Software Requirements Specification (SRS)

Pedestrian Backup Assist System (PBAS)

Authors: Ahmad Arman Newaz, Daniel Nguyen, Jacob Rutkowski, Javen Zamojcin, Sara Rezaeimanesh, Simon Situ
Instructor: Dr. Betty H.C. Cheng

1. Introduction

1.1. Purpose

This document defines the software requirements for the Pedestrian Backup Assist System. The intended audience for this document are the developers and stakeholders that are involved in this project.

1.2. Scope

The project is called Pedestrian Backup Assist System (PBAS). The objective of this software is to be an assistant for the driver when the vehicle is in reverse and the driver is aiming to back up. It will cover the driver's blind spots near the rear bumper that the driver cannot see from their position in the driver seat, referred to as the rear blindspot. Various sensors such as LiDAR and proximity sensors will be used to detect obstacles in the rear blindspot, and the system will alert the driver of nearby obstacles with the use of visual and audible alerts. To prevent injury, automatic brakes will be applied in case the driver fails to respond to an obstacle and brake the vehicle. A collision detector will detect any collision and apply brakes to immobilize the vehicle from moving in the event an obstacle collides with the vehicle.

This software project will provide communication to the respective backup sensors, logic to handle object detection with visual and auditory alerts for the driver, functionality to manually disable said alerts, an automated braking system in the event of object detection and imminent collisions, logic to check the health of sensors and other hardware components, a visual data stream for the backup camera on the display interface, a system to circumscribe obstacles in the built-in display and calculate their respective distances based on sensor readings, and logic to immobilize the vehicle in the event of a collision to prevent further damages.

1.3. Definitions, acronyms, and abbreviations

1.3.1. Table of Definitions:

Backover	It is an accident that occurs when a vehicle reverses and collides with someone or something, due to lack of visibility of the rear.
LiDAR	Acronym for “Light detection and ranging”. Projects lasers into the environment and collects returning rays to map the environment [2].
Ultrasonic	Ultrasonic sensors send sound waves into the environment and collect the reflecting waves to map the environment [3].
Collision Detector	Also known as a force sensor, it is a sensor that outputs an electrical signal if a force is applied to the sensor [4]. This can be used to detect if a collision has occurred.
Camera	Also known as image or video sensors, cameras produce video footage of the environment.
Object Detection	Using sensor data of the environment, object detection recognizes objects of interest that the vehicle may collide with.

1.4. Organization

The rest of this document contains different sections detailing the system. Overall Description (Section 2) contains the description of the system. Specific Requirements (Section 3) details requirements that the system must satisfy. Modeling Requirements (Section 4) shows different models and diagrams of the system. Finally, the Prototype section (Section 5) shows a virtual prototype or demonstration of the system. Section 2 will contain high-level services that the system will provide, as well as the target customer base, constraints that the system must comply with, and other properties of the system. Section 3 is made up of multiple enumerated lists of functional requirements, non-functional requirements, global invariants, and sources of uncertainties with their respective mitigation strategies. Section 4 consists of different diagrams that model the system. The diagrams used are the use case diagram, domain model, sequence diagrams, and the state diagram. Section 5 describes the prototype of the system and a link to the prototype where stakeholders can test the system specified by this document. It will also describe different scenarios of using the system that the prototype can demonstrate.

2. Overall Description

2.1. Product Perspective

The product is expected to be used as one part of a vehicle's software package. The hardware that we are going to need to add to a vehicle for this system to work all involve the rear of the vehicle without any consideration for other functionality such as adaptive lane control.

2.2. Product Functions

Major functions of the software system include:

a) Informing the driver of what obstacles are behind their vehicle while they are intending to drive the vehicle in reverse.
b) Detect obstacles in the rearview of the vehicle and inform the driver of these obstacles.
c) Visually and Audibly alerting the driver when an obstacle is detected that needs the driver's attention.
d) Automatic braking of the vehicle in the case of sudden collision or driver fails to apply the brakes in time to avoid an obstacle.

2.3. User Characteristics -

2.3.1. The driver must not be under the influence of alcohol or drugs.

2.3.2. The driver must be free from impairment due to any substances.

2.4. Constraints

a) *Regulatory policies:*

- Must comply with FMVSS 111 (Federal Motor Vehicle Safety Standard) in the U.S., which mandates rear visibility systems in vehicles.
- Adherence to ISO 26262 (Functional Safety for Road Vehicles) to ensure that the system is robust, failsafe, and rigorously verified, thereby minimizing the risk of critical failures and enhancing overall automotive safety integrity.

b) *Hardware limitations:*

- Must process real-time sensor data with minimal latency (~100 ms response time).
- Utilize the vehicle's existing communication network for transferring data between systems.
- Limited processing power and memory in embedded automotive systems.

c) *Interfaces to other applications:*

- Must communicate with braking, display, and sound control systems to take preventive action.
- Compatibility with the existing dashboard display systems.

d) *Parallel operation:*

- The system must operate alongside other vehicle safety features (e.g., blind-spot monitoring, adaptive cruise control) without interference.
- The system must be able to process data from all sensors in parallel.

e) *Audit functions*:

- Diagnostic self-checks on system startup to detect sensor malfunctions and report failures.

i) *Reliability requirements*:

- Retaining a relatively small false positive rate to avoid desensitizing the driver to alerts. (<5%).
- Retaining a very small false negative rate to avoid putting lives in danger. (<0.01%)
- Collision detection sensors to prevent serious injuries or damage to the car in case of false negatives.

j) Criticality of the application:

- Classified as safety-critical, requiring fail-operational mechanisms.

k) Safety-Critical Properties

- The system should provide real-time feedback to the user interface.
- The system should provide real-time reaction (alarm/brake) to the obstacles behind the car.
- The sensors must remain fully operational and continuously transmit data whenever the vehicle is engaged in reverse gear.
- The system must detect obstacles within a 3m range to prevent collisions.
- Emergency braking must be triggered when an obstacle is within 1m of the vehicle's rear or collision is detected.

l) Other Important Properties for Proper Functioning

- The system should remain functional under adverse environmental conditions such as rain or fog.
- Periodic self-checks must be performed to ensure system integrity.

2.5. Assumptions and Dependencies

One of the assumptions made about this system is that it will be implemented onto a commercial car such as a sedan, suv, or minivan. Specialty vehicles such as construction vehicles were not considered in the development of the system.

We assumed that the typical use case for a driver is using the system to assist with reversing at slow speeds, below 10 mph. The distances for the sensors to detect obstacles assume this speed to give sufficient distance for the vehicle or driver to safely react to unexpected variables during a reverse.

The operating environment for the vehicle is assumed to include all weather conditions, including adverse scenarios such as heavy snow or rain. These conditions may impair camera visibility and degrade sensor performance. Consequently, the system must be capable of functioning reliably even when some or all sensors are compromised. Additionally, the environment is assumed to have a high likelihood of mobile obstacles, such as pedestrians and other vehicles. These assumptions are intended to encompass the majority of real-world situations in which a vehicle equipped with this system would operate.

2.6. Deferred Requirements

The legal requirements that the Pedestrian Backup Assist System may need to abide by were not considered in this iteration of the requirements. The requirement engineers creating this document are primarily software developers. Besides the size of the rear blindspot, any other legal documents related to this system have not been rigorously considered.

Another requirement that could use development is the specification and installation of the hardware. There isn't a specific requirement related to how many sensors of each type should be present except that there should be at least one of each type of sensor. In addition, the quality of each sensor was determined arbitrarily. The customer might have differing requirements then what was decided in this document that will need to be addressed.

3. Specific Requirements

Functional Requirements

1. The vehicle has a built-in display unit at the front of the vehicle for the driver.
2. The vehicle will have these four sensors mounted on the rear of the car called the sensor suite. These are all used in conjunction for obstacle detection. Their specification is listed here.
    - 2.1. Rear-Facing Camera
        - 2.1.1. Should be able to see the rear blind spot. The rear blindspot is defined as 10ft wide by 20ft long area behind the vehicle as specified in FMVSS No. 111 [1]
        - 2.1.2. Rear camera should be able to see an object that is 30.2 inches tall that is 1ft behind the vehicle.
        - 2.1.3. Minimum resolution of 4k and real-time latency (<= 100ms response time)
    - 2.2. LiDAR Sensor
        - 2.2.1. Used to measure the distance from obstacles to vehicles.
        - 2.2.2. The LiDAR sensor should be able to detect objects with a 5m radius surrounding the vehicle on all-sides centered at the LiDAR sensor.
    - 2.3. Ultrasonic Sensor (proximity)
        - 2.3.1. Should cover what the rear-facing camera and lidar sensor do not see of the rear/, such as underneath the vehicle’s rear bumper. This can be tested by mapping out blind-spots in other sensors and making sure they are covered by this sensor.
    - 2.4. Collision Detector (force sensor)
        - 2.4.1. Used to determine if a collision has occurred between the car and an obstacle. If force is applied to the collision detector, then it will trigger the response for a collision. This response can be tested by manually applying force to the sensor and seeing that the response is triggered.
3. Obstacle Detection and Visual Highlighting
    - 3.1. Any obstacles in the rearview visual should be detected with a confidence of 80% or higher.
    - 3.2. When an object is detected, it should have a bounding box surrounding it on the rearview visual so that the driver can identify it as an obstacle.
        - 3.2.1. The color of this bounding box is dependent on how far the obstacle is from the rear of the vehicle.

3.2.1.1. The red bounding box means the obstacle is less than 1 meter (3.281 ft) away.

3.2.1.2. The yellow bounding box means the obstacle is between 1 and 2 meters (3.281 ft - 6.562 ft) away.

3.2.1.3. The green bounding box means the obstacle is more than 2 meters (6.562 ft) away.

3.2.1.4. To test this, we can place different objects at different distances and see if the system can correctly identify the objects.

3.3. Obstacles that could potentially crash into the vehicle that are not within the view of the rearview camera and are moving into the view of the camera within 2 seconds will trigger an audio alert.

4. Driver Alert Prioritization:

4.1. Sound Alert: A beeping sound whose frequency changes depending on how far an object is from the vehicle.

4.1.1. The sound alert should beep once every two seconds if the closest obstacle is in the green border range, once every second if the closest obstacle is in the yellow border range, and twice every second if the closest obstacle is in the red border range.

4.1.2. The driver is allowed to temporarily mute the driver alerts in case of false positives. The driver should be able to mute the sound alert by pressing an on-screen button on the built-in display that says “Mute Sound”.

4.2. Visual Alert: A red border around the camera feed will indicate the vehicle is close to an object.

4.3. To test this, we can put one object at different distances and see the different beep frequencies and the turning on/off of the red border at a very close distance.

5. In the case that the driver fails to stop the vehicle, the system should engage automatic braking to avoid hitting a detected obstacle.

5.1. The braking system should take over once an obstacle is detected within 0.5m of the rear of the vehicle.

5.2. In the case that the vehicle does accidentally hit an obstacle, the automated braking should activate to immobilize the vehicle to prevent rollover.

6. System should let the driver be able to gauge how far obstacles are from the rear of the vehicle by providing guidelines (3m line, 5m line, 10m line).

Non-Functional Requirements

1. False positives (object detection) should be mitigated (<5%) for the convenience of the driver. One way to test the rate of false positives is by giving the driver a button on the built-in display screen where they can report that an object is a false-positive. From this we'll be able to see what percentage of detected obstacles are reported as false positives by dividing the number of reports by the total number of detected obstacles.
2. The camera sensor must be reliable under a large range of weather conditions (slush, snow, rain, dust) and temperatures (-20F to 120F).
3. The system must be able to detect faulty sensors, disable emergency braking in case ⅔ of the sensor malfunctions, and alert the driver system isn't working properly.
   3.1. 2/3rds of the sensors going down is the floor for disabling emergency braking. The emergency brakes will continue to be disabled even if more or all sensors go down, and will only be re-enabled when more than 1/3rd of the sensors are operational.
4. False negatives must be nearly nonexistent to ensure that no obstacles and more importantly pedestrians go undetected.
5. The system cameras and sensors must be installed at least 12 inches above the ground to avoid false positives.
6. Emergency auto-brakes have to be applied within 0.1 seconds of obstacle detection.

Global Invariants

1. The system should only be active when the vehicle is shifted into reverse.
2. The system must process sensor and camera data with ≤ 100 milliseconds latency (hardware constraint) to ensure real-time detection.
   2.1. An entity that appears in front of the rearview camera should appear on the display in real time (≤ 100 ms).
3. The system must always allow the driver to brake.
4. The system must identify any obstacles that are greater than a 0.3m by 0.3m by 0.3m cube behind the vehicle.

Sources of Uncertainty

1. Varying environmental conditions
   1.1. Varying environmental conditions make it impossible to test for every possible environment. The environment includes location, objects or people within the surroundings, and weather conditions.

    1.2. The sensors are responsible for monitoring the environment, and the Backover Prevention System is responsible for making the decision to alert the driver and engage emergency brakes based on the environment.
    1.3. To mitigate the problem of variability in the environment, the system uses multiple types of sensor technologies (LiDAR, Ultrasonic, and Camera) to increase the confidence in the measurements taken from the environment.
2. Unpredictability of other entities' movements
    2.1. Similar to the first uncertainty, the vehicle has to account for a variety of situations when reversing. Obstacles like pedestrians and other vehicles are unpredictable in their movements.
    2.2. The Sensor Suite with all of its sensors is responsible for tracking every obstacle which includes their movement.
    2.3. The multitude of sensors should help mitigate this by providing a plentitude of information for the system to make predictions. In addition, the system has access to the vehicle's braking system. In this case it detects an imminent collision the system can automatically apply the brakes without the need for driver reaction.
3. Degradation of sensors
    3.1. The natural degradation of sensors over time can potentially inhibit the overall functionality of the entire backup system. Without the sensors (camera, proximity, LiDAR), there would no longer be means for detecting objects.
    3.2. Check for failed hardware heartbeats such as no response from the hardware in the past 5 minutes; compare sensor readings to baseline readings and flag any sensors with less than 50% of their normal baseline readings; recommend user maintenance or inspection after a specified duration provided by the manufacturer such as every 6 months or every 3,000 miles driven. These operations are delegated to the Backover Prevention System and Sensor Suite to process incoming data and determine failures.
    3.3. Upon detecting failed sensors, alert the driver through the display system that immediate maintenance is required.

Cybersecurity

1. Threat Vectors:
    1.1. Wireless interfaces on the vehicle used for diagnostics or OTA firmware updates expose a potential attack vector. Unauthorized access via poorly secured wireless channels can compromise the system.

1.2. Attackers might attempt to manipulate or spoof sensor data (from cameras, LiDAR, or ultrasonic sensors) to create false environmental readings. This could induce the system to behave unexpectedly, such as triggering unnecessary alarms or failing to engage emergency braking.

1.3. Components or firmware sourced from external vendors may carry hidden vulnerabilities if rigorous security validation is not conducted during the procurement and installation stages.

2. Threat Actors:

2.1. **Hackers:** These actors may seek to exploit known vulnerabilities in vehicle networks or software to gain unauthorized access for malicious purposes, such as causing chaos or gaining control of safety systems.

2.2. **Disgruntled Employees / Service Technicians:** Individuals with legitimate access or detailed knowledge of the system could misuse their privileges to compromise safety-critical operations.

2.3. **Industrial Espionage:** Competitors looking to gain an edge might attempt to steal intellectual property or sensitive system information through cyber intrusions.

3. Cybersecurity Vulnerabilities and Mitigation Strategies:

3.1. Unauthorized Access and Network Attacks

3.1.1. **Vulnerability:** Weak encryption or lack of proper authentication in internal communications can allow attackers to intercept or alter control signals within the vehicle network.

3.1.2. **Prevention:** Implement robust encryption (e.g., TLS) and secure authentication for all internal communications

3.1.3. **Detection:** Employ intrusion detection systems (IDS) to monitor network traffic for anomalous patterns or unauthorized access attempts.

3.1.4. **Mitigation:** Introduce fail-safe modes that disengage non-essential functions if anomalies are detected.

3.2. Software and Firmware Vulnerabilities:

3.2.1. **Vulnerability:** Programming errors, buffer overflows, or inadequate access controls in the PBAS software can be exploited by attackers to gain control or disable safety functions.

3.2.2. **Prevention:** Use secure coding practices, conduct thorough code audits, and implement static/dynamic analysis during development.
3.2.3. **Detection:** Implement continuous integrity monitoring and anomaly detection algorithms to validate software behavior.
3.2.4. **Mitigation:** Establish rapid patch management procedures and a robust response plan to handle vulnerabilities as they are discovered.

3.3. Sensor Spoofing and Data Manipulation:

3.3.1. **Vulnerability:** Physical or electromagnetic attacks might alter sensor outputs, leading to false positive or false negative detections in obstacle recognition.
3.3.2. **Prevention:** Integrate sensor fusion strategies that cross-check data from multiple sources to verify consistency.
3.3.3. **Detection:** Incorporate algorithms that monitor for sudden deviations or inconsistencies in sensor data beyond normal environmental variations.
3.3.4. **Mitigation:** Develop redundancy in sensor inputs so that the system can maintain functionality even if one sensor is compromised.

4. Modeling Requirements

Use Case Diagram

The use cases of this system are shown in Figure 1. Considering that this system is integrating with the vehicle's existing systems, the external actors of the system are the Driver, Sound System, Display, and Braking System. The primary use cases from the Driver's perspective are to be alerted of obstacles, view the rear blind spots, and be able to mute the alarm in the case of a false positive.

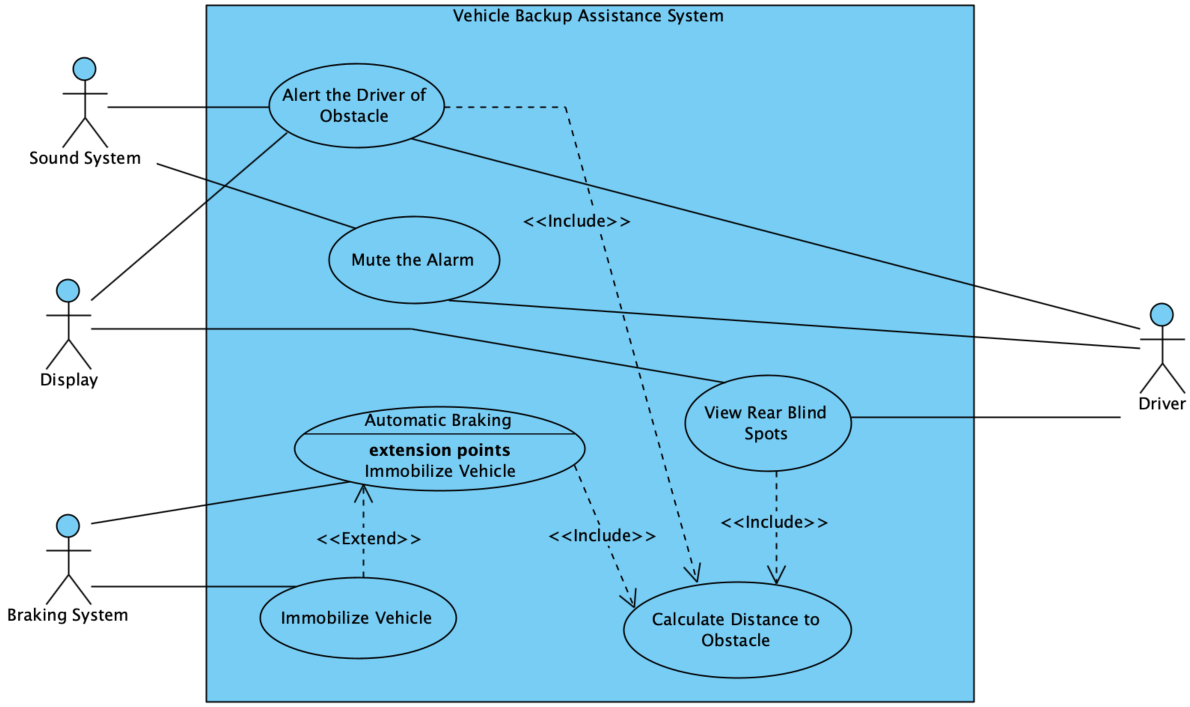

Figure 1: The use case diagram for the system. Shows the high-level services that the system provides.

Use Case Descriptions

F: Functional
N: Non-functional
G: Global Invariant

Use Case:	Automatic Braking
Actors:	Braking system, Driver
Description:	The brakes are automatically applied in case of emergency like the driver not applying brakes in time before a sudden impact
Type:	Primary and essential
Includes:	Calculate distance to Obstacle
Extends:	Manual Braking
Cross-refs:	**F:** V; **N:** VII
Use cases:	Calculate the distance to Obstacle: The system has to know how far the vehicle is from an object to decide whether to apply the brakes.

Use Case:	Mute the Alarm
Actors:	Sound System, Driver
Description:	The driver manually mutes the alarm system in response to a false-positive object detection.
Type:	Secondary
Includes:	None
Extends:	None
Cross-refs:	**F:** VI;
Use cases:	Alert the driver of Obstacle, as there must be an alarm to mute first.

Use Case:	Alert the driver of Obstacle
Actors:	Driver, Sound System
Description:	Utilize the sensor suite and rear camera to detect obstacles behind the vehicle. Once an obstacle is detected, alert the driver through a beeping sound that is distinct from the other alarm sounds of the vehicle if an obstacle is detected.
Type:	Primary
Includes:	Calculate Distance to Obstacle
Extends:	-
Cross-refs:	**F:** II, III, IV; **N:** I, II, III, IV, V
Use cases:	*Calculate Distance to Obstacle* Knowing the distance to the obstacle allows the system to alert the driver when the obstacle is a certain distance from the vehicle.

Use Case:	Immobilize Vehicle
Actors:	Braking System
Description:	When the system has detected a collision with an object, the braking system will engage to prevent further damage or rollover.
Type:	Primary
Includes:	None

Extends:	Automatic Braking
Cross-refs:	**F:** II, Vb;
Use cases:	

Use Case:	Calculate distance to obstacle
Actors:	None
Description:	This calculates the distance of the object from the vehicle
Type:	Primary and essential
Includes:	None
Extends:	None
Cross-refs:	**F:** III, V;
Use cases:	

Use Case:	View Rear Blindspot
Actors:	Driver, Display
Description:	Assist the driver in reversing by allowing them to view the rear blindspot through the camera. Should display visual indicators of obstacles detected behind the vehicle to the driver.
Type:	Secondary
Includes:	Calculate Distance To Obstacle
Extends:	None
Cross-refs:	**F:** I, II, III, IVb, VII; **N:** I, II, V
Use cases:	None

Domain Model

An overview of the system components is shown in Figure 2, along with their relationships and operations. The Backover Prevention System is the primary component of the system and acts as the central processor for the system. It is responsible for detecting the obstacles using sensor data from the Sensor Suite, displaying sensor data and highlighting obstacles on the Display, triggering the Sound System to alert the driver of close obstacles, and applying Brakes if necessary. The Sensor Suite is made up of many sensors and of different types. It collates the data provided by the different onboard sensors. A Sensor can be any one of Ultrasonic, LiDAR, Collision Detector, or Camera. The driver can use the system through viewing rear blind spots through the display, and be alerted when the vehicle is approaching an obstacle that may be collided with.

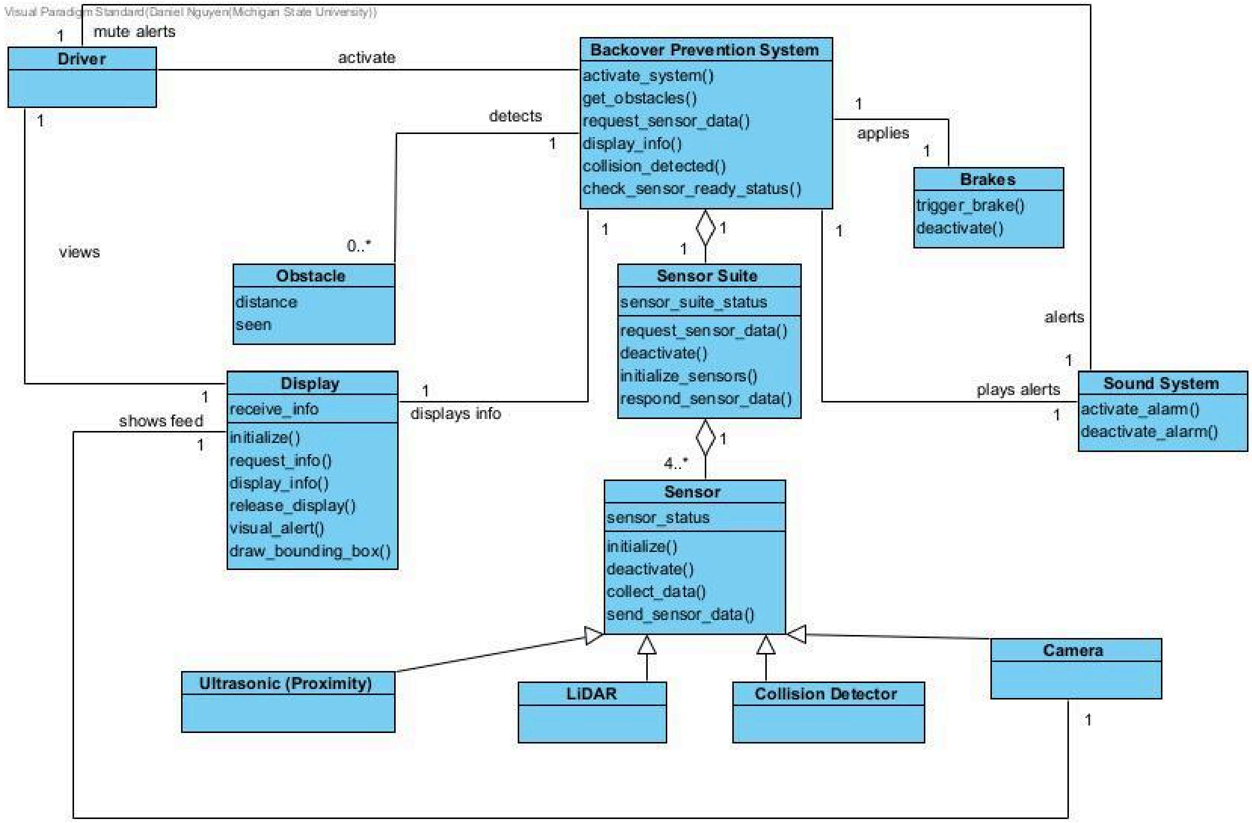

Figure 2: The domain model of the system. Shows the components of the system and their relationships.

Data Dictionary

Class	Obstacle
Name	**Description:** An object or person obstructing the vehicle backup path.
	Export Controls: Public
Relationships	**Associations:** *Backover Prevention System*
	Aggregations/Compositions: *None*
	Generalizations: *None*

List of Attributes	Distance, Seen
List of Operations	None

Class	Driver
Name	**Description:** The human operator of the vehicle.
	Export Controls: Public
Relationships	**Associations:** *Sound System, Display, Backover Prevention System*
	Aggregations/Compositions: *None*
	Generalizations: *None*
List of Attributes	None
List of Operations	None

Class	Sensor Suite
Name	**Description:** All of the sensors that the system is using to determine the presence of obstacles behind the vehicle. Calculates the distance to obstacles and gives information to the system.
	Export Controls: Private
Relationships	**Associations:**
	Aggregations/Compositions: *None*
	Generalizations: *Backup Prevention System, Sensor*
List of Attributes	sensor_suite_status
List of Operations	request_sensor_data(), deactivate(), initalize_sensors(), respond_sensor_data()

Class	Sensor
Name	**Description:** Obtains data from the vehicle's surroundings
	Export Controls: Private

Relationships	**Associations:** *None*	
	Aggregations/Compositions: *Sensor Suite*	
	Generalizations: *None*	
List of Attributes	sensor_status	
List of Operations	initialize(), deactivate(), collect_data(), send_sensor_data()	

Class	Ultrasonic (Proximity)
Name	**Description:** A redundant object detection sensor for detecting obstacles that are extremely close to the vehicle. Uses ultrasonic technology to detect obstacles near the rear bumper.
	Export Controls: Private
Relationships	**Associations:** *None*
	Aggregations/Compositions: *None*
	Generalizations: *Sensor*
List of Attributes	None
List of Operations	None

Class	LiDAR
Name	**Description:** A sensor that uses LiDAR technology. Used to report the distance of obstacles behind the vehicle. Also used to detect obstacles outside the field of view of the rear camera.
	Export Controls: Private
Relationships	**Associations:** *None*
	Aggregations/Compositions: *None*
	Generalizations: *Sensor*
List of Attributes	None
List of Operations	None

Class	Collision Detector	
Name	**Description:** Uses collision sensors to detect whether the car has hit an obstacle.	
	Export Controls: Private	
Relationships	**Associations:** *None*	
	Aggregations/Compositions: *None*	
	Generalizations: *Sensor*	
List of Attributes	None	
List of Operations	None	

Class	Camera
Name	**Description:** The main sensor used by the system for obstacle detection. Also used by the driver to view the rear blindspot of the vehicle.
	Export Controls: Private
Relationships	**Associations:** *None*
	Aggregations/Compositions: *None*
	Generalizations: *Sensor*
List of Attributes	None
List of Operations	None

Class	Sound System
Name	**Description:** Creates audio alerts to alert the driver.
	Export Controls: Public
Relationships	**Associations:** *Driver, Backover Prevention System*
	Aggregations/Compositions: *None*
	Generalizations: *None*

List of Attributes	None
List of Operations	activate_alarm(), deactivate_alarm()

Class	Backover Prevention System
Name	**Description:** The main system responsible for mitigating backup object collisions. Controls what gets displayed to the driver, and what alerts are played.
	Export Controls: Public
Relationships	**Associations:** *Sound System, Display, Driver, Brakes, Obstacle*
	Aggregations/Compositions: *Sensor Suite*
	Generalizations: *None*
List of Attributes	None
List of Operations	activate_system(), get_obstacles(), request_sensor_data(), display_info(), collision_detected(), check_sensor_ready_status()

Class	Display
Name	**Description:** The visual display interface for the backup camera datastream. Displays visual alerts and obstacles to the driver as well as any system errors.
	Export Controls: Public
Relationships	**Associations:** *Camera*
	Aggregations/Compositions: *Vehicle*
	Generalizations: *None*
List of Attributes	recieve_info
List of Operations	initialize(), request_info(), display_info(), release_display(), visual_alert(), draw_bounding_box()

Class	Brakes
Name	**Description:** The vehicle's braking system.

	Export Controls: Yes
Relationships	**Associations:** *Backover Prevention System*
	Aggregations/Compositions: *None*
	Generalizations: *None*
List of Attributes	None
List of Operations	trigger_brake(), deactivate()

Representative Scenarios

The following section is a set of 5 scenarios that illustrate the general use case of the Pedestrian Backup Assistance System. Some of the edge cases such as vehicle collision and sensors detecting that they are faulty are present as well.

Activation Of System (General Use, no detection and detection)

In Figure 3, the general use of the system will be demonstrated. Driver shifts the vehicle into reverse. This activates the Pedestrian Backup Assistance System and displays to the driver the rear blindspot. The Pedestrian Backup Assistance System continually monitors the vehicle's rear trajectory alerting the driver of potential obstacles. During the reverse, the system detects an obstacle that crosses the rear of the vehicle. The system calculates the distance to the obstacle and determines if it needs to give visual and audible warnings to the driver based on the distance. After the driver has finished reversing the vehicle, they will shift the vehicle out of reverse and the system will deactivate.

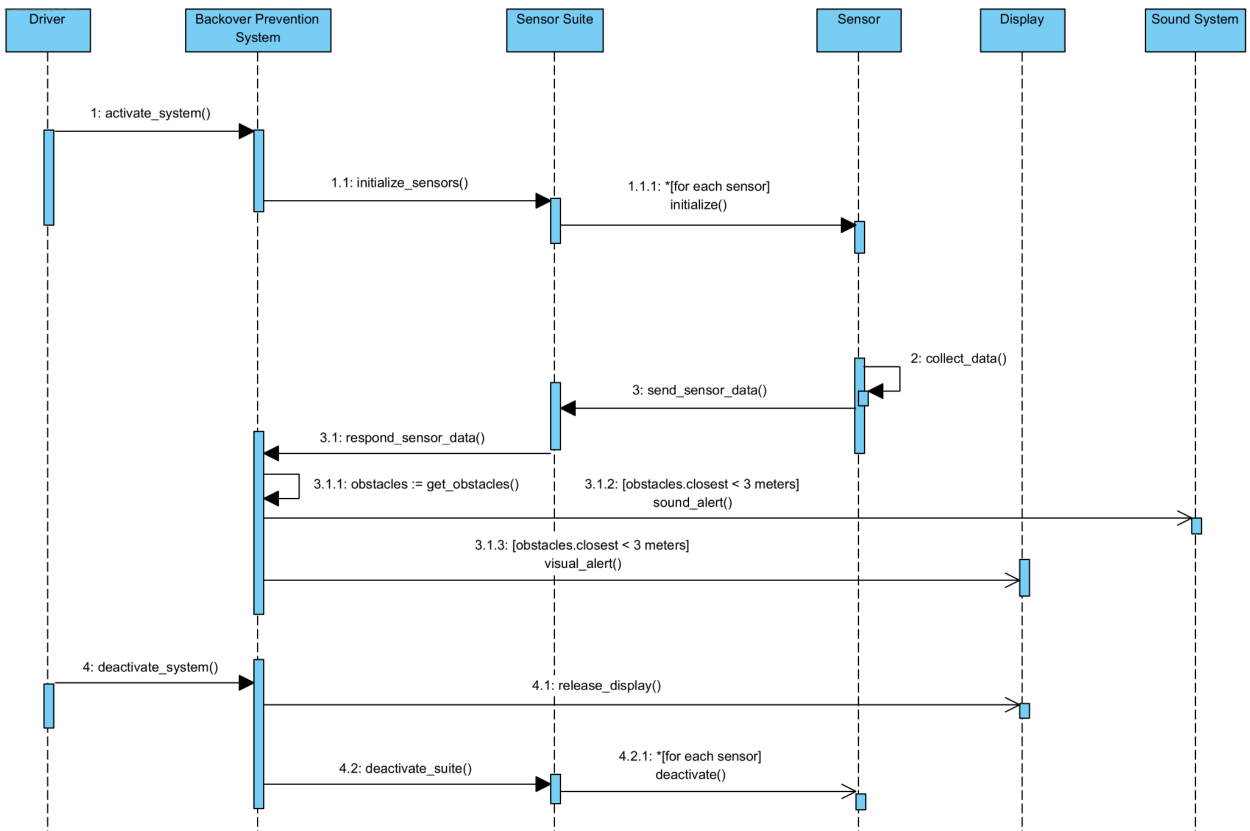

Figure 3: Sequence diagram for system activation when no obstacles are behind the vehicle

Vehicle Alerts Driver of Obstacle and Driver Fails to Stop (Automatic Braking)

In the scenario shown in Figure 4, we will show how the system will react to the Driver reversing into an obstacle. Driver shifts the vehicle into reverse. This activates the Pedestrian Backup

Assistance System and displays to the driver the rear blindspot. An obstacle crosses begins to cross the rear of the vehicle and a visual and audible warning is given to the driver. The driver is unable to react in time and continues to reverse the vehicle. Once the vehicle detects that the obstacle is within 1 meter of the rear bumper the system will automatically engage the brakes. The driver can disengage the brakes by holding the brakes and letting go. Once the driver is finished reversing and shifts out of the reverse gear, the system will deactivate.

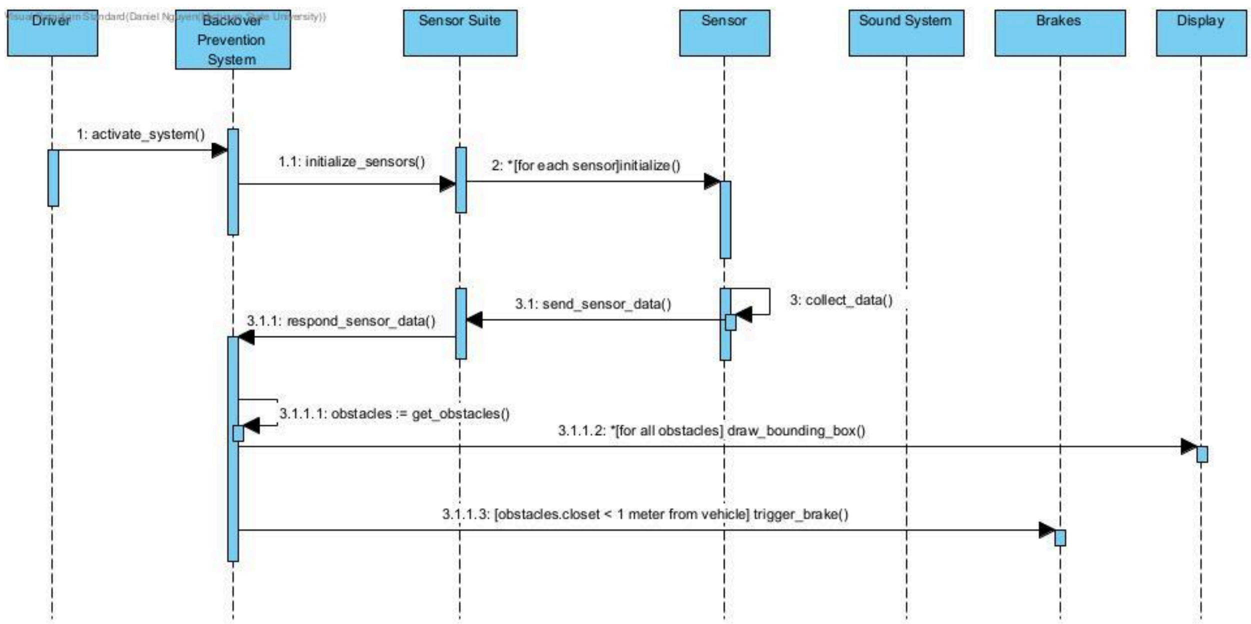

Figure 4: Sequence diagram for the case where the driver fails to stop and emergency braking activates.

Vehicle Detects Non Existent Obstacle (Temporarily Mute Alarm)

In figure 5, we show what the system does if it encounters a false positive. While the Driver is reversing the vehicle, the Pedestrian Backup Assistance System falsely detects there is an obstacle and gives visual and audible warnings to the driver. The driver checks the camera's feed to see where the obstacle is and sees there is no obstacle. They push a button on the dashboard to mute the warnings and continue reversing. The system does not continue giving warning to the driver until it sees a new obstacle enter the rear blindspot.

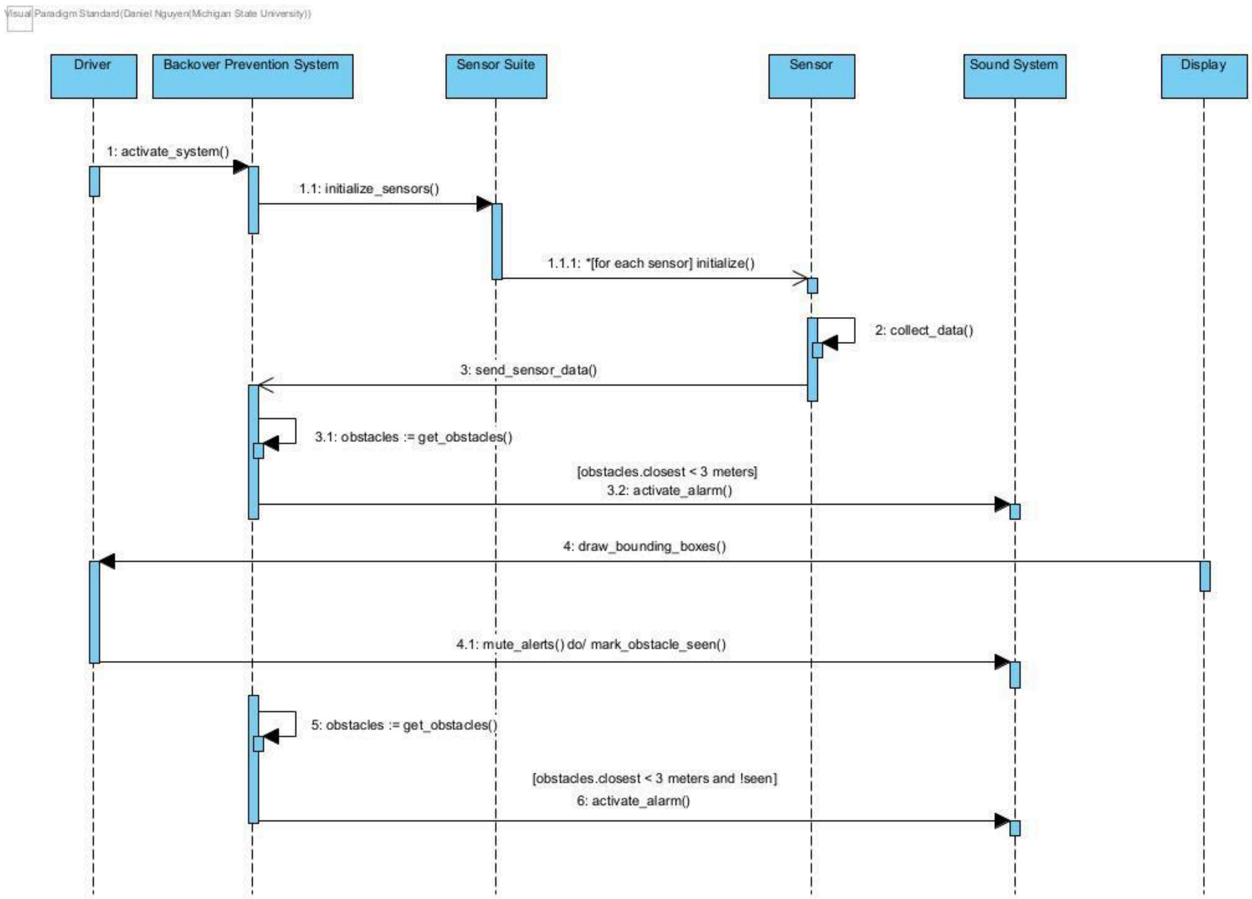

Figure 5: Sequence diagram for the case where the system detects a non-existent obstacle

Vehicle Collided with Object (Immobilize Vehicle)

Figure 6 demonstrates what happens when a collision occurs with the vehicle. In this scenario, the Driver is reversing the vehicle and an obstacle is fast approaching our vehicle and the system displays warnings to the driver. The driver stops their car, but the obstacle continues on its path and collides with the vehicle. Upon the system detecting the obstacle with the collision detector, the system automatically engages the brakes immobilizing the vehicle.

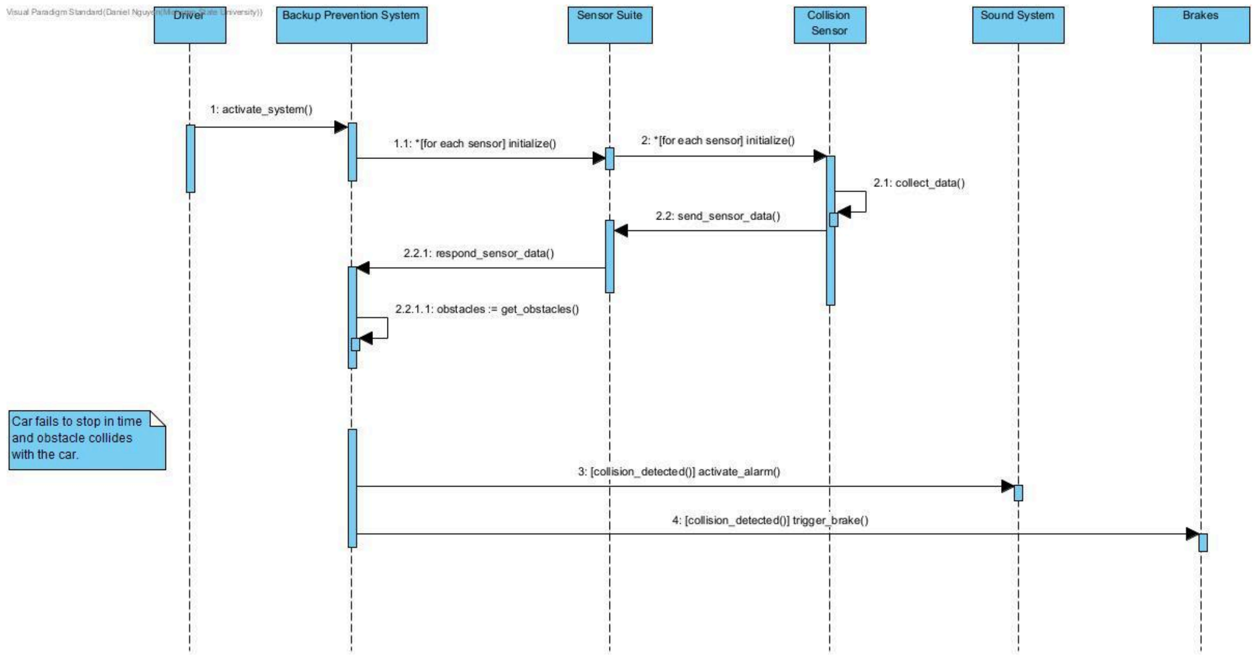

Figure 6: Sequence diagram for collision detection

System Detects Sensors are Faulty

If the system detects that less than 2/3 of the sensors are faulty, it notifies the driver that some sensors are malfunctioning but does not shut down. If the system detects that more than 2/3 of the sensors are faulty, it alerts the driver that the system is no longer functional through both an audio alert and a visual alert on the display. Then the system automatically deactivates itself. Figure 7 illustrates this scenario.

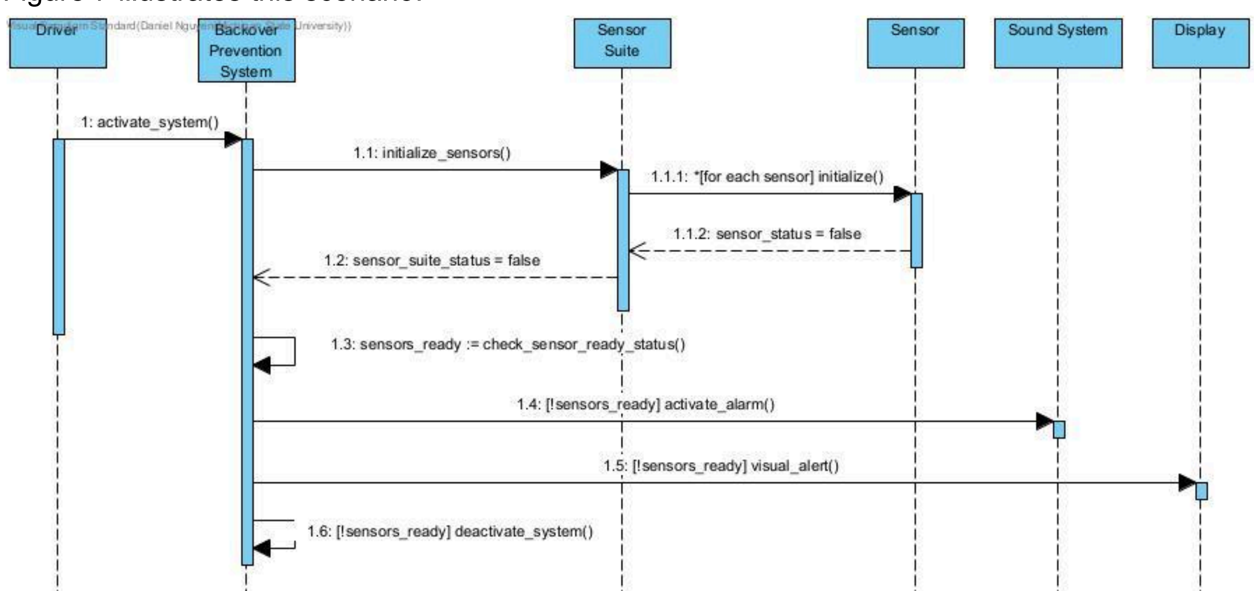

Figure 7: Sequence diagram for the case where ⅔ of sensors are faulty

View Rear Blindspot

In this simple scenario, we show what the system does in the presence of obstacles that are not a danger to the vehicle. When the system is activated, the system detects objects in the backup camera's vicinity and creates bounding boxes around the obstacles to identify each object for the driver. Figure 8 shows this scenario.

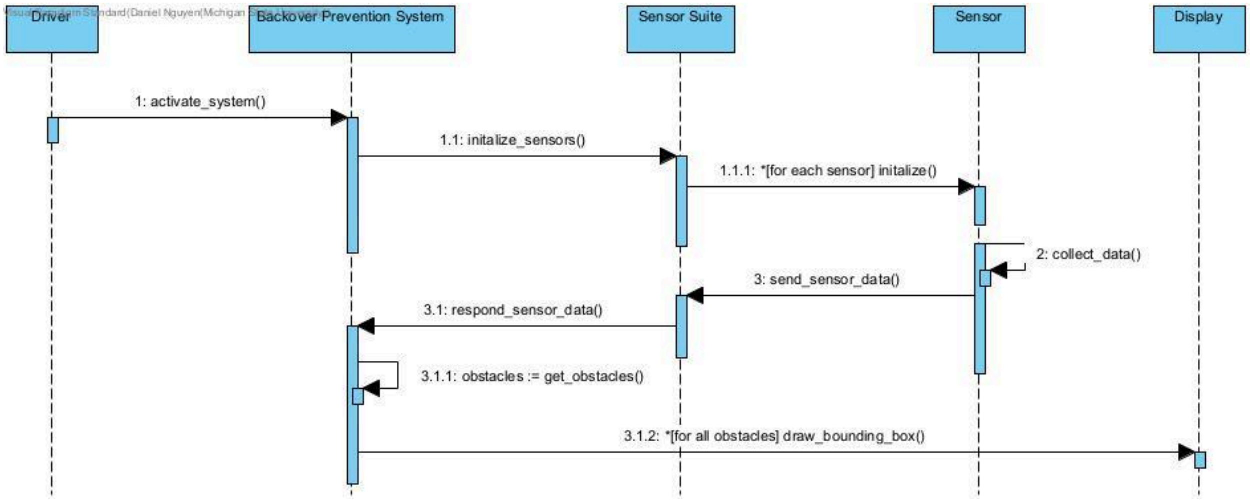

Figure 8: Sequence diagram for viewing vehicle rear blindspots and drawing bounding boxes

State Diagrams

In this section, each of the objects present in the domain model will have their states modeled with these state diagrams. The black circle in the diagram signifies which state is the starting state. Each transition consists of a response followed by any actions that are triggered as a result of that transition.

Backover Prevention System:

This is the controller of the system. Using data from the sensors, it determines the distance from the obstacles behind the vehicle and triggers the alarm/brakes when necessary. It also alerts the driver and deactivates itself if more than ⅔ of the sensors are faulty. Figure 9 shows the state diagram.

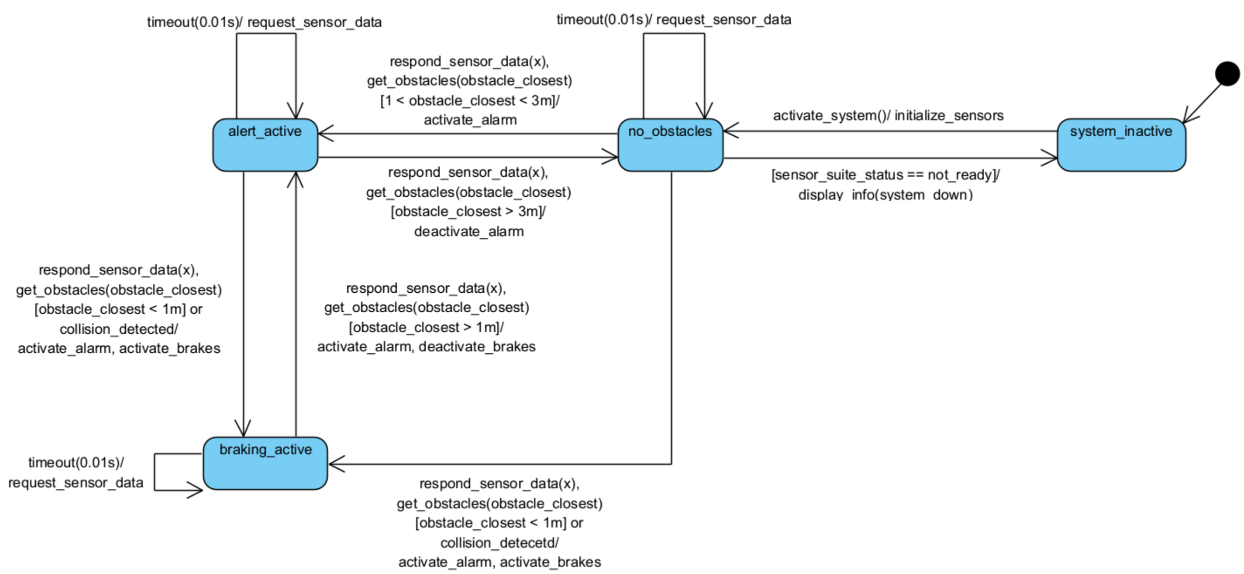

Figure 9: State diagram for Backover Prevention System

Sensor Suite:

This module collects data from all of the sensors and sends it to the BackoverPreventionSystem upon receiving a request for sensor data. If it fails to initialize sensors, it will set sensor suite status to not_ready which deactivates the BackoverPreventionSystem. Figure 10 shows the state diagram.

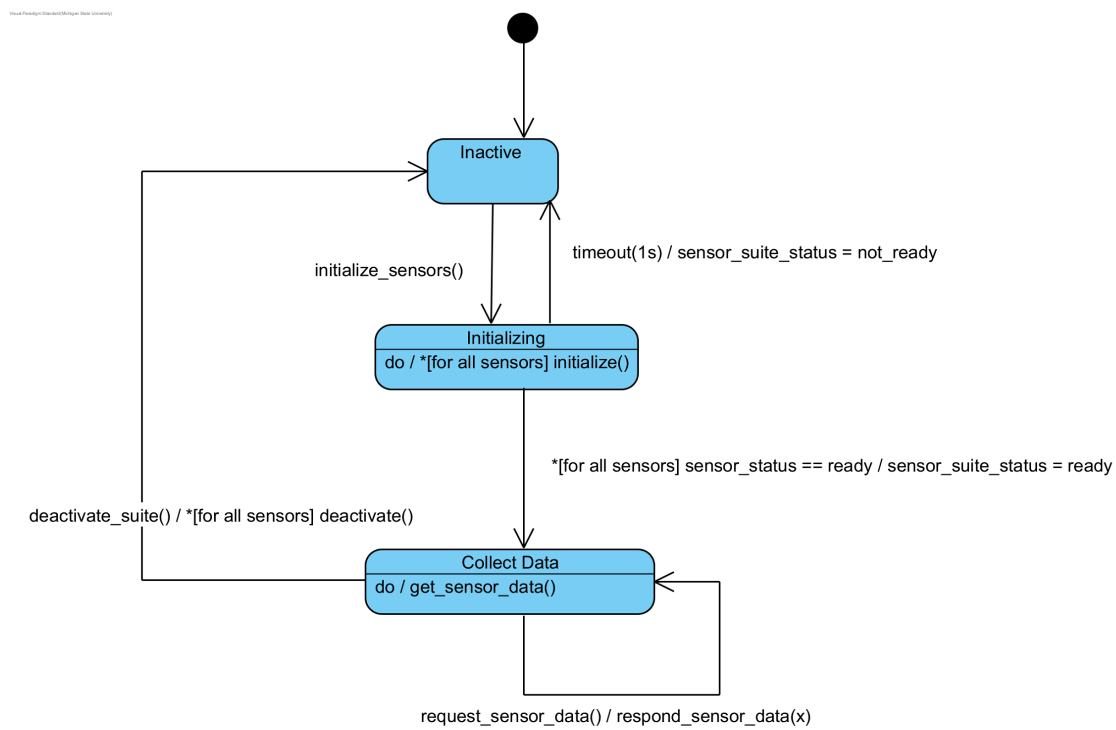

Figure 10: State diagram for Sensor Suite

Sensor:

All sensors which include the LiDAR, Collision Detection sensors, proximity sensors, and camera follow the same state diagram. The four sensor types are not modeled individually because all of the sensor types utilize the same state logic, and did not warrant repeating the diagram. They report their sensor status upon initialization and transmit sensor data when requested by the sensor suite. Figure 11 shows the state diagram.

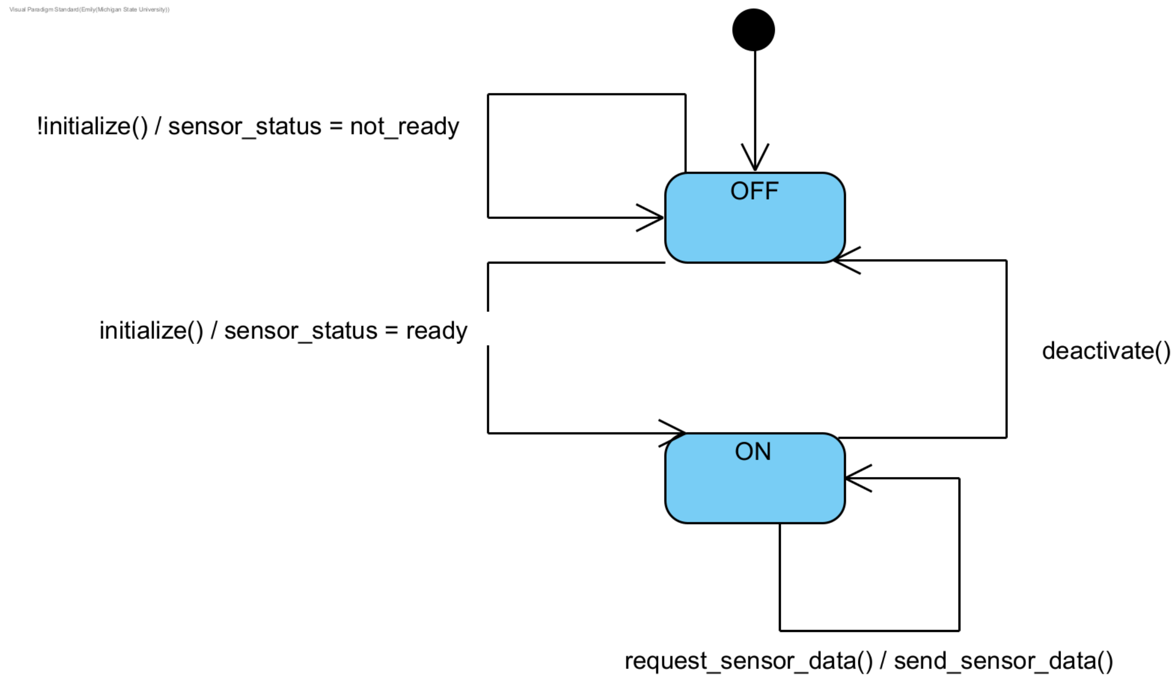

Figure 11: State diagram for Sensor

Sound System:
This module activates and deactivates the sound system to alert the driver based on the generative actions from the BackoverPreventionSystem. Figure 12 shows the state diagram.

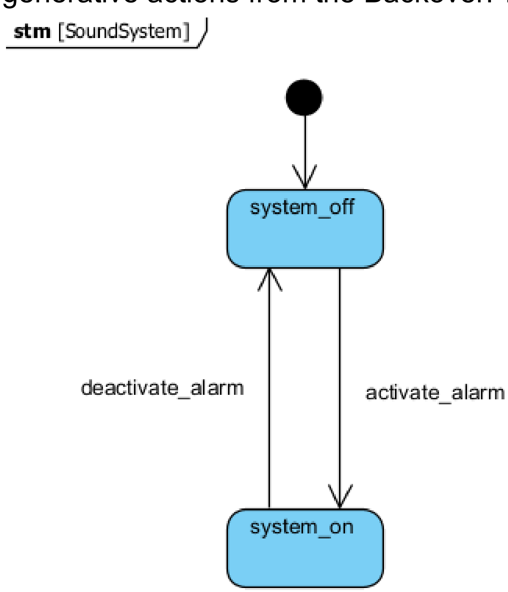

Figure 12: State diagram for Sound System

Brakes:

This module activates and deactivates the braking system based on the generative actions from the BackoverPreventionSystem. Figure 13 shows the state diagram transitions.

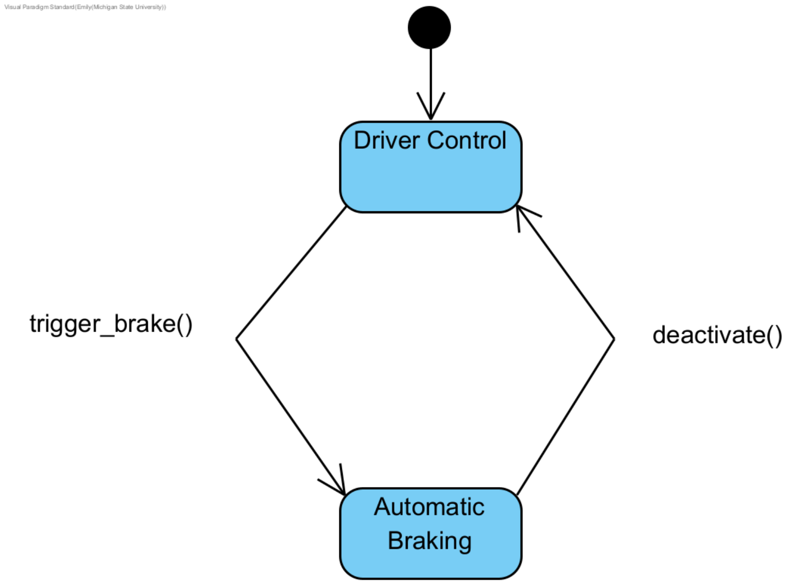

Figure 13: State diagram for Brakes

Display:
This module displays the vehicle's rear blindspots to the driver and deactivates when there is no more information to show on display or when it is deactivated by an external module. Figure 14 shows the state diagram transitions.

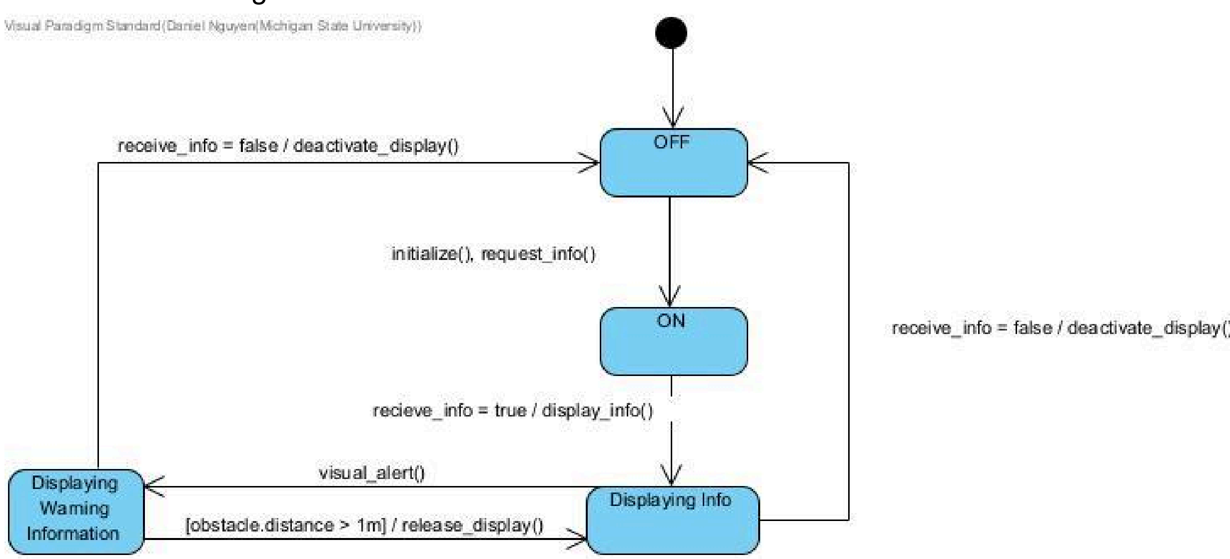

Figure 14: State diagram for Display

5. Prototype

5.1. How to Run the Prototype

The online prototype application may be found at the following location: https://javenw.itch.io/cse870-bps-prototype. Clicking on "Run game" will launch the web application version of the prototype. Alternatively, platform specific versions (Windows, MacOS, Linux) may be downloaded and run through the same webpage, which do not contain audio and lighting bugs specific to the web application version.

The following controls are available: The numbers **1-4** will change scenarios, **W** controls acceleration, **S** controls braking, **A** steers the car to the left, **D** steers the car to the right, **R** switches the car between reverse and drive modes, **V** changes the view perspective, and **M** disables the alarm and emergency braking features. The heads up display (HUD) shows these controls, vehicle state information, active scenario description (scrollable), and active scenario collision objects detected by the Backover Prevention System (BPS). Figure 15 showcases the starting position of the prototype when launched for the first time.

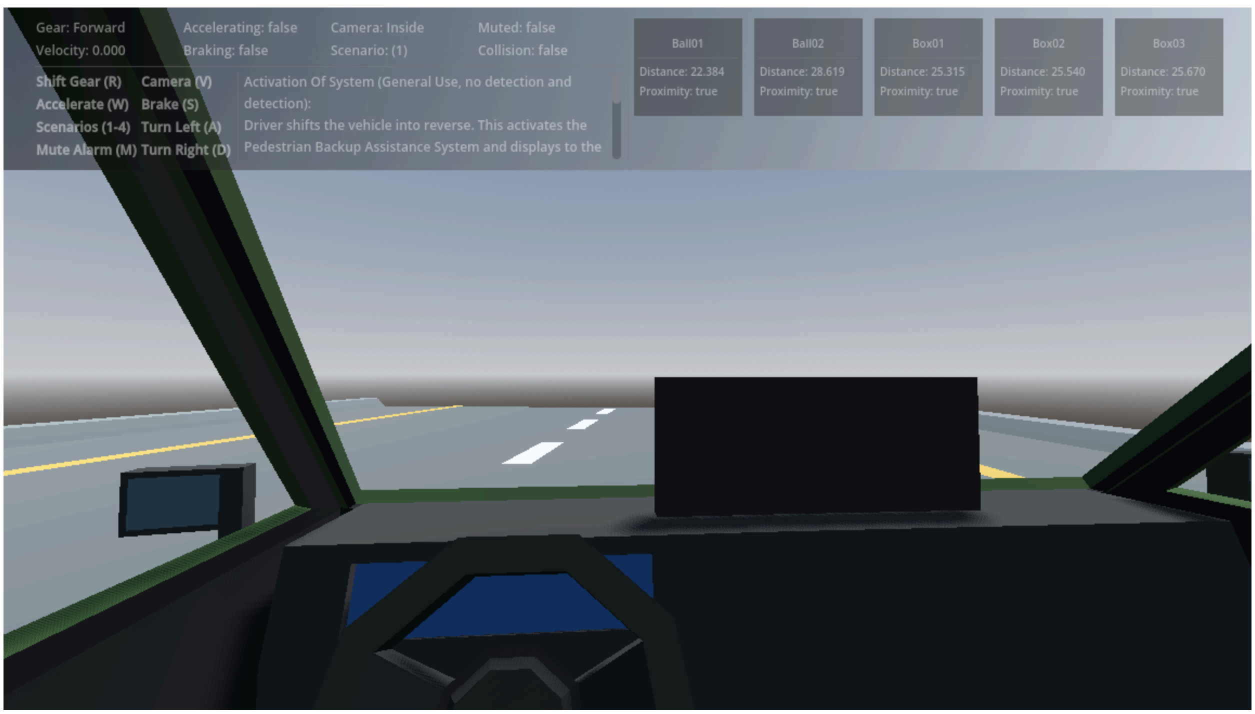

Figure 15: Screenshot of the prototype in scenario 1 in the forward mode, where the display does not show the rearview camera video feed.

5.2. Sample Scenarios

Scenario 1 demonstrates the general use of the system and how the system reacts to stationary collision objects while the BPS is active. Upon launching the application, this scenario will be the default scenario loaded and active, but may also be reset at any point (**1**). By default in this scenario, the vehicle will be in a “Forward” gear state, as indicated in the top-left HUD, which disables the BPS and only permits forward vehicle movement. In this initial position, there will be no collision objects in front of the vehicle and the vehicle center dashboard display interface will be disabled (all black screen).

Switch gears to Reverse (**R**), you will notice the display interface become active and reflect the vehicle’s rear camera datastream. In the display, you should now notice several collision objects located behind the vehicle, two blue balls and a stack of orange boxes, each with a colored rectangular bounding box indicating they are potential hazards to the vehicle. Accelerate (**W**) and steer (**A / D**) toward the objects (while still in reverse gear) and observe the color change of the bounding boxes as the vehicle minimizes the distance between them, red indicating the closest distance and green the furthest. Figure 16 shows how scenario 1 is setup on start up.

Figure 16: Screenshot of the prototype in scenario 1 where there are two stationary spheres and a stationary pile of cubes highlighted on the display with bounding boxes.

Upon entering within the specified alarm distance to any one of the objects, the alarm sound effect will begin playing (potential bug with the web version), with its frequency determined by the distance to the nearest object. Upon entering within the specified emergency braking distance to the nearest object, the Emergency Braking System (EBS) will activate and automatically brake the vehicle to a halt; no further reverse acceleration will be permitted while in this state. Switch gears to Forward (**R**) and navigate away from the collision objects. Then, switch gears back to Reverse (**R**), and navigate in-between and past the objects without activating the EBS to complete the scenario. Manual braking (**S**) may be applied while in either gear state for navigational control. Figure 15 showcases the bounding boxes changing color when the car detects that it is to

Figure 17: Screenshot of the prototype. Demonstrates the display showing the rearview camera footage.

Scenario 2 demonstrates the situation where there is a false positive object detection, and the vehicle operator has to manually mute the EBS and Alarm System to proceed. Load the scenario (**2**); the vehicle should already be in the Reverse state (switching gears is disabled for this scenario). Observe the erroneous bounding box on the display interface; this represents the falsely detected collision object by the BPS (i.e., “Ghost Object”). Upon approaching the Ghost Object, the EBS and Alarm System will still activate, preventing the vehicle from proceeding. Manually mute (**M**) these systems, and proceed past the Ghost Object to complete the scenario.

Scenario 3 demonstrates how the BPS reacts to an object collision with the vehicle. Load the scenario (**3**); the vehicle should already be in the Reverse state (switching gears and steering are disabled for this scenario). In this scenario, a moving object (a truck vehicle) will automatically approach the user’s vehicle from the rear while the BPS is active. The BPS will still detect the approaching object, attempt to activate the alarms and EBS, but the collision will occur outside of the user's control. Upon colliding with the moving object, the user's vehicle will be put into a paralyzing Collision state where a hazard message will be shown on the display interface, the EBS activates, and all mobility controls are disabled. This completes the scenario.

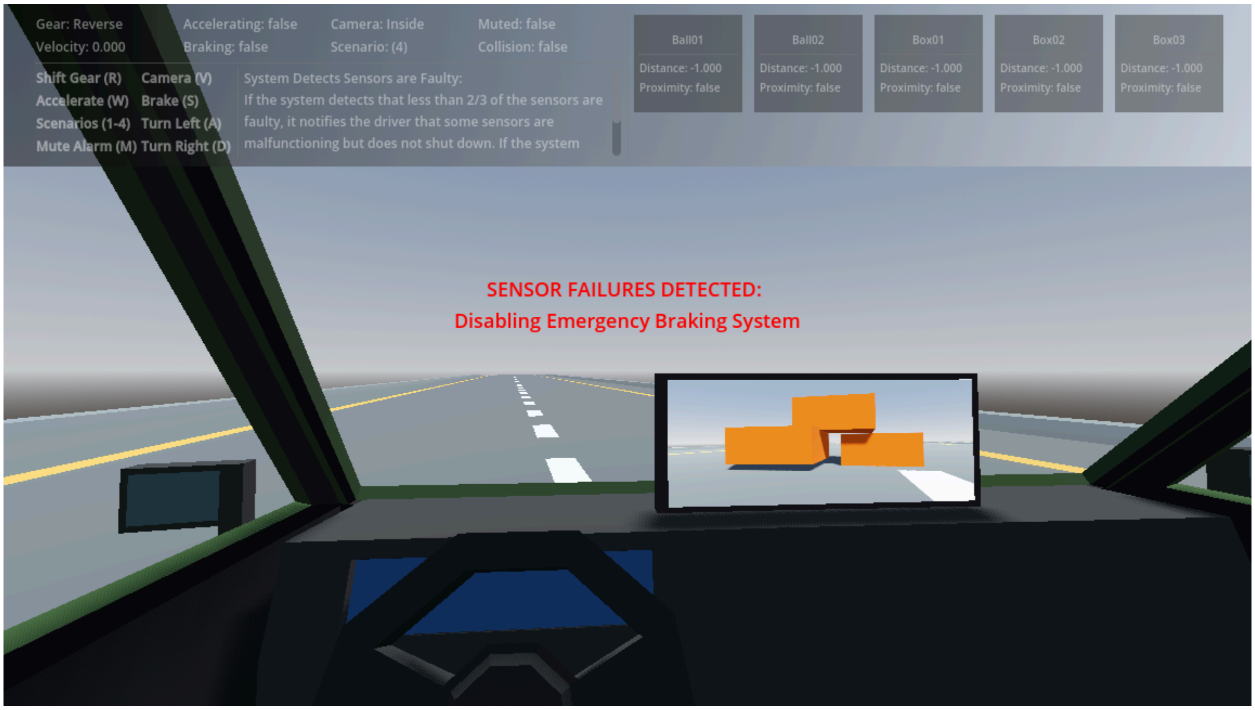

Figure 18: Screenshot of the prototype in scenario 4 showing what would occur when two thirds of the sensors are offline.

Finally, Scenario 4 demonstrates what occurs when the system detects a majority of sensors are faulty. Load the scenario (**4**) and shift to the Reverse gear (**R**). Upon entering the Reverse state in this scenario, an error message will display indicating the EBS is disabled. The error message should appear on the vehicle's display in practice; however, due to the low text quality on the display, we presented the error message as shown in Fig. 12. While in this state, bounding boxes will not appear, the EBS will not activate, and alarms will not sound. Colliding into objects, while in either the Reverse or Forward state, will still put the vehicle into a disabling Collision state. To complete this scenario, reverse and collide into an object, reset (**4**), and then attempt to navigate in reverse past the objects similar to Scenario 1.

Figures 19 and 20 showcase more screenshots of the prototype's scenario 1, to give a more clear view of the general setup used.

Figure 19: Screenshot of the prototype in scenario 1 where there are two stationary spheres and a stationary pile of cubes.

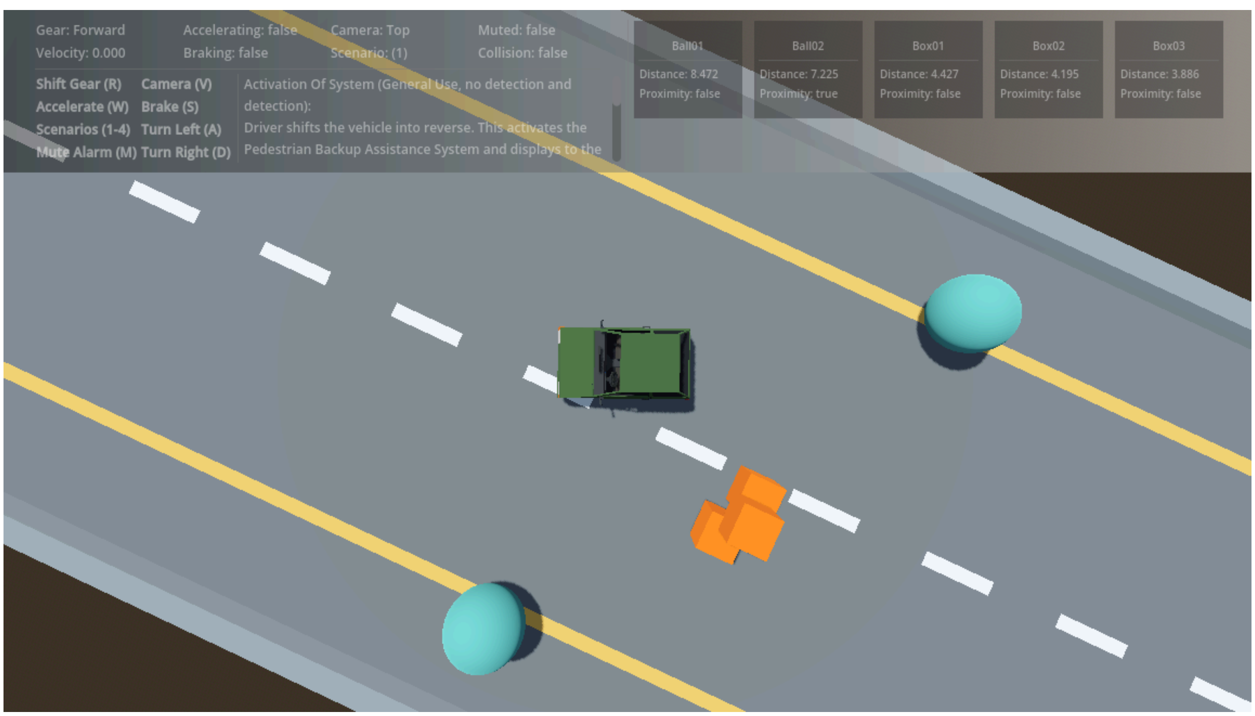

Figure 20: Screenshot of the prototype in the top-down perspective of the vehicle.

6. References -

[1] “49 cfr 571.111,” https://www.ecfr.gov/current/title-49/section-571.111

[2] IBM, “LiDAR,” *Ibm.com*, Jul. 14, 2023. https://www.ibm.com/think/topics/lidar

[3] MaxBotix, “How Ultrasonic Sensors Work,” *MaxBotix*, Mar. 01, 2023. https://maxbotix.com/blogs/blog/how-ultrasonic-sensors-work

[4] “Force Sensor | Measuring Force | How it Works | FUTEK,” *www.futek.com*. https://www.futek.com/force-sensor

[5] “STATE OF CALIFORNIA • DEPARTMENT OF TRANSPORTATION ADA Notice.” Accessed: Mar. 11, 2025. [Online]. Available: https://dot.ca.gov/-/media/dot-media/programs/research-innovation-system-information/documents/final-reports/ca10-0940-finalreport-a11y.pdf

7. Point of Contact

7.1. 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.