Software Requirements Specification (SRS)

Project - Lane Management System

Authors: Sawyer VanDyke, Ryon Baldwin-Williams, Troy Williams, Grant Bossio, Ben Grycza, Collin Heavner

Customer: Mr. Ayush Agrawal

Instructor: Dr. Betty H.C. Cheng

1 Introduction

The Lane Management System(LMS) is a system that will improve the safety of vehicles by giving the driver a safety measure when driving on the road. This is put into several sections, each with dedicated sub-sections to help visualize and make the LMS safe and operational. The document begins with the purpose of the Software Requirements Specification(SRS) and identifies the audience in. Then section 1.2 outlines the scope which will be the main objectives, key features, and limitations of the LMS software. Section 1.3 is about definitions, acronyms, and abbreviations that will then clarify the terms used throughout the document. In section 1.4 an overall description of the LMS will cover its context, functionalities, constraints, assumptions, and dependencies. The Specific Requirements section details both functional and non-functional requirements needed for proper system operation. In the Modeling Requirements section, there will be graphical representations of the LMS structure and behavior. Finally, a Prototype section that will be used to test and validate the LMS system.

1.1 Purpose

The purpose of the SRS document is to provide a detailed description of the requirements for the LMS. This document provides software functionality, operational constraints, and performance criteria to align with our customers' needs. Our general audience for the SRS is our stakeholders, including our customer Mr. Agrawal, and the developers of the LMS

1.2 Scope

The LMS is made to prevent unwanted lane changes or departures, enhancing safety by reducing the risk of collisions and swerving over lanes. The LMS is a system with multiple subsystems that consist of the Lane Keeping System, Lane Centering System, and Lane Departure Warning System. When using LMS it will notify the driver using visual indications on the dashboard, as well as sound alerts if the driver gets close to or is on the lane markings. The LMS will also help the vehicle back into its lane if it goes over the lane markings. The LMS will not be used to detect objects in the lane beside the vehicle or objects in front of the vehicle.

For a lower-level perspective, the LMS will use the camera to detect a lane beside the vehicle on the road, and then the LMS will calculate the distance the lane marking is from the vehicle. If the driver decides to go too close to the lane markings, then the Lane Centering System will go off and alert the driver that the vehicle needs to be adjusted, if the driver ignores the notification and the vehicle goes too far over the lane markings the Lane Keeping System will activate and LMS will provide torque on the steering wheel. If there is a curve the LKS will also help control the speed of the vehicle to allow for proper traction.

1.3 Definitions, acronyms, and abbreviations

Term	**Definition**
Lane Management System(LMS)	Lane Management System is the name of the overall system being built, which is the driver assistance system that includes a variety of different sub-systems
Lane Departure Warning System(LDWS)	Lane Departure Warning System issues warnings to the driver when the vehicle leaves a lane.
Lane Centering System(LCS)	Lane Centering System, allows the vehicle to track lane positioning on a given roadway.
Lane Keeping System(LKS)	Lane Keeping System is an enhancement to LDWS, where the system could intervene and try to send commands to steer and adjust the position of the vehicle.
Camera Sensing Subsystem(CSS)	Captures images on the sides of the vehicle and sends them over to the image processing unit for lane marker detection
Image Processing Subsystem(IPS)	Processes the raw images coming from the camera and identifies the lane marker
Vehicle State Estimation system(VSES)	A set of sensors that would periodically determine the speed, steering angle, and road curvature
Path prediction Subsystem(PPS)	A software subsystem receives information from the Image Processing Subsystem and Vehicle State Estimation system and then tries to predict the path of the vehicle to detect, warn and

	possibly correct any potential lane violations and avoid disabling of LMS.
User Interface system(UIS)	The driver and LMS exchange control and data information through this system.
Supervisory Control Systems(SCS)	The Supervisory Control Systems controls all the other subsystems, decides when to enable and disable other subsystems, and possibly provides diagnostic information.
Driver	The vehicle operator that will be using LMS inside the vehicle.

1.4 Organization

For further sections of the SRS, it will be organized as follows. Section 2 is the overall description of the product perspective, functions, user characteristics, constraints, assumptions and dependencies, and apportioning requirements. Section 3 is the specific requirements needed for the LMS, including functional, non-functional, invariant, and cybersecurity requirements. Section 4 will be the modeling requirements, this will include the use case diagram, class diagram and dictionary, representative scenarios with sequence diagrams, and then state diagrams. Section 5 will describe our prototype, including how to run the prototype, and give sample scenarios. Section 6 will provide the references used for our SRS. Section 7 is the point of contact for the LMS project.

2 Overall Description

This section will describe how the LMS functions. To start, the context and constraints of the system are explored. Next, an overview and explanation of the LMS's functionality is provided. A more in-depth description of the constraints is then discussed. Any assumptions made about hardware, software, the environment, and user interactions with the LMS are covered next. Finally, the document then describes what was determined to be out of scope and/or to be implemented into future versions/releases of the LMS.

2.1 Product Perspective

The LMS is a system that works with the driver of a vehicle to assist with lane keeping and lane centering. The LMS works to correct movements of a vehicle that has drifted out of the center of a lane or that has moved out of a lane that it was previously in without the proper use of a turn signal or indicator. Further, it works with the user by alerting them via audible, visual, and textile alerts and notifications when it detects that the vehicle is beginning to drift from the center of its lane or into another lane.

The primary interface utilized by the user when interacting with the LMS is the dashboard and center console as this is where a lot of the information communicated by the LMS is stored. On the dashboard, the user receives alerts and notifications from the LMS. The user can also customize the functionality and sensitivity of LMS through the center console.

For the LMS to function, the vehicle needs to be equipped with several different pieces of hardware. Most importantly, a camera for image capturing and a computer to process the images, determine the conditions of the road, compute the appropriate course of action, communicate with other systems onboard the vehicle, and control the subsystems of the LMS.

Each subsystem that the LMS is made up of will require software that allows it to function. This will involve system-specific operations as well as general control structures such as activation and deactivation processes. There will also need to be a software or set of software that allows for the communication between the LMS subsystems and other systems on the vehicle such as the steering wheel, the throttle control, and the dashboard.

Finally, memory is a large constraint on the LMS. The size and availability of the vehicle’s memory places a large constraint on how much memory the LMS can use which can make developing the system difficult.

2.2 Product Functions

The LMS is made up of several subsystems that communicate with each other for the entire LMS to properly function. To start, there is a Camera Subsystem that takes pictures of the road. This subsystem is also responsible for distance calculations performed on the images captured. Distance calculations include calculations such as how close or far the vehicle is to the lane markings on the left and right sides of the vehicle. Those pictures are then fed to the Image Processing Subsystem which takes that raw picture data and determines whether there are visible lane markings on the road. Simultaneously, the Vehicle State Estimation System is working with the vehicle's state data. Information such as speed and GPS data is obtained by this unit and fed to the Path Prediction Subsystem along with the result from the Image Processing Subsystem.

From there, the Path Prediction System estimates the path of the vehicle based on the image data fed to it by the Image Processing Subsystem and Camera Subsystem and the Vehicle’s state information provided by the Vehicle Estimation System. From there, the Path Prediction System decides whether to correct the vehicle's movement or push alerts to the user. If a notification or alert needs to be sent to the user, the Path Prediction System sends that information to the User Interface System which then communicates with the dashboard (if the alert is visual and audible) and/or the steering wheel (if the alert is haptic).

The final subsystem required for the LMS to function is a Supervisory Control Subsystem. This subsystem is responsible for controlling the other previously mentioned subsystems. It is continuously communicating with the other subsystems and is waiting for information that suggests whether a subsystem or set of subsystems needs to be turned enabled or disabled. It is also partially responsible for error handling.

If errors are detected, such as functional failure or lack of valid data, this information is to be passed to the Supervisory Control Subsystem. From there, the Supervisory Control Subsystem disables the failing subsystems or the LMS as a whole depending on the severity of failure to ensure safe functionality of the LMS and the safety of the vehicle's passengers. If the error is less severe, subsystems can be put into a passively disabled state in which the subsystems are waiting to be re-enabled as opposed to being completely off.

2.3 User Characteristics

It is expected that the user is legally licensed to drive. It is also expected that the user can interact with the alerts, notifications, and other features of the LMS in a safe and responsible manner. Finally, it is expected that the user operates the vehicle in a safe and responsible manner, operating the vehicle according to state and federal laws where applicable.

2.4 Constraints

Since the LMS interacts with and controls safety-critical systems such as the throttle and the steering wheel, the LMS needs to have a certain set of constraints that ensures the safety of the user and other vehicles, drivers, passengers, and pedestrians on and around the road.

To start, the LMS will not be active at any speeds under 35 mph.

To control the movement of the vehicle, when correcting the vehicle's movement in the event of something such as an unintentional lane departure. For example, the LMS needs to interact with and control the steering wheel. This is done in the form of applying a limited amount of torque to the steering wheel. However, the LMS must not completely control or guide the vehicle along the road as if the system were driving the vehicle itself. The vehicle will apply roughly 2 Nm of torque when correcting the vehicle position [Park, Han, Lee, Kwahk] (this value can vary slightly due to environmental conditions and varying steering angles), so if the user applies a higher level of torque to the wheel compared to that of the torque being applied by the LMS, the LMS will stop applying torque to the steering wheel and relinquish control of the vehicle to the user.

In combination with controlling the steering of the vehicle for a few moments, (only through corrective actions), the LMS must also have a limited amount of control over the throttle of the vehicle. Control is necessary only when the vehicle is being corrected along a curve as an increase or decrease in speed depending on the bank angle of the curve is required to maintain the speed at which the driver entered the curve. However, due to the safety-critical nature of this control, if the driver applies the brakes, the LMS relinquishes its control over the throttle to the user. The throttle must only be used by the LMS to maintain speed while correcting vehicle movement.

The LMS also needs a set of properly working cameras. A lack of functioning cameras can create dangerous situations for the user and other drivers and their vehicles on the road. Due to this, if the LMS detects that the cameras are not working, the LMS should be disabled. Along with this, if the LMS subsystems determine that the Image Processor and/or the Vehicle State Estimation System are not working, the LMS should be disabled as well because attempting to correct the vehicle in this hypothetical state creates incredibly dangerous situations in which loss of life, injury, or damage to property could occur.

One final constraint to consider is the speed of alerts and notifications being sent to the user. These alerts need to be quick enough for the user to correct for the events the LMS is warning the user about.

2.5 Assumptions and Dependencies

There currently exists a wide variety of assumptions and dependencies placed on the vehicle, hardware, software, environment, and user interactions and capabilities for LMS to function properly.

To start, it is assumed that the vehicle has a properly functioning steering wheel, throttle, throttle control system, and dashboard with working visual and audio interfaces.

It is also assumed that there is enough space in memory available for the LMS to be implemented in full for its base functionality. On top of this, it is assumed that the systems that control the communication between the dashboard, throttle control, and steering wheel function properly.

For the LMS to function, it is assumed that the road the vehicle is traveling on has clear lane markings that are clear and visible to the cameras and can be derived from the Image Processing Subsystem.

It is assumed that the user safely operates the vehicle according to federal and/or state laws where applicable. It is also assumed that the user knows how to operate the vehicle properly and is legally licensed to drive. Finally, users are assumed to properly interact with the LMS and react to vehicle corrections, alerts, and notifications.

2.6 Approportioning of Requirements

Currently, out-of-scope operations include obstacle detection and avoidance and blindspot detection and avoidance.

The LMS, at its current state, is not responsible for actively detecting and avoiding any obstacles in front of or around the vehicle while active. The user is still responsible for being aware of the objects that are around them on the road and operating their vehicle safely.

The LMS is also not responsible for blindspot detection and avoidance. If the user were to signal that they were merging into another lane, (actions that comply with the requirements of the LMS for it to properly function), and another vehicle was in the user's blindspot, the LMS is not responsible for alerting the driver of this vehicle or preventing a potential collision with this vehicle. The user is responsible for being aware of the vehicles around their vehicle and vehicles that could potentially be and/or vehicles that are currently in their blind spot.

These features may be explored in future iterations of the LMS.

3 Specific Requirements

1. Functional requirements
   - 1.1. The LMS shall be enabled and disabled by the driver.
     - 1.1.1. The LMS shall be active once the vehicle is in forward gear, going 35 mph or more, and the lanes are detected for 1 second using the camera sensing and image processing subsystems.
   - 1.2. The LMS shall always be able to be overridden by the driver at any time by applying more torque to the wheel than the LMS.
     - 1.2.1. The LMS shall not apply more than 2 Nm of torque.
     - 1.2.2. If any torque in the opposite direction is applied by the driver, then LMS will go into a passive state.
     - 1.2.3. If this is detected the LMS will be in a passive state until the driver is no longer applying any torque to the wheel.
   - 1.3. The LMS will use the supervisory control systems to report to the driver if any subsystem is not working, in which case the LMS will be disabled.
   - 1.4. The LMS shall use the camera sensing and image processing subsystems to detect the lane markings and relative distance to the vehicle for usage in calculating the necessary torque to apply and detecting lane departures.
     - 1.4.1. Apply corrective torque once the lane border has been detected to have been crossed.
   - 1.5. The LMS shall use the Vehicle State Estimation system to monitor:
     - 1.5.1. Speed of Car
     - 1.5.2. Steering Angle of Vehicle
     - 1.5.3. Road Curvature
   - 1.6. The LMS shall enter a passive state if the lane markings become inconsistent or undetectable.
     - 1.6.1. If the LMS enters a passive state, the LMS shall alert the driver by a vibration of the steering wheel, as well as an audible beep with a notification on the dashboard that the LMS is not currently active.
   - 1.7. The LMS will provide audible and visual warnings to the driver when it detects the vehicle moving out of its lane.

1.8. The LMS should be able to detect several different lane markings:
   1.8.1. Solid Yellow
   1.8.2. Solid Double Yellow
   1.8.3. Dashed Yellow
   1.8.4. Solid Yellow & Dashed Yellow
   1.8.5. Solid White
   1.8.6. Dashed White
1.9. The LMS shall ignore all boundaries on the same side as an active blinker.
   1.9.1. Once the blinker is deactivated, the LMS will reevaluate the lane markings before entering an active state if they are detected.
1.10. The LMS shall still give alerts even when the driver is purposely crossing a lane marking without a blinker.
1.11. The LMS shall vibrate the steering wheel to alert the driver if the LMS is attempting to apply torque in the opposite direction of the driver.
1.12. The LMS will monitor the driver's engagement with the steering wheel and will provide alerts when they are not engaged (detection is 2 seconds).
   1.12.1. If the driver is detected to not be engaged with the wheel, then the LMS will enter a passive state until the driver is engaged with the wheel.

2. Non-Functional Requirements
   2.1. Do not annoy the driver with too many unnecessary alerts.
      2.1.1. Necessary alerts are unexpected lane departures and system failures resulting in the LMS being deactivated.
   2.2. Have a clear indication of the LMS being enabled and active on the dashboard.
   2.3. Display proximity to boundaries on the dashboard visually.
   2.4. Do not send false signals.
   2.5. LMS sensitivity should be able to be customizable at the user's discretion
      2.5.1. The user can decide how much torque they wish the LMS to apply when lane-keeping (Low, Medium, High).
      2.5.2. The user can decide how to be alerted when the LMS detects the vehicle leaving its lane (Steering wheel vibration, dashboard message, audible alert, etc.). They cannot turn off any necessary alerts, however.

3. Invariant Requirements
   3.1. The LMS should always be prepared to relinquish control to the driver.
      3.1.1. The LMS shall stop applying steering torque on the steering wheel when the driver is applying any torque in the opposite direction or more torque in the same direction.
   3.2. The LMS shall be active only when driving 35+ miles per hour.
   3.3. The LMS shall not apply more than 2 Nm of torque.
   3.4. The LMS will not be active if a double yellow lane marking is detected on the right side of the vehicle (Regional setting).
   3.5. The LMS shall not work if it is disabled.

3.6. The vehicle must be stable and within the boundaries of the lane.

4. Cybersecurity requirements
    4.1. The most likely pathway for an attack is through the LMS updates, which happen over the air during vehicle updates from satellites. There is no internet or Bluetooth connectivity.
    4.2. The most likely threat actors for the LMS are white-hat hackers and cyberterrorists.
        4.2.1. White-hat hackers could be employed by the companies creating or implementing the software to close up any security breaches in the software.
        4.2.2. Cyberterrorists could be motivated to cause car accidents, by feeding the LMS false data and causing the LMS to make the vehicle swerve.
    4.3. The vulnerabilities in the LMS lie mainly in the camera and image processing subsystems.
        4.3.1. Images could be fabricated and sent to the camera subsystem instead of the intended pictures causing the vehicle to incorrectly detect the lane markings.
        4.3.2. Images could be modified going from the camera subsystem to the image processing subsystem to create a similar effect.
        4.3.3. These can be prevented by adding a verification method to the images to ensure that they were sent internally and not from outside sources.

4 Modeling Requirements

This section contains a variety of models to provide a visual and high-level overview of the proposed Lane Management System alongside its sub-systems and functionality.

4.1 Use Case Diagram

Figure 1 illustrates the relevant actors outside of the LMS alongside the interactions these actors have in relation to the LMS. With this, the interactions between the subsystems themselves are also depicted. The blue rectangle indicates the system boundary and all of the systems and functions within it. Outside of the blue rectangle are the actors or external entities the system interacts with or responds to. Within the system boundary are ovals indicating use cases or system interactions. Some use cases interact which is denoted by a dotted line indicating which type of interaction relationship is involved.

Following this is **Table 1** which lists a variety of use case descriptions for the system.

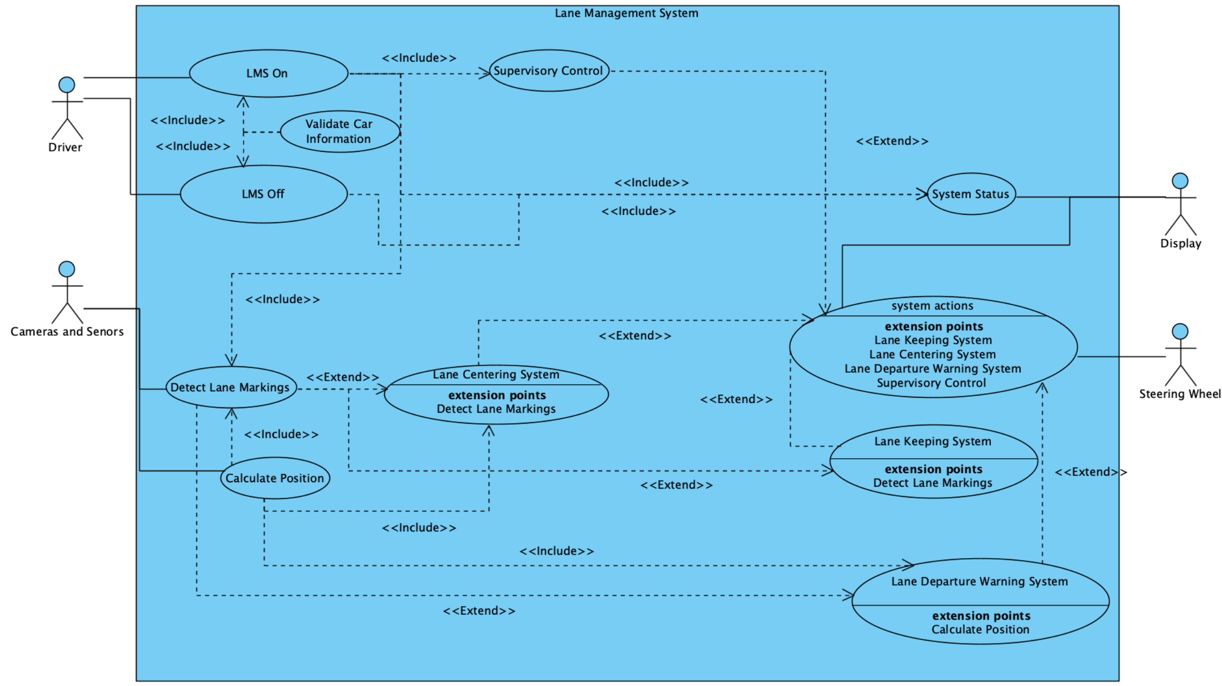


Figure 1: Use Case Diagram for proposed LMS System.

Table 1: Use Case Descriptions for the proposed LMS system.

Use Case	LMS On
Actors	Driver
Description	The LMS shall be enabled and disabled by the driver.
Type	Primary
Includes	Validate Car Information
Extends	N/A
Cross-references	4.4.
Use cases	Validate Car Information, Detect Lane Markings

Use Case	LMS Off

Actors	Driver
Description	The LMS shall be enabled and disabled by the driver.
Type	Primary
Includes	System Status
Extends	Supervisory Control
Cross-references	4.4.

Use Case	Validate Car Information
Actors	N/A
Description	The LMS periodically checks and runs tests on all sensors used in the system.
Type	Primary
Includes	N/A
Extends	N/A
Cross-references	N/A

Use Case	Detect Lane Markings
Actors	Cameras and Sensors
Description	The LMS uses sensors to accurately identify lane markers to send warnings or send steering corrections.
Type	Primary
Includes	Calculate Position
Extends	N/A
Cross-references	4.4.

Use Case	Lane Centering System

Actors	Cameras and Sensors, Steering Wheel
Description	Allows the vehicle to track lane positioning on a given roadway.
Type	Primary
Includes	Calculate Position
Extends	Detect Lane Markings
Cross-references	N/A

Use Case	Calculate Position
Actors	Cameras and Sensors
Description	The LMS shall use the camera sensing and image processing subsystems to detect the lane markings and relative distance to the vehicle
Type	Secondary
Includes	N/A
Extends	N/A
Cross-references	N/A

Use Case	Supervisory Control
Actors	N/A
Description	Controls all the other subsystems, decides when to enable and disable other subsystems
Type	Secondary
Includes	LMS ON
Extends	N/A
Cross-references	N/A

Use Case	Lane Keeping System
Actors	Steering Wheel, Cameras and Sensors
Description	Send commands to steer and adjust the position of the vehicle.
Type	Secondary
Includes	N/A
Extends	Detect Lane Markings
Cross-references	N/A

4.2 Class Diagram

Figure 2: depicts the class diagram for the proposed LMS. The purpose of the diagram is to provide a high-level overview of the LMS. More specifically, the objects that make up the LMS and the relationships between these objects. Each class/object or system is represented by a blue rectangle. Connections with diamonds on the end indicate aggregation or composition. Closed diamonds indicate composition, while open diamonds indicate aggregation meaning that the class furthest from the diamond is part of the class nearest to the diamond. Connections with triangles on the end indicate that the class nearest to the triangle will utilize functions from the class furthest from the triangle.

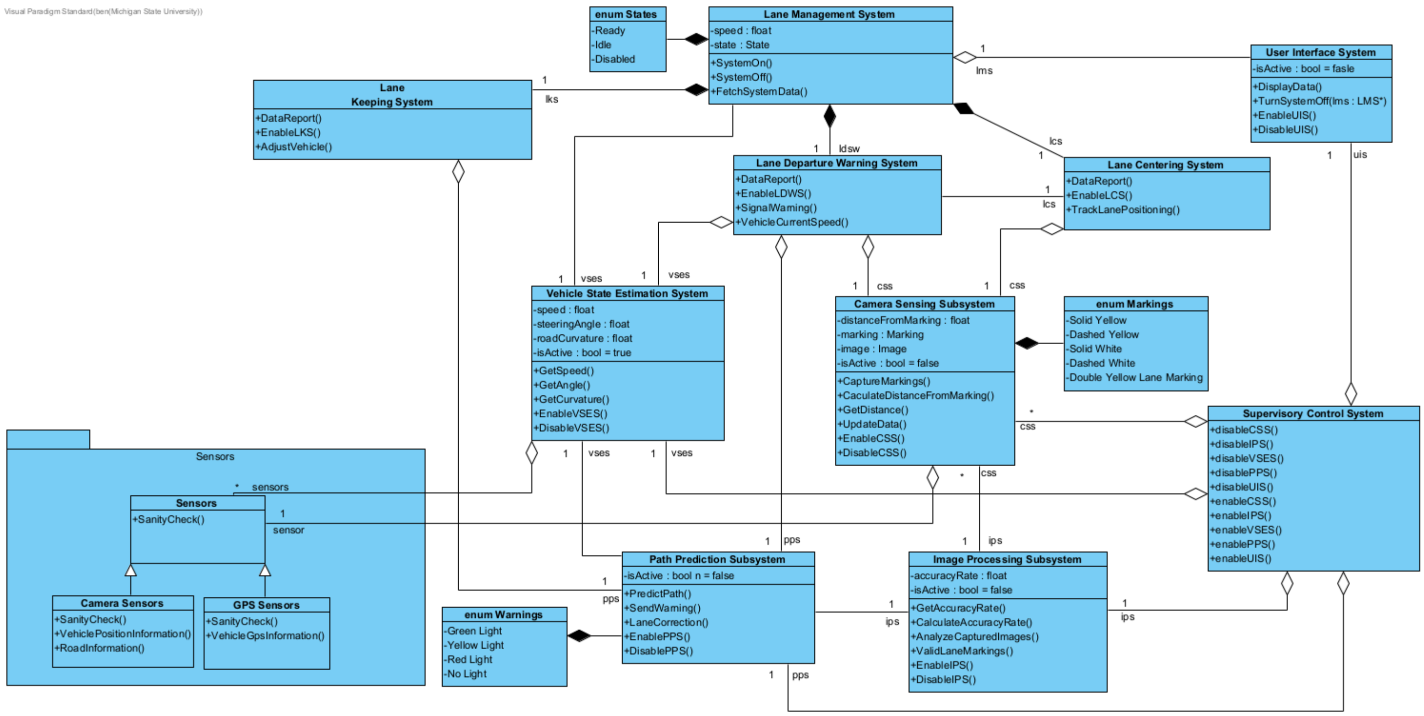


Figure 2: Class diagram for proposed LMS

Class Diagram Dictionary:

Element Name		**Description**
Camera Sensing Subsystem(CSS)		Captures images on the sides of the vehicle and sends them over to the image processing unit for lane marker detection
Attributes		
	distanceFromMarking : float	The calculated distance the car is from the lane marking
	marking : Marking	The type of lane marking

	image : Image	The image that the system receives
	isActive : bool = false	The system is active or not
Operations		
	CaptureMarkings()	This will grab the markings from the image
	CalculateDistanceFromMarking() : float	This will calculate what the distance from the car to the lane marker is
	GetDistance()	This will get the distance from the camera to the lane marker
	EnableCSS()	Enable the CSS when no errors have occurred
	DisableCSS()	Disable the CSS if errors have occurred
	UpdateData()	Update the distance data and image data of the class
Relationships	The Image Processing Subsystem is directly connected to the CSS, LDWS,LCS, Supervisory Control System, and Sensors classes.	
UML Extensions	Enumerated Markings	

Element Name		**Description**
CameraSensors		This will be the built-in camera sensors for the vehicle
Attributes		
	None	

Operations		
	SanityCheck()	Check if the camera is working
	VehiclePositionInformation()	The image of the road
	RoadInformation()	The image of the road
Relationships	CameraSensors is an inheritance of the class Sensors	
UML Extensions	None	

Element Name		**Description**
GPSSensors		This will give the location of the vehicle
Attributes		
	None	
Operations		
	SanityCheck()	Check if the GPS is working
	VehicleGPSInformation()	This will give the location of the vehicle
Relationships	GPS Sensor is an inheritance of the class sensors	
UML Extensions	None	

Element Name		**Description**
Image Processing Subsystem(IPS)		Processes the raw images coming from the camera, and detects a lane marker
Attributes		
	accuracyRate : float	This determines if the image is clear enough to be used

	isActive : bool = false	This is the indication if the subsystem should be active because of accuracy
Operations		
	GetAccuracyRate() : float	This will get the accuracy rate from the attribute accuracyRate
	CalculateAccuracyRate()	This will calculate the accuracy of the image having a lane marking
	AnalyzeCapturedImages()	This function will analyze the image
	EnableIPS()	This will enable the subsystem to function
	Disable()	This will disable the subsystem
Relationships	The IPS has direct connections to the Path Prediction Subsystem, Camera Sensing Subsystem, and Supervisory Control System	
UML Extensions	None	

Element Name		**Description**
Lane Centering System		This is used to track lane positioning on a given roadway.
Attributes		
	None	
Operations		
	DataReport() : void	Sends reports of lane data
	EnableLCS() : void	Sending the alert to the LDWS

	TrackLanePositioning()	Tracks the lane position of the vehicle
Relationships	Lane centering is connected to LMS, LDWS, and CSS	
UML Extensions	None	

Element Name		**Description**
Lane Departure Warning System		Issues warnings to the driver when the vehicle leaves a lane.
Attributes		
	None	
Operations		
	DataReport(): void	This will report the data to the system
	EnableLDWS(): void	Activates the System
	SignalWarning(): void	Will send a warning signal
	VehicleCurrentSpeed():void	Grabs the current speed of the vehicle
Relationships	LDWS is connected with VSES, PPS, CSS, LCS, and LMS	
UML Extensions	None	

Element Name		**Description**
Lane Keeping System		Send commands to steer and adjust the position of the vehicle
Attributes		
	None	
Operations		

	DataReport()	Reports the data to the system
	EnableLKS()	Enables the LKS to operate
	AdjustVehicle()	Adjust vehicle position
Relationships	The LKS is connected to the LMS and the PPS	
UML Extensions	None	

Element Name		**Description**
Lane Management System		Controls the user interface system, LCS, LDWS, and LKS
Attributes		
	Speed : float	The speed of the vehicle
	state : State	The state of the entire system
Operations		
	SystemOn(): void	Turns the system on
	SystemOff(): void	Turns the system off
	FetchSystemData()	Takes in the data of the system
Relationships	LMS is connected to LKS, User Interface System, LDWS, and LCS	
UML Extensions	Enumeration States	

Element Name		**Description**
Path Prediction Subsystem		Predicts the path of the vehicle to detect, warn, and correct any potential lane violations

Attributes		
	isActive : bool = false	This is for if the PPS is active
Operations		
	PredictPath() : void	Predicts the path of the vehicle
	SendWarning()	Sends warning to connected classes
	LaneCorrection()	Corrects the vehicle to stay within the lane
	EnablePPS()	Enables the PPS to operate/be used
	DisablePPS()	Disables the PPS so it cannot be used
Relationships	PPS is connected to VSES, LKS, IPS, LDWS, and Supervisory Control system	
UML Extensions	Enumerated Warnings	

Element Name		**Description**
Supervisory Control System		Responsible for controlling all of the LMS subsystems
Attributes		
	None	
Operations		
	disableCSS()	Disables the CSS
	disableIPS()	Disables the IPS
	disableVSES()	Disables the VSES
	disablePPS()	Disables the PPS
	disableUIS()	Disables the UIS

	enableCSS()	Enables the CSS
	enableIPS()	Enables the IPS
	enableVSES()	Enables the VSES
	enablePPS()	Enables the PPS
	enableUIS()	Enables the UIS
Relationships	The Supervisory Control System is connected with the User Interface System, CSS, VSES, IPS, and PPS	
UML Extensions	None	

Element Name		**Description**
User Interface System		This system will interact with the driver, providing warnings and interactions
Attributes		
	isActive : bool = false	Is working and operating
Operations		
	DisplayData() : void	Displays the warnings on the dashboard
	TurnSystemOff(lms : LMS)	The user can choose to turn the system off
	EnableUSI() : void	Used to make the system operable
	DisableUSI() : void	Used to disable the system
Relationships	UIS is connected with the LMS, and SCS	
UML Extensions	None	

Element Name	**Description**

Vehicle State Estimation System		Uses a set of sensors that would periodically determine the speed, steering angle, and road curvature.
Attributes		
	speed : float	Speed of the vehicle
	steeringAngle : float	Steering angle of the vehicle's steering wheel
	roadCurvature : float	The curvature of the road
	isActive : bool = true	The System activity status
Operations		
	GetSpeed() : float	Gets the speed from the sensors
	GetAngle() : float	Gets angle from Sensors
	GetCurvature() : float	Gets the curvature of the vehicle from the path prediction system
	EnableVSES()	Enables the VSES
	DisableVSES()	Disables the VSES
Relationships	VSES is connected to the Sensors, LMS, LDWS, PPS, and SCS	
UML Extensions	none	

4.3 Representative Scenarios of System

Scenario: System On

Description:

The Vehicle Estimation System checks if the vehicle speed is at least 35 miles per hour before starting the LMS system.

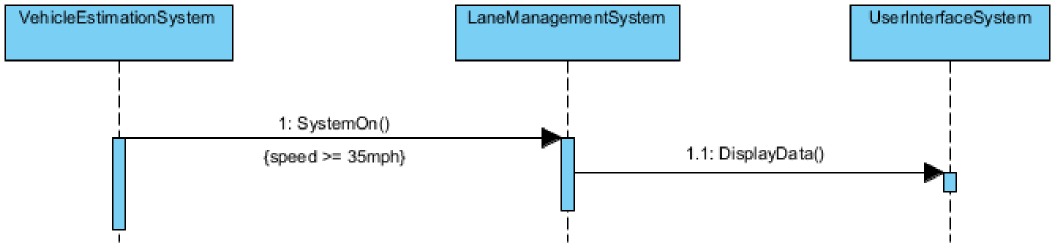


Figure 3: Sequence Diagram for system on

Scenario: Sensor Sanity check

Description:

The Vehicle Estimation System performs a system senor sanity check across all of the sensors that are used in the system. If all passes the system will be set to a ready state meaning the sensors respond correctly. If any fail the system will be set to a disabled state and will notify the user.

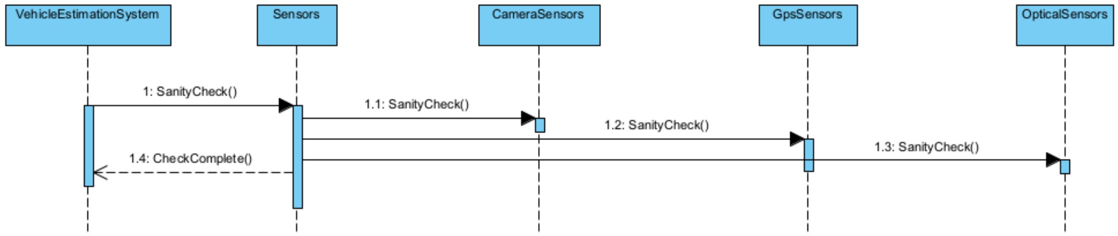


Figure 4: Sequence Diagram for sensor sanity checks

Scenario: Valid Lane Markings

Description: The Camera Sensing Subsystem and the Image Processing Subsystem capture and validate the lane markings on the road if the markings are not valid the system sends a warning to the user.

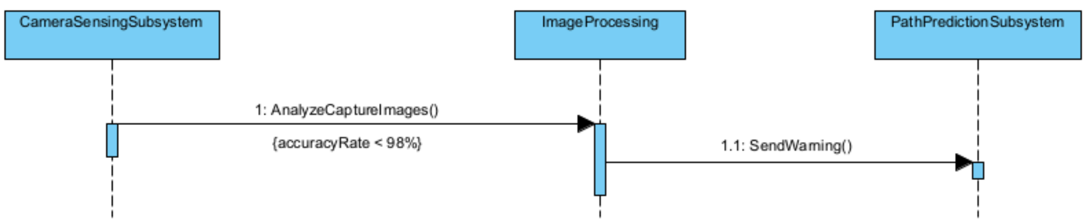


Figure 5: Sequence Diagram for validating lane markings

4.4 State Diagram

In Figure 6, the Image Processing subsystem remains idle until it receives an image to process. The system will process the image, find relevant lane markings, and send the accuracy rate to the Path Prediction System, then return to idle.

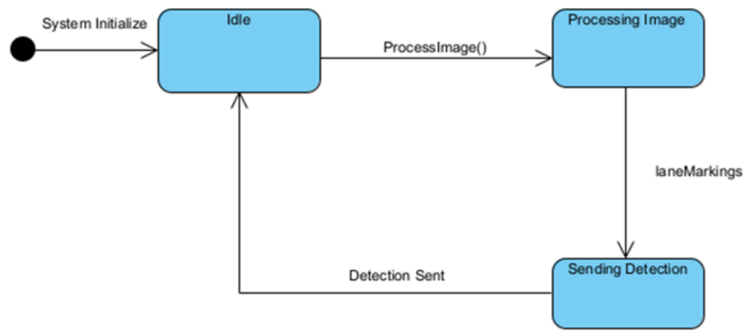


Figure 6: Image Processing Subsystem state diagram

In Figure 7, the supervisory control subsystem turns on and awaits a signal to turn off or enter an idle state. The system then enters the appropriate state based on the call given.

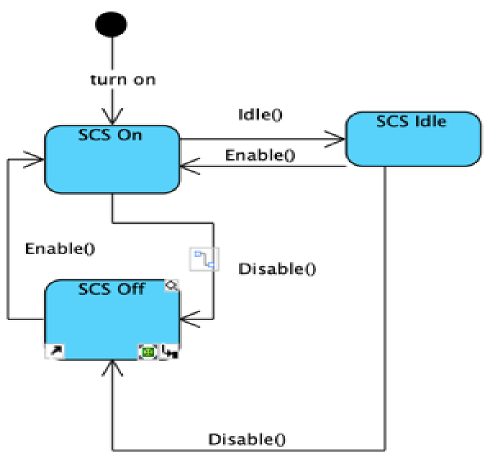


Figure 7: Supervisory Control Subsystem state diagram

In Figure 8, the path prediction subsystem initializes in the off state. Upon the vehicle reaching a speed of 35 mph or greater, the subsystem enters Idle state. The path prediction subsystem will then predict and subsequently correct the path of the vehicle with breaking and steering adjustments.

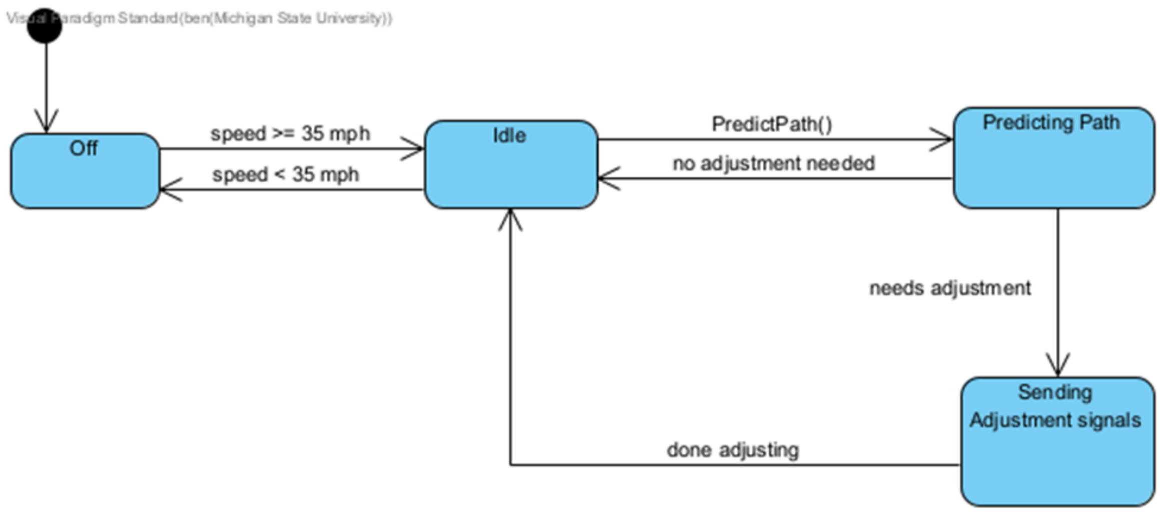


Figure 8: Path Prediction Subsystem state diagram

As illustrated in Figure 9, the camera subsystem is initialized to be off. At the call of the SetActivity function the camera will begin taking image data until the SetActivity function turns the camera off.

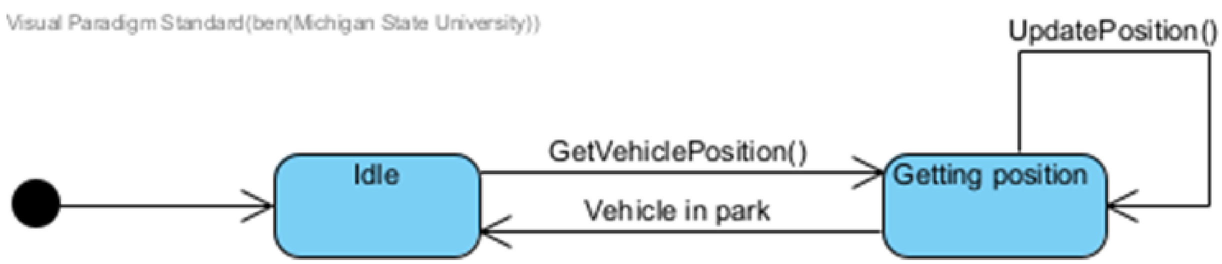


Figure 9: Camera Subsystem state diagram

Figure 10 depicts the lane-centering subsystem initialized in an idle state. Once it is set active by the supervisory subsystem, it collects the lane positioning data and determines if markers are present. If they are, the status of the lane is sent off to be used for lane adjustments, otherwise, the subsystem is returned to idle.

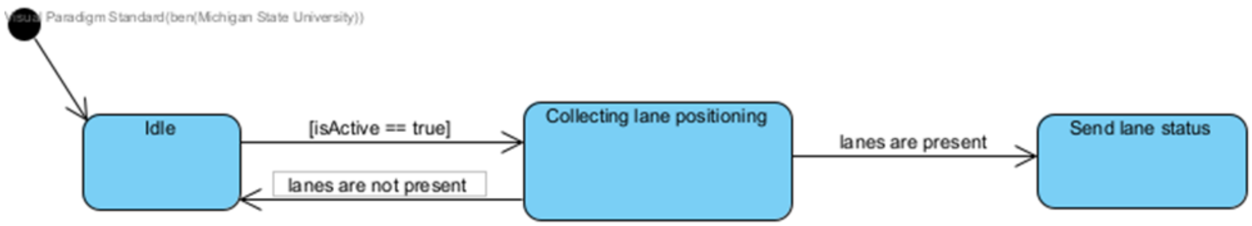


Figure 10: Lane Centering subsystem state diagram

Figure 11 shows the subsystem for lane keeping initialized in the off state. At the speed threshold lane keeping enters the idle state, detects lane warnings, and measures torque in order to ensure that the vehicle is below the torque threshold. If these measurements are within bounds, the vehicle steering and speed (if applicable) are corrected and the subsystem returns to idle. If the vehicle speed drops below the threshold, the subsystem is turned off.

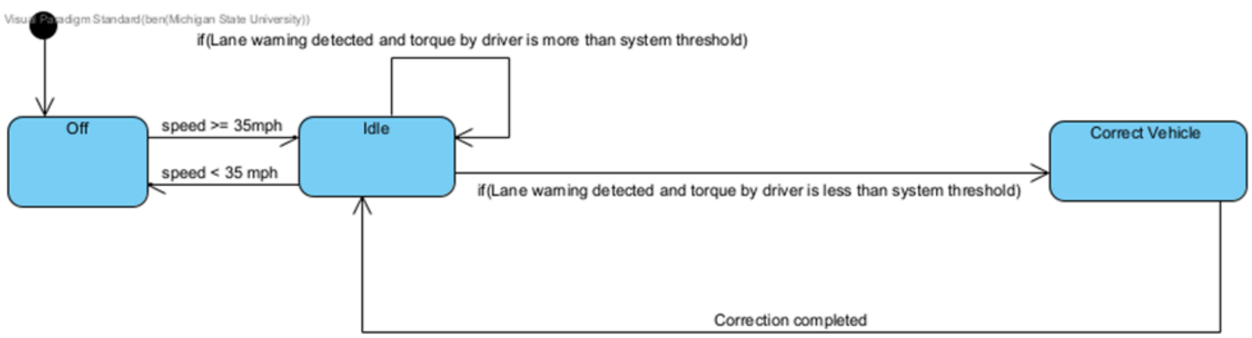


Figure 11: Lane Keeping Subsystem state diagram

Figure 12 begins in the off state. Idle state is entered when the speed threshold is met. If sensor info is available path information is collected from other subsystems to determine if a warning is necessary to be sent. If a warning is necessary, a warning message will be sent and the system will return to the information collection state until a call to turn the warning system off is completed.

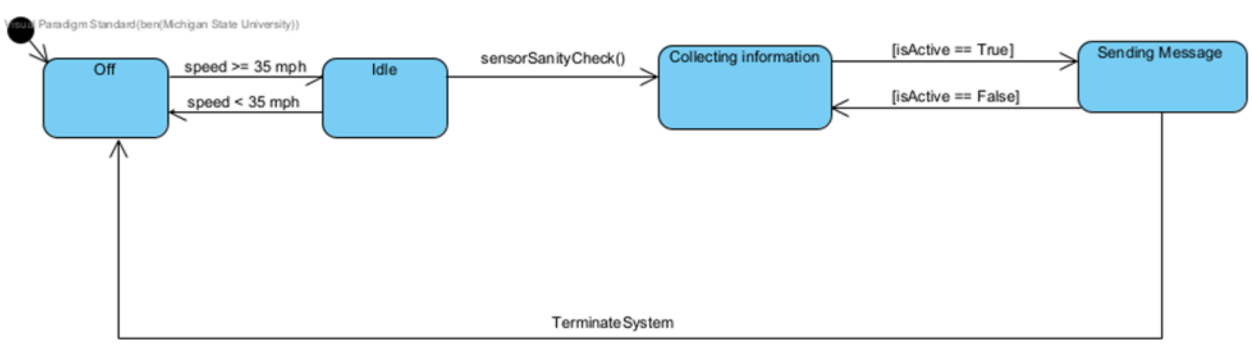


Figure 12: Lane Departure Warning Subsystem state diagram

5 Prototype

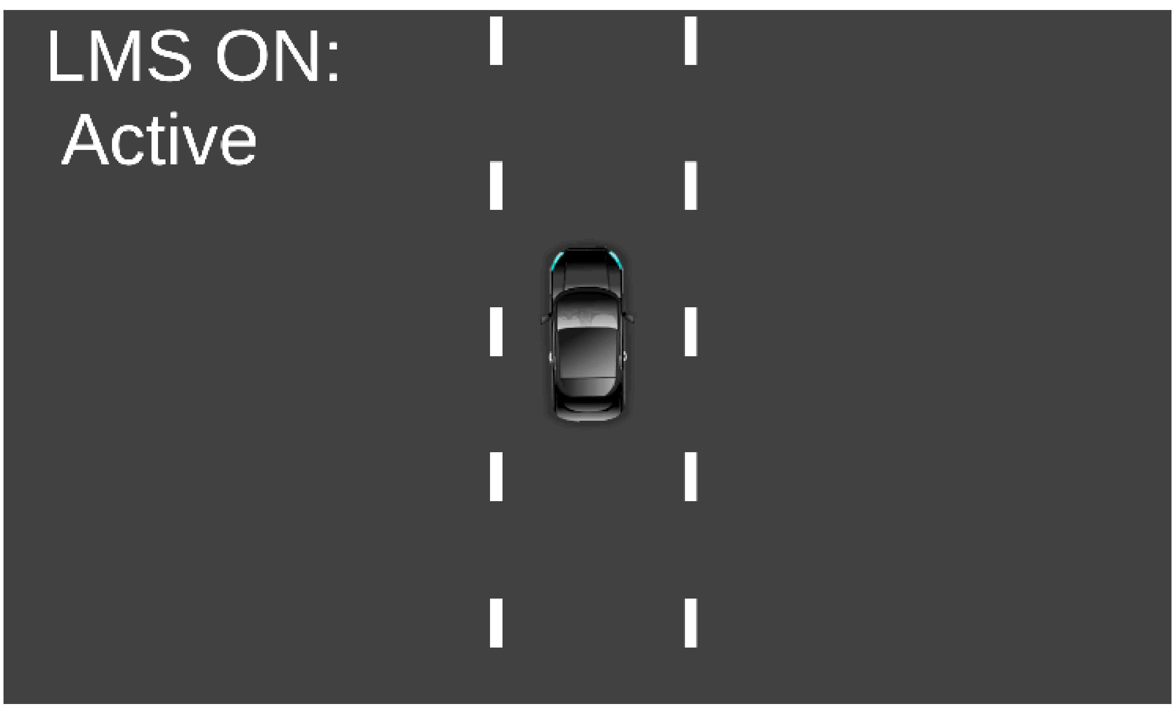
The prototype associated with this project will demonstrate how the LMS should function under different scenarios. There will be four scenarios. The first will be the ideal scenario in which the vehicle is going down a straight road at a constant speed. The second will be of a road with non-zero curvature to demonstrate the functionality of the LMS under more stress. Third will be the failure scenario where lane markings become undetectable or unclear. Lastly, there will be another set of lanes showing the LMS entering back into an active state once lanes are detected again.

5.1 How to Run Prototype

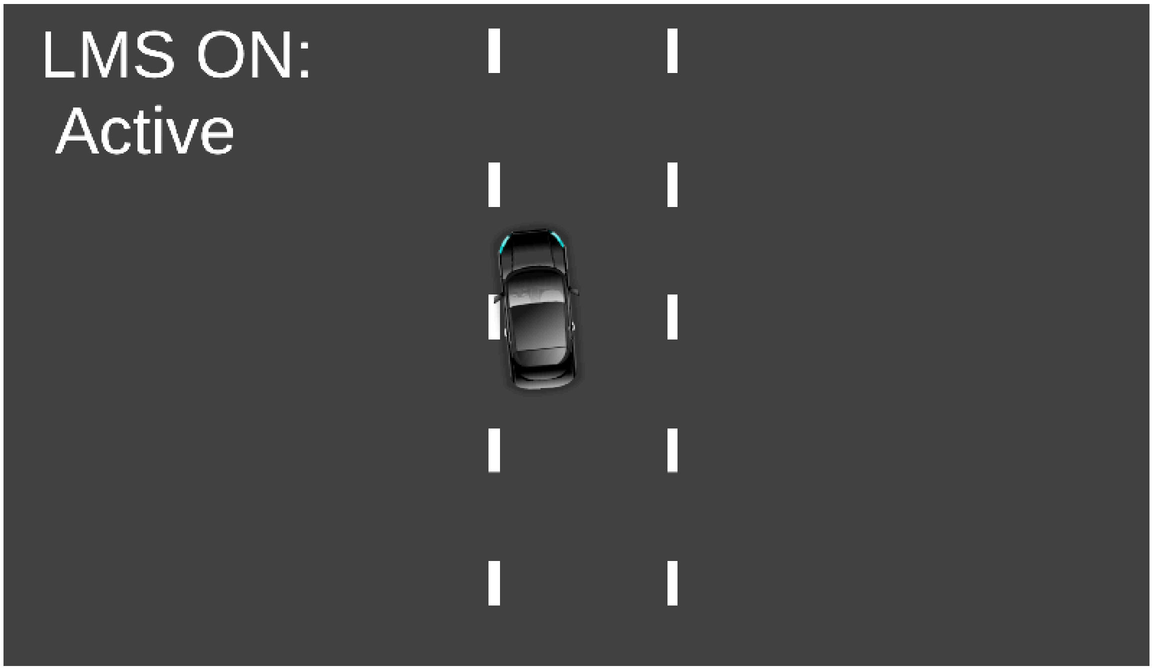
To run the prototype, follow the link from the project website to be taken to a page with an embedded Unity web player. The controls for the prototype are the arrow keys using up and down for acceleration and left and right for steering. The LMS is turned on and off using the spacebar. To ensure the prototype runs properly, use any modern up-to-date browser. Below is the link to the prototype: https://cse.msu.edu/~baldw266/Prototype/proto.html

5.2 Sample Scenarios

The ideal scenario:

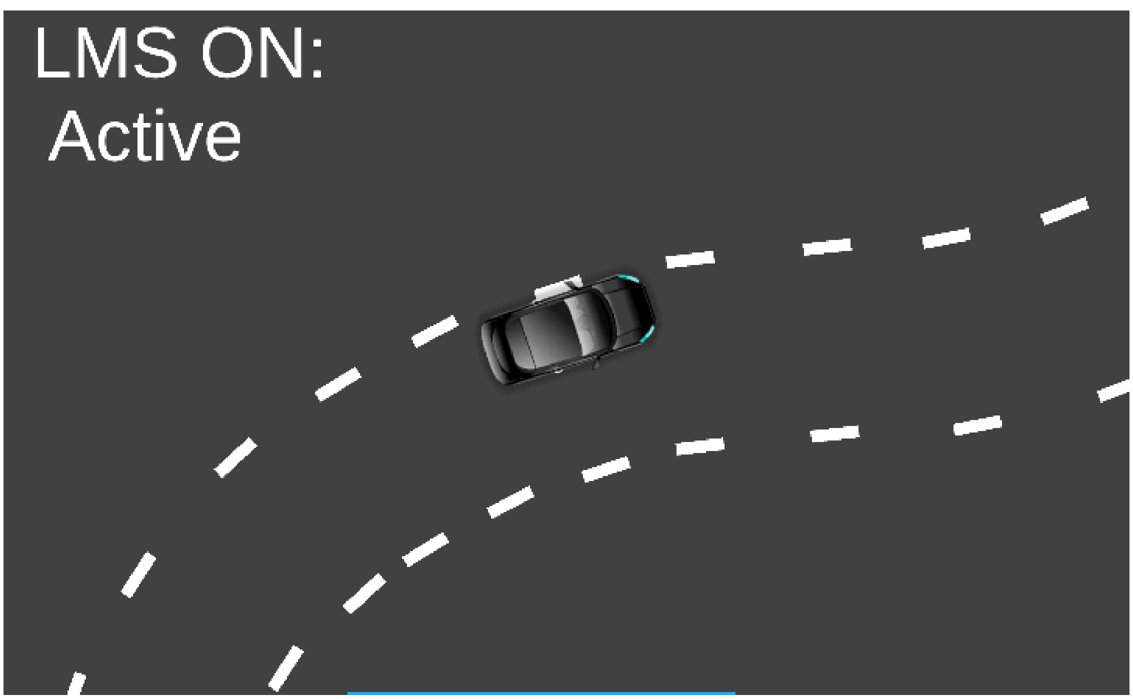


In this scenario, the LMS is on, the vehicle is moving forward, and the lanes are being detected, so the LMS is active. The vehicle is not making any unannounced lane violations, so the LMS is not applying any corrective torque to the wheel.



At this point, the LMS is enabled, lanes are detected, and an unexpected lane departure is occurring. Torque is then applied to the wheel to steer the vehicle back into the lane.

Road curvature scenario:

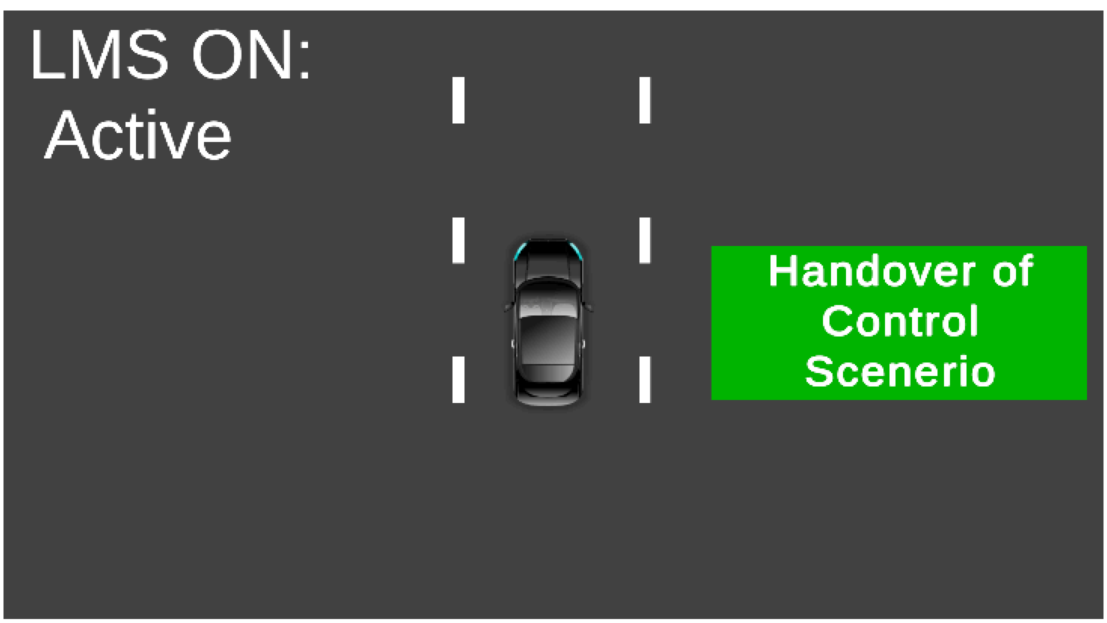


In this scenario, the road is not uniform and is curving. This scenario demonstrates the LMS continuing to work in the same fashion as the ideal scenario but under more stress.

Failure scenario:

In this scenario, the LMS is on, but the road markings are undetectable or unclear. Once the lane markings are no longer being detected, the LMS enters a passive state until lane markings are once again detected.

Change of Control Scenario:



In this scenario, the vehicle is coming from the previous failure state. In that state, the LMS was on but in a passive state. Now that lines are being detected once again, however, the LMS enters an active state.

6 References

Khattak, Z. H., Smith, B. L., Fontaine, M. D., Ma, J., & Khattak, A. J. (2022). Active lane management and control using connected and automated vehicles in a mixed traffic environment. In Transportation Research Part C: Emerging Technologies (Vol. 139, p. 103648). Elsevier BV. https://doi.org/10.1016/j.trc.2022.103648

Maserati. “Lane Keeping Assist System.” *Maserati.com*, 2021, www.maserati.com/us/en/ownership/maserati-manuals/safety/lane-keeping-assist. Accessed 30 Sept. 2024.

National Safety Council. “Lane Keeping Assist: MyCarDoesWhat.org.” *My Car Does What*, mycardoeswhat.org/safety-features/lane-keeping-assist/.

Ford. “Lane-Keeping System | Ford Co-Pilot 360™ Technology.” *Ford Motor Company*, www.ford.com/technology/driver-assist-technology/lane-keeping-system/.

Mopar, “Active Lane Management | How To | 2023 Jeep Vehicles,” *YouTube*, Jan. 26, 2023. https://www.youtube.com/watch?v=gBuJ6gD-nuA (accessed Sep. 30, 2024).

“Lane Keep Assist Safety Feature | Vehicle Support,” *GMC*. https://www.gmc.com/support/vehicle/driving-safety/driver-assistance/lane-keep-assist-departure-warning

“Lane Management System Customer: Ayush Agrawal, formerly with AI/ML GMIT (now with Amazon).” Accessed: Sep. 30, 2024. [Online]. Available: http://www.cse.msu.edu/~cse435/Projects/F2024/ProjectDescriptions/2024-LMS-GM-Amazon-Agrawal.pdf

K. Park, S. H. Han, H. Lee, and J. Kwahk, “Shared steering control: How strong and how prompt should the intervention be for a better driving experience?,” *International Journal of Industrial Ergonomics*, vol. 86, p. 103213, Nov. 2021, doi: https://doi.org/10.1016/j.ergon.2021.103213.

Leitgeb, Erich, et al. *Optical Wireless Communications and Optical Sensing and Detection Technologies for Increasing the Reliability and Safety in Autonomous Driving Scenarios*. 1 July 2018, https://doi.org/10.1109/icton.2018.8473718. Accessed 25 Oct. 2024.

Kim, Yongdae. “Hacking Sensors.” *Www.usenix.org*, 2017, www.usenix.org/conference/enigma2017/conference-program/presentation/kim.

Project Website Link:

https://cse.msu.edu/~baldw266/

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.