Integrated Lane Departure Warning and Adaptive Cruise Control for Enhanced Driving Safety in Advanced Driver Assistance Systems: A Simulation Study

Tanakorn Mongkondatsawat[1*], Panumat Rergngam[2], Wutthichai Imwong[3], Namchai Sokai[4], and Nakorn Indra-Payoong[5]

[1,2,3,4,5]Center for Future Mobility (NDrive), Faculty of Logistics, Burapha University, Chonburi

E-mail: tanakorn.mongkondatsawat@gmail.com*, phanumatrakngam@gmail.com, wutthichaiimwong@gmail.com, namchai48@gmail.com, nakorn@go.buu.ac.th

Received: June 13, 2025 / Revised: January 1, 2026 / Accepted: February 2, 2026

***Abstract*—This research aims to develop and test Advanced Driver Assistance Systems (ADAS), specifically Adaptive Cruise Control (ACC) and Lane Departure Warning (LDW), utilizing the "CARLA Simulator," an open-source driving environment simulation software for autonomous driving technology research and development. Testing results showed that the ACC system achieved 97.916% accuracy in following the speed of the vehicle ahead, maintained an average distance of 10.651 meters from the leading vehicle (exceeding the specified safety distance), and exhibited an automatic braking response time of 0.3 seconds. The LDW system correctly detected vehicle lane departures with over 75% accuracy when lane markings were complete or straight. However, in heavy rain or on roads with incomplete or dashed lane markings, the system produced false positives—incorrectly warning of lane departures departures when the vehicle remain in lane—a rate exceeding 30%.**

***Index Terms*—Advanced Driver Assistance Systems, Adaptive Cruise Control, Lane Departure Warning, CARLA Simulator**

I. Introduction

Advanced Driver Assistance Systems (ADAS) have assumed a pivotal role in the evolution of the automotive industry, with the primary objectives of enhancing safety and optimizing driving convenience. As ADAS technologies such as Adaptive Cruise Control (ACC) and Lane Departure Warning (LDW) continue to develop, they require rigorous testing and performance evaluation before real-world implementation to ensure system safety and stability.

Assessing ADAS in real environments presents numerous constraints, including substantial costs, potential accident risks during testing, and environmental limitations. Consequently, vehicle simulators are an optimal alternative for testing, as they can simulate diverse driving scenarios and allow control over various parameters. This includes the capability to evaluate ADAS performance in potentially hazardous situations or circumstances that cannot be reliably replicated in real-world environments, such as navigating dense traffic conditions, executing lane changes, or encountering obstacles.

This research employs the CARLA simulator, an open-source vehicle simulation platform with robust capabilities for realistic environmental simulation [1], [2]. Using the CARLA simulator enables the development and assessment of ADAS technologies across diverse, controlled environmental conditions. In developing countries such as Thailand, implementing and evaluating Advanced Driver Assistance Systems (ADAS) faces several challenges, including limited access to real-world testing environments, high implementation costs, diverse traffic conditions, and adverse weather scenarios. While previous studies have demonstrated the effectiveness of individual ADAS functions, such as Adaptive Cruise Control (ACC) or Lane Departure Warning (LDW), most evaluations have been conducted under controlled conditions or in developed-country contexts. Thus, a research gap remains in evaluating integrated ADAS functions under complex driving scenarios using simulation-based platforms.

This research's main contribution is developing and evaluating an integrated ADAS framework that combines Adaptive Cruise Control (ACC) and Lane Departure Warning (LDW) within the CARLA simulation environment. Unlike previous studies that primarily focus on single-function ADAS systems or theoretical modeling, this work emphasizes system-level integration, scenario-based testing, and performance evaluation under diverse conditions, including varying traffic behaviors. This study also provides a detailed analysis of velocity-tracking

accuracy, following-distance behavior, and LDW false-positive rates, highlighting practical limitations and safety implications. The findings improve understanding of ADAS performance in simulation environments and inform future improvements for real-world testing and deployment in developing countries.

II. Literature Review

A. ADAS Technology

ADAS has undergone continuous development within the automotive industry with the primary objective of enhancing driving safety [3]. These systems comprise multiple subsystems, including Adaptive Cruise Control (ACC), Lane Departure Warning (LDW), Autonomous Emergency Braking (AEB), and Lane Keeping Assist (LKA). Dosovitskiy et al. [6] showed a significant reduction in accident rates, particularly when accurate object detection and distance estimation from preceding vehicles or obstacles are achieved using diverse sensor technologies such as cameras, radar systems, and LiDAR. Furthermore, ADAS technologies have been shown to mitigate driver fatigue in high-traffic-density conditions and long-distance driving by helping drivers maintain safe vehicle operation.

B. Adaptive Cruise Control (ACC)

Adaptive Cruise Control (ACC) is an advanced driving assistance function that automatically regulates vehicle speed while maintaining a safe distance from a preceding vehicle. Anand and Ohol [1] proposed a methodology for designing and simulating ACC systems by integrating Model Predictive Control (MPC) with Decision Tree algorithms to analyze acceleration–deceleration behavior and overtaking assistance in realistic traffic conditions. Their experimental results demonstrated the effectiveness of combining radar sensor technology with intelligent control algorithms in maintaining appropriate safety distances. However, the study primarily focused on ACC as a standalone system and did not consider its interaction with other ADAS functions in an integrated framework.

C. Lane Departure Warning (LDW)

Lane Departure Warning (LDW) systems aim to detect unintended lane departures and alert drivers to potential safety risks. Ping et al. [13] developed a vision-based LDW framework using Hough transform techniques and lateral deviation ratio calculations. The system was evaluated using real-world datasets collected under both daytime and nighttime conditions. The reported results showed 94.71% lane-detection accuracy and 81.18% lane-departure detection precision. Nevertheless, the study identified performance limitations in scenarios with worn lane markings and low illumination, which increased false positive rates. These findings highlight the sensitivity of vision-based LDW systems to environmental and road conditions.

D. CARLA Simulator for ADAS Development and CARLA Selection

Driving simulators have been widely adopted for ADAS and autonomous vehicle research because they provide safe, repeatable, and cost-effective testing environments. Previous studies have used several simulation platforms, including PreScan, SUMO, LGSVL Simulator, and AirSim. PreScan offers high-fidelity vehicle dynamics and sensor modeling; however, it requires commercial licensing, which limits its accessibility for academic and research applications. SUMO is designed for traffic flow simulation and lacks the realistic sensor modeling capabilities required for perception-based ADAS evaluation. LGSVL Simulator and AirSim support sensor simulation and autonomous driving research but offer less flexibility for large-scale urban scenario customization and adverse-weather modeling than CARLA.

Dosovitskiy et al. [6] introduced the CARLA simulator as an open-source urban driving simulation platform that emulates various sensor modalities, including RGB cameras, radar, and LiDAR, within highly realistic virtual environments. In addition. Markofsky et al. [12] proposed a framework for virtual calibration of ADAS and Automated Driving Systems (ADS) using multi-scenario simulations in CARLA, demonstrating that simulation-based testing can significantly reduce real-world testing costs while enabling iterative performance optimization without safety risks.

Table I presents a comparative analysis of driving simulators used in ADAS development and identifies CARLA as the most suitable platform for developing and evaluating Adaptive Cruise Control (ACC) and Lane Departure Warning (LDW) systems. CARLA's suitability stems from its open-source architecture, realistic urban road networks, comprehensive sensor emulation, dynamic traffic generation, and advanced weather-condition modeling, including heavy rain. These features enable scenario-based testing and systematic performance evaluation of integrated ADAS functions under complex driving environments.

Unlike previous studies that primarily focus on individual ADAS functions, such as ACC and LDW, this research emphasizes the integrated evaluation of ACC and LDW within a unified ADAS framework using a simulation-based approach. The proposed study focuses on system-level performance analysis under diverse traffic and environmental conditions, which is particularly relevant for developing countries

such as Thailand, where cost, infrastructure, and safety concerns constrain real-world ADAS testing. This simulation-based evaluation provides practical insights into the applicability and limitations of integrated ADAS technologies prior to real-world deployment.

TABLE I
COMPARES CAR SIMULATOR FOR ADAS (ACC/LDW) DEVELOPMENT

Simulator	Developer	ADAS (ACC / LDW)	Reality	Sensors Support	Compatibility	Cost	Limitation
CARLA	Open-sourced by the research community	ACC and LDW in a dynamic 3D environment	High	LiDAR, Radar, Camera	Python and ROS	Free	Requires high computing resources and is difficult to set up
LGSVL Simulator	LG Electronics	ACC and LDW in urban area using ROS and Autoware	Moderate	LiDAR, Camera	ROS and Autoware	Free	Restricted vehicle models and environment
Gazebo	Open-source (OSRF)	ACC and LDW in a simulated environment using ROS	Low	Camera, LiDAR	ROS	Free	Poor graphics and user interface
PreScan	Siemens	ACC and LDW using MATLAB/Simulink with accurate sensor simulation	High	LiDAR, Radar, Camera	MATLAB and Simulink	High	Proprietary software
Ansys VRXPERIENCE	Ansys	ACC and LDW using advanced sensors and vehicle dynamics	High	LiDAR, Radar, Camera	Development across multiple platforms	High	Difficult to set up
IPG CarMaker	IPG Automotive	ACC and LDW using Hardware-in-the-loop and MATLAB	High	LiDAR, Radar, Camera	MATLAB, Hardware-in-the-loop	High	Focus on automotive engineering rather than AI development
AirSim	Microsoft Research	Unreal Engine for LDW ACC in vitural environment	High	RGB, Depth, LiDAR	Python and C++	Free	Fewer users than CARLA, and the simulation has a limit
Nvidia Drive	Nvidia	ACC and LDW using Ray-tracing technology and AI simulation	Very high	LiDAR, Radar, Camera	Nvidia Drive	High	Only simulated with Nvidia hardware

III. METHODOLOGY

As automotive technology transitions toward autonomous driving, ADAS development is expected to be a crucial component in reducing accidents and improving road safety, especially in developing countries with high accident rates. These systems must be calibrated to reflect local driver behavior across different regional contexts. However, testing ADAS on real-world roads faces numerous limitations regarding cost, time, and passenger safety risks. Therefore, this research uses the CARLA simulation program to design, test, and refine ADAS systems to ensure both safety and effectiveness.

This section examines the CARLA framework, sensor and vehicle simulation methodologies, and the ADAS development process using a simulation-driven design approach that emphasizes development based on simulated data and logic generated within the simulator environment.

A. CARLA Architecture

CARLA (Car Learning to Act) is an open-source platform for autonomous driving research and development. The platform uses Unreal Engine 4 to create highly realistic virtual environments, including sophisticated visual elements, sound reproduction, weather-condition simulation, and accurate vehicle physics. The architectural framework of CARLA is segmented into three main components [8]:

1) Server: functions as the virtual world simulator, encompassing scene rendering, map generation, actor instantiation, and diverse meteorological conditions, all implemented through Unreal Engine 4.

2) Middleware (RPC Layer): Serves as the communication intermediary between the Client and Server through a Remote Procedure Call system specifically engineered to maintain low latency values.

3) Client (Python API): The component that developers utilize to implement vehicle control algorithms, receive sensor data input, and process various behavioral parameters.

B. CARLA components

1) Actors: Vehicles, pedestrians, motorcycles, or obstacles. (Used: Any CARLA vehicle catalogue)

2) Maps: Standard simulated urban maps, such as Town01–Town10 within CARLA, or imported maps from OpenDRIVE. (Used: Town04 map)

3) Sensors: RGB camera, LiDAR, Radar, GNSS, IMU, etc. (Used: RGB Camera and Radar)

4) Traffic Manager: Traffic behavior management systems, such as braking responses to vehicle incursions or lane-changing maneuvers. (Used: Spawn-point to spawn all vehicles, Lead Vehicle: Autopilot)

IV. Experimental Results

A. Adaptive Cruise Control (ACC)

Adaptive Cruise Control (ACC) represents a critical function within Advanced Driver Assistance Systems (ADAS), automatically regulating vehicle velocity according to traffic conditions while maintaining safe distances from preceding vehicles across all speed ranges [9]-[11]. This research focuses on developing and evaluating ACC functionality within the CARLA Simulator environment, using virtual radar sensors to detect the velocity of leading vehicles and implementing PID control algorithms with safety-oriented decision logic for acceleration and braking responses. The system responds precisely to simulated scenarios that accurately reflect real-world vehicle behavior. The ACC system relies primarily on the virtual Radar Sensor (Sensor.other.radar), which detects forward objects at extended ranges and measures critical parameters including Distance, Relative Velocity, and Azimuth. The research team positioned the radar sensor at the front section of the subject vehicle (Ego Vehicle) to simulate ACC functionality in automotive applications. The system processes data frame by frame and uses median filtering to eliminate anomalous velocity readings, maintaining system stability.

The developed ACC system uses Proportional-Integral-Derivative Control (PID) algorithms to keep the Ego Vehicle's velocity within safety parameters, primarily based on the distance to the Lead Vehicle, supplemented by additional conditions that simulate human decision-making processes, as illustrated in Fig. 1.

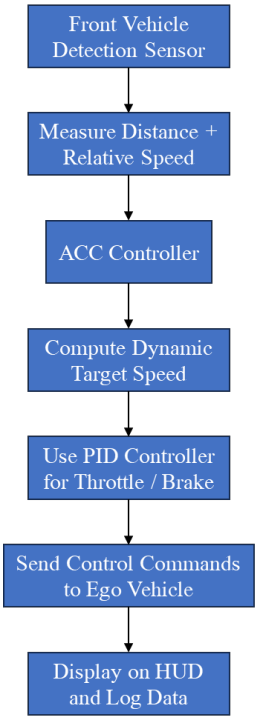

Fig. 1. Flowchart of ACC

1) Decision Criteria

• If the distance from the Lead vehicle is less than 5.5 meters, the system commands the Ego vehicle to initiate emergency braking procedures immediately.

• In conditions of extremely low velocity with proximate Lead vehicle positioning, the system instructs the Ego vehicle to execute a complete stop-and-hold maneuver.

• When the intervehicular distance falls below the calculated safety threshold (derived from velocity parameters), the system activates braking protocols. Safe_distance = max (10.0, ego_vehicle_speed x 2.0)

• In scenarios where substantial intervehicular distance exists, the system implements acceleration commands governed by PID control mechanisms.

• Under standard operating conditions, the system maintains appropriate velocity parameters.

2) PID Controller

The PID control component has been calibrated to enable rapid ego-vehicle response while maintaining smooth driving characteristics. Through parameter tuning, we determined optimal values of Kp = 1.2 (primary control), K_i = 0.08 (steady-state error reduction), and K_d = 0.15 (overshoot mitigation). The PID controller generates commands that regulate the Ego Vehicle's throttle and brake functions in real time.

The developed ADAS was tested in the CARLA Simulator, simulating driving conditions with ACC mode activated and using radar and PID control components. The acc_metrics_logger.py utility recorded all movement frames and automatically generated logs and graphical representations. Fig. 2 illustrates ACC system operations



Fig. 2. Testing ACC in CARLA simulator

Fig. 3 presents the performance assessment results for the developed ACC, and Table II summarizes the Performance Matrices for the ACC system.

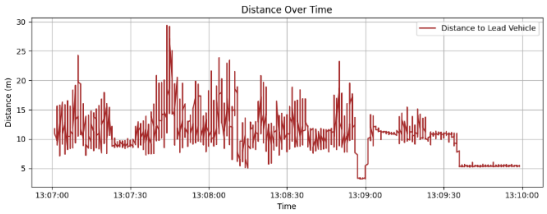

Fig. 3 (a). Distance to lead vehicle

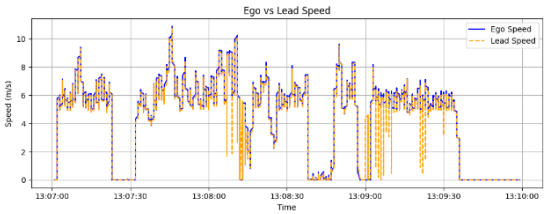

Fig. 3 (b). Ego vs. lead speed

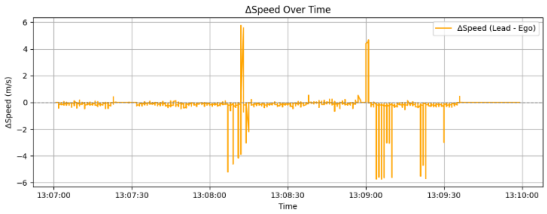

Fig. 3 (c). Delta speed

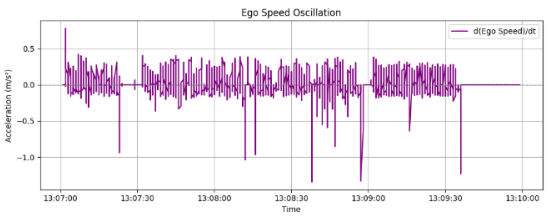

Fig. 3 (d). Eco speed oscillation

TABLE II
SUMMARY OF ACC PERFORMANCE MATRICES

Metric	Value
mean_delta_speed	-0.095
std_delta_speed	0.558
mean_ego_acceleration	0.000
std_ego_acceleration	0.131
max_ego_speed	10.877
max_lead_speed	10.793
speed_following_accuradcy (%)	97.916
afjust_reaction_match_rae (%)	58.306
emergency_brake_count	740
mean_distance	10.651
min_distance	3.191
max_distance	29.376

Testing results from the developed ACC system in the CARLA Simulator demonstrate its ability to maintain safe distances in dynamic situations. The system achieved 97.916% velocity accuracy in tracking the Lead Vehicle and maintained an average distance of 10.651 meters from the Lead Vehicle, exceeding the specified safety distance threshold. Additionally, the system exhibited an automatic braking Reaction Time of 0.3 seconds, indicating rapid response when detecting braking actions from the Lead vehicle.

This PID with Logic-Based Control approach is appropriate for implementation as an automotive engineering prototype and presents opportunities for future expansion. Potential development pathways include integration with Autonomous Emergency Braking (AEB) systems and incorporation into Full Autopilot systems or Lane Keeping Assist (LKA) technologies.

B. Lane Departure Warning (LDW)

LDW is another essential ADAS system that aims to reduce accidents caused by drowsiness, unintentional lane deviation, or driver inattention. An effective LDW system must accurately detect the vehicle's position on the road, respond in real-time, and provide precise warnings [2], [9]. This research focuses on developing and evaluating an LDW system using the CARLA Simulator, with a simulated front-facing RGB camera as the primary sensor. The system incorporates Computer Vision techniques and geometric logic to assess lane deviation and transmits warning signals through a Heads-Up Display (HUD), color indicators, and auditory alerts in situations where the vehicle is at risk of departing from its lane Operational Principles of LDW.

1) The system utilizes images from the sensor. camera.rgb mounted at the front of the vehicle (with a Field of View of 110° at a resolution of 1280 x 720 pixels) to capture real-time road and lane marking data [5].

2) The LDW employs the following image processing procedures:

• Grayscale → Gaussian Blur → Canny Edge Detection to decrease noise and detect lane markings efficiently

• Perspective Transform (Bird's Eye View) to adjust the image viewpoint to render lane markings as straight lines.

• Sliding Window Search + Polynomial Fitting to utilizes image window segmentation to detect left and right lane markings, followed by curve fitting to determine lane curvature [10].

• To compute the lane center point in relation to the vehicle position to measure lane deviation.

Fig. 4 illustrates the flowchart of the developed LDW.

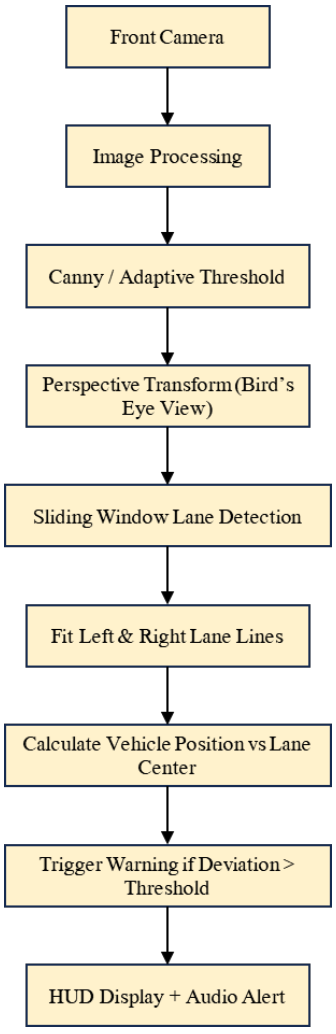

Fig. 4. Flowchart of LDW

Notification and Heads-up Display (HUD): When the system detects lane deviation exceeding the predetermined threshold values, it initiates driver notifications through multiple channels:

- Notification message "LDW: Lane Departure!" on HUD (red text).
- A flashing red bar displayed at the side of the screen to indicate danger.
- Beep.wav sound to notify the driver.
- Lane Overlay in green to follow the correct lane (bird's-eye overlay).

We tested the LDW system on the Town04 map in CARLA, simulating a route with extended continuous curves and implementing periodic transitions between driving modes, alternating between Manual mode (user-controlled) and Autopilot (automated driving system). We incorporated both intentional and unintentional lane deviations to observe LDW functionality across various scenarios. Figs. 5-7 illustrate examples of LDW system alerts during vehicle lane deviation. Fig. 8 shows the LDW simulation over time.

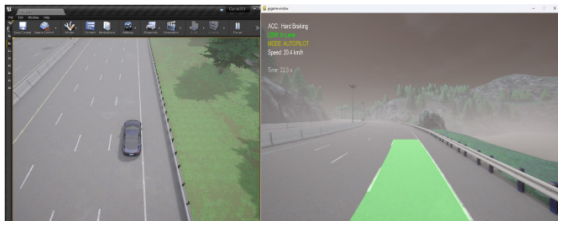

Fig.5. LDW test: Vehicle in lane (with complete or straight lane markings: Show Display = LDW: In Lane)



Fig. 6. LDW test: Lane departure (with complete or straight lane markings: Show Display = LDW: Lane Departure! and display a red frame to the right and left of the display frame.)

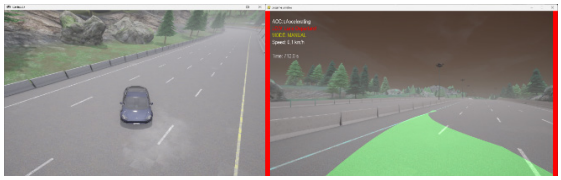

Fig. 7. LDW test: Lane departure (with incomplete or dashed lane markings: Show Display = LDW: Lane Departure! and display a red frame to the right and left of the display frame.)

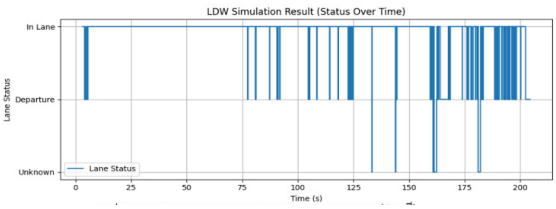

Fig. 8. LDW simulation over the time period

In Lane = vehicle in lane, Lane Departure = vehicle deviates from lanes, Unknown = LDW cannot detect, caused by unclear images or poor lighting, etc.

The developed LDW system demonstrates effective lane-deviation detection using data from a single RGB camera, employing geometric principles alongside fundamental image-processing techniques. The system provides real-time feedback through a Heads-Up Display (HUD) and auditory alerts, and is compatible with all driving modes. However, testing revealed that under certain conditions, such as heavy precipitation or roads with indistinct lane markings, the LDW system generates frequent false positives—error notifications exceeding 30% of instances. This error rate results from the system's inability to differentiate between clear and indistinct lane markings, particularly during adverse weather conditions like heavy rainfall. Future development of the LDW system could incorporate Deep learning methodologies within constrained computational resources.

V. Conclusions

This research focuses on developing Advanced Driver Assistance Systems (ADAS), specifically two primary systems: Adaptive Cruise Control (ACC) and Lane Departure Warning (LDW). Both ACC and LDW systems were developed and evaluated using the CARLA Simulator, a platform that provides a highly realistic automotive driving environment simulation and facilitates efficient algorithm development for control and sensor data processing.

The ACC system incorporates acceleration and braking force calculations utilizing data from a Radar sensor capable of detecting the distance and velocity of the Lead vehicle. A PID controller manages the driving system in real time to maintain safe distances from the Lead vehicle, enhancing driving convenience and safety. Testing results show that the system effectively adjusts the ego vehicle's behavior in response to the lead vehicle's movement, achieving a velocity tracking accuracy of 97.916%. The system maintains an average distance from the Lead vehicle of 10.651 meters, exceeding the specified safety threshold of 10.0 meters, with an automatic braking response time of 0.3 seconds. Additionally, the ACC system can initiate automatic braking when necessary, resulting in safe driving performance and appropriate responses to traffic conditions across all testing scenarios.

The LDW system uses forward-facing camera imagery and image-processing techniques, including Canny edge detection and Hough transform, to identify lane markings. The system incorporates a Heads-Up Display (HUD) and auditory alerts when it detects vehicle lane departure. Testing results indicate that the LDW system can continuously monitor lane positioning and departures throughout the testing period. When lane markings are intact or straight, the system demonstrates correct lane departure warning accuracy exceeding 75%. However, in heavy rainfall or on roads with incomplete or dashed lane markings, the LDW system produces false positives—incorrect warnings when the vehicle has not left its lane—exceeding 30%. Future LDW development could incorporate Deep learning in resource-constrained processing environments to improve detection accuracy by implementing confidence-level assessment for each frame. Because of the limitations of RGB camera technology, rainy weather degrades performance, resulting in over 30% false lane departure warnings. This could potentially be addressed by using deep learning methods instead of the Canny edge detection and Hough transform used in this experiment.

References

[1] S. Anand and S. S. Ohol, "Modelling and simulation of adaptive cruise control and overtake assist system," *Mater. Today Proc.*, vol. 72, pp. 1353-1360, 2023, https://doi.org/10.1016/j.matpr.2022.09.330

[2] Angelkim88, "CARLA-Lane_Detection," *GitHub repository*, 2020. [Online]. Available: https://github.com/angelkim88/CARLA-Lane_Detection [Accessed: Dec. 10, 2024].

[3] CARLA, "Quickstart start package instalbtion," *Carla Documentation*. [Online]. Available: https://carla.readthedocs.io/en/latest/start_quickstart [Accessed: Dec. 10, 2024].

[4] Carla Simulator, "scenario_runner", *GitHub repository*, 2018. [Online]. Available: https://github.com/carla-simulator/scenario_runner [Accessed: Dec. 10, 2024].

[5] Da-frank2, "Carla-object-detection-using-yolo," *GitHub repository*, 2022. [Online]. Available: https://github.com/Da-frank2/Carla-Object-Detection-Using-YOLO [Accessed: Dec. 10, 2024].

[6] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An open urban driving simulator," in *Proc. 1st Annu. Conf. Robot Learn. (CoRL)*, 2017, pp. 1-16.

[7] *Intelligent transport systems—Adaptive cruise control systems—Performance requirements and test procedures*, ISO 15622:2018, Sep. 2018. [Online]. Available: https://www.iso.org/standard/71515.html

[8] *Intelligent transport systems—Forward vehicle collision warning systems—Performance requirements and test procedures,* ISO 15623:2013, Jul. 2013. [Online]. Available: https://www.iso.org/standard/56655.html

[9] *Intelligent transport systems—Lane Departure Warning systems—Performance requirements and test procedures*, ISO 17361:2017, Jun. 2017. [Online]. Available: https://www.iso.org/standard/72349.html

[10] C. Kedmanee, "Kalman filter modeling for trajectory prediction of cartraveling through Straight and curved lane boundaries," *Sripatum Rev. Sci. Technol*., vol. 5, no.1, pp. 57-66, Dec. 2013.
[11] KochlisGit, "Autonomous-Vehicles-Adaptive-Cruise-Control," *GitHub repository*, 2021. [Online]. Available: https://github.com/kochlisGit/Autonomous-Vehicles-Adaptive-Cruise-Control [Accessed: Dec. 10, 2024].
[12] M. Markofsky, M. Schäfer, and D. Schramm, "Use cases and methods of virtual ADAS/ADS calibration in simulation," *Vehicles*, vol. 5, no. 3, pp. 802-829, Jul. 2023, https://doi.org/10.3390/vehicles5030044
[13] E. P. Ping, J. Hossen, F. Imadudin, and W. E. Kiong, "Vision-based lane departure warning framework," *Heliyon*, vol. 5, no. 8, Art. no. e02169, Aug. 2019, https://doi.org/10.1016/j.heliyon.2019.e02169
[14] Siraphobk, "xd_senior2019_adas_selfdrive", *GitHub repository*, 2020. [Online]. Available: https://github.com/siraphobk/xd_senior2019_adas_selfdrive/tree/master [Accessed: Feb. 5, 2025].



Tanakorn Mongkondatsawat received a Bachelor of Science (B.Sc.) in Logistics and Supply Chain Management from Burapha University, Thailand, in 2024. He is currently pursuing a Master of Science program in Logistics and Supply Chain Management. His research interests include artificial intelligence development, logistics, and supply chain management.



Phanumat Rakngam received her B.Sc. degree in Logistics and Supply Chain Management from Burapha University, Thailand, in 2024. She currently works in the private sector in logistics and supply chain management. Her interests span intelligent transportation and supply chain process design.



Wutthichai Imwong earned a B.Sc. degree in Logistics and Supply Chain Management from Burapha University, Thailand, in 2024. As an undergraduate, he gravitated toward data-center projects—building proof-of-concept ADAS functions in the CARLA simulator, writing Python scripts for data cleaning and visualisation, and experimenting with machine-learning models for small research assignments. He is now looking at graduate study and early-career roles that combine data analytics, AI modelling, and software development. His current interests include applied machine learning, open-source software engineering, digital-twin simulation, and data engineering.



Namchai Sokai received a Master of Science in Logistics and Supply Chain Management from Burapha University, Thailand. He was a lecturer at the Faculty of Logistics at Burapha University, Thailand. His research interests are in future mobility, artificial intelligence, and optimization.



Nakorn Indra-Payoong received a Ph.D. in Computing from the University of Leeds. He is currently with the Faculty of Logistics, Burapha University, Thailand, where he serves as an associate professor in logistics and supply chain management. His research interests include large-scale network optimization and future mobility.