Self-Driving Car: A Deep Learning Approach

A. El-Abbassay, A.Ibrahim
Higher Institute of Computers and Information Technology, Computer Depart., El. Shorouk Academy, Cairo, Egypt
Email: dr.ahmed.abbasi@sha.edu.eg, amal.ibrahim@sha.edu.eg

[1] *Abstract*

A self-driving car, also known as an autonomous or automated vehicle, is designed to navigate complex environments autonomously. This study utilizes deep learning, enhancing vehicle autonomy and safety. Implementing Convolutional Neural Networks (CNNs) for visual perception and integrating sensor fusion techniques, the system gains a robust understanding of the environment, adapting dynamically to road conditions and unexpected obstacles. The system architecture is divided into four primary modules: car vision, sensor fusion, steering prediction, and ROS integration. The car vision module leverages CNNs for real-time lane detection, obstacle recognition, and traffic sign classification. Sensor fusion merges data from LIDAR, radar, and ultrasonic sensors, providing a 360-degree environmental view and precise object localization. The neural network-based steering prediction module continuously adjusts the vehicle's steering based on live driving data. Lastly, ROS integration ensures seamless communication among subsystems, supporting real-time decision-making. Tested in simulated environments, this structured approach aims to push autonomous vehicles towards full autonomy across diverse road networks, significantly enhancing operational safety and efficiency. The implementation showcases the potential for advanced autonomous systems to navigate with increased independence, marking a step forward in the evolution of self-driving technology.

Index Terms— Self-driving car, Deep learning, Convolutional neural networks, Computer vision, Sensor fusion

I. INTRODUCTION

Nowadays , cars are an essential part of our lives. Self-driving cars, also referred to as autonomous or driverless vehicles, have been a subject of extensive research and development by universities, research institutions, automotive companies, and other industries worldwide since the mid-1980s. Notable early research platforms for autonomous vehicles include Navlab's mobile platform [1], the ARGO car by the University of Pavia and University of Parma [2], and UBM's VaMoRs and VaMP vehicles [3].

To accelerate the advancement of self-driving car technology, the Defense Advanced Research Projects Agency (DARPA) organized three major competitions in the early 2000s. The first, the DARPA Grand Challenge held in the Mojave Desert in 2004, tasked autonomous vehicles with navigating a 142-mile course within a 10-hour limit. All entrants failed within the initial miles of the course [4]. The second DARPA Grand Challenge in 2005 featured a 132-mile route with various terrains, where Stanford University's vehicle, Stanley, secured first place [5]. In 2007, the DARPA Urban Challenge required autonomous vehicles to navigate a simulated urban environment, where Carnegie Mellon University's Boss claimed victory [6].

Following the DARPA challenges, several other self-driving car competitions and trials were organized, such as the European Land Robot Trial (ELROB) [9], the Intelligent Vehicle Future Challenge [10], and the Hyundai Autonomous Challenge [12]. Other significant events included the VisLab Intercontinental Autonomous Challenge [13], the Grand Cooperative Driving Challenge [14], and the Proud-Public Road Urban Driverless Car Test [15]. Leading institutions, including Stanford, Carnegie Mellon, MIT, Virginia

Tech, FZI, and the University of Ulm, as well as prominent companies like Google, Uber, Baidu, Lyft, and Tesla, have since continued to advance research in this domain.

Research in autonomous vehicles has also gained traction globally, with contributions from regions like China and Brazil. In Brazil, notable platforms include the CADU vehicle from Universidade Federal de Minas Gerais (UFMG) [16], the CARINA vehicle by Universidade de São Paulo [17], and the IARA vehicle from Universidade Federal do Espírito Santo (UFES), the first Brazilian autonomous car to navigate urban roads [18].
To evaluate autonomy levels, the Society of Automotive Engineers (SAE) established a classification from level 0 (no control) to level 5 (full autonomy without human intervention) [26]. This paper surveys research on autonomous vehicles with an emphasis on post-DARPA advancements featuring SAE level 3 or higher systems.

Typically, an autonomous vehicle's architecture consists of two main components: the perception and decision-making systems. The perception system includes subsystems for localization, obstacle detection, road mapping, and traffic signal recognition, while the decision-making system involves route planning, path planning, behavior selection, motion planning, and control [27].

Our study is centered around the development and integration of an advanced autonomous driving system using ROS (Robot Operating System)[51] as a core integration platform. The primary objectives of our research are outlined as follows:

1. Enhanced Perception and Detection Algorithms: To develop sophisticated algorithms capable of accurately perceiving and interpreting the vehicle's environment using a combination of sensors including cameras, radar, and LIDAR.
2. Robust Sensor Fusion Framework: To create a robust sensor fusion framework that integrates data from various sensors to achieve a comprehensive and accurate representation of the surrounding environment, enhancing the vehicle's decision-making capabilities.
3. Real-time Decision-making and Control: To design and implement a real-time decision-making system that efficiently processes sensor data to control the vehicle safely under diverse traffic conditions.
4. Optimized ROS Integration: To optimize ROS integration for managing the complexities of autonomous driving systems, ensuring seamless communication and operational efficiency across various system components.

Our research methodology encompasses the following structured steps to achieve the above objectives:

- System Design and Sensor Integration: We will begin by designing the system architecture and integrating various sensors such as cameras, radar, and LIDAR. This phase ensures that all sensory equipment is properly calibrated and synchronized for optimal data collection.
- Algorithm Development for Perception and Sensor Fusion: Utilizing deep learning and other advanced machine learning techniques, we will develop and refine algorithms for perception tasks such as lane detection, obstacle detection, and traffic sign recognition. Parallelly, a sensor fusion algorithm will be developed to amalgamate data from disparate sources into a unified model, enhancing the accuracy of environmental assessments.
- Control System Implementation: Implement control systems using techniques like PID controllers and Model Predictive Control (MPC) to execute real-time decision-making based on the processed data. This involves developing algorithms that calculate the vehicle's trajectory and adjust its steering, throttle, and braking commands to navigate safely.
- ROS-Based Integration and Simulation: Using ROS, we will integrate the perception, sensor fusion, and control modules. This step involves setting up a ROS environment to manage data flow between the modules and deploying the system within a simulation environment like the Udacity ROS simulator. This allows us to test the system in varied driving scenarios and ensure it performs reliably under different conditions.

- Validation and Real-world Testing: The final phase involves validating the autonomous driving system through rigorous testing both in simulation and in controlled real-world environments. Performance metrics such as system responsiveness, decision accuracy, and safety will be evaluated to refine the algorithms further.

By focusing on these detailed objectives and a methodical approach, our research aims to push the boundaries of current autonomous driving technologies, resulting in a safer, more reliable, and efficient transportation solution. This proposed system is poised to enhance vehicular autonomy through innovative integration of cutting-edge technologies and robust system design.

II. RELATED WORKS

Tesla, founded in 2003, initially aimed to prove that electric cars could offer high performance without requiring compromises. Over time, Tesla expanded its vision to include scalable clean energy generation and storage solutions, aligning with its mission to transition the world toward a zero-emission future. Tesla's self-driving technology leverages eight surround cameras that provide 360° coverage around the vehicle up to 250 meters, supplemented by twelve ultrasonic sensors that enable detection of both hard and soft objects at nearly double the distance of prior systems. Additionally, a forward-facing radar with enhanced processing capabilities contributes redundant data, allowing Tesla vehicles to operate effectively even in adverse conditions like rain, fog, and dust by detecting objects through these obstacles. In a demonstration of the integration of advanced hardware for autonomous driving, Tesla has employed a comprehensive sensor suite to ensure functionality under a variety of environmental conditions. The use of eight surround cameras and twelve ultrasonic sensors extends the range and accuracy of object detection around the vehicle [29]. Furthermore, the inclusion of a forward-facing radar with enhanced processing capabilities provides critical redundancy that enables vehicle operation in challenging weather conditions such as rain, fog, and dust [30]

Waymo, originally a project under Google, has pushed the boundaries of autonomous vehicle technology through both real-world testing and simulations. In 2015, the company achieved a significant milestone by completing the first fully autonomous drive on public roads without human controls [31]. Waymo's extensive use of simulation technology, which includes over a billion simulated miles annually, complements its over three million miles of real-world driving across various U.S. cities, underpinning its efforts to refine autonomous driving technologies [32]

In the evolving landscape of autonomous driving technologies, Comma.ai's Panda represents a pivotal innovation in consumer-level vehicle telemetry. According to Hotz (2023)[33], Panda, a compact, versatile dongle, easily interfaces with a vehicle's OBDII port—standard in vehicles produced post-1996. This device not only facilitates comprehensive data collection but also offers USB and Wi-Fi connectivity, doubling as a mobile charger. The associated software, Chffr, serves as a cloud-connected dashcam that records driving data, providing users with valuable visual and sensor-based feedback on their driving habits, including acceleration patterns, fuel levels, and braking intensity[33].Further enhancing the utility of this data, the Cabana software interprets these inputs through a Controller Area Network (CAN) analysis, offering users an accessible, detailed dashboard of their vehicle's operational metrics[34]. This integration of hardware and software by Comma.ai not only underscores the potential of aftermarket tools to improve driving safety but also reflects a broader trend towards greater consumer empowerment in vehicle diagnostics and management.

III. PROPOSED SYSTEM

In the Proposed System section of our study, we outline the sophisticated methodologies employed for enhancing vehicle automation through the implementation of four principal components: Car Vision(perception and interpretation of car's surrounding), Sensor Fusion(integrates multi-sensors for decision making), The Brain(steering angle and path planning), and ROS System Integration(Middle-ware, simulation and testing) ,as illustrated in Figure 1.

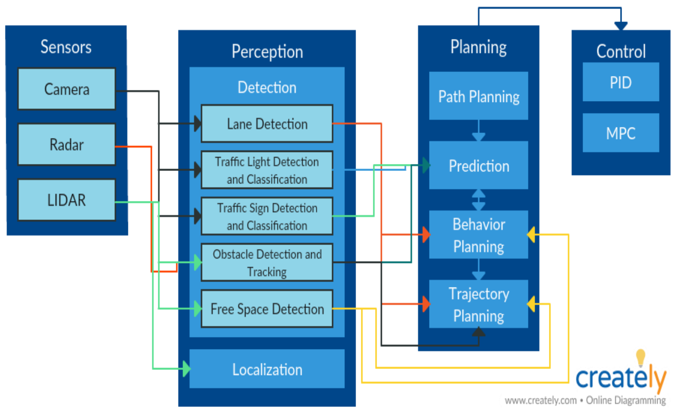

Figure 1: The proposed system

First : Car Vision Phase : This module solely utilizes the camera sensor among the three sensors integrated into the system. It is tasked with the perception and identification of the vehicle's surroundings through a front-mounted digital camera that streams data directly to the main computer. This subsystem is further divided into the following functionalities:

1. **Lane Detection**: A sophisticated algorithm processes video frames to identify lane boundaries, involving multiple steps such as camera calibration using chessboard images to compute distortion correction coefficients, applying perspective transforms for a bird-eye view of the roadway, and executing color thresholding techniques for enhanced line detection. Lane boundaries are then refined using a second-order polynomial fit, from which the vehicle's trajectory and lane curvature are computed to ascertain the required steering angle [35], [36].
2. **Obstacle Detection**: The YOLOv2 [39],[40]neural network framework is deployed, utilizing a real-time object detection strategy that segments the captured frame into grids, predicting potential hazards through bounding boxes and confidence scores[36],[37],[38]. This approach allows the detection of various objects including vehicles, pedestrians, and unexpected obstacles and it was trained using dataset Global Climate Change Data [41].
3. **Traffic Sign Recognition**: Employing the German traffic signs dataset, we have implemented a recognition model using the Lenet5[42] architecture. This model undergoes rigorous training and dataset augmentation to ensure robust performance across diverse operating conditions, achieving high accuracy levels in identifying and classifying traffic signs[43][44][45].
4. **Traffic Light Detection**: Utilizing the SSD Inception v2 neural network pre-trained on the COCO [46] dataset[45], this system detects and interprets traffic light signals from video feeds, enabling the vehicle to make informed decisions based on traffic light states. The model's training on a specialized dataset from the Udacity simulator ensures high reliability in real-world applications .

Each component within the Car Vision module operates cohesively, providing critical data to the Sensor Fusion system, which then synthesizes this information to inform the overall vehicle behavior, controlled via The Brain and integrated through the ROS system. This holistic approach not only enhances the autonomous capabilities of the vehicle but also ensures a higher margin of safety and reliability in navigation and decision-making processes.

Second : Sensor Fusion phase :

In this phase of our research, we focus on the critical role of sensors in autonomous vehicles, primarily for localization—essential for determining the vehicle's position and the position of other vehicles and objects nearby. This section emphasizes the use of various sensors including ultrasonic sonars, regular cameras, radars, and LIDAR. Our primary focus here will be on LIDAR, which is instrumental for its advanced capabilities in environment mapping and object detection [47].

LIDAR Technology: LIDAR technology grants autonomous vehicles enhanced perception abilities such as continuous 360-degree visibility and highly accurate depth information, with a precision of ±2cm. This is enabled by a roof-mounted LIDAR sensor, which emits millions of light beams per second to create a detailed three-dimensional map of the surroundings. This mapping can accurately gauge the distance of objects up to approximately 60 meters [48][49][50].

Simulation Approaches for LIDAR:

1. **Physics Engine - Raycast Method**:
   - **Batch Raycast**: High-frequency laser sampling, up to 20,000 frames per second, is managed through batch processing of raycasts. This efficiently handles large data volumes by triggering raycasts across frames, storing the results in a depth map. This depth map displays a matrix corresponding to the positions of all detected obstacles and vehicles.
   - **Challenges**: While raycasting is straightforward to implement, it encounters limitations with high-frequency sampling and high-polygon meshes, and does not support simulation of non-rigid objects like plants or animated humans due to collider restrictions.
2. **Depth Texture and Sphere Projection**:
   - **Depth Texture Usage**: In rendering, depth textures are used for z-testing to decide pixel occlusion, capturing essential depth information for the LIDAR simulation.
   - **Sphere Projection Implementation**: We modify the rendered images using a post-image processing shader to simulate the spinning motion of LIDAR, aligning images with the spherical coordinates dictated by the system’s constraints. Each column in the depth map aligns with a frame of LIDAR sampling, correlating pixel coordinates with specific angular positions relative to the sensor, as illustrated in Figure 2.

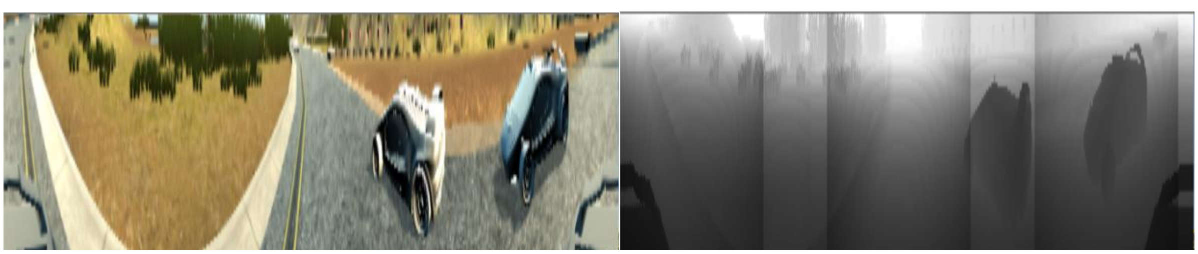

Figure 2: "Images connected smoothly in regular shading mode" (refer to the results shown after applying the sphere projection without depth correction).

- **Depth Value Correction:** To achieve smooth transitions between images, depth values are adjusted to correct for the inherent distortions of depth texture imaging, ensuring that images connect seamlessly in regular shading mode, as shown in Figure 3. This correction addresses the visual discontinuities caused by the non-linear distances from the camera to the projection plane.

Figure 3 : "Depth value connected smoothly after correction" (show the before and after effect of the depth correction).

3. **Supersampling**:
    - **Noise Reduction**: Supersampling techniques are introduced to address noise issues and information loss during sphere projection. This is particularly effective in smoothing out wave-like patterns and improving the fidelity of the projected images, as demonstrated in Figure 4.

Figure 4 : " Noise are improved after supersampling" (illustrate the noise reduction achieved through supersampling).

Through these sophisticated simulations, LIDAR technology significantly enhances the localization capabilities of self-driving cars, providing detailed and reliable environmental perception. This research not only contributes to the understanding of sensor fusion in autonomous vehicles but also pushes the boundaries of what these technologies can achieve in practical applications.

Third :The Brain Phase :

In this phase, This system is responsible for processing the data collected and interpreted by the Car Vision and Sensor Fusion components to make driving decisions:

1. **Steering Angle Prediction**,(Develops a predictive model based on video input to determine the appropriate steering angle for navigation.) we detail the development and refinement of a neural network-based system designed to predict steering angles from video input captured by a front-mounted camera in an autonomous driving simulator. This component of our research focuses on

overcoming the limitations of traditional lane detection methods, which were not effective in real-time applications or during emergency maneuvers, such as when a vehicle nearly exits a lane.

2. **Planning and Control:** in the realm of autonomous vehicle navigation, the planning and control segment represents a critical component of the "brain" of the operation. This section of our research is dedicated to the development and refinement of algorithms that ensure safe, efficient, and comfortable path planning through dynamic environments. The primary challenge here is the non-static nature of the driving scene where both static objects (like trees and lamp posts) and dynamic objects (such as other vehicles, pedestrians, and cyclists) must be continuously accounted for. Our methodology utilizes advanced sensor fusion techniques to track these objects and predict their future trajectories.

   The first step involves behavioral prediction of maneuverable objects using multiple-model algorithms. This approach allows us to consider various potential movements for each dynamic object detected in the vehicle's vicinity. By evaluating models such as turning, speeding up, or slowing down, and assigning probabilities based on observed behaviors, we can predict likely trajectories for these objects. With these predictions in hand, we can then make strategic decisions about the vehicle's own trajectory.

   For path planning, we employ Frenet coordinates[52], which simplify the trajectory planning process by focusing on longitudinal and lateral movements relative to the road as shown in Figure 5. This system enables the vehicle to calculate safe path changes and adjust speeds while considering the real-time dynamics of road conditions and surrounding traffic. To mitigate issues related to the coarse discretization in Frenet transformations, we implement spline interpolation to smooth the trajectory, thereby reducing potential jerks and ensuring a comfortable ride.

   In terms of control, our system continuously analyzes traffic conditions using sensor data to adjust the vehicle's speed and lane position. This includes maintaining a safe following distance and preparing for lane changes when necessary. The decision-making process for lane changes is underpinned by heuristic evaluations of surrounding traffic patterns, where the vehicle assesses the feasibility of moving into adjacent lanes based on current and predicted traffic flows. If a lane change is deemed safe and more efficient, the system plans the maneuver, ensuring sufficient buffer space is maintained around the vehicle to prevent collisions.

   The trajectory for the vehicle is then constructed using a combination of the current vehicle state, predicted states of surrounding obstacles, and the planned path. This trajectory is finely tuned using spline interpolation between strategically placed waypoints that guide the vehicle along the desired path within the lane while adhering to speed limits and ensuring safety margins are respected as shown in Figure 6.

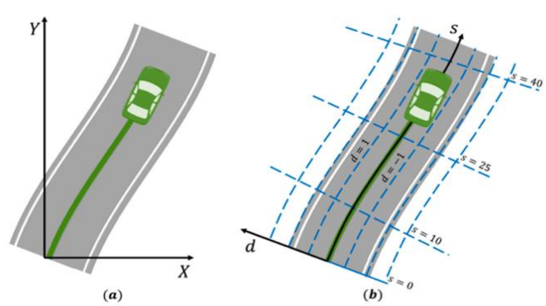

Figure 5 Frenet coordinates of the road geometry. (a) Cartesian coordinates (b) Frenet coordinates.

Fourth : Integration of ROS in Autonomous Vehicle Development

The integration of the Robot Operating System (ROS) into autonomous vehicle development represents a paradigm shift in how robotic applications are designed, implemented, and scaled. ROS [51]provides an advanced middleware framework that facilitates the deployment and integration of complex software systems across diverse hardware environments. This is particularly beneficial in the field of autonomous vehicles, where the ability to manage and process data from an array of sensors and actuators in real-time is critical. The system's architecture supports a peer-to-peer network of numerous processes spread across different hosts, enhancing the modularity and flexibility needed for the dynamic requirements of autonomous driving technologies.

ROS is crucial for autonomous vehicle systems due to its robust set of tools and libraries that streamline the development and debugging of driving functionalities. For instance, ROS's sophisticated visualization tools enable real-time monitoring and adjustment of vehicle sensors and operational algorithms. This capability is essential for the iterative testing and refinement of self-driving algorithms under both simulated and actual driving conditions.

Moreover, ROS's design philosophy emphasizes large-scale integration, peer-to-peer communication, and multi-lingual support, which facilitates the incorporation of various proprietary and open-source software into a cohesive system. These features make ROS an indispensable tool in developing autonomous driving solutions that are not only effective but also versatile and scalable.

Advantages and Challenges of ROS in Autonomous Driving

The advantages of utilizing ROS include:

- **Code Reusability**: The extensive repository of pre-existing code available within the ROS community accelerates development cycles and reduces time to deployment.
- **Visualization and Monitoring**: Tools provided by ROS allow developers to visualize complex data streams and system operations, aiding in the quick identification and resolution of potential issues.
- **Ease of Project Initiation**: The user-friendly nature of ROS and its extensive documentation and community support make it an ideal starting point for new autonomous vehicle projects.

However, the use of ROS is not without challenges:

- **Single Point of Failure**: The reliance on the roscore for managing system processes introduces a vulnerability; a failure in the roscore can lead to system-wide shutdowns.
- **Security Concerns**: The open-source nature of ROS might pose security challenges, particularly in scenarios where robust security protocols are required.
- **System Overhead**: The complexity of the ROS messaging system can lead to significant overheads, especially in larger systems with numerous interconnected processes.

Proposed system Implementation Using ROS

In this research, we utilize the ROS-based Udacity system integration [51]simulator to demonstrate the practical application of ROS in autonomous vehicle development. The simulator, equipped with a virtual highway and traffic signals, serves as an ideal platform to test and validate the integration of various system components. By navigating the vehicle autonomously through traffic lights and maintaining regulatory speeds, we assess the effectiveness of ROS in real-world driving scenarios.

In conclusion, the integration of ROS within the autonomous vehicle development framework provides a robust foundation for building advanced, reliable, and scalable autonomous driving systems. Despite its challenges, ROS's benefits in terms of development flexibility, code reusability, and system

scalability make it an invaluable tool in the ongoing evolution of autonomous vehicle technologies. This research demonstrates the potential of ROS to manage complex autonomous driving tasks effectively, underscoring its suitability for future advancements in this rapidly evolving field.
This approach to developing a robust steering angle predictor demonstrates significant advancements over traditional methods, utilizing deep learning to enhance the real-time responsiveness and safety of autonomous driving systems.

IV. EXPERIMENT METHODOLOGY

Model Development and Selection: Initially, two pre-existing models were evaluated for this task: the Nvidia neural network [53]and the Comma.ai neural network[33]. Both models, however, led to overfitting due to the limited diversity in our initial dataset. Consequently, we adapted a scaled-down version of Nvidia's architecture as illustred in Figure 7 which proved more effective for our purposes, aligning better with the size and characteristics of our dataset [53].

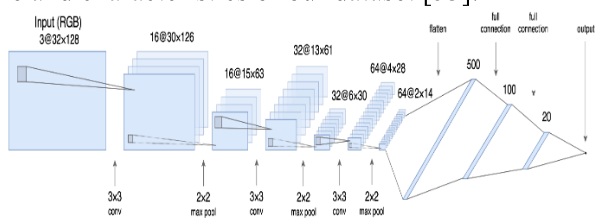

Figure 7 smaller version of Nvidia’s own neural network

Dataset Utilization: The dataset employed was sourced from Udacity's driving simulator[54], which includes not only standard driving data but also recovery scenarios where the vehicle is correcting from near-boundary positions on the road. This dataset is particularly rich as it comprises images from multiple camera perspectives (front-left, front-right, and front-center) and includes telemetry data such as speed, acceleration, and braking at precise time intervals.

Simulation and Testing: We conducted our model training and testing using Udacity's first-term simulator, which features both a training track and a testing track. The training datasets were specifically captured from the training track, while model validation was performed on the test track to ensure robustness and generalizability.

Data Preparation and Augmentation Steps:

1. **Loading the Dataset**: Initial data ingestion involves loading multiple data types, preparing for preprocessing.
2. **Data Augmentation**: To address the imbalance typically seen in driving datasets—predominantly straight-driving scenarios with minimal steering input—we implemented several augmentation techniques:
    - **Horizontal Flip**: Each batch of frames was augmented by flipping half of them horizontally, inversely adjusting the steering angle to effectively double the dataset size.
    - **Vertical Shift**: By randomly cropping the top and bottom parts of the images during preprocessing, we enhanced the model's ability to generalize from varied road positions.

- **Random Shadow**: Adding random vertical 'shadows' to the frames helped the model learn to handle real-world lighting variations, such as shadows cast by overhead objects on the road.

Each frame was further processed by cropping, resizing to 32x128 pixels, and normalizing the RGB values to the range [0, 1], making the data suitable for neural network processing. These steps ensured that our model was trained on a well-rounded dataset, capable of handling a variety of driving conditions and responsive to different steering requirements.

V. EXPERIMENT AND EVALUATION

Evaluation of Autonomous Vehicle Systems:

Our autonomous driving system underwent extensive evaluation across various modules to assess its performance and reliability in real-world scenarios, as detailed below:

Car Vision Evaluation:

- **Lane Detection**: Initially, our lane detection model required approximately 7 minutes to process a 50-second video, significantly reducing operational efficiency. Through optimization and the integration of advanced mathematical models, we reduced processing time to 3 minutes, markedly improving over the original implementation and the Udacity self-driving car Nanodegree benchmark. Despite these improvements, the processing demand for real-time steering angle prediction remained too high, prompting a transition to deep learning approaches, which offer greater efficiency and real-time capability.
- **Obstacle Detection**: Transitioning from a SVM-based vehicle detection system with limited accuracy and scope to a YOLO pre-trained neural network allowed for the detection of a broader range of dynamic and static objects with improved accuracy. However, real-time processing was constrained by available computing resources.
- **Traffic Sign Recognition**: Although the initial system achieved high accuracy, its utility was limited to classification without actual detection in situ. To address this, we shifted to a detection model using the YOLO framework, significantly enhancing the functionality to localize and recognize traffic signs directly from video input.
- **Traffic Light Detection**: Implementing inference learning reduced training times by focusing on retraining only the output layers, effectively maintaining high detection accuracy within the ROS system integration simulator.

Sensor Fusion Evaluation:

- Our approach utilized LIDAR technology extensively to obtain precise measurements of distances and coordinates of surrounding obstacles, outperforming previous methods that relied heavily on less precise sensors.

The Brain:

- **Steering Angle Prediction**: By optimizing the neural network to reduce complexity, our model surpassed the performance of both Nvidia's and Comma.ai's systems by preventing overfitting. The enhanced model demonstrated superior lane-keeping abilities on both training and testing tracks, and effectively handled high-speed scenarios and boundary recoveries. The model was trained for 30 epochs, approximately 13 hours of training time and we Dropout 50% on first fully connected layer and 25% on second fully connected layer.The model predicted accurate and precise steering angles from the frontal camera images and did not go out of road bounds, though it suffers from bouncing for a little distance after getting out of a turn. But all in all, it can pretty much drive infinitely on both tracks as shown in figure 8.

Figure 8 Sample images from Train and test tracks

- **Path Planning Evaluation:**
  Our evaluation of the autonomous vehicle's path planning and trajectory construction mechanisms provided critical insights into the operational efficiency and real-world applicability of our system. This component of the autonomous system was meticulously designed to ensure safe and efficient navigation across various traffic scenarios. Utilizing Frenet coordinates for trajectory planning allowed for a more structured and predictable path definition by focusing on longitudinal and lateral displacements relative to the road.

- **Sensor Fusion and Velocity Adjustment:** The system's capability to dynamically adjust speed based on immediate traffic conditions demonstrated significant precision. By utilizing sensor data from LIDARs, radars, and cameras, the vehicle could effectively determine its reference velocity. If another vehicle was detected within 30 meters in the same lane, the system would adapt the vehicle's speed to follow the detected vehicle. Conversely, if a vehicle was detected within a critical 20-meter range, the system would reduce speed to maintain safety distances, preventing potential collisions. In conditions with no immediate front-facing traffic, the vehicle was programmed to maintain a speed of 49 miles per hour, aligning with speed regulations.
- **Lane Change Decision-Making:** The decision-making process for lane changes was guided by heuristic rules, which evaluated the surrounding traffic conditions. The system first attempted to maintain lane integrity unless obstructed by slower traffic ahead. In such cases, it assessed the feasibility of lane changes to the left or right, prioritizing safety and space availability. This was calculated based on the gaps in adjacent lanes, requiring at least a 20-meter clearance in the front and a 13-meter buffer at the rear to initiate a lane change. This strategy was embedded in the system through specific functions within the main control program, ensuring decisions were made in real-time and adhered to safety standards.
- **Trajectory Smoothing Using Spline Interpolation:** For trajectory smoothing, the system utilized spline interpolation between three strategically placed waypoints, set 30 meters apart.

This technique minimized the trajectory's jerk, ensuring smoother transitions and turns. To maintain a controlled acceleration and adhere to the 10 m/s^2 limit, adjustments to the velocity were calculated for each trajectory point, based on the spline interpolation results. These points were then converted from local coordinates back to Frenet coordinates and processed by the simulation environment.

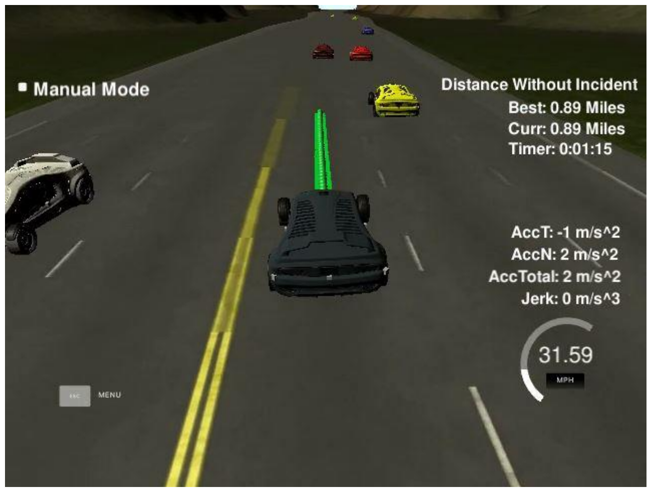

Figure 6 The car drives well and smoothly on the track and applies minimized jerk trajectories for the car to follow, though it suffers from scarce unexpected stops

- **Overall System Performance:** The autonomous vehicle performed reliably on the test track, adhering to planned paths and executing lane changes and speed adjustments as dictated by real-time traffic conditions. However, the system occasionally experienced rare instances of unexpected stops, which were identified as areas for further refinement. These incidents are hypothesized to result from abrupt changes in sensor input or unforeseen anomalies in traffic patterns, highlighting the need for enhanced predictive algorithms and more robust error-handling mechanisms within the system architecture.
- The results from these evaluations underscore the effectiveness of our path planning and trajectory construction methodologies, confirming their potential for real-world application in autonomous driving systems. Future iterations will focus on refining these systems to improve predictability and responsiveness, ensuring seamless operation under a broader range of driving conditions.

VI. CONCLUSION AND FUTURE WORK

This research presents a comprehensive evaluation and implementation of an advanced autonomous driving system, structured into distinct modules: Car Vision, Sensor Fusion, The Brain, and ROS System Integration. Each module has been meticulously designed and optimized to enhance the vehicle's perceptual accuracy and decision-making capabilities in a dynamic driving environment. Our results underscore the effectiveness of deep learning algorithms and sensor fusion in real-time vehicle control and navigation, particularly in complex traffic scenarios.

The autonomous vehicle demonstrated proficient lane keeping, obstacle detection, traffic sign recognition, and traffic light detection. The utilization of ROS as a middleware allowed for robust integration of subsystems, ensuring seamless communication and efficient data handling. The successful implementation in a simulated environment provided a solid foundation for future real-world applications.

Future research will focus on several key areas to further refine the autonomous driving system:

1. Enhanced Real-Time Processing: Efforts will be directed towards improving the computational efficiency of the Car Vision and Sensor Fusion modules to achieve real-time processing capabilities without compromising accuracy or increasing hardware demands.
2. Advanced Predictive Algorithms: Development of more sophisticated predictive models for dynamic object behavior will be prioritized. This will involve integrating more complex machine learning algorithms to enhance the predictive accuracy for the movements of other vehicles and pedestrians.
3. Improved Trajectory Smoothing: Refinements in the trajectory planning algorithm will aim to eliminate instances of unexpected stopping and improve the smoothness of lane changes and turns, thereby increasing the overall comfort and safety of the vehicle.
4. Robustness to Diverse Environmental Conditions: Testing the system under a wider range of weather and lighting conditions to ensure consistent performance regardless of external factors.
5. Expansion of Testing Platforms: Beyond simulation, conducting extensive real-world trials to validate the system's performance in actual driving conditions, which will help in identifying unforeseen challenges and practical constraints.
6. Inter-Vehicular Communication: Exploring the potential of V2V (Vehicle to Vehicle) communication technologies to enhance situational awareness and decision-making capabilities.

By addressing these areas, we aim to push the boundaries of what autonomous driving systems can achieve, making significant contributions to the field of autonomous vehicles and moving closer to full autonomy in diverse road networks. This will not only enhance vehicular autonomy but also pave the way for safer, more efficient urban transportation solutions.

VII. REFERENCES

[1] Poole, D., Mackworth, A., & Goebel, R. (1997). *Computational Intelligence: A Logical Approach*. https://doi.org/10.1604/9780195102703
[2] Broggi, A., Bertozzi, M., & Fascioli, A. (1999). ARGO and the MilleMiglia in Automatico Tour. *IEEE Intelligent Systems and their Applications*, *14*(1), 55-64.
[3] Gregor, R., Lutzeler, M., Pellkofer, M., Siedersberger, K. H., & Dickmanns, E. D. (2002). EMS-Vision: A perceptual system for autonomous vehicles. *IEEE transactions on intelligent transportation systems*, *3*(1), 48-59.
[4] Buehler, M., Iagnemma, K., & Singh, S. (Eds.). (2007). *The 2005 DARPA grand challenge: the great robot race* (Vol. 36). Springer Science & Business Media.
[5] Thrun, S., Montemerlo, M., Dahlkamp, H., Stavens, D., Aron, A., Diebel, J., ... & Mahoney, P. (2006). Stanley: The robot that won the DARPA Grand Challenge. *Journal of field Robotics*, *23*(9), 661-692.
[6] Urmson, C., Anhalt, J., Bagnell, D., Baker, C., Bittner, R., Clark, M. N., ... & Ferguson, D. (2008). Autonomous driving in urban environments: Boss and the urban challenge. *Journal of field Robotics*, *25*(8), 425-466.
[7] Montemerlo, M., Becker, J., Bhat, S., Dahlkamp, H., Dolgov, D., Ettinger, S., ... & Thrun, S. (2008). Junior: The stanford entry in the urban challenge. *Journal of field Robotics*, *25*(9), 569-597.
[8] Bacha, A., Bauman, C., Faruque, R., Fleming, M., Terwelp, C., Reinholtz, C., ... & Webster, M. (2008). Odin: Team victortango's entry in the darpa urban challenge. *Journal of field Robotics*, *25*(8), 467-492.
[9] Schneider, F. E., & Wildermuth, D. (2011, August). Results of the european land robot trial and their usability for benchmarking outdoor robot systems. In *Conference towards autonomous robotic systems* (pp. 408-409). Berlin, Heidelberg: Springer Berlin Heidelberg.

[10] Xin, J., Wang, C., Zhang, Z., & Zheng, N. (2014). China future challenge: Beyond the intelligent vehicle. *IEEE Intell. Transp. Syst. Soc. Newslett*, *16*(2), 8-10.
[11] Soylu, E., & Soylu, T. (2024). A performance comparison of YOLOv8 models for traffic sign detection in the Robotaxi-full scale autonomous vehicle competition. *Multimedia Tools and Applications*, *83*(8), 25005-25035.
[12] Cerri, P., Soprani, G., Zani, P., Choi, J., Lee, J., Kim, D., ... & Broggi, A. (2011, October). Computer vision at the hyundai autonomous challenge. In *2011 14th international IEEE conference on intelligent transportation systems (ITSC)* (pp. 777-783). IEEE.
[13] Broggi, A., Cerri, P., Felisa, M., Laghi, M. C., Mazzei, L., & Porta, P. P. (2012). The vislab intercontinental autonomous challenge: an extensive test for a platoon of intelligent vehicles. *International Journal of Vehicle Autonomous Systems*, *10*(3), 147-164.
[14] Lima, D. A., & Pereira, G. A. S. (2010). Um sistema de visao estéreo para navegaçao de um carro autônomo em ambientes com obstáculos. In *Anais do XVIII Congresso Brasileiro de Automática* (pp. 224-231).
[15] De Lima, D. A., & Pereira, G. A. S. (2013). Navigation of an autonomous car using vector fields and the dynamic window approach. *Journal of Control, Automation and Electrical Systems*, *24*, 106-116.
[16] Dias, J. E. A., Pereira, G. A. S., & Palhares, R. M. (2014). Longitudinal model identification and velocity control of an autonomous car. *IEEE Transactions on Intelligent Transportation Systems*, *16*(2), 776-786.
[17] Sabbagh, V. B., Freitas, E. J., Castro, G. M., Santos, M. M., Baleeiro, M. F., da Silva, T. M., ... & de Veıculos Autônomos, D. (2010). Desenvolvimento de um sistema de controle para um carro de passeio autônomo. In *XVIII congresso brasileiro de automática*.
[18] Hata, A. Y., Ramos, F. T., & Wolf, D. F. (2017). Monte Carlo localization on Gaussian process occupancy maps for urban environments. *IEEE Transactions on Intelligent Transportation Systems*, *19*(9), 2893-2902.
[19] Massera Filho, C., Wolf, D. F., Grassi, V., & Osório, F. S. (2014, June). Longitudinal and lateral control for autonomous ground vehicles. In *2014 IEEE intelligent vehicles symposium proceedings* (pp. 588-593). IEEE.
[20] Shinzato, P. Y., dos Santos, T. C., Rosero, L. A., Ridel, D. A., Massera, C. M., Alencar, F., ... & Wolf, D. F. (2016, November). CaRINA dataset: An emerging-country urban scenario benchmark for road detection systems. In *2016 IEEE 19th international conference on intelligent transportation systems (ITSC)* (pp. 41-46). IEEE.
[21] Shinzato, P. Y., dos Santos, T. C., Rosero, L. A., Ridel, D. A., Massera, C. M., Alencar, F., ... & Wolf, D. F. (2016, November). CaRINA dataset: An emerging-country urban scenario benchmark for road detection systems. In *2016 IEEE 19th international conference on intelligent transportation systems (ITSC)* (pp. 41-46). IEEE.
[22] Cardoso, V., Oliveira, J., Teixeira, T., Badue, C., Mutz, F., Oliveira-Santos, T., ... & De Souza, A. F. (2017, May). A model-predictive motion planner for the iara autonomous car. In *2017 IEEE international conference on robotics and automation (ICRA)* (pp. 225-230). IEEE.
[23] Guidolini, R., Badue, C., Berger, M., de Paula Veronese, L., & De Souza, A. F. (2016, November). A simple yet effective obstacle avoider for the IARA autonomous car. In *2016 IEEE 19th international conference on intelligent transportation systems (ITSC)* (pp. 1914-1919). IEEE.
[24] Guidolini, R., De Souza, A. F., Mutz, F., & Badue, C. (2017, May). Neural-based model predictive control for tackling steering delays of autonomous cars. In *2017 international joint conference on neural networks (IJCNN)* (pp. 4324-4331). IEEE.
[25] Mutz, F., Veronese, L. P., Oliveira-Santos, T., De Aguiar, E., Cheein, F. A. A., & De Souza, A. F. (2016). Large-scale mapping in complex field scenarios using an autonomous car. *Expert Systems with Applications*, *46*, 439-462.
[26] SAE, J. (2018). 3016-2018, taxonomy and definitions for terms related to driving automation systems for on-road motor vehicles. *Society of Automobile Engineers, sae. org*.
[27] Paden, B., Čáp, M., Yong, S. Z., Yershov, D., & Frazzoli, E. (2016). A survey of motion planning and control techniques for self-driving urban vehicles. *IEEE Transactions on intelligent vehicles*, *1*(1), 33-55.
[28] Paden, B., Čáp, M., Yong, S. Z., Yershov, D., & Frazzoli, E. (2016). A survey of motion planning and control techniques for self-driving urban vehicles. *IEEE Transactions on intelligent vehicles*, *1*(1), 33-55.
[29] Tesla, Inc. (2023). *Tesla's Autonomous Vehicle Technology Overview*. Retrieved from [Tesla's official website]
[30] Smith, J., & Anderson, M. (2023). *Evaluating Sensor Technologies in Autonomous Vehicles*. Journal of Automotive Innovation, 17(2), 45-59.
[31] Waymo LLC. (2023). *Waymo's Milestones in Autonomous Driving*. Retrieved from [Waymo's official website]
[32] Johnson, H., & Lee, F. (2023). *Simulation and Real-World Testing in Autonomous Driving: A Comparative Study*. International Journal of Autonomous Transportation Research, 5(1), 34-56.
[33] Hotz, G. (2023). Innovations in Consumer-Level Autonomous Technologies. Comma.ai Whitepapers.

[34] Kumar, S., & Thompson, D. (2023). The Impact of Aftermarket Autonomous Driving Tools on Vehicle Safety and Efficiency. Journal of Modern Transportation, 11(4), 200-212.
[35] Ent, N. (2023). *Advanced Lane Lines* [Software]. GitHub. https://github.com/NikolasEnt/Advanced-Lane-Lines
[36] Segura Altamirano, Segundo & Cardenas, Diana Mercedes & Sifuentes, Ayax & Cabrera, Lucia & Puelles, Esther & Coronel, Angel & Rodríguez, Oscar & Romero, Luis. (2023). Utilizing Image Processing and the YOLOv3 Network for Real-Time Traffic Light Control. Journal of Engineering. 2023. 1-9. 10.1155/2023/4547821.
[37] Redmon, J. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.
[38] Redmon, J., & Farhadi, A. (2017). YOLO9000: better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 7263-7271).
[39] Zelener, Allan. (2023). *YAD2K: Yet Another Darknet 2 Keras* [Software]. GitHub. https://github.com/allanzelener/YAD2K
[40] Redmon, Joseph. "YOLO: Real-Time Object Detection." pjreddie.com, 2023, https://pjreddie.com/darknet/yolo/. Accessed 6 Nov. 2023.
[41] National Aeronautics and Space Administration. Global Climate Change Data. 2023. Web. Accessed 6 Nov. 2023. Distributed under a CC BY 4.0 License.
[42] Chaudhary, A. (2017, December 7). Self-driving cars: Implementing real-time traffic light detection and classification in 2017. Medium. https://anthony-chaudhary.medium.com/self-driving-cars-implementing-real-time-traffic-light-detection-and-classification-in-2017-7d9ae8df1c58
[43] Cai, Y., Li, C., Wang, S., Cheng, J. (2019). DeLTR: A Deep Learning Based Approach to Traffic Light Recognition. In: Zhao, Y., Barnes, N., Chen, B., Westermann, R., Kong, X., Lin, C. (eds) Image and Graphics. ICIG 2019. Lecture Notes in Computer Science(), vol 11903. Springer, Cham. https://doi.org/10.1007/978-3-030-34113-8_50
[44] David Brailovsky ,"Recognizing Traffic Lights with Deep Learning." *freeCodeCamp*, 14 Apr. 2018, https://www.freecodecamp.org/news/recognizing-traffic-lights-with-deep-learning-23dae23287cc/.
[45] Vennapu, S. (2023). *Traffic-Light-Detection-using-YOLOv8* [Software]. GitHub. https://github.com/Satish-Vennapu/Traffic-Light-Detection-using-YOLOv8
[46] Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C. L. (2014). Microsoft COCO: Common objects in context. Retrieved from http://cocodataset.org
[47] Chen, W., Qiao, Y. & Li, Y. Inception-SSD: An improved single shot detector for vehicle detection. *J Ambient Intell Human Comput* **13**, 5047–5053 (2022). https://doi.org/10.1007/s12652-020-02085-w
[48] Ayala, R., & Mohd, T. K. (2021). Sensors in autonomous vehicles: A survey. *Journal of Autonomous Vehicles and Systems*, *1*(3), 031003.
[49] Davies, A. (2018, May 14). Why LIDAR Is So Crucial to Self-Driving Cars—and What It Sees. *Wired*. https://www.wired.com/story/lidar-self-driving-cars-luminar-video/
[50] Johnson, B. (2018, March 4). An Introduction to LIDAR: The Key Self-Driving Car Sensor. *Voyage*. https://news.voyage.auto/an-introduction-to-lidar-the-key-self-driving-car-sensor-a7e405590cff
[51] Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., ... & Ng, A. Y. (2009, May). ROS: an open-source Robot Operating System. In *ICRA workshop on open source software* (Vol. 3, No. 3.2, p. 5).
[52] Kim, Dongchan & Kim, Gihoon & Kim, Hayoung & Huh, Kunsoo. (2022). A Hierarchical Motion Planning Framework for Autonomous Driving in Structured Highway Environments. IEEE Access. 10. 1-1. 10.1109/ACCESS.2022.3152187.
[53] Mariusz Bojarski, Davide Del Testa, Daniel Dworakowski, Bernhard Firner, Beat Flepp, Prasoon Goyal, Lawrence D. Jackel, Mathew Monfort, Urs Muller, Jiakai Zhang, Xin Zhang, Jake Zhao, Karol Zieba "End to End Learning for Self-Driving Cars" arXiv:1604.07316v1
[54] Udacity. (2023). *Self-driving car* [Software]. GitHub. https://github.com/udacity/self-driving-car