

Software Design Specification

for

Autonomous Vehicle System: IntelliDrive

Prepared by Joshua Concepcion, Logan Scully, Xiangming Xie

13 October 2023

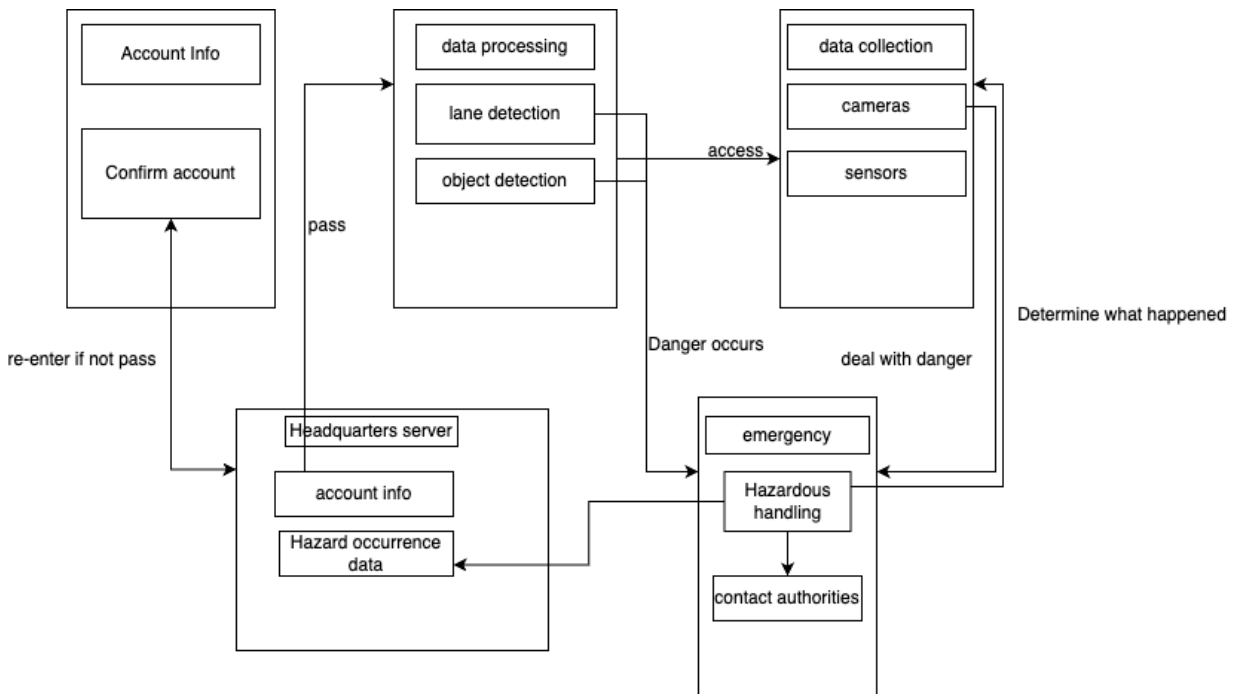
1. System Description

IntelliDrive is a software system designed to keep the driver, passengers, and others on the road safe. Our system will utilize lane and object detection, road stability, and real-time data gathering through the use of a variety of cameras, sensors and other similar components. In the event of an emergency, the system will send signals via satellite to the nearest rescue center. This feature also detects obstacles in the car's path and sends that information to the vehicle's computer to adjust road stability.

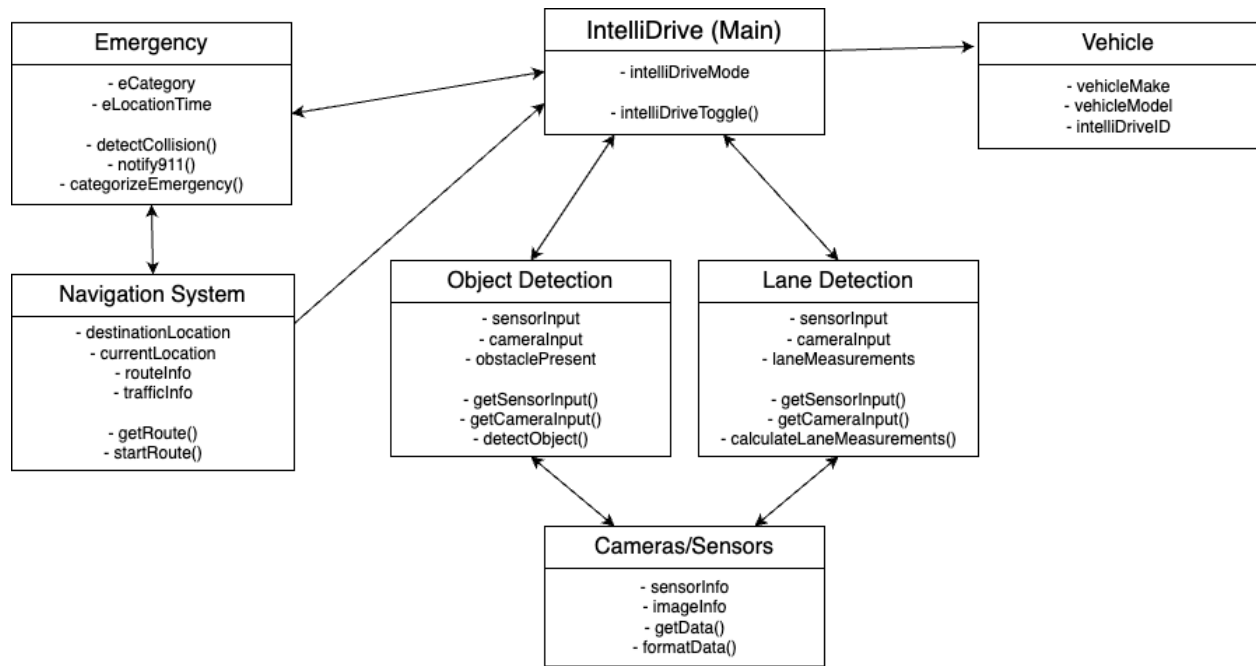
However, there are limits to our system that we do intend to expand upon in the future. For the time being, our AVS does not provide complete self-driving functionality, and users will be expected to handle a variety of aspects when operating the vehicle; the user is responsible for reacting to traffic signs, making turns, merging/exiting freeways, as well as accelerating or braking.

2. Software Architecture Overview

2.1 Architectural Diagram of all Major Components



2.2 UML Class Diagram



2.3 Description of Classes

IntelliDrive (Main)

- Contains method toggling of autonomous driving mode, calls classes such as object detection, lane detection, camera/sensors, and navigation system

Navigation System

- Contains methods for retrieving current location, determining efficient route

Vehicle

- Contains information regarding the vehicle's make, model, and registered IntelliDrive account

Lane detection

- Calls cameras and sensors
- Contains methods for processing information from the cameras and sensors, as well as methods for determining shape of lane

Object detection

- Calls cameras and sensors
- Calls Satellite class for information on set route

Camera/Sensors

- Provides information used for lane and object detection
- Contains methods for reading/processing information about scanned areas or images

Emergency

- Calls Navigation System to update route info and notify of accidents/obstacles
- Contains methods for determining accidents, system failures, and notifying authorities

2.4 Description of SWA diagram

- The system detects whether Account Info matches when booting Auto pilot. If the account does not match, you need to re-enter the account.
- When IntelliDrive Mode starts, Lane Detection and Object Detection will automatically start and run passively. Automatically detect whether Account Info matches when booting Auto pilot. Object Detection is responsible for using cameras/sensors to detect other vehicles/objects/pedestrians/and anything else on the road.
- Cameras and sensors are responsible for capturing information and sending it to Lane Detection and Object Detection
- If Object Detection and Lane Detection detect danger, emergency will be automatically enabled. When emergency is used, cameras and sensors are automatically called to detect what happened and call the appropriate authorities.
- Also when a vehicle is in danger, the server will automatically receive the information.

2.5 Description of Attributes

IntelliDrive (Main)

- intellidriveMode - a boolean attribute that would determine whether or not the autonomous driving mode is on

Navigation System

- destinationLocation - the address of desired destination

- `currentLocation` - the address/coordinates that indicate the vehicle's current location
- `routeInfo` - the route that the vehicle will take to reach the `destinationLocation`
- `trafficInfo` - provides information on current/expected traffic status

Vehicle

- `vehicleMake` and `vehicleModel` - provides basic information regarding the make and model of the vehicle
- `intelliDriveID` - the registered ID for the IntelliDrive user's vehicle

Lane Detection

- `sensorInput` - receives information from the sensors
- `cameraInput` - receives information from the camera
- `laneMeasurements` - information regarding size and shape of the lane

Object Detection

- `sensorInput` - receives information from the sensors
- `cameraInput` - receives information from the camera
- `obstaclePresent` - boolean that indicates whether there is an object in the road

Camera/Sensors

- `sensorInfo` - information regarding what the sensors have detected
- `imageInfo` - information regarding what the cameras have detected

Emergency

- `eCategory` - information regarding type of emergency (i.e. accident, system failure, etc)
- `eLocationTime` - information regarding the location of the accident/failure, as well as timestamp

2.6 Description of Operations

IntelliDrive (Main)

- `intelliDriveToggle()` - toggles on and off the autonomous driving mode

Navigation System

- `getRoute()` - determines route for vehicle to take
- `startRoute()` - begins navigation on screen

Lane Detection

- `getSensorInput()` - retrieves and processes input from sensor
- `getCameraInput()` - retrieves and processes input from cameras
- `calculateLaneMeasurements()` - uses input from sensor and camera to determine size and shape of lane

Object Detection

- `getSensorInput()` - retrieves and processes input from sensor
- `getCameraInput()` - retrieves and processes input from cameras
- `detectObject()` - uses input from sensors and camera to determine if there is an obstacle on the path, and notifies the system of obstacle if present

Camera/Sensors

- `getData()` - enables scanning and imaging of sensors and cameras
- `formatData()` - formats the data for processing by lane and object detection systems

Emergency

- `detectCollision()` - calls cameras and sensors to detect collision to car, or collision nearby on the road
- `notify911()` - calls/notifies authorities in the area to collision or accident
- `categorizeEmergency()` - uses information from cameras and sensors to determine the type of accident

3. Development Plan and Timeline

3.1 Partitioning of tasks

- Develop Software (9-12 months)
 - IntelliDrive (Main)
 - intelliDriveMode
 - intelliDriveToggle
 - Navigation System
 - destinationLocation
 - currentLocation
 - routeInfo
 - trafficInfo
 - getRoute()
 - startRoute()
 - Vehicle
 - vehicleMake
 - vehicleModel
 - intelliDriveID
 - Lane Detection
 - sensorInput
 - cameraInput
 - laneMeasurements
 - getSensorInput()
 - getCameraInput()
 - calculateLaneMeasurements()
 - Object Detection
 - sensorInput
 - cameraInput
 - obstaclePresent
 - getSensorInput()
 - getCameraInput()
 - detectObject()
 - Cameras/Sensors
 - sensorInfo
 - imageInfo
 - getData()
 - formatData()
 - Emergency
 - eCategory
 - eLocationTime
 - detectCollision()
 - notify911()

■ categorizeEmergency()

- Debugging (6-9 months)
 - Crosscheck colleagues code for errors
- Implementation (4-6 months)
 - Install hardware into select vehicles to run software
 - Install software into select vehicles for testing
- Testing (2-3 years)

3.2 Team member responsibilities

- Development
 - Joshua - Main, Emergency
 - Tony - Navigation System, Vehicle
 - Logan - Lane Detection, Object Detection, Cameras/Sensors
- Debugging
 - Logan review Main Emergency
 - Joshua reviews Navigation System, Vehicle
 - Tony review Lane Detection, Object Detection, Cameras/Sensors
- Implementation
 - Outsource hardware installation
 - Tony is in charge of installing the software
- Testing
 - Logan and Joshua will come up with ways to test the software properly
 - Employee beta testers will receive versions of the software to test on the road