Advanced Graph Editor for Autonomous Vehicle Collision Avoidance Simulation

Project Reference No.: *47S_BE_0918*

College ***:*** *REVA University,Bengaluru*
Branch ***:*** *Department of Computer Science and Engineering*
Guide(s) ***:*** *Prof. Shilpa. V*
Student(S) ***:*** *Ms. Bhanu Priya J. S.*
Ms. Bhavya Shivani
Ms. Lavanya C. S.

Keywords:

Autonomous vehicles, collision avoidance, simulation framework, neural networks, real-world data integration.

Introduction:

Simulation of Urban Mobility (SUMO) Overview: SUMO is a traffic simulation tool designed to model urban mobility. It offers capabilities to simulate complex traffic scenarios, evaluate different transportation modes, and assess traffic management strategies. SUMO's flexibility and extensibility make it suitable for research, education, and transportation planning. It integrates with other tools and frameworks, enhancing its functionality for various applications. CARLA: Urban Driving Simulator: CARLA is an open-source simulator focused on urban driving scenarios. It supports autonomous driving research with its realistic rendering engine, high-fidelity sensor simulation, and diverse built-in environments. CARLA's integration with reinforcement learning algorithms allows training and evaluating autonomous agents, supported by a robust community. Realistic Simulation Environments: Real-world testing challenges for autonomous vehicles include vehicle dynamics, sensor behavior, and environmental conditions. Simulation platforms, especially open-source ones with machine learning integration, are crucial for addressing these challenges. Collision Avoidance Algorithms: Evaluating collision avoidance algorithms in real-time driving simulators under various scenarios provides insights to enhance these systems. Neural network-based control systems predict collision risks and generate control actions, improving safety and reliability. These approaches enhance collision avoidance performance, offering adaptive capabilities for safer navigation. Enhancing Simulation Realism Employing graph editing techniques in simulation environments improves fidelity, dynamic interactions, and scenario diversity, crucial for robust autonomous vehicle testing.

Objectives:

Develop a user-friendly graph editor tool for creating complex and realistic simulation environments, accurately replicating real-world road layouts, traffic patterns, and environmental conditions. Integrate neural network architectures for AI-driven navigation, enabling the training and evaluation of autonomous vehicles with advanced perception and control. Incorporate real-world data sources like OpenStreetMap to enhance simulation fidelity. Implement genetic algorithms to optimize collision avoidance algorithms, allowing continuous improvement and ensuring robustness and effectiveness in diverse driving scenarios.

Methodology:

The car's functionality revolves around managing its dynamics and control mechanisms. It responds to user inputs for forward, reverse, left, and right, adjusting its speed and angle. Friction gradually slows the car when controls are inactive, updating its position based on its current speed and angle. The car can also operate autonomously via neural network integration, using sensor readings to guide its actions.

The Road class focuses on rendering the road, including borders and lane lines. The Sensor class updates readings based on ray intersections with road borders or vehicles. The Car class handles collision detection by evaluating intersections with road borders or other cars, marking damage upon detection. Multiple Car instances, each with its neural network and sensors, simulate traffic, enabling interactions with other vehicles and the road.

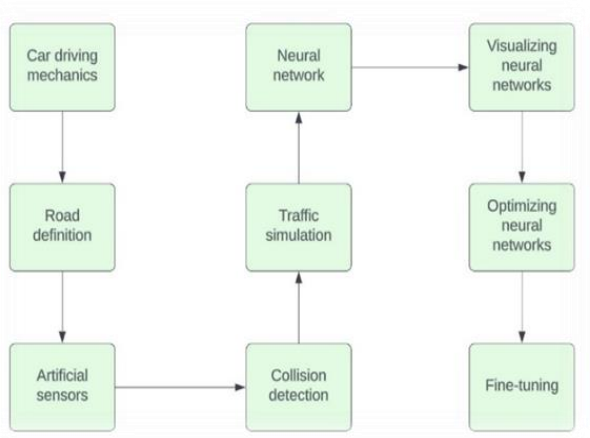

Fig 1. Flowchart

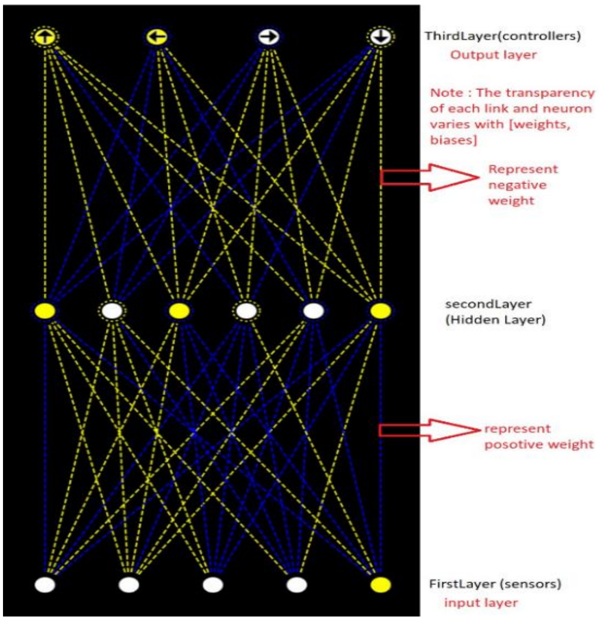

Fig 2. Neural Network

The Neural Network class, the AI system's core, consists of multiple Level objects representing layers. The feedForward method processes inputs through the network, producing outputs. The mutate method, using a genetic algorithm, randomly adjusts neuron weights and biases to create variations. The Visualizer class visually represents the neural network, using distinct visual cues for different weights and biases.

Optimization relies on the mutate function, which adjusts biases and weights via small random variations, similar to a genetic algorithm. This introduces diversity into the solution space, though it may be less efficient than techniques like gradient descent. The project uses a neural network to control a simulated self-driving car, processing sensor inputs and outputting control signals. The car interacts with a simulated road and other cars, and its behavior is determined by its physical properties and network control signals. Visualization aids analysis and debugging, enhancing the network's performance and adaptability in the simulation.

Conclusion:

The results showcase the successful development and implementation of an advanced graph editor and tailored explicitly for autonomous vehicle (AV) collision avoidance simulations. Through meticulous design and rigorous testing, the tool offers researchers and developers a comprehensive platform for creating and evaluating complex scenarios in AV environments. Utilizing the developed simulation framework, various real-world scenarios were simulated, encompassing diverse road layouts, dynamic traffic patterns, and unpredictable environmental conditions. These simulations yielded promising outcomes, highlighting the effectiveness and robustness of collision avoidance algorithms under different circumstances.

In conclusion, the development of an advanced graph editor tailored for autonomous vehicle collision avoidance simulations offers significant contributions to the field of AV technology. The project successfully addresses key challenges in simulation frameworks by providing a user-friendly interface, efficient graph manipulation features, and seamless integration with neural networks. By minimizing user input and optimizing mouse interactions, the tool enhances user experience and streamlines the process of creating intricate road networks. The integration of real-world data from sources like OpenStreetMap enriches the simulation environment, enabling researchers to develop accurate and immersive simulations for AV testing. Furthermore, the project's emphasis on neural network integration facilitates AI-driven navigation within the simulation, empowering researchers to develop and validate collision avoidance algorithms effectively.

Scope for future work:

This project has significant potential for future enhancements. Integrating advanced sensors such as LiDAR, radar, and cameras can improve environmental perception and data accuracy. Incorporating cutting-edge neural networks and deep learning models can refine decision-making and vehicle control. Utilizing real-world driving data for training and validation will increase reliability and performance. Expanding the simulation to include more complex and varied traffic scenarios, pedestrian interactions, and multi-vehicle coordination will enhance scalability. Improving the user interface for intuitive interaction and ensuring seamless cross-platform compatibility will further broaden its accessibility. Additionally, implementing V2X (vehicle-to-everything) communication technology can enhance the simulation's realism and interactivity, paving the way for more comprehensive autonomous driving research and development.