Design and Implementation of Real-Time Obstacle Detection and Automatic Braking for Collision Prevention

Gowda Yashvi Manjunath, Ranjitha H, Impana P.S, Guide: Chaithra K, Assistant Professor
Department of Electronics and Communication Engineering
P.E.S. College of Engineering, Mandya , Karnataka, India

Abstract — Road safety is a very important problem in today's transportation systems. Rear-end collisions are among the most frequent and dangerous accidents on highways and city streets. The paper proposes a MATLAB based simulation of an intelligent obstacle detection and autonomous lane changing system for collision avoidance. We consider a scenario where an ego vehicle drives on a two-lane road and continuously observes a slower vehicle in front and a fast-approaching vehicle in the adjacent lane. The system features front obstacle detection, blind spot monitoring, adaptive speed control, and smooth lane-change execution based on a rule-based decision-making algorithm. When the vehicle in front enters the pre-set detection range of 45 m, the system checks the side lane for safety before beginning an overtaking maneuver. In case of a detected speed-risk vehicle in the blind spot zone, the lane change is delayed, and a warning alert is issued. Simulation results show successful collision avoidance, safe gap maintenance, and smooth overtaking operations, validating the effectiveness of the system as a simplified Advanced Driver Assistance System (ADAS) prototype. Index Terms — Collision prevention, lane change, blind spot detection, ADAS, V2V communication, obstacle detection, MATLAB simulation.

Keywords— ADAS, autonomous vehicles, blind spot detection, collision prevention, lane change, MATLAB simulation, obstacle detection, V2V communication.

I. INTRODUCTION

Road safety is a major concern across the globe because of the rising number of vehicular accidents resulting from delayed driver reaction, poor awareness of obstacles, and unsafe overtaking choices. Rear-end collisions are among the most common and deadly accidents on highways and urban roads. According to the World Health Organization, approximately 1.35 million people (about the population of Maine) die each year in road traffic crashes, with a significant proportion of those fatalities caused by human error in detecting and responding to obstacles. Such accidents are mitigated by modern Advanced Driver Assistance Systems (ADAS) that continuously observe the surrounding environment and assist drivers in making timely and safe decisions. Vehicle-to-Vehicle (V2V) communication also enhances situational awareness by enabling vehicles to exchange real-time information on position, speed, and intent. Based on these ideas, this paper develops a MATLAB simulation of smart obstacle detection and lane-changing assistance system for autonomous vehicles.

The simulation considers an ego vehicle driving on a two-lane road while keeping track of a slower vehicle in front and a rapidly approaching vehicle in the neighboring lane. By continuously measuring the distance between vehicles and checking blind spot conditions, the system can assess the surrounding traffic situation and support safe lane-changing decisions. The system independently evaluates the surrounding traffic conditions and determines the most appropriate action, such as continuing at the current speed, reducing speed, waiting for a safer opportunity, or performing an overtaking action when conditions permit. To achieve this, the simulation uses a rule-based decision-making approach that continuously monitors the vehicle's environment. It identifies obstacles ahead, maintains a safe distance from other vehicles, checks blind spot areas, and determines the right time to change lanes. The system also ensures smooth lane transitions and adjusts the vehicle's speed appropriately during overtaking, contributing to safer and more efficient driving behavior.

While the proposed model does not include advanced features such as sensor fusion, detailed vehicle dynamics, or AI-driven path planning, it successfully demonstrates the core concepts behind intelligent driving assistance systems used in modern autonomous vehicles. The simulation provides a clear understanding of how vehicles can detect obstacles, assess surrounding traffic conditions, and make safe driving decisions in real time. The rest of this paper is structured as follows. Section II presents a review of related studies, Section III outlines the problem statement and objectives, Section IV describes the proposed methodology, Section V explains the system architecture and block diagram, Section VI discusses the simulation results and analysis, and Section VII concludes the paper with future research directions and possible enhancements.

II. LITERATURE SURVEY

Guo et al. [1] explored the application of MATLAB/Simulink for designing and testing intelligent vehicle systems, including obstacle detection and adaptive cruise control. Their study showed that simulation environments provide an efficient and reliable way to evaluate collision avoidance strategies before moving to real-world hardware implementation.

Thompson et al. [2] introduced an adaptive nonlinear Model Predictive Control (MPC) approach for obstacle avoidance in autonomous vehicles. The proposed method intelligently calculated the required braking force while maintaining passenger comfort. Simulation results indicated better stopping performance and greater accuracy when compared with traditional threshold-based braking techniques.

Oke et al. [3] developed a robust control system aimed at improving vehicle safety and ride quality through active suspension control and safe following-distance maintenance. By continuously monitoring the relative distance and speed between vehicles, the system adjusted acceleration and braking actions accordingly. Both simulation and experimental results confirmed a reduction in collision risk along with enhanced passenger comfort.

Lan et al. [4] proposed a real-time obstacle detection framework for urban traffic environments, emphasizing Time-to-Collision (TTC) as an important indicator for predicting potential collisions. Their approach demonstrated the ability to adapt effectively to different vehicle speeds, offering more reliable performance than conventional fixed-distance methods.

The studies reviewed above demonstrate that MATLAB-based simulation is a practical, reliable, and cost-effective tool for developing and evaluating Advanced Driver Assistance Systems (ADAS). While many existing approaches primarily concentrate on either vehicle braking and speed regulation or depend heavily on complex sensor fusion techniques, fewer studies address both aspects together. To address this limitation, the present work combines longitudinal control, such as speed adjustment, with lateral control through lane-changing decisions in a single, lightweight simulation framework, providing a more comprehensive representation of autonomous driving assistance behavior.

III. PROBLEM STATEMENT AND OBJECTIVES

1. Problem Statement

In dynamic traffic scenarios, safe navigation for autonomous vehicles necessitates the capability of accurately detecting objects, tracking, and decision-making. When overtaking and changing lanes, such factors as blind spots, vehicles in the vicinity, different speeds and the possibility of a collision may compromise driving safety.

2. Objectives

- To design a MATLAB-based autonomous overtaking system for safe lane-changing operations on a two-lane road.
- To implement blind spot detection and collision avoidance logic during overtaking maneuvers.
- To simulate adaptive speed control and safe following distance maintenance under varying traffic conditions.
- To provide real-time graphical visualization and audio-visual warning alerts for system behavior monitoring.

IV. METHODOLOGY

The proposed system is designed as a MATLAB script with a continuous simulation loop. The method is predicated on the following sequence of steps:

1. Starting the vehicle

The three vehicles are initialized at the following predefined positions, velocities and lane assignments: the ego vehicle (blue, Lane 1, 0 m, 19.5 m/s), the front obstacle vehicle (red, Lane 1, 90 m, 8 m/s) and the rear overtaking vehicle (green, Lane 2, −35 m, 25 m/s). Simulation parameters are dt = 0.1 s, detection range = 45 m, safe gap = 20 m, blind zone front = 12 m and blind zone rear = 18 m.

2. Detection and distance monitoring of objects

At each time-step, the distance to the front is calculated as: front_dist=front_pos−vehicle_pos. This value is then compared to the threshold of the detection range to find the potential for collision. If the front distance is equal to or less than 45 m the system requests an overtaking maneuver.

3. Blind Spot Sensing

The system checks for the presence of the rear vehicle in the blind spot zone next to the ego vehicle. rear_in_zone = (

rear_lane== 2) & & (rear_pos > vehicle_pos - blind_zone_back) & & (rear_pos < vehicle_pos + blind_zone_front) ; If the speed of the following vehicle is higher than the current ego vehicle speed, a speed risk flag is raised. If rear_in_zone and speed_risk are both true, then the lane change is blocked and a WARNING alert is raised.

4. Passing Judgments

The decision logic works as follows: (1) If the front vehicle is detected and the ego is in Lane 1 and is not changing lanes, a lane change request is set. (2) If a lane change request is detected and the blind spot is empty (no speed risk), the ego vehicle smoothly transitions to Lane 2. (3) If the blind spot is unsafe, the system moves to the 'Blind Spot Waiting' state and produces audio and visual warnings through the play Alert () function.

5. Adaptive Speed Control

Speed is dynamically adjusted based on driving state: during lane change, vehicle_speed = max (base_speed − 3, front_speed + 2); if front_dist ≤ safe gap, the ego matches the front vehicle's speed (Maintaining 20 m Gap); if front_dist ≤ detection_range, speed is reduced by 3 m/s (Waiting to Overtake); otherwise, base speed is maintained (Cruising).

6. Lane Change Execution

This gives a smooth lateral movement. What we do is interpolate the y - position of the vehicle: current_y = current_y + 0.16 * (target_y - current_y) . This incremental update mimics the realistic side-to-side dynamics. Once the ego vehicle has fully entered Lane 2 and passed the front vehicle, it returns to Lane 1 with the same smooth interpolation.

7. Visualization and Warning System

The road, lane markings, vehicles, blind spot bounding box and status information are displayed through MATLAB graphical functions at each time step. The info panel displays status, speed, and front distance. If the blind spot or safe gap condition is violated, a red 'WARNING!' The label appears. The function plays Alert () generates and plays a beep of 1000 Hz using the MATLAB sound () function.

V. SYSTEM ARCHITECTURE AND BLOCK DIAGRAM

The overall system architecture is organized into six functional blocks that operate in a repeating loop at each simulation time step

1. Vehicles entered in

The input layer is the surrounding traffic environment including the ego vehicle, the front obstacle vehicle (red) and the rear overtaking vehicle (green). Each transmits position, speed and lane information necessary for sensor and decision modules.

2. Sensor and Detection Unit

Constantly tracks the locations and speeds of all vehicles. Calculates front vehicle distance, blind-spot occupancy detection, and detections of vehicles within detection or blind zone thresholds.

3. Decision-Making Block

It looks at sensor data to determine if an overtake is necessary and if the lane next door is safe. Generates warning flags if the rear vehicle in the blind spot is determined to have a faster speed than the ego vehicle.

4. Lane Change Controller

Executes smooth lateral movement from the current lane to the adjacent lane when safe conditions are confirmed and manages the return to the original lane after successful overtaking.

5. Speed Controller

Adjusts ego vehicle speed according to traffic conditions and lane-change state: maintains safe following distance, reduces speed during overtaking preparation, and restores base cruising speed when the road ahead is clear.

6. Vehicle Update and Visualization

The system continuously updates the position, speed, and lane information of all three vehicles based on motion equations over time. These updates are then visualized in a graphical interface that represents the complete road scenario. The visualization shows vehicle movement, lane placement, blind-spot areas, distance between vehicles, and warning messages, allowing real-time monitoring of system behavior.

VI. RESULTS AND DISCUSSION

1. Initial Conditions

The simulation begins by placing three vehicles in defined initial conditions. The ego vehicle (blue) starts at position 0 m, moving at 19.5 m/s in Lane 1. Ahead of it, the front vehicle (red) is positioned at 90 m with a slower speed of 8 m/s in the same lane. The rear vehicle (green) is placed at −35 m in Lane 2, traveling at a higher speed of 25 m/s. These initial settings establish the relative spacing and speed differences between vehicles, which are then used for analyzing interaction behavior during the simulation. Table I summarizes these initial conditions.

Table I: Initial Vehicle Parameters

Vehicle	Position	Speed	Lane
Ego (Blue)	0 m	19.5 m/s	Lane 1
Front Car (Red)	90 m	8 m/s	Lane 1
Rear Car (Green)	−35 m	25 m/s	Lane 2

2. Simulation Flow
The simulation progresses through seven distinct phases, with each stage verified using graphical output:

Cruising Mode:
The ego vehicle moves normally in Lane 1 at 19.5 m/s while maintaining a front distance of 73.9 m. The system remains in a "Cruising" state, indicating stable and safe motion

Obstacle Detection:
When the front distance reduces to 45 m or below, the system detects a potential overtaking requirement. The ego vehicle reduces its speed to 16.5 m/s and enters a "Waiting to Overtake" state.

Blind Spot Check:
A safety check is performed to evaluate the rear vehicle presence in the blind zone. Since the rear vehicle is within unsafe limits (front: 12 m, rear: 18 m), lane change is blocked. The system displays "Blind Spot Waiting" status along with a RED WARNING and beep alert.

Safe Gap Maintenance:
When the front distance reduces to 20 m or less, the ego vehicle adapts its speed to match the front vehicle (8 m/s), maintaining a constant safe gap. This behavior represents Adaptive Cruise Control (ACC) functionality.

Lane Change Execution:
Once the rear vehicle moves away and the blind spot becomes clear; the system initiates a smooth lane change. The status transitions from "Lane Clear – Changing" to "Changing Lane".

Lane Change Speed Control:
During overtaking, the system regulates speed to 16.5 m/s to ensure controlled and safe acceleration, avoiding aggressive maneuvering.

Overtaking Complete:
After successfully passing the slower vehicle, the ego vehicle fully transitions into Lane 2 and resumes its cruising speed of 19.5 m/s. The system returns to "Cruising" state, indicating normal operation (Fig. 6).

3. Performance Summary
The system demonstrated effective performance across all simulated scenarios, successfully avoiding collisions throughout the tests. It achieved smooth lane transitions without introducing abrupt lateral motion, ensuring stable vehicle behavior during overtaking. The controller also maintained a consistent safe distance of approximately 20 m through adaptive speed adjustments. In addition, the blind spot detection module reliably prevented unsafe lane changes by identifying vehicles in critical zones before initiating any maneuver.

The rule-based control strategy executed in real time within the MATLAB environment, with a graphical refresh rate of 0.05 s, enabling responsive visualization of system dynamics. These results indicate that the proposed model is suitable for rapid prototyping and preliminary validation of V2V-based driving scenarios.

VII. CONCLUSION

This paper presented a MATLAB-based simulation of an intelligent obstacle detection and autonomous lane-changing system designed for collision prevention. The proposed system integrates key functionalities such as front obstacle detection, blind spot monitoring, adaptive speed control, and smooth lane-change execution within a unified rule-based framework. Simulation results confirmed the ability of the system to perform collision-free overtaking, maintain safe inter-vehicle gaps, and generate real-time warnings across all tested scenarios.

Overall, the developed model effectively demonstrates the core principles of Advanced Driver Assistance Systems (ADAS) and V2V-inspired driving assistance using rule-based decision logic. It provides a simplified yet effective framework for understanding vehicle interaction and safety-aware maneuvering in dynamic traffic conditions

.

Future work can extend this model to more complex traffic environments involving multiple lanes and higher vehicle density. Additional improvements may include sensor fusion using radar and camera data, as well as the integration of deep learning-based object detection techniques. Hardware implementation using a Hardware-in-the-Loop (HIL) setup with embedded microcontrollers is also a potential direction. Furthermore, incorporating real-world V2V communication standards such as DSRC and C-V2X would enable evaluation of the system under realistic network conditions.

REFERENCES

1. W. Guo, Y. He and X. Ling, "Modeling and Simulation of Intelligent Vehicle Systems Using Simulink and MATLAB Programming for Advanced Automotive Application," 2025 11th International Conference on Computing and Artificial Intelligence (ICCAI), Kyoto, Japan, 2025, pp. 627-631, doi: 10.1109/ICCAI66501.2025.00100.
2. M. Thompson, J. Dallas, J. Y. M. Goh and A. Balachandran, "Adaptive Nonlinear Model Predictive Control: Maximizing Tire Force and Obstacle Avoidance in Autonomous Vehicles," in IEEE Transactions on Field Robotics, vol. 1, pp. 318-331, 2024, doi: 10.1109/TFR.2024.3497925.
3. P. Oke, J. Tawaketini and N. Singh, "Robust Control of Active Suspension System Using MATLAB/Simulink," 2023 International Conference on Sustainable Technology and Engineering (i-COSTE), Nadi, Fiji, 2023, pp. 1-7, doi: 10.1109/i-COSTE60462.2023.10500785.
4. Lan, J., Jiang, Y., Fan, G., Yu, D., & Zhang, Q. (2016). Real-Time Automatic Obstacle Detection Method for Traffic Surveillance in Urban Traffic. 82(3), 357–371. https://doi.org/10.1007/S11265-015-1006-4
5. L. Claussmann, M. Revilloud, D. Gruyer and S. Glaser, "A Review of Motion Planning for Highway Autonomous Driving," in IEEE Transactions on Intelligent Transportation Systems, vol. 21, no. 5, pp. 1826-1848, May 2020, doi:10.1109/TITS.2019.2913998.
6. C. Wang, Y. Cao, T. Sun and X. Guo, "Design and Evaluation of a Forward Collision Warning System Based on Time-to-Collision Estimation," IEEE Transactions on Intelligent Transportation Systems, vol. 20, no. 9, pp. 3465-3476, Sep. 2