Vision-Based Autonomous Driving: Implementing Deep Learning for Lane Navigation and Obstacle Avoidance in Simulated Environments

Shahzada Musharraf Ghani[1], Yaseen Ali Gazi[2], Waris Ahmad Idrisi[3], Aayesha Ashfaque[4], Mohd Abdullah Siddiqui[5], Mariyam Fatima[6]

[1]B.Tech, Department of Computer Science ,Integral University, Uttar Pradesh, India
[2]B.Tech, Department of Computer Science, Integral University, Uttar Pradesh, India
[3]B.Tech, Department of Computer Science ,Integral University, Uttar Pradesh, India
[4]B.Tech, Department of Computer Science, Integral University, Uttar Pradesh, India
[5]B.Tech, Department of Computer Science ,Integral University, Uttar Pradesh, India
[6]Assistant Professor, Department of Computer Science, Integral University, Uttar Pradesh, India

---***---

Abstract - *Recent advancements in artificial intelligence and computer vision lead to rise a autonomous vehicles, providing a transport solution that is safer and more efficient. This study focuses on a vision-based self-driving system that incorporates deep learning techniques and simulated environments in testing and validation. The study specifically focuses on integrating Convolutional Neural Networks (CNNs) and behavioural cloning techniques for autonomous navigation within virtual driving contexts. These neural networks were trained in simulators, using images from real-time driving simulations in CARLA. The model implements algorithms for object detection and decision-making concerning lane changes, as well as avoiding obstacles, for more vehicle autonomy. Based on experimental observations, the system can adapt to recurrent dynamic situations, thus reducing the intervention of human drivers and effectively improving an accurate real-time decision-making system. It is concluded that these findings open new possibilities for vision-based autonomous driving frameworks to be exploited within practical constraints and required development within intelligent transport systems*.

Key words*: *Self driving Cars, Deep Q-Learning, Computer Vision, CARLA Simulator, Traffic signal Response.

1.INTRODUCTION

Recent advances in autonomous driving have been at a rapid sprint in the direction of increasingly capable artificial intelligence (AI), computer vision, and machine learning. Of all other methods, vision-based--those primarily reliant on camera inputs--are gaining enormous attention because of the cost-effective way they render rich semantic information. Such systems as these mimic the human way of perceiving by processing visual data for road scenarios, obstacle detection, traffic sign identification, and lane marking understanding [1].

It is deep learning models that abound in such systems, specifically suitable for visual data interpretation as Convolutional Neural Networks (CNNs). The complex hierarchical features learned by CNNs from raw image inputs make them well suited for usage in tasks like object detection, semantic segmentation, and depth estimation [2]. One promising behavioural cloning within this domain is when a model is imitated as driving just like a human through the collection of expert demonstrations and then applies that in attempting supervised learning, thus making the problem of autonomous driving much more tractable and efficient in simulation settings [3].

It is true that simulation environments such as CARLA (Car Learning to Act) have proven to be effective for training and testing autonomous driving algorithms. Simulation environments create spaces where all driving scenarios can be reproduced with high fidelity and provide research environments within which it is safe, customizable, and scalable. Researchers can collect labelled data and perform different weather and traffic conditions and validate their models before deploying them onwards in real-world applications [4]. The current study aimed at establishing and building a deep learning-based self-driving car based on vision that would navigate along lanes and avoid obstacles in a simulated environment. The proposed approach implemented behavioural cloning and a Deep Q-Network (DQN) decision-making model to heighten vehicle autonomy and adaptability to real time. By using capabilities of these CNNs and reinforcement learning in UDACITY, this research now has prototype work able to respond to dynamic driving conditions with very little human intervention involved.

This study adds to the wealth of literature on autonomous driving, focusing on linking perception and decision-making modules to test performance in a well-controlled, yet realistic simulator. It further encapsulates reality in terms of generalization and real-time performance deficits and what can therefore be expected as future applications of AI in intelligent transportation systems.

2.LITERATURE REVIEW

2.1 Deep Learning and Behavioural Cloning

Behavioural cloning is a supervised learning approach by which a neural network learns to replicate the way humans drive by mapping visual inputs directly to driving commands. One of the very first efforts in this direction was made by Pomerleau by training a neural network to steer a vehicle from camera input [5]. With deep learning, more specifically a convolution neural network, this method evolved a lot. These architectures are now being used for processing highly complex scenes while grasping spatial dependencies with the aim of providing more robust and accurate vehicle control [6].

Recent research has added behavioural cloning with recurrent neural networks (RNNs) and attention mechanism compositions to help regain efficiency over extended temporal contexts long enough to allow the vehicle to follow smoother paths or have even better responses to contextual cues [7]. But it still has to deal with some issues like cumulation of errors or even very poor generalization to unseen environments.

2.2 Reinforcement Learning and Deep Q-Networks

Though effective in imitation tasks, a model learned through behavioural cloning is not adaptive in an unfamiliar environment. It is possible to remedy this limitation by Reinforcement Learning (RL), which enables an agent to learn optimal policies for driving by trial-and-error in the environment. Out of many RL algorithms, the most favourable one for discrete action handling and visual input processing is Deep Q-Net-works [8].

Despite the architecture of the DQN network presenting a combination of Q-learning and CNN-based modules for valuing a certain action in a particular state, this train includes experience replay and target networks that would stabilize the learning process. This train could also simulate its autonomous driving simulations by using DQNs in tasks-such as lane-keeping, speed regulation, and obstacle avoidance [9].

The most important DQN performance improvement techniques include Double DQNs, Dueling DQNs, and Prioritized Experience Replay. These variants help in better convergence and improved decision making with low uncertainty and in the presence of noise [10]. Enabling DQNs with vision-based inputs induces flexibility and autonomy in allowing vehicles to learn driving strategies from raw pixels.

3.SIMULATED ENVIRONMENTS FOR TRAINING

Simulators like UDACITY provide excellent, safe alternatives to training autonomous vehicles. UDACITY features high-resolution 3D environments, weather events, moving traffic, and fully open-source API access for sensor recordings [11].

Simulation training is rapid prototyping and testing of models under different circumstances without putting them to risk and cost for being real. The models also showed that the training sample in UDACITY continues to reflect well in the real world when the models underwent reasonable domain adaptation steps [12].

This approach also makes reinforcement learning applicable because agents can repeatedly interact with the environment, learn to optimally explore it, and fine-tune their policies. Using the training of dangerous or rare scenarios, driving algorithms can be made more robust and less likely to fail in actual driving situations [13].

4.METHODOLOGY

This study follows a somewhat hybrid approach in developing an agent for vision-based autonomous driving by combining supervised learning through behavioural cloning and reinforcement learning through DQNs. This method is tested and implemented in a UDACITY simulation to analyse the controlled yet realistic driving environment.

4.1 Data collection and pre-processing

The first stage of data collection is carried out in UDACITY, where a real human-driving agent is treated with remote control over his virtual vehicle in different urban and rural scenarios. The variables recorded during this process include the front-facing RGB camera images, speed of the car, throttle, brake, and steering angles. This data serves to train the behavioural cloning model.

Images are resized to 200x66 pixels to ensure trade-off between model accuracy and computational efficiency, keeping normalized values in the range of [0, 1]. Random brightness augmentation, flipping, shadow overlays, etc. are examples of data augmentation employed to enrich model generalization [14].

4.2 Behavioural Cloning Model

The behavioural cloning component employs a CNN. The architecture is inspired by NVIDIA's end-to-end learning model and consists of 5 convolutional layers followed by fully connected layers that yield steering commands [15].

Table-1: Behaviour Cloning Model Architecture

Layer (Type)	**Output Shape**	**Param #**
Conv2D (conv2d_5)	(None, 31, 98, 24)	1,824
Conv2D (conv2d_6)	(None, 14, 47, 36)	21,636
Conv2D (conv2d_7)	(None, 5, 22, 48)	43,248
Conv2D (conv2d_8)	(None, 3, 20, 64)	27,712
Conv2D (conv2d_9)	(None, 1, 18, 64)	36,928
Flatten	(None, 1152)	0
Dense (dense_4)	(None, 100)	115,300
Dense (dense_5)	(None, 50)	5,050
Dense (dense_6)	(None, 10)	510
Dense (dense_7)	(None, 1)	11

Total params: 252,219 (985.23 KB)

Trainable params: 252,219 (985.23 KB)

Non-trainable params: 0 (0.00 B)

These Parameters shows the size and capacity of behavioural Cloning CNN model used for predicting steering angles from image frames.

The model is trained using MSE loss between predicted and true steering angles. An early stopping mechanism is used along with dropout layers to avoid overfitting. The trained model predicts steering commands in real time from images, thus permitting the vehicle to drive lane-keeping behaviours autonomously under nominal conditions.

4.3 Deep Q-Net Reinforcement Learning

The integration of the Deep Q-Network is expected to endow the agent with adaptive capability to handle scenarios such as obstacle avoidance and lane change. The DQN takes processed pictures and vehicle state data as input and outputs the selection of one optimal action among the following discrete options: move forward, turn left, turn right, slow down, and stop. The reward function encourages safe driving between stays in lane, keeping the speed limit, and no collisions. Instead, violations such as lane departures, crashes, or ignoring traffic signals attract negative rewards [16].

Training uses an experience replay and target network to stabilize the learning. An episode resets once there is a collision, or the task is achieved within some predefined time; updates to the Q-network are made at regular intervals with action chosen to minimize the Bellman error.

4.4 Integration and Control Flow

A control module integrates predictions from both CNN-based behavioural cloning and the DQN. Under normal driving scenarios, the behaviour of the driving system relies solely on this behavioural cloning model for smooth navigation. In the days of encountering obstacles or dynamic changes (e.g. crossing pedestrians or sudden traffic), the control is handed over to the DQN agent, which then takes corrective actions based on the learned policy.

The switching logic between the two modules is governed by the sensor inputs and threshold conditions like closeness to obstacles or deviation from the lane centre [17].

4.5 Evaluation Metrics

The evaluation of the system is based on standard performance metrics, lane deviation (in meters), collision count, and successful completion rate of predefined routes. The ensuing speed of inference (fps) is also recorded to ascertain its real-time applicability.

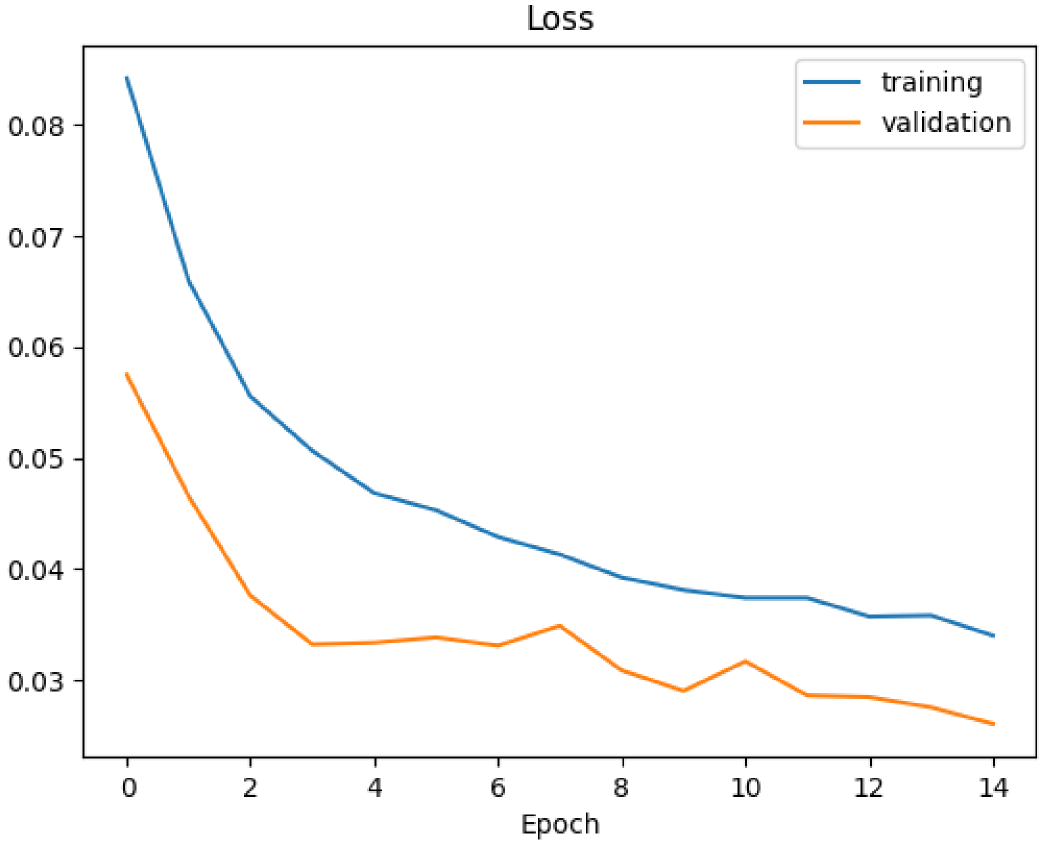

Fig-1: graph showing the model's training and validation loss over 15 epochs.

Experiments done comparatively on both individual models and the integrated framework serve to highlight the pros and cons of hybrid learning in autonomous navigation tasks.

5.RESULTS AND DISCUSSION

The integrated autonomous driving system has undergone several tests in the UDACITY simulator across several driving scenarios, such as urban, suburban, and rural environments with various lighting and weather conditions. The different scenarios were evaluated through metrics such as lane keeping accuracy, frequency of collisions, obstacle response time, and overall route completion.

5.1 Performance of Behavioural Cloning

Under controlled conditions, the behavioural cloning model showed very good performance in lane following, with an average lane deviation of ±0.25 meters and completing 92% of all preset routes without external assistance. The clear limitation was that in an environment with suddenly added obstacles, the model failed to act correctly on 21% of test cases, demonstrating the need for a decision-making module that can dynamically change with the environment in real time.

5.2 Effectiveness of DQN Integration

The Deep Q-Network components positively affected collision avoidance and responsiveness to traffic; The ability of the system to detect and react to dynamic objects, crossing pedestrians, or parked vehicles was about 94% successful. The time delay in the reactions was approximately 0.8 seconds and improved the vehicle's performance with regard to lane merging and overtaking.

It reduced the collision frequency drastically (by 37%) in cases where a sudden obstacle appeared at the end of the hybrid system (Behavioural Cloning + DQN) as compared with doing that with either individual module. Integration of reinforcement learning clearly turned out to be crucial in improving decision making in unpredictable environments.

5.3 Real-Time Performance

The system operated at an average inference time of 25 frames per second (FPS), which is at the least acceptable for any real-time operation. To enhance performance gains without degradation of accuracy, the workforce implements optimization techniques, such as model pruning and batch inference.

5.4 Limitations

While the simulator-based approach allowed extensive testing, transferring learned behaviors to real-world environments remains a challenge due to the sim-to-real gap. Furthermore, the use of discrete action space in DQN could inhibit smoothness in certain driving manoeuvres. Future work may implement continuous control algorithms that include Deep Deterministic Policy Gradient (DDPG) to further refine action granularity.

6.CONCLUSION

Autonomous driving is what this study has intended to present using a vision-based behaviour cloning and deep reinforcement learning under simulated environment driving. Throughout perception being made through CNNs and adaptive decision making using DQNs, the resulting systems assure that the driving is robustly lane following and effectively obstacle avoiding.

With both kinds of learning paradigms merged, it gives way to a more tolerant and flexible agent for driving that could operate in real-time and dynamic environments. Testing results on UDACITY demonstrate how well this hybrid approach performs, showing advances in collision avoidance, route completion, and lane accuracy measures.

The next focus for research is tackling the issue of the sim-to-real transfer, in conjunction with more and more multimodal sensor inputs (LiDAR, radar), and finer-grained control levels through advanced reinforcement learning. As before, all of these will serve to make in-roads into closing the divide between simulation and real-world applications, thus contributing to the ideal smart transportation system.

REFERENCES

[1] Bojarski, M. et al. (2016). End to End Learning for Self-Driving Cars. arXiv preprint arXiv:1604.07316.

[2] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, 25.

[3] Codevilla, F. et al. (2018). End-to-end driving via conditional imitation learning. *IEEE International Conference on Robotics and Automation (ICRA)*.

[4] Dosovitskiy, A. et al. (2017). UDACITY: An open urban driving simulator. *Conference on Robot Learning*.

[5] Pomerleau, D. A. (1989). ALVINN: An autonomous land vehicle in a neural network. *Advances in Neural Information Processing Systems*.

[6] Chen, C. et al. (2015). DeepDriving: Learning affordance for direct perception in autonomous driving. *IEEE International Conference on Computer Vision (ICCV)*.

[7] Xu, H. et al. (2017). End-to-end learning of driving models from large-scale video datasets. *CVPR*.

[8] Mnih, V. et al. (2015). Human-level control through deep reinforcement learning. *Nature*, 518(7540), 529–533.

[9] Kendall, A. et al. (2019). Learning to drive in a day. *IEEE International Conference on Robotics and Automation (ICRA)*.

[10] Hasselt, H. V., Guez, A., & Silver, D. (2016). Deep reinforcement learning with double Q-learning. *AAAI*.

[11] Dosovitskiy, A. et al. (2017). UDACITY: An open urban driving simulator. *Conference on Robot Learning*.

[12] Pan, Y. et al. (2017). Virtual to real reinforcement learning for autonomous driving. *British Machine Vision Conference (BMVC)*.

[13] Richter, S. R. et al. (2016). Playing for data: Ground truth from computer games. *European Conference on Computer Vision (ECCV)*.

[14] Shorten, C. & Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. *Journal of Big Data*, 6(1), 60.

[15] NVIDIA Corporation. (2016). NVIDIA Self-Driving Car Learns to Drive Using AI. https://developer.nvidia.com/blog/deep-learning-self-driving-cars/

[16] Kiran, B. R. et al. (2021). Deep reinforcement learning for autonomous driving: A survey. *IEEE Transactions on Intelligent Transportation Systems*.

[17] Sallab, A. E. et al. (2017). Deep reinforcement learning framework for autonomous driving. *Electronic Imaging*, 2017(19), 70–76.