Software Requirements Specification (SRS)

Project LMS 1

Authors: Brant McCormick, Andre Sasser, Kerry Dai, Lance Motowski, Augustine Adomako

Customer: Ayush Agrawal

Instructor: Betty Cheng

1 Introduction

In 2015, nearly 13,000 people died in single-vehicle run-off-road, head-on, and sideswipe crashes where a passenger vehicle left the lane unintentionally [1]. The prevalence of fatal accidents caused by lane departure highlights the utility of automated lane management systems which help mitigate unintentional lane departure.

This document details the requirements, functionality, elements, and uses of the Lane Management System (LMS) that helps prevent automobile drivers from making unintentional driving maneuvers that endanger themselves and their surroundings.

1.1 Purpose

The purpose of the software requirements specification is to communicate to the customer the team's interpretation of the product and design which attempts to satisfy the requirements. The document contains use case diagrams which demonstrate how a user might interact with the system. The document also contains a domain model which identifies the key elements of the system and their relationships with each other. The document goes over scenarios describing the use cases provided earlier in natural language. The last model used to describe the system is a state model which rigorously describes the state transitions the system undergoes under each scenario. Finally, the document describes the prototype developed to provide the customer with a tangible demonstration of the team's interpretation of the product.

1.2 Scope

LMS is an embedded system in a vehicle that provides electronic steering assistance and warnings which prevent the driver from executing unintentional driving maneuvers. To achieve this goal, the system utilizes a vehicle-mounted camera, sensor detection, electronic steering, and utilizes the sound system, infotainment screen, and steering wheel vibration to communicate with the driver. The system implements a lane departure warning subsystem which produces warnings when the driver is potentially making an unintentional maneuver and prompts them to correct their steering. The system implements a tiered warning system which produces different warnings based on their severity. The system tries to be as unobtrusive to the user as possible while also keeping them safe.

1.3 Definitions, acronyms, and abbreviations

- LMS: Lane Management System
- LDWS: Lane Departure Warning System
- LKAS: Lane Keeping Assisting System (general term for lane assistance software)
- LCS: Lane Centering System
- LDS: Lane Detection System
- LKS: Lane Keeping System
- ECU: Electronic Control Unit: an embedded system in a vehicle
- ECS: Electronic Control System
- CAN: Control Area Network
- AI: Artificial Intelligence
- ADAS: Advanced Driver Assistance System
- DFD: Data Flow Diagram
- LDAI: Lane Detection Artificial Intelligence
- CSS: Camera Sensor System
- OTA: Over-the-Air

1.4 Organization

Section 1 introduces the customer to the scope and purpose of the product as well as the purpose and organization of the software requirements specification document. Section 2 gives a high-level description of how the LMS works. Section 3 contains an enumerated list of requirements divided into the categories hardware, software, and security. Section 4 contains sequence, state, use-case, and domain models which give an abstract view of how the system executes the requirements. Section 5 describes the prototype which gives a tangible insight into the system's behavior. Section 6 contains a list of references cited in the IEEE format. Section 7 contains contact information for the team's instructor, Betty Cheng.

2 Overall Description

This section provides an overview of the Lane Management System, its role within a vehicle, and the general characteristics of its users and constraints. It also summarizes the main software functions, assumptions, and any requirements that may be deferred for future versions.

2.1 Product Perspective

The LMS system can talk to the car through a number of built-in interfaces. The camera interface sends a constant video feed that is used to find lanes, and the steering interface gets control signals from the LMS to make small corrections. The torque sensor interface records what the driver does, and the alert system interface sends dashboard lights, haptic feedback, or audio warnings when they are needed. The CAN bus interface connects the LMS to other vehicle modules, which allows for communication between all parts of the system.

The user interacts with the system mostly through the dashboard lights and icons, audible alerts, and the physical on/off switch. There is no need for a separate software interface. The system needs both hardware and software to work. The hardware includes the camera, torque sensor, steering actuators, and processing units. The software includes the lane-detection AI and the LMS controller's built-in control logic. All communication with outside electronic control systems uses standard CAN messaging and follows safety rules for cars. Finally, the system needs to work within the limits of memory and processing power that allow for real-time image processing and sensor analysis.

2.2 Product Functions

At a high level, LMS performs three main functions: The LMS system uses its camera and AI subsystem to continuously monitor the road. It finds lane markings and gives each reading a confidence score. The system keeps the car in the lane by using this information and applies gentle corrective steering torque whenever it sees the car drifting without a turn signal. The alert system gives the driver different types of feedback based on how bad the situation gets. First, there are visual cues, then haptic feedback, and finally audio alerts for the worst cases.

When a driver activates a turn signal, the LMS temporarily pauses its corrections to allow intentional maneuvers. If sensors fail or lane confidence drops below 95%, the system disables itself and alerts the driver until conditions improve.

The following diagrams use a simpler version of the DFD notation. In this notation, rectangles stand for things outside the system, like the driver or sensors, and rounded rectangles stand for processes or subsystems inside the system, like the LMS, LDAI, and Alert System. Arrows show which way data or control flows between parts. This notation shows how information goes into, through, and out of the LMS system.

Flow:
Driver → On/Off Switch → Camera → Lane Detection AI → LMS → Steering + Alert System → Driver

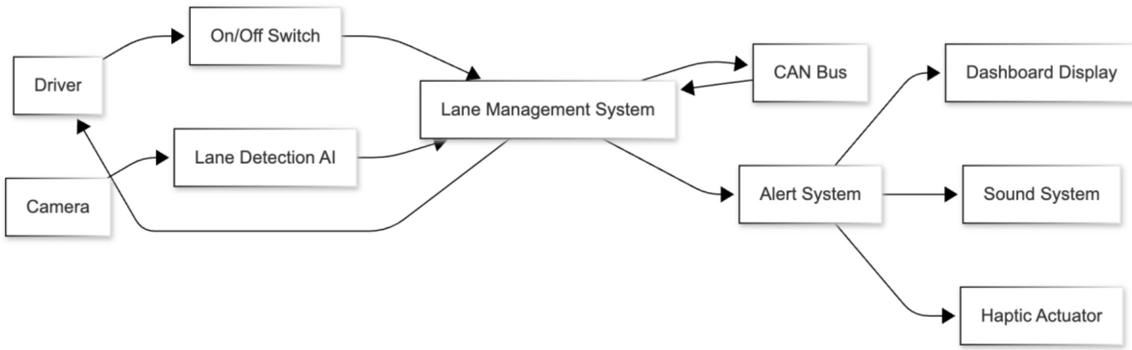

Fig1: flow chart of product functions

Figure 1 presents the high-level flow of data between the LMS-1 controller, its sensors, and its output systems. Figure 2 provides a refined view that explicitly includes the CAN bus and separates the alert outputs (dashboard, sound, and haptic feedback). These two versions

2.3 User Characteristics

The expected user is a licensed driver operating a vehicle equipped with LMS. No special training is required beyond standard driving experience. Drivers should be able to interpret dashboard symbols and auditory cues. The system assumes that users will maintain control of the vehicle at all times and may override the system by steering or braking as needed.

2.4 Constraints

The LMS only works when the car is going faster than 35 mph and the lane markings are easy to see. The system must turn off if lane confidence drops below 95%, the driver applies too much torque, or any subsystem fault is found. All parts must work reliably in normal car environments, which can include changes in temperature, vibration, and light. The driver must always be in charge of the car, and the LMS cannot change what the driver says. Lane detection and steering corrections must also happen in real time, and the system must respond in less than 100 milliseconds.

2.5 Assumptions and Dependencies

The proper operation of the LMS-1 system relies on several environmental, hardware, software, and user-related assumptions. These include the following:

- Lane markings are present and visible under normal daylight or illuminated conditions.
- The camera, torque sensor, and steering systems are fully functional.
- The AI model has been trained and calibrated for standard US road conditions.
- The CAN network is reliable and free of major communication latency.
- The driver will not intentionally disable safety features while the system is active.

2.6 Approportioning of Requirements

Some advanced capabilities are beyond the current project scope but may be considered for future releases:

- Automatic lane-change assist using predictive trajectory modeling.
- Integration with adaptive cruise control for semi-autonomous highway driving.
- Enhanced performance in poor weather or unmarked roads using map data fusion.
- Over-the-air software updates for AI model retraining and calibration.

3 Specific Requirements

1. Global Invariant Requirements

1.1. The LMS turns on when the user presses the LMS button located on the steering wheel.
1.2. The LMS turns off when the user presses the LMS button located on the steering wheel.
1.3. The user can override the LMS by manual steering, activating turn signal, or braking.
1.4. The LMS disables itself if lane boundaries are ambiguous or undetected.
1.5. The LDWS must always work when the LMS is active
1.6. The LMS disables when the LDWS fails.
1.7. The LMS must alert the user on activation, interruption, or failure events.
1.8. The LMS must go on standby on user intervention.

1.8.1. The LMS goes on standby while the turn signal is active.
1.8.2. The LMS goes on standby when the user applies brake.
1.8.3. The LMS goes on standby when the user applies torque greater than the threshold of 1.5 N·m.

1.9. In case of software or sensor failures, the LDWS should notify drivers with a visual and auditory alert.

1.10. The LMS transitions to standby if detection confidence goes below 0.95.

1.11. The LDWS issues system failure alert to user within 100 ms when LMS is deactivated due to low confidence value.

1.12. The LMS must not apply torque to the vehicle while the LMS is not in the active state.

1.13. Steering correction should not exceed 1.5 N·m.

2. Primary Requirements

2.1. Hardware

2.1.1. Camera – The LMS shall have a camera sensor.

2.1.1.1. The camera shall be mounted at the front center of the vehicle, positioned behind the rearview mirror at the top of the windshield.

2.1.1.2. The camera shall have a range of ten feet ahead of the vehicle.

2.1.1.3. The camera data is used by the LDS to detect lane boundaries.

2.1.2. **Electronic Steering** - The LMS shall have access to electronic steering capable of exerting 1.5 N/m of force on the steering wheel.

2.1.3. **On/Off Button** – The LMS will have a button mounted on the steering wheel which toggles the on/off status of the LMS.

2.1.4. **Automatic Steering Button** – The LMS will have a button mounted on the steering wheel which enables and disables the automatic steering applied by the system.

2.1.5.

2.1.6. **Torque Sensor** – There will be a torque sensor in the steering wheel to detect user steering input in case of user override of the system.

2.1.7. **Haptic Actuator** – There will be a haptic actuator in the steering wheel to alert the user of situations which are safety critical and require attention.

2.1.8. **Sound System** – The user interface must be able to send messages to activate the sound system in the event of a serious warning.

2.2. Software

2.2.1. The LMS turns on when the user presses the on button located on the steering wheel.

2.2.2. The LMS is divided into four subsystems, each responsible for a different function of the LMS.

2.2.2.1. The LDS is responsible for identifying lane boundaries and their positions.

2.2.2.2. The LCS centers the vehicle in the lane boundaries.

2.2.2.3. The LKS orients the vehicle back into the lane boundaries if it exits the lane boundaries.

2.2.2.4. The LDWS sends warnings to the user when the user commits a lane violation.

2.2.3. Lane Detection System (LDS)

2.2.3.1. The LDS shall identify the lane the vehicle is in and the distance of the vehicle from the right and left lane boundaries within tolerance of 5cm.

2.2.3.2. The LDS shall detect lane boundaries of both solid and discontinuous (dashed) lane markings regardless of curvature or color.

2.2.3.3. The LDS shall compute the positions of lane boundaries with a confidence value of 0-1, indicating the percent confidence that the LDS has identified the lanes accurately.

2.2.3.4. The LDS receives lane positions from and computes the curvature of the lane.

2.2.3.5. The LDS periodically receives sensor data every 100hz to compute current vehicle speed within a tolerance of 1 km/h.

2.2.3.6. The LDS periodically receives sensor data to compute current steering angle within tolerance of 0.02 radians.

2.2.4. Lane Centering System

2.2.4.1. While active, the LCS will apply corrective torque to the vehicle to bring the vehicle back to the centerline of the lane.

2.2.4.2. While active, the LCS will send a message to the electronic steering actuator to apply corrective torque to the steering wheel every 100 ms.

2.2.4.3. When encountering road curvature, the LCS will use relative lane boundary position, steering angle, and road curvature values to predict the path of the vehicle.

2.2.4.4. The requisite torque will be calculated as a function of the current state of the vehicle, the predicted path of the vehicle, and the curvature of the road with the intention of bringing the vehicle closer to the centerline of the current lane.

2.2.4.5. The LCS will send a message after computing the torque to the user interface to apply the computed corrective torque to the electric steering of the vehicle.

2.2.5. Lane Keeping System

2.2.5.1. The Lane Centering system activates when lane boundary confidence is above 0.95 and vehicle is between lane boundaries.

2.2.5.2. While active, the LKS will attempt to bring the vehicle back within lane boundaries.

2.2.5.3. When encountering road curvature, the LCS will use relative lane boundary position, steering angle, and road curvature values to predict the path of the vehicle.

2.2.5.4. The LKS will maintain a vehicle steering angle of 0.5 rad relative to the lane centerline when the vehicle has crossed the left lane boundary.

2.2.5.5. The LKS will maintain a vehicle steering angle of -0.5 rad relative to the lane centerline when the vehicle has crossed the right lane boundary.

2.2.5.6. The LKS will use the most recent steering angle data of the vehicle to compute the necessary torque to apply to steering actuator to bring the vehicle back to 0.5 or –0.5 rad steering angle.

2.2.5.7. The LKS will send a message after computing the torque to the user interface to apply the computed corrective torque to the electric steering of the vehicle.

2.2.6. Lane Departure Warning System

2.2.6.1. The LDWS will have a tiered warning system with light warnings and serious warnings.

2.2.6.1.1. The light warning will generate haptic and visual feedback to the user.

2.2.6.1.2. The serious warning will generate haptic, visual, and auditory feedback to the user.

2.2.6.1.3. The intensity of serious warning haptic feedback will be 1.5 times more intense than haptic feedback of light warning.

2.2.6.2. The LDWS will generate haptic feedback in the steering wheel's haptic actuator on the side where the LDS detects the lane violation.

2.2.6.3. The LDWS will generate auditory feedback in the vehicle's sound system.

2.2.6.4. The LDWS will generate visual feedback on the vehicle's infotainment screen.

2.2.6.4.1. The light warnings will display a yellow symbol on the infotainment screen to indicate mild severity.

2.2.6.4.2. The serious warnings will display a red symbol on the infotainment screen to indicate greater severity.

2.2.6.5. The LDWS will issue a light warning when the LDWS is active, and the LDS detects that the vehicle has entered 0.75 m (2.5 ft) of the right or left lane boundary [2].

2.2.6.6. The visual feedback of a light alert must be issued before the lane departure exceeds 0.3 m (1 ft.) [2].

2.2.6.7. The LDWS must issue a light alert prior to, or concurrent with, the start of the intervention by LCS or LKS [2].

2.2.6.8. The LDWS will issue a single serious warning while the LDWS is active, and the LDS detects that the vehicle has crossed one of the lane boundaries.

2.2.7. User Interface

2.2.8. The user interacts with the user interface by pressing buttons located on the steering wheel.

2.2.8.1. The user interacts with the user interface by pressing the LMS button located on the steering wheel which toggles the LMS on and off.

2.2.8.2. The user interacts with the user by pressing a button called steering assist which enables or disables automatic steering.

2.2.9. The dashboard should clearly show when lane assist is active, standby, or turned off.

2.2.10. The user interface will act as the interface for messages sent from LDWS to system actuators.

2.2.11. Deactivating automatic steering prevents LMS from activating LCS or LKS.

2.2.12. System States

2.2.12.1. The LMS enters the on state when the user presses the LMS button on the steering wheel if the system is initially in the off state.

2.2.12.2. The LMS enters the off state when the user presses the LMS button on the steering wheel if the system is initially in the on state.

2.2.12.3. The LMS enters the standby state when the system state transitions from off to on.

2.2.12.4. The LMS transitions from the standby state to the active state when the LDS detects a confidence value greater than 0.95 and the LDS detects the vehicle's speed is 56.327 Km/h (35 mph) or greater.

2.2.12.5. The LMS transitions from the active state to the standby state when the LMS receives a message from the user interface that the user input is overriding the system.

- 2.2.12.5.1. The LMS transitions from active to standby when the torque sensor in the steering wheel detects the user exerts a torque over 1.5 N/m on the steering wheel.
- 2.2.12.5.2. The LMS transitions from active to standby when the brake sensor detects any input from the user.
- 2.2.12.5.3. The LMS transitions from active to standby when the turn signal is active.

2.2.12.6. The LMS sets LCS to the active state while the LMS is in the centering state.

2.2.12.7. The LMS sets LKS to the active state while the LMS is in the keeping state.

2.2.12.8. The LMS changes from active to centering when the LDS detects that the vehicle is within lane boundaries.

2.2.12.9. The LMS changes from active to keeping when the LDS detects that the vehicle is within lane boundaries.

2.2.12.10. The LMS changes from keeping to centering when the LDS detects that the vehicle is within lane boundaries.

2.2.12.11. The LMS changes from centering to keeping when the LDS detects that the vehicle is within lane boundaries.

2.3. Security

2.3.1. When receiving a message from an over-the-air update, verify that the message is valid and not from a false or malicious source.

2.3.2. Data communicated between subsystems should be encrypted and handled appropriately to ensure that the information they contain has not been altered by an external source.

4 Modeling Requirements

This section outlines multiple models used to describe the LMS. These models include the Use Case Diagram, Domain Model, Sequence Diagrams, and State Diagrams. To best describe system functions internally and interaction with the driver and subsystems, each diagram will follow Unified Modeling Language (UML) notation.

4.1 Use Case Diagram

The use case diagram visualizes how the system will work from the user's perspective, describing major use cases. The actors include the driver, camera, ECU, and steering system, which are shown as stick figures outside of the system boundary. Each actor represents an outside factor that interacts with LMS. The use cases are represented as ovals, connected to their associated actors with solid lines. Dotted arrows labeled «include» represent actions that are mandatory sub-steps of a larger use case and are always performed whenever that use case occurs. Dotted arrows labeled «extend» represent optional or conditional behaviors that are triggered only when specific conditions arise, thereby extending the behavior of a primary use case.

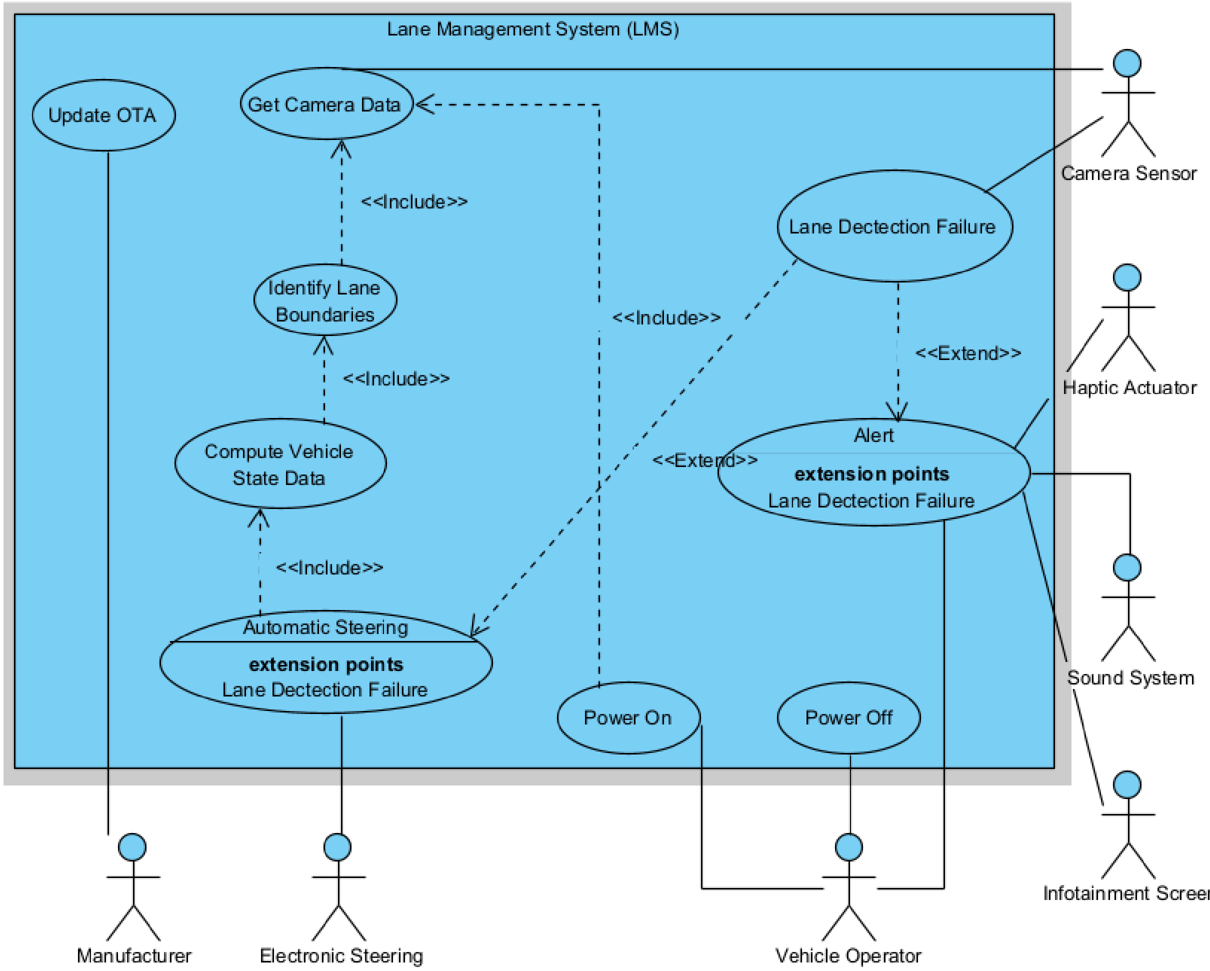

Fig2: Use Case Diagram

Figure 2 shows the main use cases of the LMS system and the external actors that interact with it. The Vehicle Operator controls system activation through the Power On and Power Off use cases, while the Camera Sensor and Electronic Steering System exchange data with the LMS during lane detection and steering corrections. Mandatory subsystem operations are retrieving camera data, identifying lane boundaries, and computing vehicle states. These are shown with include, indicating tasks that always occur during Automatic Steering. The Lane Detection Failure use case extends Automatic Steering and Alert to represent exceptional conditions that modify normal behavior. The Manufacturer interacts only through the update over the air use case. Overall, the diagram highlights how LMS integrates sensing, processing, and alerting functions to support lane-keeping.

b. Use case descriptions (in alphabetical order)

Use Case:	Alert
Actors:	Sound System, Infotainment Screen, Vehicle Operator, Haptic Actuator
Description:	When lane detection indicates deviation, boundary crossing, or subsystem failure, the LDWS alerts the driver using visual, auditory, and haptic feedback. Light warnings occur when the vehicle approaches lane boundaries, and serious warnings occur after a lane crossing or critical failures.
Type:	Primary (essential)
Includes:	None
Extends:	Lane Detection Failure
Cross-refs:	1.6, 1.8, 2.2.6.1-2.2.6.8, 2.2.10
Use cases:	Automatic Steering

Use Case:	Automatic Steering
Actors:	Electronic Steering System, Vehicle Operator
Description:	The LMS uses LCS and LKS to apply corrective torque while active. LCS centers the vehicle when within lane boundaries, and LKS returns it to the lane when crossing. Automatic steering disengages on driver override, braking, turn signal activation, LDWS failure, low confidence, or any condition causing a transition to standby.
Type:	Primary (essential)
Includes:	None
Extends:	Lane Detection Failure
Cross-refs:	1.3, 1.7, 1.9-1.12, 2.1.2, 2.2.4.1-2.2.4.5, 2.2.5.1-2.2.5.6, 2.2.12.4-2.2.12.11
Use cases:	Alert

Use Case:	Compute Vehicle State Data
Actors:	None (internal LMS computation triggered via <<include>> by Automatic Steering)
Description:	The LMS computes current vehicle state parameters, including vehicle speed, steering angle, curvature, lateral offset, and driver-applied torque. These values are required for lane-centering and lane-keeping control during Automatic Steering.
Type:	Primary (support)
Includes:	Identify Lane Boundaries
Extends:	None
Cross-refs:	2.2.4.1–2.2.4.5, 2.2.5.1–2.2.5.6, 2.2.12.4–2.2.12.11
Use cases:	Automatic Steering

Use Case:	Get Camera Data
Actors:	Camera Sensor

Description:	The LMS continuously retrieves real-time image data from the forward-facing camera. This data serves as the input for lane detection, distance estimation, and steering computation.
Type:	Primary (essential)
Includes:	None
Extends:	None
Cross-refs:	2.1.1, 2.1.1.1-2.1.1.3
Use cases:	Compute Vehicle State Data, Identify Lane Boundaries, Lane Detection Failure

Use Case:	Identify Lane Boundaries
Actors:	Camera Sensor, Vehicle Operator
Description:	The LMS analyzes the camera feed to identify left and right lane boundaries, determine road curvature, and compute the lane-confidence value. This provides boundary geometry needed for state computation and steering control.
Type:	Primary (essential)
Includes:	Get Camera Data
Extends:	None
Cross-refs:	2.2.3.1-2.2.3.6
Use cases:	Automatic Steering, Alert, Compute Vehicle State Data

Use Case:	Lane Detection Failure
Actors:	Camera Sensor, Infotainment Screen, Vehicle Operator
Description:	Triggered when lane markings cannot be reliably detected or when the confidence level drops below 0.95. The system transitions to standby, disables automatic steering, and issues appropriate alerts within required timing.
Type:	Secondary
Includes:	None
Extends:	Alert, Automatic Steering
Cross-refs:	1.4, 1.9-1.10
Use cases:	Alert, Automatic Steering

Use Case:	Power Off
Actors:	Vehicle Operator
Description:	The driver presses the LMS button to turn off the system. All steering assistance and warnings deactivate. Any active steering

	torque is removed immediately, and the user is notified of the state's change if necessary.
Type:	Secondary
Includes:	Lane Detection Failure
Extends:	None
Cross-refs:	1.2, 1.11, 2.2.12.2
Use cases:	None

Use Case:	Power On
Actors:	Vehicle Operator
Description:	The driver presses the LMS button to turn the system on. The LMS initializes sensors, retrieves camera data, checks LDWS, and prepares for lane monitoring. The system begins in standby before becoming active if speed and confidence conditions are met.
Type:	Primary (essential)
Includes:	Get camera data
Extends:	None
Cross-refs:	1.1, 2.2.12.1,2.2.12.3
Use cases:	Lane Detection Failure, Identify Lane Boundaries, Compute Vehicle State Data

Use Case:	Update
Actors:	Manufacturer
Description:	The manufacturer transmits software or calibration updates. The LMS verifies the source and integrity of the update using security protocols before applying changes. Invalid or malicious messages are rejected.
Type:	Secondary
Includes:	None
Extends:	None
Cross-refs:	2.3.1, 2.3.2
Use cases:	Power On

4.2 Domain Model

The domain model defines the key subsystems, their responsibilities, and the relationships between them, providing a high-level structural view of how the LMS-1 system organizes and exchanges information across its components.

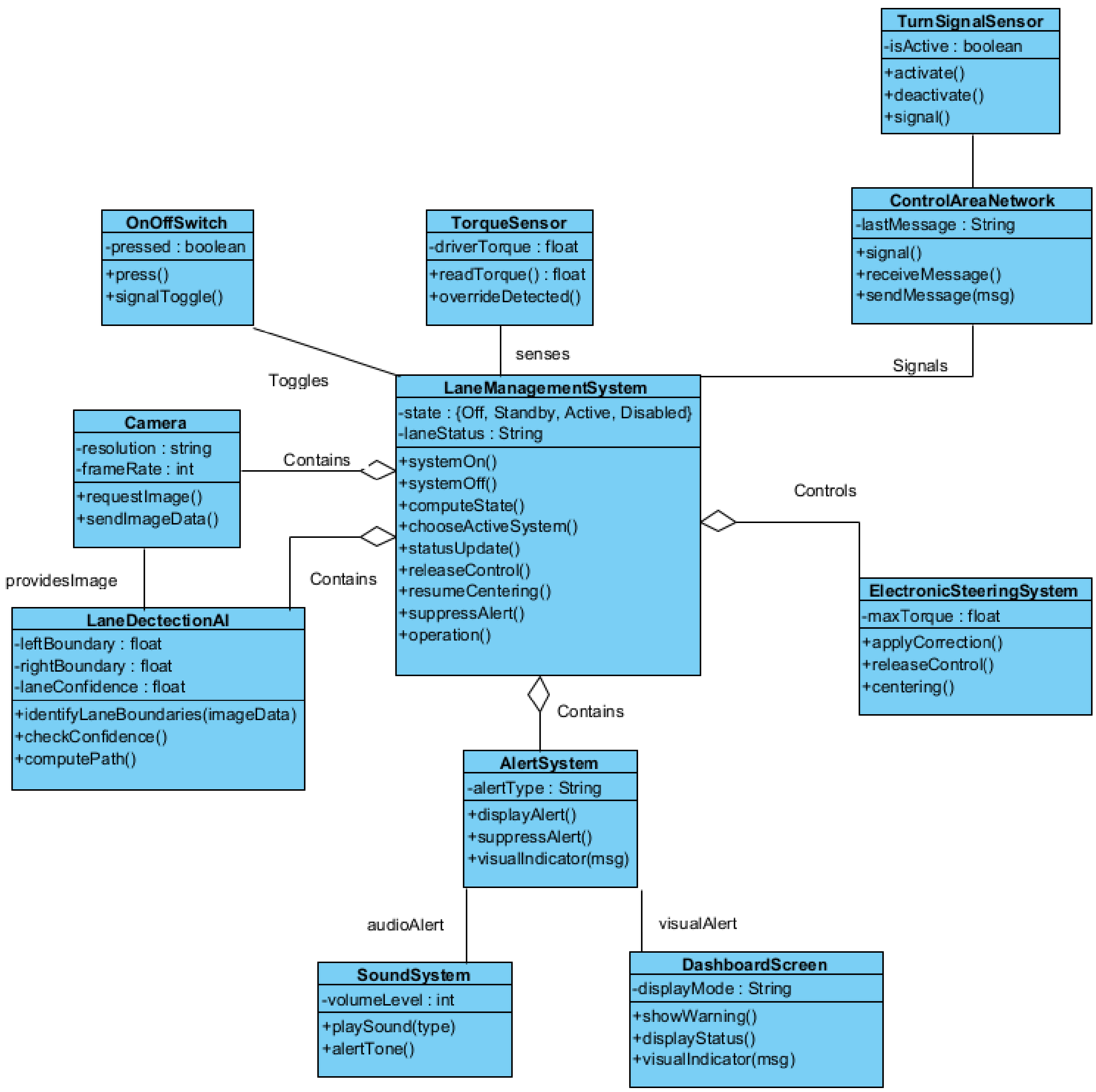

Fig3: domain model

Figure 3 shows the domain model for the LMS-1 system and shows the major subsystems and their relationships. The Camera and Lane Detection AI provide lane-position information to the Lane Management System, which serves as the central coordinating component. The LMS controls the Electronic Steering System for lane-centering responses and communicates with the Alert System to issue visual, audio, and haptic warnings through the Dashboard Screen, Sound System, and Haptic Actuator. The Control Area Network provides turn-signal data to the LMS, while the On/Off Switch and Torque Sensor allow the driver to toggle the system and override steering when necessary. Overall, the diagram illustrates how LMS-1 integrates sensing, processing, steering control, and driver feedback into a unified lane-management architecture.

a. Class data dictionary

Class	On/Off Switch	
Name	***Description (responsibilities)*** Physical switch that allows the driver to toggle the LMS on or off.	
	Export control (public: yes/no) yes	
	Relationships	Associations: Lane Management System
		Aggregations: None
		Generalization: None
	List of attributes and their primitive types Pressed: boolean – indicates if the switch has been pressed	
	List of operations (include parameters and results) press() - called when driver presses the switch signalToggle() - sends toggle signal to LMS	

Class	Lane Detection AI
Name	***Description (responsibilities)*** Detects lane boundaries, computes lane confidence, and interprets lane geometry.
	Export control (public: yes/no) yes
	Relationships Associations: Camera Aggregations: Lane System Management Generalization: None
	List of attributes and their primitive types: leftBoundary: float – Distance to left lane boundary rightBoundary: float – Distance to right lane boundary laneConfidence: float – Confidence score (0-1) for lane detection
	List of operations (include parameters and results): identifyLaneBoundaries(imageData) – Detects left/right boundaries checkConfidence() - Computes and returns lane confidence computePath() - Predicts vehicle path based on lane curvature

Class	Camera
Name	***Description (responsibilities)*** Captures forward-facing road images for processing by the Lane Detection AI. ***Export control (public: yes/no) yes*** ***Relationships*** Associations: Lane Detection AI Aggregations: None Generalization: None ***List of attributes and their primitive types:*** resolution: string – Image resolution FrameRate: int – Frames per second ***List of operations (include parameters and results):*** RequestImage() - LMS request new camera frame SendImageData() - Sends captured image data outward

Class	Lane Management System
	Description (responsibilities) Central system managing lane detection, steering control, lane centering, lane keeping, and warnings. ***Export control (public: yes/no) no***

Name	***Relationships***	Associations: Torque Sensor, Electronic-Steering System
		Aggregations: Control Area Network, Alert System, Lane Detection AI, Camera
		Generalization: None
	Attributes: state: {off, standby, active, disabled} - State LSM is in laneStatus: string – Indicates lane position status (centered, drifting, crossed)	
	Methods: systemOn() - Activates the LMS systemOff() - Deactivates the LMS computeState() - Updates lane/vehicle state chooseActiveSystem() - Select LCS or LKS on lane status statusUpdate() - Sends status to UI or logs releaseControl() - Disables steering control (e.g., during turn signal) resumeCentering() - Restores lane-centering after interruption suppressAlert() - Prevents alerts during intentional lane changes	

Class	Torque Sensor
	Description (responsibilities) Detects driver-applied torque for override detection

	Export control (public: yes/no) no	
Name	***Relationships***	Associations: Lane System Management
		Aggregations: None
		Generalization: None
	List of attributes and their primitive types:	
	driverTorque: float – Torque applied by driver	
	List of operations (include parameters and results):	
	readTorque() - Returns measured driver torque	
	overrideDetected() - Returns true if driver override occurs	

Class	Alert System	
	Description (responsibilities)	
	Provides feedback (visual, audio, haptic) to inform the driver of lane departures or errors.	
	Export control (public: yes/no) no	
Name		Associations: Lane Management System, Sound System, Dashboard Screen

Relationships

Aggregations: None

Generalization: None

List of attributes and their primitive types

alertType : string – Light or serious warning

List of operations (include parameters and results)

displayAlert() - Issues active warning

suppressAlert() - Temporarily disables warnings

visualIndicator(msg) - Displays icon/staus indicator

systemFailureWarning() - Issues system failure warning

lightWarning() - Issues a light warning

seriousWarning() - Issues a serious warning

Class	Control Area Network

Description (responsibilities)

Vehicle communication network conveying sensor signals (turn signal, braking, etc.).

Export control (public: yes/no) no

Name	**_Relationships_**	Associations: Turn Signal Sensor
		Aggregations: Lane Management System
		Generalization:
	List of attributes and their primitive types lastMessage: string – Most recent CAN message recieved	
	List of operations (include parameters and results) signal(On/Off) - Sends turn signal event to LMS sendMessage(msg) - Broadcasts CAN message recieveMessage() - Receives incoming CAN message	

Class	Infotainment Screen
	Description (responsibilities) Displays visual indications and warnings from the LMS and Alert System.
	Export control (public: yes/no) yes
Name	Associations: Alert systems
	Aggregations: None

	Relationships Generalization: None
	List of attributes and their primitive types displayMode: string – Current display state
	List of operations (include parameters and results) showWarning() - Displays warning icons/messages visualIndicator(msg) - Displays specific visual alerts displayLightWarning() - Displays light warning onto the infotainment screen displaySeriousWarning() - Displays serious warning onto the infotainment screen
Class	Electronic-Steering System
	Description (responsibilities) Applies torque corrections as commanded by the LMS.
	Export control (public: yes/no) yes

Name	**_Relationships_**	Associations: Lane Management System
		Aggregations:
		Generalization:
	List of attributes and their primitive types maxTorque: float – Maximum steering torque actuator supports	
	List of operations (include parameters and results) applyCorrection() - Applies LCS/LKS torque releaseControl() - Stops applying torque centering() - Performs lane centering action	

Class	Sound System
	Description (responsibilities) Generates audio alerts for lane departures and critical warnings.
	Export control (public: yes/no) yes

Name	**_Relationships_**	Associations: Alert System
		Aggregations: None
		Generalization: None

List of attributes and their primitive types

volumeLevel: int – Current volume setting

List of operations (include parameters and results)

playSound(type) - Plays a specific alert sound

alertTone() - Default tone for serious warnings

Class	Turn Signal Sensor	
Name	***Description (responsibilities)*** *Detects turn signal activation, used by LMS to determine driver intent.*	
	Export control (public: yes/no) yes	
	Relationships	Associations: Control Area Network
		Aggregations: None
		Generalization: None
	List of attributes and their primitive types isActive: boolean – Indicattes turn signal state	

List of operations (include parameters and results)

activate() - Activates turn signal

deactivate() - Deactivates turn signal

signal() : boolean – Returns current signal state

4.3 Sequence Diagrams

This part shows the sequence diagrams that show how the different parts of the LMS system work together over time to manage lanes. These diagrams show how messages are sent between sensors, control units, and system actors. They also show the flow of data, decision points, and the conditions under which core behaviors like activating, correcting steering, and alerting happen.

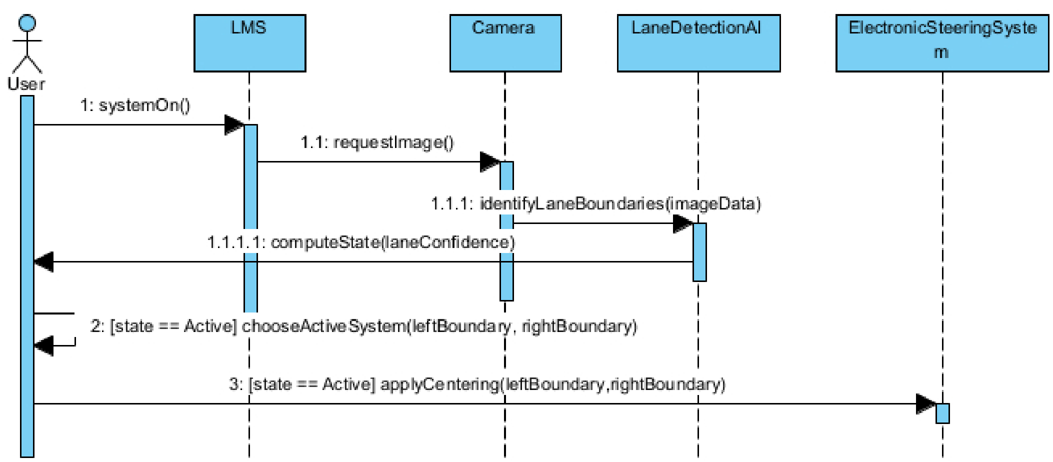

Fig4: scenario1 (normal operation) lane centering v1

Figure 4 shows the normal lane-centering sequence, beginning when the user toggles the system on. The LMS requests camera data, identifies lane boundaries, computes the vehicle state, and selects the lane-centering subsystem. The system then retrieves the vehicle state and applies centering control as long as the lane distances remain within acceptable thresholds. This diagram illustrates the step-by-step interactions between the user, sensors, processing units, and control components during standard operation.

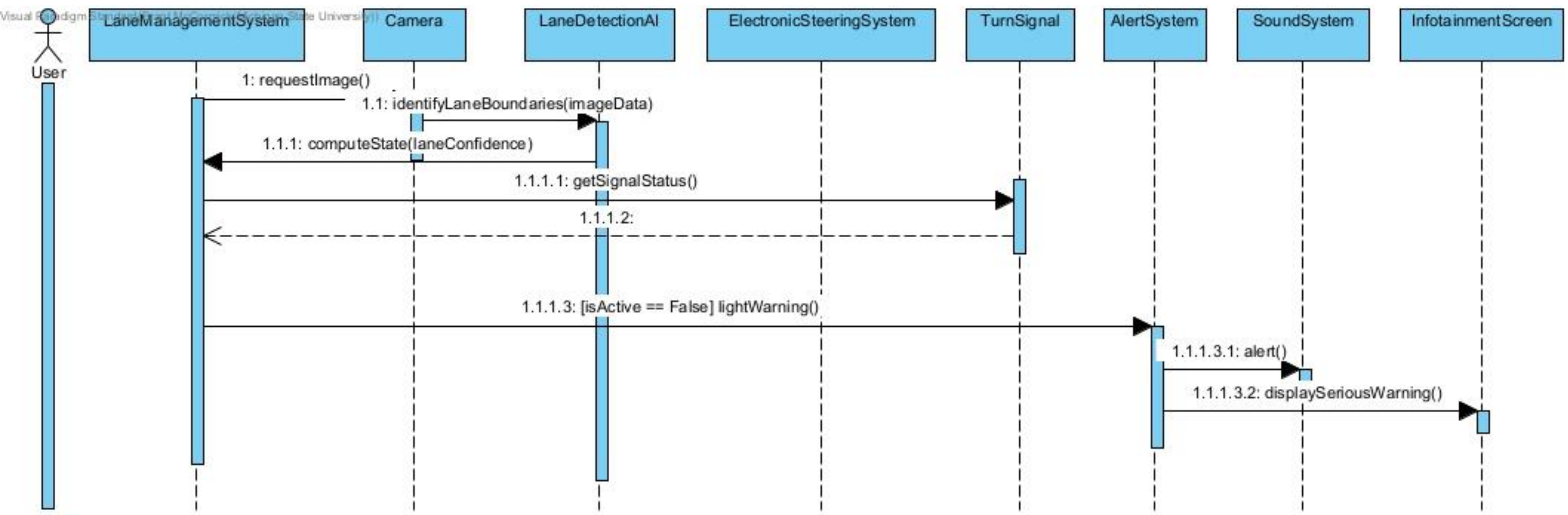

Fig5: scenario2 (exceptional case1) lane centering v1

Figure 5 shows what happens when you get a lane-departure warning. The LMS periodically requests the camera's current sensor data, which prompts the camera to send a message to the Lane Detection AI to identify the lane boundaries. The Lane Detection AI then sends the lane data to the LMS in a message to compute the new state of the LMS. Using the lane data communicated by this message, the LMS identifies that the vehicle has surpassed the lane boundary. The LMS then requests the status of the turn signal. After the LMS receives the inactive status of the turn signal, the LMS assumes the maneuver across the lane boundary was unintentional, and thus the LMS then sends a severe alert through the alert system to warn the user about the lane violation.

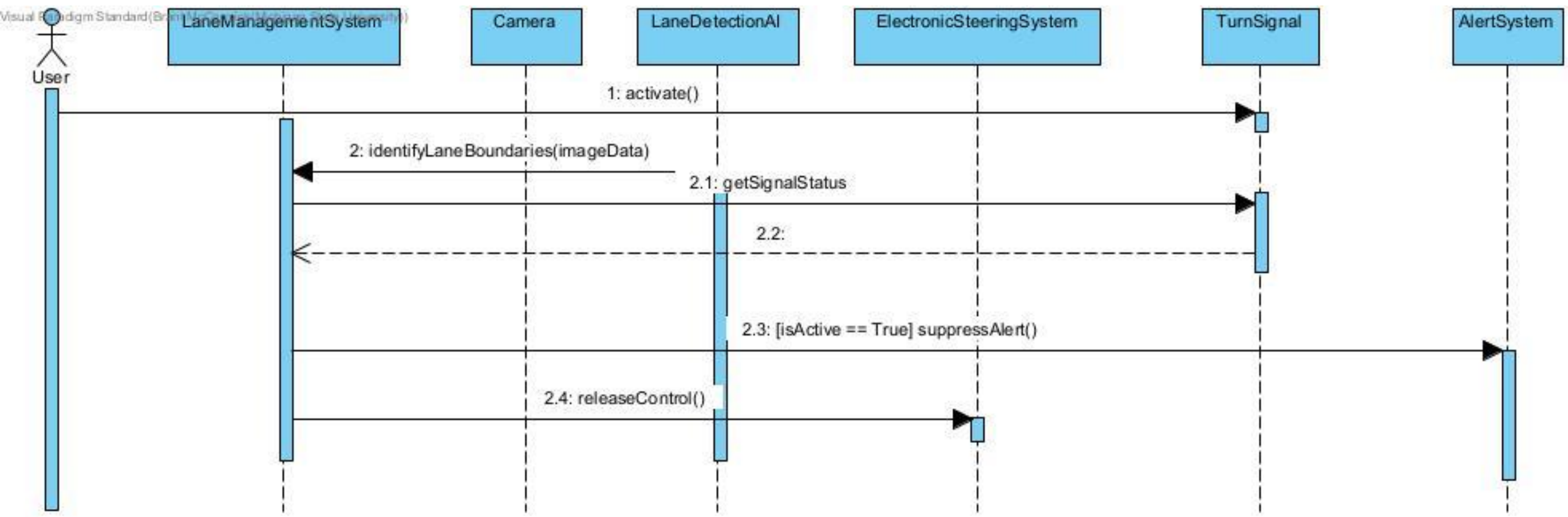

Fig6: scenario3 (exceptional case2) lane change

Figure 6 shows the situation where someone changes lanes on purpose. When the driver turns on the turn signal, the LMS lets go of steering control and stops sending alerts while the turn signal is active, allowing the driver to complete their turn uninhibited. This sequence diagram shows how LMS works when you change lanes on purpose.

4.4 State Diagrams

This section shows the state diagrams for the LMS system and how it changes between important operational modes based on what the driver does, what the sensors pick up, and how the system is doing. Each diagram shows the main states, the events that cause changes between them, and the actions that happen during those changes.

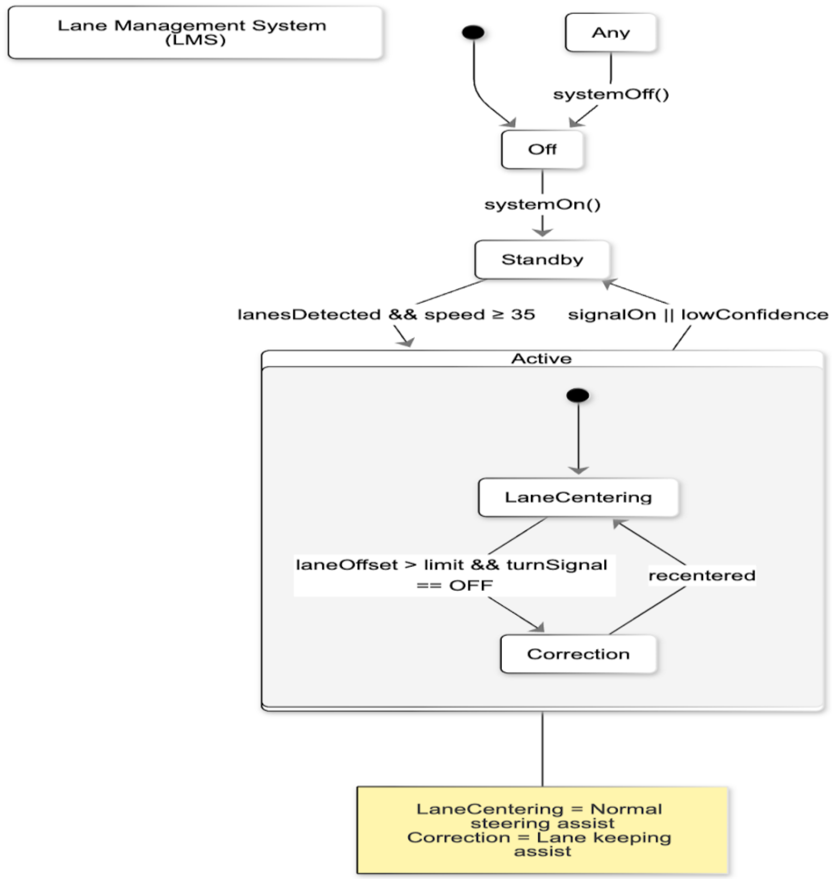

Fig7: lms state diagram

Figure 7 shows the main LMS state diagram. When the system is turned on, it transitions from Off to Standby, and then to Active once lanes are detected and the vehicle speed meets the activation threshold. Within the Active state, the system moves between LaneCentering and Correction depending on lane offset and turn-signal status. When the vehicle is recentered, the system returns to LaneCentering.

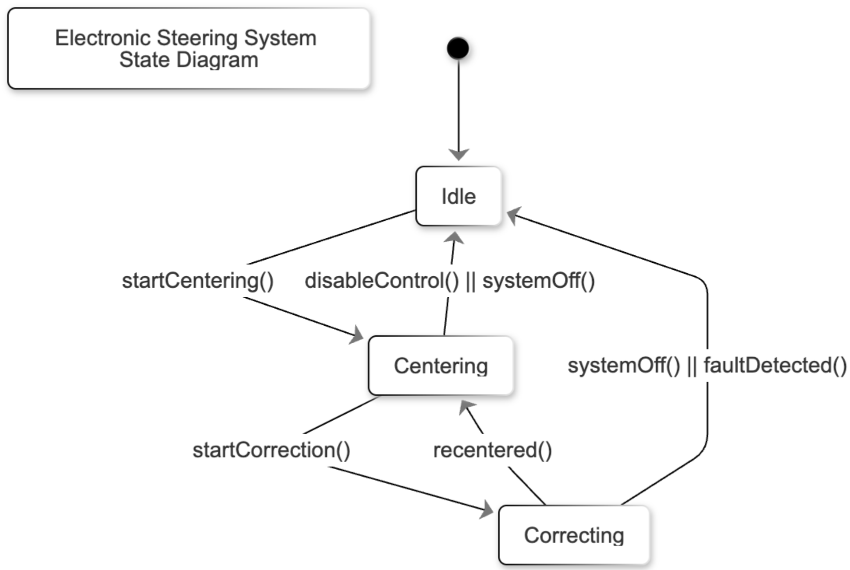

Fig8: steering state diagram

Figure 8 shows the steering subsystem state diagram. The system starts in Idle, enters Centering when normal steering assist is requested, and transitions to Correcting when a correction command is issued. It returns to Centering once the vehicle is recentered, and moves back to Idle if the system is turned off, control is disabled, or a fault is detected.

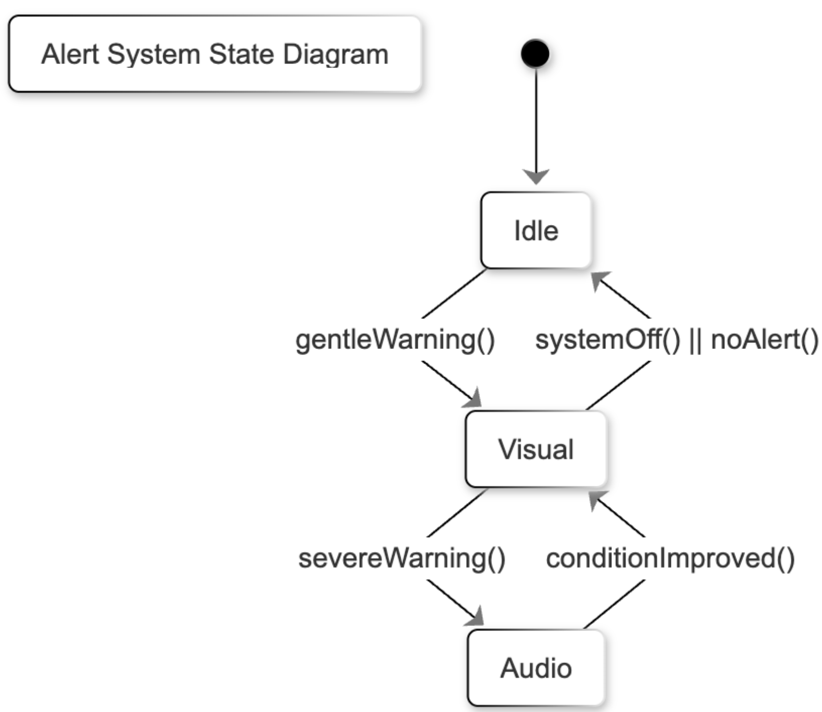

Fig9: alert system state diagram Figure 9 shows the alert system state diagram. The system begins in Idle, enters the Visual state when a gentle warning is issued, and

transitions to Audio if a severe warning is required. It returns to Visual or Idle when conditions improve or when no alert is needed.

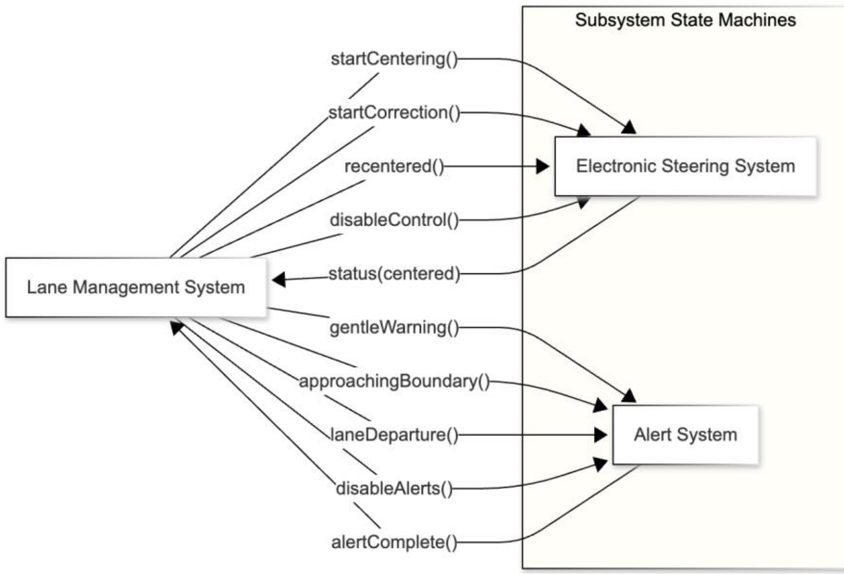

Fig10: intersecting state diagrams

Figure 10 shows how the LMS interacts with the subsystem state machines. The LMS issues commands to the steering and alert systems, such as starting centering, initiating corrections, or generating warnings. The subsystems return status updates back to the LMS

5. Prototype

The purpose of this scenario is to show off the prototype of the LMS model, and how it performs in various scenarios. Below is the sample scenario of how the prototype is used. The scenario is the LMS being at its normal state with no other additions. This prototype provides the customer with an opportunity to see how the team visualizes the system's action. The prototype does not serve as a fully accurate representation of how the system works in real life.

5.1 How to Run Prototype

The prototype is hosted on our website https://cse435.netlify.app/. You can also access the prototype by clicking this link: https://play.unity.com/en/games/32c3fdce-0d18-4b59-9bf8-676288e5cf6e/updated-LMS-prototype-web. When you open the prototype, you are presented with the scenario once you get click the "Normal LMS Behavior" button. The scenario plays a small animation of the vehicle acting in normal LMS behavior. There's also a "STOP" button that you can click to pause the animation at any time. The three scenarios numbered 2, 3, and 4 which are described below can be accessed under the following links:

Prototype Scenario 2 Link: https://play.unity.com/en/games/472a36cc-9126-46fb-b13e-2c09f81224c2/lms-1-prototype-scenario-2

Prototype Scenario 3 Link: Link can be found on our website

Prototype Scenario 4 Link: https://play.unity.com/en/games/b286d407-b058-42d4-9d13-16967eddf7e8/lms-1-prototype-scenario-4

5.2 Sample Scenario(s)

Below are the sample scenarios for the LMS model. There are screen captures that illustrate the prototype at each stage of the animation along with a description.

5.2.1 Scenario 1: Normal LMS Behavior (Perfect Condition)

The normal driving scenario describes how the LMS will operate when driving normally on a clear road. The vehicle remains in the center of the lane at the start and then veers off to the left or right randomly. This scenario shows how the system remains active and that there is no need for adjustments or warnings.

Starting State

This is the starting state of the LMS as the vehicle is in park and is ready to be driven. Once the animation is played, the car moves as normal, driving straight and in between the lanes on a clear day with no distractions.

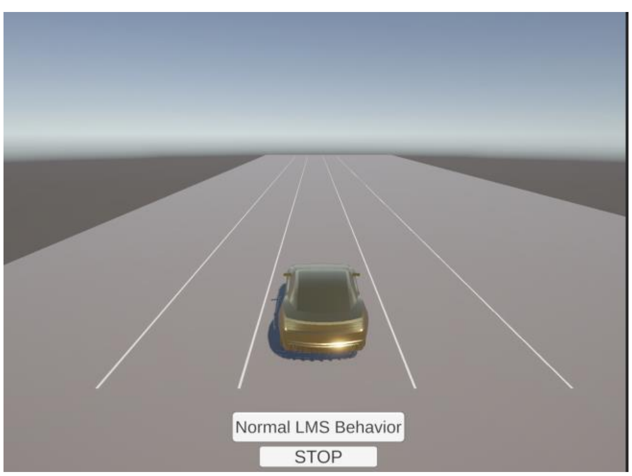

Figure 11: The vehicle is currently in the starting state

Veering off to the left

The second screen capture shows the vehicle steering off to the left at random, with nothing in its way, showing that the vehicle or user itself may have a hard time being in the lane or losing control of the car. Shortly after, the car goes back into the middle lane like it was in.

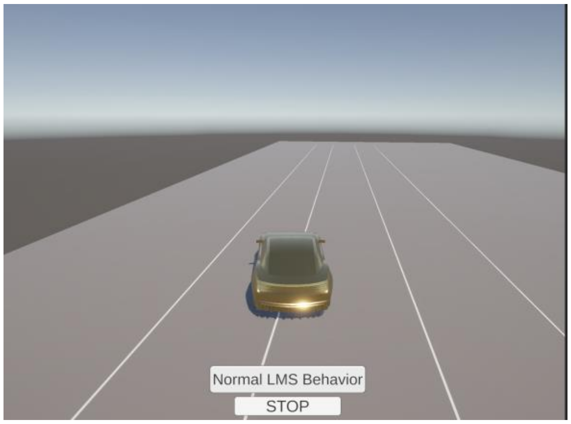

Figure 12: The vehicle is seen steering off to the left

Veering off to the right

The third screen capture shows the vehicle steering off to the right at random, with nothing in its way, showing that the vehicle or user itself may have a hard time being in the lane or losing control of the car. Shortly after, the car goes back into the middle lane like it was in.

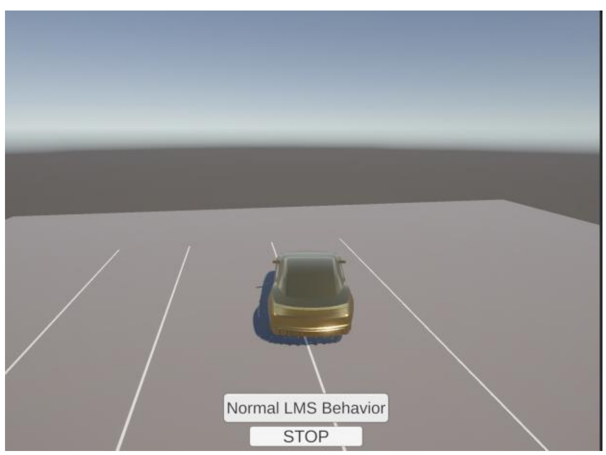

Figure 13: The vehicle is seen steering off to the right

5.2.2 Scenario 2: Road Curvature Scenario

The second scenario of the prototype illustrates the behavior of the LMS and LCS when it encounters a lane with various values of curvature. The vehicle starts in the center line and as the vehicle detects curved lanes, it uses data from the LDS to predict the path of the vehicle and apply compensatory torque to keep the vehicle centered, regardless of the curvature.

The car drives around in a curved lane and tries to stay centered. However if it goes out of the lane, the LMS activates and centers itself back into its respected lane.

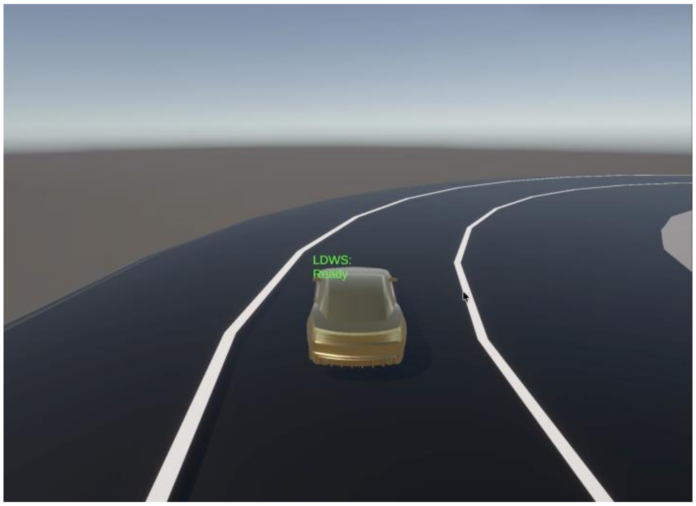

Figure 14: The vehicle is centered in the curved lane

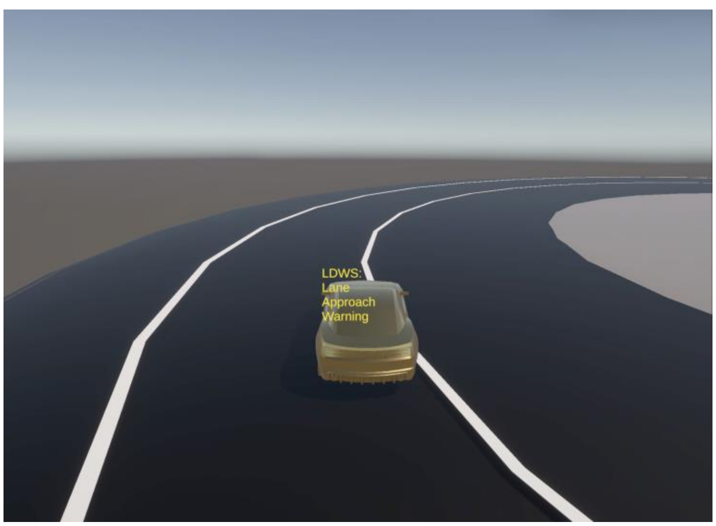

Figure 15: The vehicle is steering off to the right, triggering a lane approach warning.

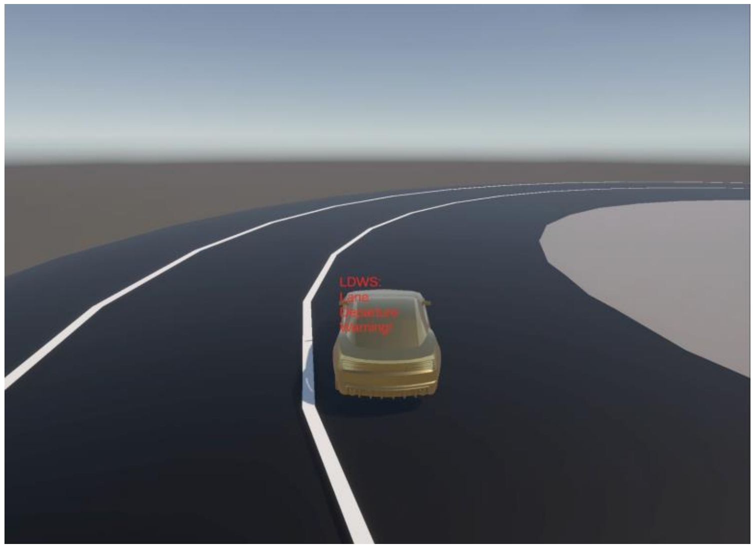

Figure 16: The vehicle is now in a completely new lane, triggering a lane departure warning.

5.2.4 Scenario 4: User Override Scenario

The fourth scenario of the prototype illustrates the behavior of the LMS and LDWS when the user attempts to steer the vehicle outside the lane to enter the adjacent lane. As the vehicle approaches the lane boundary, the LDWS sends a lane approach warning. When the user faces resistance from the system, the user activates the turn signal, putting LMS in standby mode, which disables all systems except LDS. The user will then cross the lane without interference and center themselves in the adjacent lane. After they center themselves in the other lane, the user will deactivate the turn signal, reactivating the LMS and the LCS.

The vehicle starts off with the LMS being turned on. Then once you press the L key, it turns it off and the vehicle is free to go in any lane they choose without the LMS interfering. Once the L key is pressed again, the LMS tuns back on and centers itself back into its respected lane.

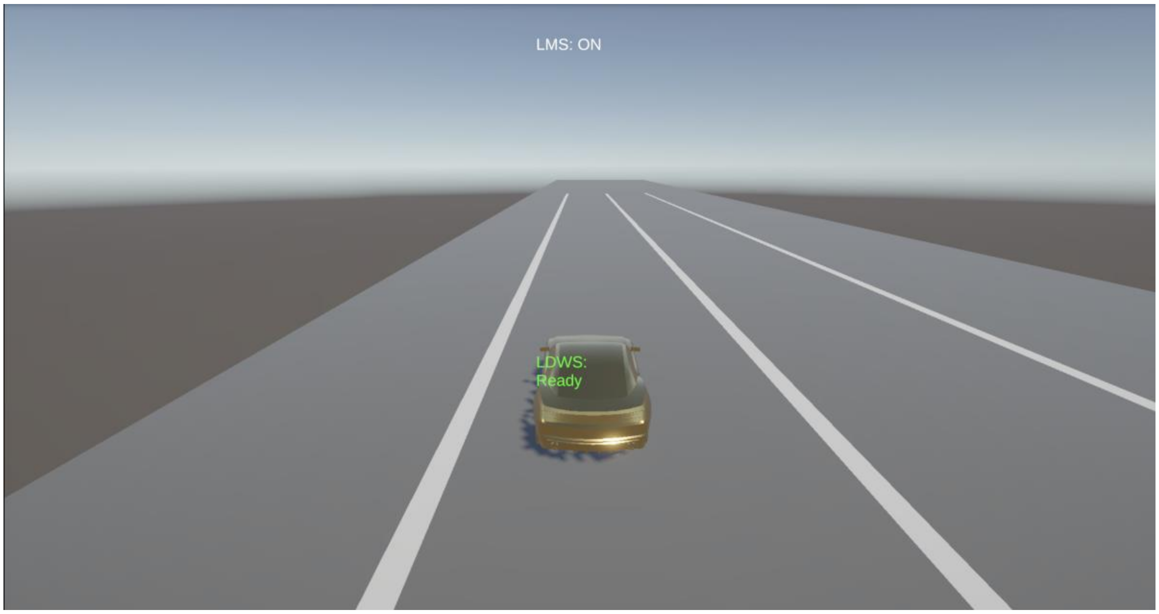

Figure 17: The LMS is on and the vehicle is centered.

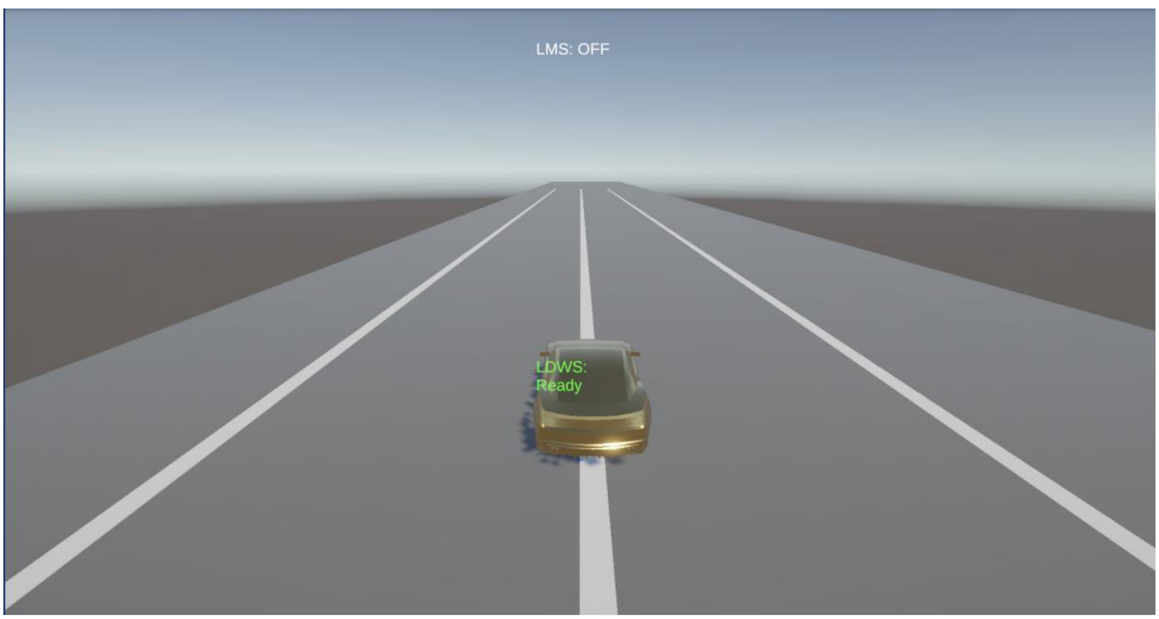

Figure 18: The LMS is off and now the vehicle is free to roam.

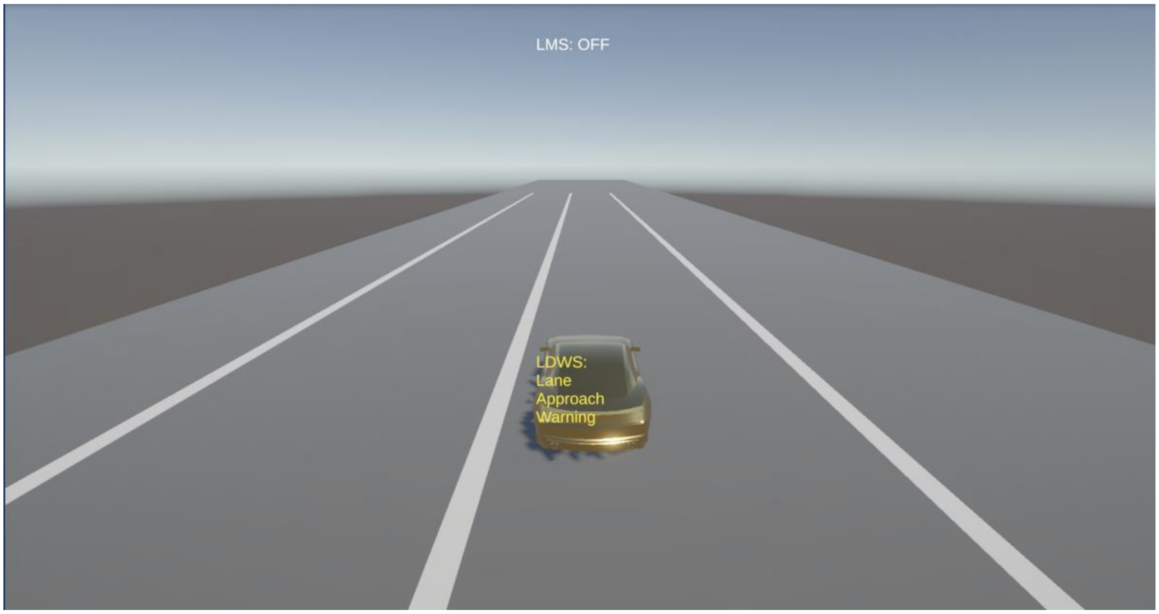

Figure 19: The LMS gives a lane approach warning, but since it is off, it doesn't do anything.

Figure 20: The LMS gives a lane departure warning, but doesn't center itself since the feature is off.

6 References

[1] J. B. Cicchino, "Effects of lane departure warning on police-reported crash rates," *Journal of Safety Research*, vol. 66, pp. 61–70, 2018, doi: https://doi.org/10.1016/j.jsr.2018.05.006.

[2] *Lane Keeping Assist and Lane Departure Warning System Confirmation Test Procedure*, National Highway Traffic Safety Administration, Washington, DC, USA, 2024.

7 Point of Contact

For further information regarding this document and project, please contact **Prof. Betty H.C. Cheng** at Michigan State University (chengb at msu.edu). All materials in this document have been sanitized for proprietary data. The students and the instructor gratefully acknowledge the participation of our industrial collaborators.