinforcement Learning (HRL) in addition to Proportional–Integral–Derivative (PID) controller for path scheduling. The proposed HRL aims to divide entire process of self-driving into various levels of sub tasks. The introduction of sub-tasks reduces the meeting time and allows the learned rules to be repeatedly used for other situation and perfect trajectories by keeping the noisy perception system of the autonomous car constant. The PID controller is included for keeping track of the waypoints, these waypoints give rise to proper trajectories, this reduces any kind of sudden movements like jerks. By using a LSTM layer the problem of incomplete observations is solved. Results are obtained using CARLA simulator which shows this method helps in reduction of convergence time, and also generates continuous trajectories, and is able to work in real time condition.

Yuanfu Luo [7] introduces SUMMIT which is a high-fidelity simulator, SUMMIT is used for simulation of dense and unregulated city traffic conditions at locations that are supported by the Open Street Map. SUMMIT is a multi-agent movement model, GAMMA is used to model the behaviors of multidimensional traffic elements, and a real-time POMDP schedule, this acts as a driving professional. SUMMIT is an extension of CARLA simulator and has the qualities inherent including physical and visual realism for self-driving simulation. Validation of models that are used in traffic motion prediction accuracy on different types of popular data sets is carried out. Majid Moghadam [8] presents a complete deep reinforcement learning methodology used in self-driving cars intended for decision-making and motion planning. The algorithm generates regular spatio temporal trajectories using the Frenet that is used by the feedback controller, simulations carried out on CARLA of this method is compared and tested on various real time traffic situations. Majid Moghadam [9] discusses a method for planning and path decision making for highway driving task and intelligent driving models to develop sustained decisions depending on the traffic scenario. Optimization of trajectory is done using the Frenet space. Trials on CARLA simulator proves its scalability and potential implementation in various driving scenarios. Sadique Adnan Siddiqui [10] addresses the problem of dense depth predictions using sparse distance sensor data and a single image on different types of weather conditions and explores the significance of different sensors such as camera, Radar, and Lidar for determining roximity by implementing Deep Learning technique. In this work, a deep regression network is proposed which utilizes a transfer learning approach having an encoder where a high performing pretrained model is used to initialize it for extracting dense features and a decoder for upsampling and predicting required depth. The results are demonstrated on Nuscenes, KITTI, and a dataset which was generated using the CARLA simulator. Tianyu Wang [11] explains inverse reinforcement learning for self-driving using distance and semantic category evaluation. A map encoder that concludes semantic probability using the observation pattern. A deep neural network used for semantic features is synthesized as a cost reduction encoder, the different components of this model is also demonstrated using the CARLA simulator.

Jinkun Cao [12] describes effective learning of driving rules in a real time environment having multiple elements. An Instance-Aware Predictive Control (IPC) method, used to predict the interaction between obstacles, elements and structures. An instance based event prediction algorithm to predict the possible interaction that might take place between elements under autonomous driving environment, which is regulated by the selected action sequence of autonomous vehicle is used. This paper also aims towards design of sequential sampling approach for better instance prediction.

Tanmay Agarwal [13] describes the implementation of deep reinforcement learning in order to study optimal control policy for the tasks of lane-following and lane assistance, driving around intersections as well as stopping in front of other agents, obstacles or traffic lights even in the dense traffic setting. This was implemented on CARLA simulator and evaluated using No Crash benchmarks. It was also observed that the use of reinforcement learning yielded a much better performance than convolutional encoders. Nishanth Rao [14] explained the use of an RNN which is termed as the Memory Neuron Network (MNN), this model is requires lesser computation power and has a simple architecture in comparison with LSTMs and GRU. Performance is evaluated using the data set generated by CARLA simulator. This model is found to have around 20%

lesser RMSE value compared to other models. The model performed appreciably well in relatively straighter paths, it needs refinement in the case of curvature and junction situations. Yuan Shen [15] explained that the behaviour of self-driving cars differs from people's expectations. Hence, a simple but effective framework called Auto Preview is implemented using CARLA simulator. This helps users get a preview of autopilot and autonomous driving cars before actually using a self-driving car, by trying to understand the overall driving style, deployment preference, and predicting the timing of actions. Zijian Zhang [16] analyzed that sensing ranges and accuracies can improve appreciably in networked vehicles when dynamic map fusion is used. The proposed method was tested on the CARLA simulator. Hironobu Fujiyoshi [17] explains the use of deep learning in image recognition and the latest technologies in the field. It involves the problem of choosing suitable mapping functions from a huge collection of data and training labels. Multitask learning, which allows for parallel processing of these problems is also discussed. The authors also opine that the end-to-end deep learning technologies which can possibly help in the judgement and control of self driving cars has a much better industry relevance than common image recognition algorithms and methodologies. Jigang Tang [18] analyzed the different image recognition methods that can be used to solve the problem of lane detection. This paper presented an overview of several existing network architectures which includes both object classification and object detection models, segmentation based methods. It also discussed the various ways of optimizing parameters of a training function and the related loss functions for each of these methods. CNN was found to have a comparatively higher accuracy when compared to other algorithms but the problems of lack of generalization ability and the difficulty to be deployed on mobile devices still exist as noted by the authors. Gustavo A.P. de Morais [19] implemented a hybrid architecture for vision based control of an autonomous vehicle. The architecture is an amalgamation of Deep Reinforcement Learning (DRL) and Robust Linear Quadratic Regulator (RLQR). In order to model the uncertainties that the vehicle may face in certain scenarios, evolutionary estimation was used. The Deep Reinforcement Learning method was compared with three of these hybrid controllers with the input being fed by an RGB camera which aided in the incessant calculation of steering angles and steering actions. It was found that the combination of DRL, evolutionary estimation and RLQR showed good results for autonomous vision-based driving. Mario Gluhaković [20] discussed a method to prevent collisions through accurate vehicle detection and potential collision warnings for an autonomous vehicle. This method was implemented on ROS. The object detection algorithm used was YOLO v2 while an ROS node was used to assess the distance between the vehicles in an image [41] [42]. This was implemented both on the CARLA simulator and in the real world environment. The results showed that the algorithm worked accurately for distances under 50 metres while the increase in distance above 50 metres caused a drop in the prediction accuracy. The recommended solution to this problem was to use higher resolution imagery for training the model. Akhil Agnihotri [21] implemented a level 2 autonomous vehicle using a CNN approach. It gives feedback to steering controls from the obtained camera input. This was tested on the CARLA. It was realized by using Arduino Mega and Raspberry Pi to control the motors and also to process signals to adjust the steering angle. It also proposed to extend this prototype to use point RCNN instead of CNN for 3D object recognition to plan the path better and to avoid obstacles. Anton Agafonov [22] observed that 3D object detection and localization is a major task in a vehicle control system. Therefore, existing object detection methods [39] [40] which use camera, depth sensor and LiDAR data are compared and analyzed. By implementing these algorithms on CARLA, the effectiveness of these methods was studied. It was observed that models trained on simulated data do not work effectively on real world data. The need for realistic simulation data produced using generative neural networks was also mentioned. Le-Anh Ten [23] discussed that autonomous driving and surveillance are important fields of work in self driving vehicle research. A novel method was devised to detect lane markings on roads. A front-view camera is used to provide the input images. These images are then fed into a semantic segmentation network to extract features to detect lane markings. A U-Net architecture, which is a CNN developed for biomedical image segmentation is used. The effectiveness of this model was validated using CARLA simulator. Ali Baheri [24] explained an approach for control of an autonomous vehicle through CARLA simulator. Several deep learning models were used and compared. Firstly, a variational autoencoder was trained to create a latent representation by encoding the input image. Next, a RNN was used to predict the representation of the next frame and also to maintain temporal data. At the end, an evolutionary-based reinforcement learning algorithm trained the controller to identify the action to be taken. Yinfeng Gao [25] analysed methods such as imitation learning and reinforcement learning that can be used in various subdomains of autonomous driving. The results show that Dagger is much suited for lane keeping problems while InfoGAIL can efficiently differentiate between several unique driving styles.

III. CARLA SIMULATOR

CARLA (Car Learning to Act) simulator is an open-source simulator built for autonomous driving research. It consists of two main modules: the CARLA simulator and CARLA Python API. It was designed with the purpose of democratizing autonomous driving research by providing a platform which models real world driving scenarios and use cases. It is built on Unreal Engine 4 and allows users to control the simulations through the Python API. It provides users with simulated sensor and camera data which can be used to train object detection models or reinforcement learning algorithms to realize autonomous driving systems.

Features of CARLA simulator

- Features of CARLA include integration of various sensor packages for autonomous driving including multi-camera, LIDAR and GPS.
- It also has a flexible API to control simulator.
- Fast execution via fast simulation for planning and control, and users can generate the custom maps from tools like Road runner.
- It scenario Runner allows the user simulate different traffic situation based on modular behaviour.

- This includes the Autoware agent and conditional learning agent along with ROS integration via ROS-bridge.

IV. Neural network design in autonomous driving research

A. GNN (Graph Neural Networks)

A set of connected vehicles that are classified into different scopes such as road network, corridors or segments is termed as a connected autonomous vehicle (CAV) network. In such a network, traffic information is shared between the vehicles to make better decisions in coordination. With technologies such as vehicle to vehicle communication using 5G, this is becoming more possible by the day. However, multiple agents are involved in this network and processing information from all these agents to take coordinated decisions is very difficult and computationally intensive. Hence, the use of algorithms and models like Graph Neural Networks has been proposed to accomplish the decision making process faster and simultaneously integrate sensor information from multiple agents in the network [27] [30] [31].

B. LSTM (Long short-term memory) networks

Motion planning is a very important process in autonomous driving. Given a route, motion planning helps in finding the best path for the vehicle to take, based on real-time traffic conditions and size and speed constraints of the vehicle. However this requires a model which can process and understand both spatial and temporal data simultaneously in order to plan the paths. Hence spatiotemporal LSTMs have been used for this purpose [28].

C. QCNN (Quantum Convolutional Neural Networks)

Quantum computing research is growing in leaps and bounds in the past few years. Quantum computing has even had its implementation in Deep learning. One such example is a Quantum Convolutional Neural Network (QCNN). It is based on the model of CNN but the convolutions are performed on qubits, which are quantum analogues of classical bits. Such QCNNs have been successfully used for image classification by identifying super pixels, which are groups of pixels which have similar characteristics like intensity. QCNNs are capable of computing at the pixel-level by extracting RGB information [33].

D. RNN (Recurrent Neural Networks)

The process of predicting the future states of other vehicles in the environment of an autonomous vehicle based on current and past observations is called Vehicle Behaviour Prediction. This allows the vehicle to take precautions such as braking and decelerating on time to avoid probable collisions. For this purpose deep learning frameworks, especially RNNs are extensively used as they have higher efficiencies in complex environments than traditional algorithms. Since phenomena like gradient exploding and gradient vanishing restrict the training of RNNs through very long sequences of data, they are used along with LSTMs.

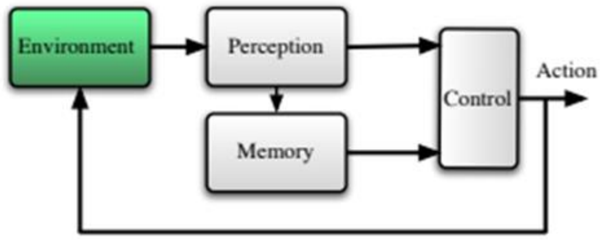

Fig.2. Model Learning approach.

Figure 2 shows model learning approach. A RNN is used to predict spatial and temporal data by interfacing it with the inputs from perception systems (sensors, LiDAR and camera). It can also be connected to the control systems to generate the actions to be performed by the automobile. The changes in the environment act as a feedback to further improve the predictions of the RNN.

E. Edge based frameworks and edge computing

Realizing the importance of edge computing, a group of inter-industry global companies have come together to form the "Automotive Edge Computing Consortium" (AECC) to drive best practices for the convergence between the vehicle and computing ecosystem. Autonomous driving edge computing systems are used to efficiently integrate the functional modules such as edge servers and V2X infrastructure. This is to process a large amount of data in real time, and without compromising the security of the users. , in order to make the autonomous driving more safe the edge computing should be fast enough to predict the uncertainties a few seconds in advance this also include high level integration of sensing, localization, perception, decision making, and the smooth interaction with cloud platforms for high-definition (HD) map generation and data storage technologies [34].

V. Popular simulators, dataset and performance metrics used for autonomous driving research

A. Open source simulators

LGSVL - Developed by LG Electronics America R&D centre, can generate HD maps.
SUMMIT - Simulator for urban Driving in massive mixed traffic is a high fidelity simulator, promotes testing of crowd-driving algorithms.
FLOW - used for traffic microsimulation developed by mobile sensing Lab at UC Berkeley.

B. Commercial or professional license simulators

PG Drive - This simulator integrates the key feature of procedural generation (PG), built on Panda3d and Bullet engine.
Deepdrive - Feature include support for linux and windows function based on speed, safety and legality.
Airsim - Developed by Microsoft Built on unreal engine supports simulation with popular flight simulator.
DataViz - This simulator used by Uber and is a proprietary tool of the company
GYM-Duckietown - simulator by duckietown university it is a fully functional simulator it can even train and trust classical robotics algorithms.
Some other Simulators are: SUMO, TORCS, Apollo, GTA V Force-base, Autono ViSim.

C. Popular available dataset

Udacity dataset - Built as a joint venture along with Google this dataset is mainly composed of video frames taken from urban road conditions. It has a total number of 404,916 video frames for training and 5,614 video for testing. Challenging conditions such as severe lighting changes, sharp road curves and busy traffic. Some other dataset such as Nvidia, Kitti, Landmarks, CARLA, LANDMARKS-V2, Waymo Open Dataset, Astyx Dataset HiRes2019, Berkeley Deep Drive, Open Images V5, Oxford Radar Robot Car Dataset, Pandaset, nuScenes Database.

D. Performance metrics

Table I shows the popular performance metrics.

TABLE I POPULAR PERFORMANCE METRICS

Criteria	Abbreviations	Formulas
Accuracy Rate	ACC	(TP+TN)/(N+P)
Sensitivity - True Positive Rate	TPR	TP/(TP+FN)
Specifity - True Negative Rate	TNR	TN/(TN+FP))
False Positive Rate	FPR	FP/(FP+TN)
False Negative Rate	FNR	FN/(FN+TP)
Positive Pred Value - Precision	PPV	TP/(TP+FP)
F Score	F	2*((PPV*TPR)/(PPV+TPR))
Error Rate	ERR	(FP+FN)/(N+P)

VI. OBJECT DETECTION ALGORITHMS

SSD MobileNet - Mobilenet is a framework released in 2017 which has an efficient CNN structure that allows it to run efficiently on mobile phones and embedded systems. SSD Mobilenet is the application of the Single SHot Multibox Detector algorithm to the Mobilenet framework

SSD ResNet - Resnet or Residual Network is a technology developed in 2015, which solves the issue of vanishing or exploding gradient through a technique known as skip connections. SSD implemented on the ResNet framework is termed as SSD ResNet

Faster RCNN - It is an object detection framework which is a region based CNN. It has 2 stages, in which a Region Proposal Network is used to detect areas of interest and the model is trained based on such regions.

Mask RCNN - Mask RCNN adds another layer to the faster RCNN framework which outputs a binary mask which analyses if a pixel is a part of an object or not apart from the use of bounding boxes.

YOLO - YOLO is an algorithm which detects objects in real time. It considers the object detection problem as a regression problem and outputs the class probabilities of the detected objects [43] [44] [45].

CenterNet - Developed in 2019, it is a deep convolutional neural network based model which interprets objects as a triplet of key points hence increasing both precision and recall.

VII. DESIGN AND IMPLEMENTATION

The dataset was created from different scenarios on the CARLA simulator as shown in figure 3. Five scenarios from different town and weather settings were considered. The images were later labelled for training purposes. The classes of objects considered were: Vehicles (cars and trucks), Bikes, Motorbikes, Traffic Lights and Traffic signs.

Fig.3. Dataset creation

The dataset was created by capturing images from CARLA simulator at certain intervals. A total of 1028 images of 640x380 pixels were taken. The images were taken from 5 different town settings in CARLA simulator with different kinds of weather conditions (by altering percentage of cloudiness, precipitation and sun altitude angle). Different traffic conditions are also present in each image. The 1028 images were split into the test and train dataset with 208 test and 820 training examples.

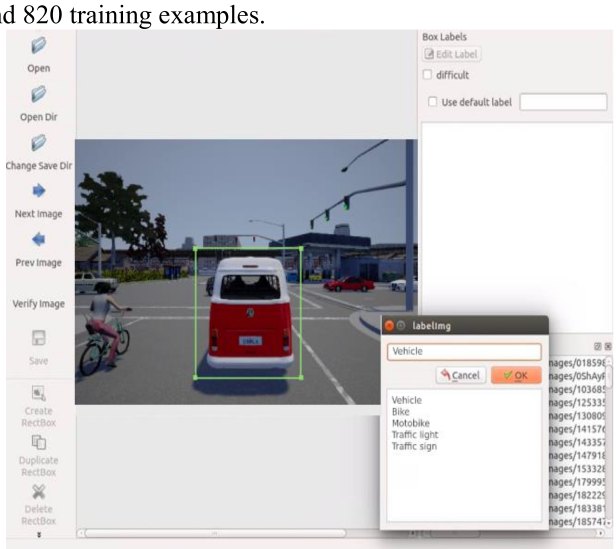

Fig. 4. Image Annotation

Image annotation was done with a Python GUI tool called labelmg as shown in figure 4, identified 5 classes of objects for our object detection problem: Vehicles (Cars, Trucks), Bike (Cycle) Motorbike, Traffic light and Traffic Sign. For each image an associated label .xml file in the pascal VOC format is hence created. SSD MobileNet (Single Shot Multibox

Detector) algorithm was used for object detection. SSD is a neural network based object detection algorithm in which the entire classification function is completed in a single pass through the network. It is known to be one of the fastest running object detection algorithms. The training parameters considered for the SSD algorithm are given below.

TABLE II. DESIGN SPECIFICATIONS OF SSD MOBILENETV1

SSD MobileNet V1: Design Specifications	
Initial Learning Rate	0.004
Activation Function	ReLU (Rectified Linear Unit)
Batch Size	10
Regulariser	L2 Regulariser
Epochs	4000
Loss Function	RMSE (Root Mean Squared error)

Table II shows the design specifications of SSD MobileNetV1. And figure 5 the graph of total RMSE loss curve.

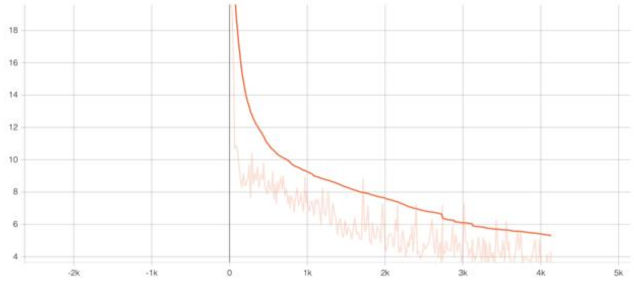

Fig.5. Total RMSE Loss Curve

VIII. SIMULATION RESULTS AND ANALSIS

A. Simulation and real-world environment

Just like humans think, plan and then control a car while driving, autonomous driving also involves these steps. These steps aims towards minimal or low human intervention

Perception - This involves how the computing system gets information regarding external environment in real world conditions primarily through Computer vision in case of camera, LiDARs (Light Detection and ranging) sensors and Radars

Planning - This part includes the strategy making of how to overtake, bypass the upcoming traffic. What is the plan of action if a “Y”, “+” junction is encountered regulation of speed and finally prediction of brake utilization as and when required Coordination: This is synchronous with the planning and deals with how the car behaves in relation to other smart cars on the street. It requires communication with other vehicles and infrastructure, examples are: Platooning: this is a car following lane on highways giving benefit of low air resistance and also reduces traffic congestion. Real world environment also comprises of static , dynamic elements and restrictions let us look into each of them Static elements are tree , lamp post traffic light and even parked vehicles and buildings, they do not move and self-driving cars has to plan accordingly and proceed whereas Dynamic elements are the one which moves for example pedestrians other cars, truck and vehicles on street The autonomous vehicle has to expect these elements in motion which can take erratic movements and plan accordingly. Another element is restrictions that are put up using sign board signals and diversions and weather restrictions as well.

B. Simulation Results

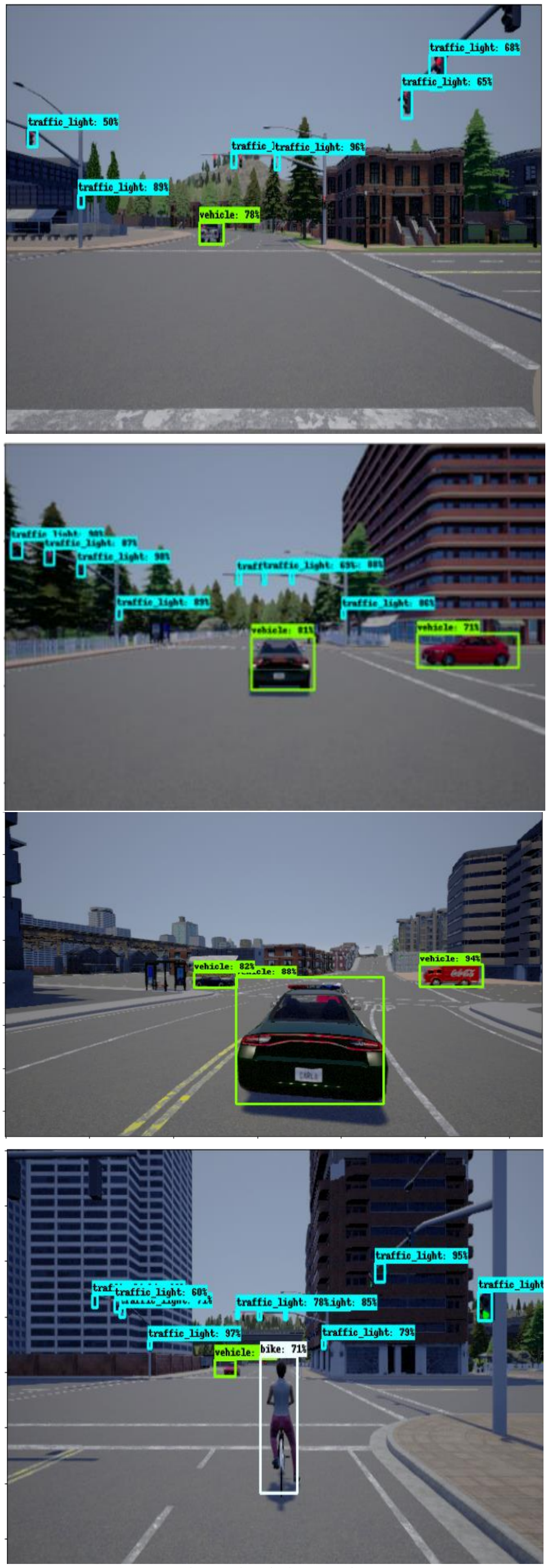

Fig.6. Simulation results.

Figure 6 shows the simulation results of SSD model. Test results obtained showed that the SSD algorithm successfully classified the objects on CARLA simulator test images with an average accuracy of 82.81%.

IX. Research and Real Time Implementation Challenges

Road conditions - Roads can be of varying types in different regions of a country or in the world. While some roads are smooth and have lane markings, others might be rough and undulated with no defined lanes. This is a major challenge while implementing autonomous vehicles, especially in countries like India.
Traffic conditions - Though we will have several autonomous cars on the road in a few years with vehicle-to-vehicle communication amongst each other to prevent accidents, there will still be a significant amount of human drivers on the road. A person breaking traffic rules might cause problems for many other autonomous vehicles.
Weather conditions - Autonomous vehicles are trained to run efficiently in different types of weather conditions such as sunny, overcast and stormy. However, the weather is still unpredictable and zero scope for error in autonomous vehicles even in extreme weather conditions is difficult to realize.
Accountability in case of accidents - The question of who is responsible for accidents caused by self-driving cars is still being pondered upon in the legal corridors of several countries. This remains a major liability to autonomous car manufacturers.

X. Key Players in Research and Development

All leading automobile giants such as Audi, Mercedes-Benz, Toyota, Honda and Volvo are working on autonomous driving technologies in collaboration with top tech companies such as:
Tesla - Autopilot is Tesla's famed autonomous driving software which has been implemented in several models of Tesla cars. Though fully autonomous driving on Autopilot is on hold due to legal restrictions and technical limitations, the company claims that the hardware on Tesla cars is fully capable of handling such technologies in the future.
Nvidia - Nvidia's autonomous driving technology project is named DRIVETM. The DRIVE AGX platform is an in-vehicle AI computer which integrates sensor fusion, and also accelerates performance of algorithms using hardware accelerators. Nvidia is working in collaboration with automobile companies such as Volvo and Daimler to realize self-driving cars.
Qualcomm - Qualcomm Vision Enhanced Precise Positioning (VEPP) is a software which provides lane-level positioning and also has immense sensor fusion capabilities with the integration of GNSS, camera, IMU and vehicle odometry information in real-time. Auto-tech firm Veoneer is also working with Qualcomm to use the VEPP onboard vehicles.
Intel - Intel acquired Jerusalem-based Mobileye which manufactures 80% of the world's driver assistance systems. Now, through Intel's technological prowess in in-vehicle computing, camera and sensor technologies, the company has expanded to manufacturing a full sensor suite for autonomous driving including radars and lidars. They plan to launch self-driving cars to the market by 2025
Waymo - Waymo is a subsidiary of Alphabet Inc (parent of Google Inc) which runs a self-driving taxi service in Arizona which demonstrated one of the first fully autonomous driving systems. Volvo and Stellantis are two automobile companies that have chosen Waymo's platform for their self-driving car fleets.
Apple - Apple's work on self-driving cars has been kept confidential for many years. Apple acquired autonomous driving tech company Drive.ai in 2019. Reports suggest that Apple is working with Hyundai and LG to roll out self-driving cars by 2024.

XI. Conclusion and Future Scope

From experimental results, observed that the CARLA simulator can be utilised effectively train and test object detection algorithms. CARLA simulator has an advantage over other simulators based on its open source nature, robustness, compatibility and real world environment modelling. Autonomous driving is attracting a lot of research and attention by different technology companies which are investing in and developing autonomous driving technologies in partnership with major automobile companies. Algorithms such as RNN, QCNN, GNN, and LSTM have been successfully implemented in the field of self-driving in various autonomous functions such as Vehicle behaviour prediction and Motion planning. Though there are several real time implementation challenges in autonomous driving, we can see that several technologies such as 3D object detection, edge based frameworks and LiDAR detection are being developed to mitigate the risks. In conclusion, SSD Model is implemented and obtained an accuracy of 82.81%. Autonomous driving research still has a long way to go, but progress has been tremendous in this field and open source software's such as CARLA simulator have immensely aided the development of this field.

References

[1] Gutierrez, Rodrigo et al., "A Waypoint Tracking Controller for Autonomous Road Vehicles Using ROS Framework" *Sensors. 20. 4062,* 2020.

[2] Kaur P et al., "A Survey on Simulators for Testing Self-Driving Cars", *arXiv e-prints,* 2021.

[3] Khaitan et al.,"Safe planning and control under uncertainty for self-driving", *arXiv e-prints*, 2020.

[4] Pulver et al.,"PILOT:Efficient Planning by Imitation Learning and Optimisation for Safe Autonomous Driving", *arXiv e-prints,* 2021.

[5] Mavrogiannis et al., " Multimodal Trajectory Prediction via Topological Invariance fo Navigation at Uncontrolled Intersections", *arXiv e-prints,* 2020.

[6] Naveed et al.,"Trajectory Planning for Autonomous Vehicles Using Hierarchical Reinforcement Learning",*arXiv e-prints,* 2020.

[7] Yuanfu Luo et al.,"SUMMIT: Simulating Autonomous Driving in Massive Mixed Urban Traffic"*arXiv e-prints,* 2020.

[8] Moghadam et al., "An End-to-end Deep Reinforcement Learning Approach for the Long-term Short-term Planning on the Frenet Space" ,*arXiv e-prints,* 2020.

[9] Moghadam and Elkaim.,"An Autonomous Driving Framework for Long-term Decision-making and Short-term Trajectory Planning on Frenet Space",*arXiv e-prints,* 2020.

[10] Siddiqui et al., "Multi-Modal Depth Estimation Using Convolutional Neural Networks", *arXiv e-prints,* 2020.

[11] Wang et al., "Inverse reinforcement learning for autonomous navigation via differentiable semantic mapping and planning", *arXiv e-prints*, 2021.

[12] Cao1 et al..,"Instance-Aware Predictive Navigation in Multi-Agent Environments",*arXiv e-prints,* 2021.

[13] A. Agarwal et al., "Affordance-based Reinforcement Learning for Urban Driving",*arXiv e-prints,* 2021.

[14] Rao and Sundaram et al., "Spatio-Temporal Look-Ahead Trajectory Prediction using Memory Neural Network",*arXiv e-prints*, 2021.

[15] Shen, Wijayaratne et al.,"AutoPreview: A Framework for Autopilot Behavior Understanding",*arXiv e-prints,* 2021.

[16] Zhang,Wang et al.,"Distributed Dynamic Map Fusion via Federated Learning for Intelligent Networked Vehicles",*arXiv e-prints,* 2021.

[17] Fujiyoshi, Hirakawa et al.,"Deep learning-based image recognition for autonomous driving", Chubu University, 1200 Matsumoto-cho, Kasugai, Aichi 487-8501, Japan.

[18] Tanga, Li et al., "A review of lane detection methods based on deep learning",www.elsevier.com/locate/patcog 2021.

[19] Morais Marcos et al., "Vision-based robust control framework based on deep reinforcement learning applied to autonomous ground vehicles",*www.elsevier.com/locate/conengprac* 2020.

[20] Gluhaković et al.,.,"Vehicle Detection in the Autonomous Vehicle Environment for Potential Collision Warning",*UNIOS-ZUP 2018*-6.

[21] Agnihotri,Saraf et al.,"A Convolutional Neural Network Approach Towards Self-Driving Cars" IEEE, 2019.

[22] Agafonov et al.,"3D Objects Detection in an Autonomous Car Driving Problem", *IEEE,* 2020.

[23] Le-Anh Tran et al., "Robust U-Net-based Road Lane Markings Detection for Autonomous Driving", International Conference on System Science and Engineering (ICSSE),2019.

[24] Baheri Kolmanovsky, et al., "Vision-Based Autonomous Driving: A Model Learning Approach", *American Control Conference Denver, CO, USA,*2020.

[25] Gao Liu et al.,"Comparison of Control Methods Based on Imitation Learning for Autonomous Driving", *International Conference on Intelligent Control and Information Processing Marrakesh, Morocco, 2019.*

[26] Mozaffari, S., et al., "Deep Learning-based Vehicle Behaviour Prediction For Autonomous Driving Applications: A Review", *arXiv e-prints,* 2019.

[27] Chen, S et al., "Graph neural network and reinforcement learning for multi-agent cooperative control of connected autonomous vehicles", *Comput Aided Civ Inf.* 2021.

[28] Bai, Zhengwei et al. "Deep Learning Based Motion Planning For Autonomous Vehicle Using Spatiotemporal LSTM Network."Chinese Automation Congress (CAC), 2018.

[29] Alexey Dosovitskiy, et al. "CARLA: An Open Urban Driving Simulator", *arXiv:1711.03938v1*, 2017.

[30] Pradhyumna P et al.,"Graph Neural Network (GNN) in Image and Video Understanding Using Deep Learning for Computer Vision Applications," *2nd International Conference on Electronics and Sustainable Communication Systems(ICESC)*, 2021.

[31] Mohana et al.,"Artificial (or) Fake Human Face Generator using Generative Adversarial Network (GAN) Machine Learning Model," *Fourth IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT 2021),* 2021.

[32] Shreyas E et al.," 3D Object Detection and Tracking Methods using Deep Learning for Computer Vision Applications," *6th International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT),* 2021.

[33] Varadi Rajesh et al.,"Quantum Convolutional Neural Networks (QCNN) Using Deep Learning for Computer Vision Applications," *6th International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT),* 2021.

[34] Biswas A et all., "Survey on Edge Computing–Key Technology in Retail Industry" Lecture *Notes on Data Engineering and Communications Technologies, vol 58, Springer,* Singapore, pp.97-106, 2021.

[35] Mohana et al.,"Performance Evaluation of Background Modeling Methods for Object Detection and Tracking," *Fourth International Conference on Inventive Systems and Control (ICISC)*, 2020.

[36] R. J. Franklin et al.,"Anomaly Detection in Videos for Video Surveillance Applications using Neural Networks," *Fourth International Conference on Inventive Systems and Control (ICISC),* 2020.

[37] R. J. Franklin et al.,"Traffic Signal Violation Detection using Artificial Intelligence and Deep Learning," *Fifth International Conference on Communication and Electronics Systems (ICCES),* 2020.

[38] H. Jain et al.,"Weapon Detection using Artificial Intelligence and Deep Learning for Security Applications," *International Conference on Electronics and Sustainable Communication Systems (ICESC),* 2020.

[39] R. K. Meghana et al.," Background-modelling techniques for foreground detection and Tracking using Gaussian Mixture Model," *Third International Conference on Computing Methodologies and Communication (ICCMC),* 2019.

[40] N. Jain et al.,"Performance Analysis of Object Detection and Tracking Algorithms for Traffic Surveillance Applications using Neural Networks," *International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, 2019.

[41] C. Kumar B et al.,"YOLOv3 and YOLOv4: Multiple Object Detection for Surveillance Applications," *International Conference on Smart Systems and Inventive Technology (ICSSIT),* 2020.

[42] C. Kumar B et al.," Performance Analysis of Object Detection Algorithm for Intelligent Traffic Surveillance System," *International Conference on Inventive Research in Computing Applications (ICIRCA)*, 2020.

[43] Mohana et al.,"Object Detection and Tracking using Deep Learning and Artificial Intelligence for Video Surveillance Applications" *International Journal of Advanced Computer Science and Applications(IJACSA), 10(12), 2019.*

[44] Ajeet Sunil et al.," Usual and Unusual Human Activity Recognition in Video using Deep Learning and Artificial Intelligence for Security Applications*," Fourth IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT 2021),* 2021.

[45] Umesh Parameshwar Naik et al.,"Implementation of YOLOv4 Algorithm for Multiple Object Detection in Image and Video Dataset using Deep Learning and Artificial Intelligence for Urban Traffic Video Surveillance Application," *Fourth IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT 2021),* 2021.