Developing and Operationalizing a Testbed of Connected Automated Shuttles to Test and Develop CAV Applications in North Carolina

NCDOT Project 2022-16
Technical Report #: FHWA/NC/2022-063
Sep 2024

North Carolina Agricultural and Technical State University

Ali Karimoddini, Ph.D.,
Autonomous Cooperative Control of Emergent Systems of Systems (ACCESS) Laboratory
Electrical and Computer Engineering Department
North Carolina A&T State University

RESEARCH & DEVELOPMENT

Technical Report Documentation Page

1. Report No. FHWA/NC/2022-16	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle Developing and Operationalizing a Testbed of Connected Automated Shuttles to Test and Develop CAV Applications in North Carolina		5. Report Date: Sep 30, 2024 6. Performing Organization Code
7. Author(s): Ali Karimoddini		8. Performing Organization Report No.
9. Performing Organization Name and Address: North Carolina A&T State University, Autonomous Cooperative Control of Emergent Systems of Systems (ACCESS) Laboratory, Electrical and Computer Engineering Department, 1601 East Market Street, Greensboro, NC, 27411		10. Work Unit No. (TRAIS) 11. Contract or Grant No.
12. Sponsoring Agency Name and Address North Carolina Department of Transportation Research and Development Unit 104 Fayetteville Street Raleigh, North Carolina 27601		13. Type of Report and Period Covered Final Report; Aug 1, 2021 – Sep 30, 2024 14. Sponsoring Agency Code: RP2022-16
Supplementary Notes:		

16. Abstract

With advances in communication, computation, and control technologies, fleets of Connected Autonomous Vehicles (CAV) can be deployed to flexibly provide on-demand transportation services in the near future. Therefore, it is important to start the development of CAV testbeds and piloted projects for the deployment of CAVs to complement some of the on-going efforts in North Carolina on the developed infrastructure for AVs such as Connected Autonomous Shuttle Supporting Innovation (CASSI) program. Therefore, the goal of this project is to *develop and operationalize a flexible and customizable open hardware/software testbed for connected automated microtransit vehicles consisting of three automated shuttles*. This will be achieved by developing and operationalizing a CAV testbed of three automated shuttles through the following objectives: (1) Installing, testing, and operationalizing the hardware components of the CAV testbed, in which all hardware components are modifiable according to the application in mind, (2) Developing and testing software components of the CAV testbed, (3) Installing, testing, and operationalizing the communication devices of the CAV testbed with the capability of communicating with other vehicles and infrastructure which follow SAE and IEEE standards for CAVs, (4) Integration of the components of the developed CAV testbed, and (5) Implementation and deployment of the CAV testbed. This project provided an opportunity for the CAV research community in North Carolina to have access to this unique, comprehensive, and flexible testbed of CAVs with a focus on microtransit vehicle systems. This project also provided information needed to design future transportation that remains relevant and ready for future CAV technologies.

17.	18. Distribution Statement		
19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages 34	22. Price

Form DOT F 1700.7 (8-72) **Reproduction of completed page authorize**

DISCLAIMER

The contents of this report reflect the views of the author(s) and not necessarily the views of the University. The author(s) are responsible for the facts and the accuracy of the data presented herein. The contents do not necessarily reflect the official views or policies of either the North Carolina Department of Transportation or the Federal Highway Administration at the time of publication. This report does not constitute a standard, specification, or regulation.

Acknowledgements

This research project was sponsored by the North Carolina Department of Transportation (NCDOT) and their continued support is greatly appreciated. The research team would like to express their appreciation to the following:

- The chair of the Steering and Implementation Committee, Sarah Searcy, for continuous engagement, support, and guidance throughout the project.
- The NCDOT personnel serving on the Steering and Implementation Committee for this research study including Robert Sawyer, Dana Magliola, Amanda Good, and Ryan Brumfield during the course of this project is greatly appreciated.
- NCDOT Research and Development personnel, particularly the project manager, John Kirby.
- Former members of the Steering and Implementation Committee including Kevin Lacy.

EXECUTIVE SUMMARY

With advances in communication, computation, and control technologies, fleets of Connected Autonomous Vehicles (CAV) can be deployed to flexibly provide on-demand transportation services in the near future. Therefore, it is important to start the development of CAV testbeds and piloted projects for the deployment of CAVs to complement some of the on-going efforts in North Carolina on the developed infrastructure for AVs such as Connected Autonomous Shuttle Supporting Innovation (CASSI) program. Therefore, the goal of this project is to develop and operationalize a flexible and customizable hardware/software testbed for connected automated microtransit vehicles consisting of three automated shuttles. This will be achieved by developing and operationalizing a CAV testbed of three automated shuttles through the following objectives: (1) Installing, testing, and operationalizing the hardware components of the CAV testbed, in which all hardware components are modifiable according to the application in mind, (2) Developing and testing software components of the CAV testbed, (3) Installing, testing, and operationalizing the communication devices of the CAV testbed with the capability of communicating with other vehicles and infrastructure which follow SAE and IEEE standards for CAVs, (4) Integration of the components of the developed CAV testbed, and (5) Implementation and deployment of the CAV testbed. This project provided an opportunity for the CAV research community in North Carolina to have access to this unique, comprehensive, and flexible testbed of CAVs with a focus on microtransit vehicle systems. This project also provided information needed to design future transportation that remains relevant and ready for future CAV technologies.

Contents

List of Tables

List of Figures

1 Introduction

This project proposes to develop and operationalize a state-of-the-art CAV testbed that will provide researchers with an opportunity to design, prototype, and test advanced transportation technologies, foster new multidisciplinary collaborations with universities and industry partners across North Carolina and the nation. More specifically, this project aims to develop and operationalize a flexible and customizable open hardware/software testbed for connected autonomous microtransit vehicles to enable researchers across North Carolina to prototype and test CAV applications and technologies. This is achieved by developing and operationalizing a CAV testbed of three automated shuttles through the following objectives: (1) Installing, testing, and operationalizing the hardware components of the CAV testbed, in which all hardware components are modifiable according to the application in mind, (2) Developing and testing software components of the CAV testbed, (3) Installing, testing, and operationalizing the communication devices of the CAV testbed with the capability of communicating with other vehicles and infrastructure which follow SAE and IEEE standards for CAVs, (4) Integration of the components of the developed CAV testbed, and (5) Implementation and deployment of the CAV testbed. This provides the CAV research community in North Carolina to have access to a unique and comprehensive testbed of CAVs whose open hardware/software architecture makes it completely modifiable and customizable, enabling the implementation of different CAV applications scenarios.

2 Research Objectives

The goal of this multidisciplinary project is to develop and operationalize a flexible and customizable hardware/software testbed for connected automated microtransit vehicles to enable researchers across North Carolina to prototype and test CAV applications and technologies. This will be achieved through the following objectives:

- **Objective 1:** Installing, testing, and operationalizing the hardware components of the CAV testbed, in which all hardware components are modifiable according to the application in mind,
- **Objective 2:** Developing and testing software components of the CAV testbed with an open-source architecture and documentation, facilitating the software modification,
- **Objective 3:** Installing, testing, and operationalizing the communication devices of the CAV testbed with the capability of communicating with other vehicles and infrastructure which follow SAE and IEEE standards for CAVs,

- **Objective 4:** Integration of the components of the developed CAV testbed, and testing the integrated components via simulation and actual implementation.
- **Objective 5:** Implementation and deployment of the CAV testbed in NCAT test tracks for different application scenarios,

3 Literature Review and Gap Analysis

The advancement of Connected and Automated Vehicles (CAVs) technology has undergone a remarkable acceleration over the last few decades, propelled by the growing demand for a transformation shift in transportation paradigms. These CAVs are set to capitalize on the progress of mobile networks, enabling seamless communication with their surroundings, including other vehicles and roadside equipment [1]. This evolution is characterized by the integration of robust automated vehicle capabilities with sophisticated communication networks, offering significant potential for enhancing transportation efficiency, safety, and reliability [2]. As society increasingly embraces the concept of smart transportation, the challenges inherent in CAV development continue to be rigorously investigated from fundamental and technological perspectives. Aligned with conventional development processes, real-world vehicle testbeds are pivotal in facilitating system-level verification and validation procedures [3]. These measures are imperative for ensuring the dependability of system operations and afford comprehensive insights into the implications of CAV integration on practical traffic operations.

The advent of 5G technology brings significant capacity, security and privacy advancements to CAV development, primarily through increased capacity, lowered latency, enhanced encryption and privacy measures, network slicing for isolated vehicular communications, improved identity management and access control, and real-time threat detection [4]. These features ensure increased capacity, stronger protection of data transmission, establish dedicated network segments for secure vehicular communication, authenticate entities within the network rigorously, and facilitate immediate identification and mitigation of cybersecurity threats [5]. Together, these aspects of 5G technology create a more secure and reliable environment for CAV operations, paving the way for safer autonomous driving experiences [6].

Taking into account logistics, safety, and cost considerations, simulation-based testing has gained widespread adoption while providing the opportunity for comprehensively testing a wide range of operational scenarios [7]. Moving beyond pure simulation and pushing for more realistic testing approaches, hardware-in-the-loop simulations provide an environment that closely approximates real-world conditions, enabling meticulous observation of the physical driving behaviors [8]. Nevertheless, physical tests on actual autonomous vehicles within authentic real-world conditions are still necessary to validate their performance before deployment.

A comprehensive review of the literature reveals that existing testbeds have demonstrably accelerated the development and deployment of CAVs. These platforms have facilitated the exploration of challenging scenarios and the seamless integration of critical hardware and software components. However, there are still significant challenges in terms of practical use of CAVs which requires the continuation of the effort on the development of CAV testbeds to incorporate real-time communication capabilities, enable system-level evaluation, and possess the scalability to accommodate multiple vehicles. Such advancements would represent a significant step toward replicating real-world CAV deployment scenarios, thereby further accelerating the path of the technology to practical application.

This research centers on the development and evaluation of a CAV testbed, which includes a fleet of three electric vehicles with connected and autonomous driving capabilities. Consequently, the contributions of this work are described as follows:

- The development of a CAV testbed based on the Gem E6 Polaris electric vehicles by providing an integrated hardware and software architecture that facilitates a wide range of connectivity and autonomous driving capabilities, enabling researchers to investigate, prototype, implement, and test various CAV concepts and driving scenarios.
- The integration of a comprehensive suite of sensors for enhanced sensing and perception, the supporting computing unit and power distribution system, and the 5G communication devices, allowing for the test and evaluation of connectivity and automated driving systems in real-world scenarios, which is crucial for the development of reliable and efficient autonomous driving technologies.
- The demonstration of the deployment of the CAV testbed components and 5G connectivity through real use cases is a critical step towards the future deployment of CAVs, allowing the assessment of V2X communication performance in real driving environments.

The remainder of this document is organized as follows: Section 4 outlines the development and setup of the CAV testbed, delving into its hardware and software frameworks. Section 5 examines deployment results, emphasizing the enhancements in connectivity brought about by 5G technology. The article wraps up with Section 6, offering a conclusion of the key findings and proposing future works for research within the realm of CAV technology.

4 Overview of the CAV Testbed

4.1 Hardware Architecture of the Autonomous Vehicles of the CAV Testbed

CAVs require various sensor types to effectively perceive the surrounding environment, enabling the vehicle to gather accurate and precise information. The data collected from these sensors is then processed using high-performance onboard computing units and subsequently used for control and decision-making about the driving tasks. To optimize the CAV performance, an autonomy software stack is utilized to generate and interpret maps, plan optimal routes, avoid obstacles, and control the lateral and longitudinal dynamics of the vehicle to track the planned routes in compliance with transportation laws and safety standards while communicating with other CAVs, road users, and infrastructure. Subsequently, the drive-by-wire unit transmits the control commands from the computing unit to adjust the acceleration, brake, and steering wheel to execute different driving maneuvers. Ensuring the reliable and stable operation of these hardware components requires a dependable power source, which is achieved through connection to the battery of the vehicle via a power regulator, with an emergency power cut-off implemented to enhance safety and facilitate the driver to take control of the vehicle at any time. In this section, the sensing, computing, intra-vehicle networking, power distribution, and drive-by-wire system architectures of the developed AVs of the CAV testbed are explained in detail.

4.1.1 Sensing Architecture

In this section, the characteristics of the sensors installed on the vehicles are discussed. The sensing architecture comprises different sensors, which primarily include Light Detection and Ranging (LiDAR), Radio Detection And Ranging (RADAR), cameras, and Global Navigation Satellite Systems (GNSS), used for the perception of the driving scene, mapping, localization, and navigation. The placement and coverage areas of sensors are shown in Figures 1. a-b.

LiDAR Sensor: A LiDAR sensor projects infrared laser pulses to detect and measure the distance to objects in the surrounding environment by measuring the time it takes for the light to travel back to the receiver from reflections. The typical LiDAR sensor covers an operating range of hundreds of meters with a 360-degree field of view, which can scan the area to generate 3D information about the environment. However, the performance of the sensor can be impacted by adverse weather conditions. Specifically, rain and fog attenuate the sensor visibility range and cause a reduction in target detection performance [9].

Given the capabilities of LiDARs, one VLP-32C Velodyne LiDAR and three VLP-16 Velodyne LiDAR are utilized for each shuttle in our CAV testbed. The VLP-32C sensor with 32 channels

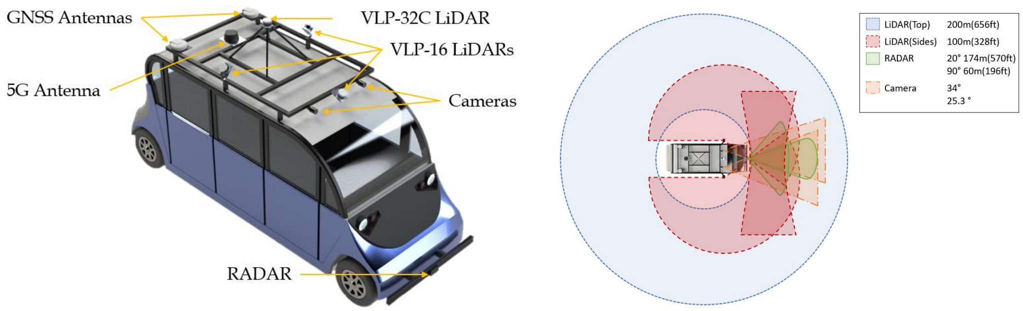
Table 1. Velodyne LiDAR VLP-32C and VLP-16 Specs

Specification	LIDAR	
	VLP-32C	VLP-16
Channels	32	16
Measurement Range (m)	200	100
Horizontal Field of View (°)	360	360
Vertical Field of View (°)	40 (-25to +15)	30 (-15 to +15)
Angular Resolution (°)	0.1 to 0.4	0.1 to 0.4
Rotation Rate (Hz)	5 to 20	5 to 20
Wavelength (nm)	903	903
Operating Temperature (°C)	-20 to +60	-10 to +60
Points Per Second (Points)	600,000/1,200,000	300,000/600,000

has an operating range of $200m$, covering a vertical field of view from -25 to 15 degrees and sweeping 360 degrees on the horizontal field of view. The VLP-16 sensors with 16 channels have an operating range of $100m$, covering a vertical field of view from -15 to 15 degrees and sweeping 360 degrees on the horizontal field of view. The rotation rate of these sensors can be adjusted from $5Hz$ to $20Hz$ for various purposes, including mapping and navigating. The technical specification of VLP-32C and VLP-16 Velodyne LiDAR are provided in Table 1.

The VLP-32C LiDAR generates 600,000 points per second with single return mode and 1,200,000 points per second with dual return mode, while VLP-16 LiDAR generates 300,000 points per second with single return mode and 600,000 points per second with dual return mode. These sensors generate two types of packets: data and position packets. The data packets consist of components expressed in spherical coordinates, which include the laser channel represented as the polar angle, the data point represented as the radial distance and reflectivity, and the azimuth angle. Additionally, the scanning sequence generated when all laser channels are fired and the time stamp marking on the first data point in the first scanning sequence is included in these packets. The position packets include a time stamp, point per second status, and the General Purpose Recommended Minimum Navigation Information (GPRMC). The LiDARs are synchronized with a GNSS-supplied Pulse-Per-Second (PPS) for time computing of scanned data points.

The VLP-32C VLP-16 LiDARs were installed in the roof rack as depicted in Figure 1.a. On one hand, the VLP-32C is mounted on the top-center of the vehicle to sense the surrounding environment in medium and long ranges. On the other hand, the VLP-16s are mounted on the top rack of the vehicle above the windshield, at right, left, and center, pointing downward with a 70-degree angle to cover the blind spot in a close-range area as shown in Figure 1.b.



(a) (b)

Figure 1. (a) Location of the installed sensors and (b) sensor coverage around the vehicle: VLP-32C (blue), VLP-16 (red), RADAR (green), and cameras (yellow).

Table 2. Radar Specification

Specification	Range	
	Long Range	Mid Range
Measurement Range (m)	174	60
Horizontal Field of View (°)	20 (-10 to +10)	90 (-45 to +45)
Vertical field of view (°)	4.75	4.75
Update Rate (ms)	50	50

Radar Sensor: The automotive industry has integrated radar sensors to provide safety and driving assistance features including emergency brake systems and adaptive cruise control. Radar emits Radio waves to detect their reflections and locate objects in the environment. Additionally, the capability of object trajectory tracking can be utilized for lateral and longitudinal vehicle following. One advantage of radar over other sensors relies on its ability to operate effectively in adverse weather conditions, including fog and rain [9].

A Delphi ESR RADAR was installed on each one of the shuttles in our CAV testbed. This sensor provides a 4.75-degree vertical field of view with two scanning beam angles, including a wide horizontal field of view at mid-range and a narrow horizontal field of view at long-range. The mid-range operates at a 90-degree angle with a range of 60 meters, while the long-range operates at a 20-degree angle with a range of 174 meters. The combination of these scanning range modes is beneficial as long-range scanning focuses on the area along the vehicle's trajectory, providing accurate range and velocity data. The mid-range is effective for detecting objects across the width of the vehicle, thereby eliminating unnecessary information from the environment. The technical specifications of the Delphi ESR RADAR are provided in Table 2.

The Delphi ESR RADAR was installed on the front bumper for frontal detections as depicted in Figure 1.a.

Camera Sensors: The cameras are critical components of autonomous vehicle technology providing visual information about the surrounding environment. This information is processed using computer vision algorithms to enable navigation and safety measures in various aspects, including but not limited to, lane detection and keeping, object identification, traffic sign and light detection, and potential hazard detection. While cameras allow visual information, they can encounter difficulties recognizing the environment in backlight and rainy conditions [10].

Two Lucid Triton IMX265 cameras are used on each shuttle in our CAV testbed to achieve reliable visualization of the environment. The cameras have 2048 x 1536 resolution, 35.4 frames per second, and 3.2 MP images. Employing different fields of view to maximize the collection of relevant data, one 12 mm focal length lens camera was used to capture wide areas (including traffic lights), while another 16 mm focal length lens camera was used to target narrower views. The technical specifications of the Lucid Triton IMX265 cameras are provided in Table 3.

Image distortion is an effect made on images, usually as the result of lens imperfections, which makes the images different from real-world appearance. To efficiently facilitate the sensor image data, the cameras in this testbed are calibrated with an OpenCV camera calibration tool. This procedure requires the camera to observe a planar pattern, an 8x6 checkerboard with 108mm squares, in different orientations and positions. The calibrated configuration is then stored as a calibration parameter to be used with the camera driver [11, 12].

The data packets of the cameras provide a timestamp, frame counter, offset X/Y, width/height, exposure time, gain, black level, line status, and sequencer set.

The Lucid Triton IMX265 cameras were mounted side by side above the windshield as depicted in Figure 1.a.

GNSS Sensor: A Novatel PwrPak7D-E2 is used as a secondary localization source, which receives GNSS signals (e.g., GPS, GLONASS, and Galileo) and Real-Time Kinematic (RTK) service provided by ground stations, allowing centimeter-level accuracy in positioning. In addition, the device is equipped with an Inertial Navigation System (INS) which employs accelerometers and

Table 3. Camera Specification

Specification	Range
Resolution	2048 x 1536 px, 3.2 MP
Framerate [FPS]	35.4
Sensor Size [mm]	8.9
Operating Temperature [°C]	-20 to 55

Table 4. GNSS Specifications

	Signal Tracking	Accuracy [cm]
GNSS	GPS	150
	GLONASS	150
	BeiDou	150
	Galileo	150
Assisted GNSS	SBAS	60
	DGPS	40
	TerraStar-L	40
	TerraStar-C PRO	2.5
	RTK	1

gyroscopes to measure the acceleration and rotation rate in three-dimensional space, respectively. By integrating GNSS and INS, the limitation of GNSS in low data rates can be compensated with the position generated by INS, improving the accuracy and reliability of the navigation. The specifications of the Novatel PwrPak7D-E2 are provided in Table 4.

The sensor data packet can provide latitude, longitude, velocity, date, and time. A fixed dual-antenna is integrated with the GNSS+INS to improve the orientation accuracy [13].

The Novatel PwrPak7D-E2 is installed at the back under the last passenger seat row, and it is connected to the two antennas on the top-back at each side of the vehicle as shown in Figure 1.a. To maintain accuracy in severe weather conditions, the shuttles are prepared for blending GNSS with radar and onboard sensors, using algorithms like the extended Kalman filter to ensure precise positioning, as suggested by [14].

4.1.2 Computing Architecture

CAVs are subjected to environmental conditions that can affect their driving performance, such as vibration and weather variation depending on the time of the day and seasons. To address these challenges, a computer with industrial-grade design and high-end graphics card support was chosen for the testbed onboard PCs.

The computing unit uses the Intel XEON E2278G 8th Generation Central Processing Unit (CPU). The CPU has 8 cores, a base frequency of 3.4 GHz, and the ability to boost up to 5 GHz. Its high-performance computing capability ensures the capacity to handle input data from various sensors with minimal latency and simultaneously process high-resource-consuming autonomy software.

In addition, an NVIDIA RTX A4000 video card, featuring 6,144 CUDA cores based on the

NVIDIA Ampere architecture, 192 third-generation Tensor Cores, 48 second-generation RT Cores, and 16 GB of GDDR6 memory with error-correction code (ECC), is utilized for managing intensive computation and graphics workloads. This upgrade significantly enhances the capability of the system to handle demanding applications. For processing power, the system incorporates an Intel i9-12900E CPU, known for its high performance with a 30M cache and speeds of up to 5.00 GHz, ensuring smooth and efficient operation across tasks.

A 1 TB NVMe M.2 2280 solid-state drive is employed, offering sequential read and write speeds of up to 3,500 MB/s and 3,300 MB/s, respectively. This storage capacity adequately accommodates the operating system, application software, and operational data.

The computing unit allows 10 Ethernet connectors, including 2 onboard ports and an additional 2 sets of 4-port gigabit Ethernet network cards. This configuration allows the shuttles to efficiently and accurately process data, analysis, decision-making, planning, and navigation.

4.1.3 Intra-vehicle Networking Architecture

One LiDAR and two cameras are connected to the computer using high-speed 100 Mbps Cat6 Ethernet cables to avoid data transfer bottlenecks due to the large volume of data generated by these sensors. The GNSS is also connected via Ethernet for reliability and low-latency communication. The radar, on the other hand, uses a Controller Area Network (CAN) to transmit data and is connected to the computer through a Kvaser CAN to USB converter. The CAN bus is a widely used standard in the automotive industry for reliable and efficient communications, as well as, providing the capability for multiple equipment expansion. The speed transferring rate of the CAN bus is at 1 Mbps. The Kvaser CAN to USB converter device enables the CANbus to read all data traffic and transfer it using USB cables. The proposed configuration for intra-vehicle communications is as shown in Figure 2.

4.1.4 Power Distribution Architecture

Electric vehicles (EVs) have emerged as a promising alternative to traditional internal combustion engine vehicles, and are increasingly being considered for use in the development of CAVs. An advantage of EVs is a quick response to the actuator control including steering, brake, and acceleration pedal. In addition, EVs are capable of efficiently providing a stable power source, which is well-suited to power the sensing and computing hardware required for autonomous driving.

The Gem E6 Polaris employed in this work is a fully electric vehicle that utilizes Absorbed Glass Mat (AGM) batteries installed in the trunk space to power the electric drivetrain, drive-by-wire system, and autonomous driving hardware.

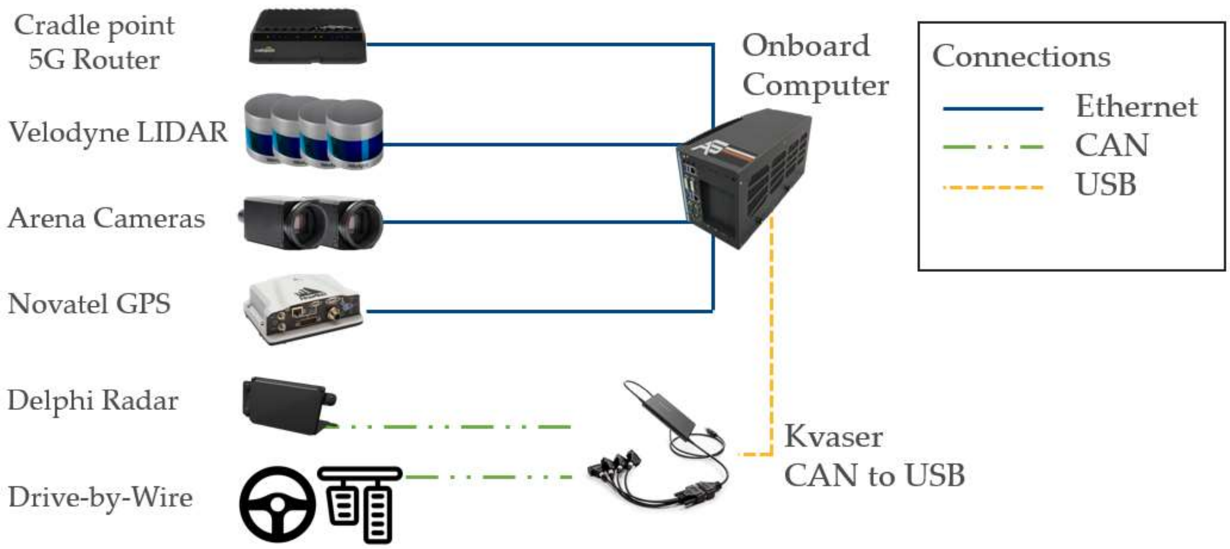


Figure 2. Intra-vehicle communication configuration for data transfer between sensors, actuators, and the computing unit.

To meet the required electric motor power rating, the GEM e6 Polaris is equipped with eight 12V AGM batteries connected in the form of two sets of series-connected batteries, wherein each set, four 12v batteries are connected in series, resulting in a total voltage of 48v.

The electric vehicle possesses a battery pack capable of delivering a driving range of up to 40 miles on a full charge. To initiate vehicle operation, the 12V auxiliary battery supplies power to the key switch, which in turn activates a relay to engage the main 48V battery. With the battery switch set to "ON," the entire system becomes operational upon ignition key activation. A DC-to-DC converter steps down the 48V source to 12V, catering to the voltage requirements of the hardware components. The drive-by-wire system incorporates an emergency stop button situated on the vehicle console, facilitating immediate power disconnection in exigent circumstances. A relay box enables independent power control for the sensors and onboard computer, while appropriately rated fuses safeguard these components from electrical overload. A detailed illustration of the power distribution scheme for the vehicle's components can be found in Figure 3.

4.1.5 Drive-by-Wire Architecture

The Platform Actuation and Control Module (PACMOD) is a drive-by-wire system built by AutonomousStuff that transfers CAN messages from the on-board computer into control signals that govern the movement of a vehicle, including acceleration, brake, steering wheel, turning lights, and gear shifting. The system incorporates a closed-loop control mechanism that returns feedback messages to regulate the operation of the system. The PACMOD is linked to the Electronic Control Unit (ECU) of the GEM e6 Polaris to regulate the movement of the vehicle, being able to automatically

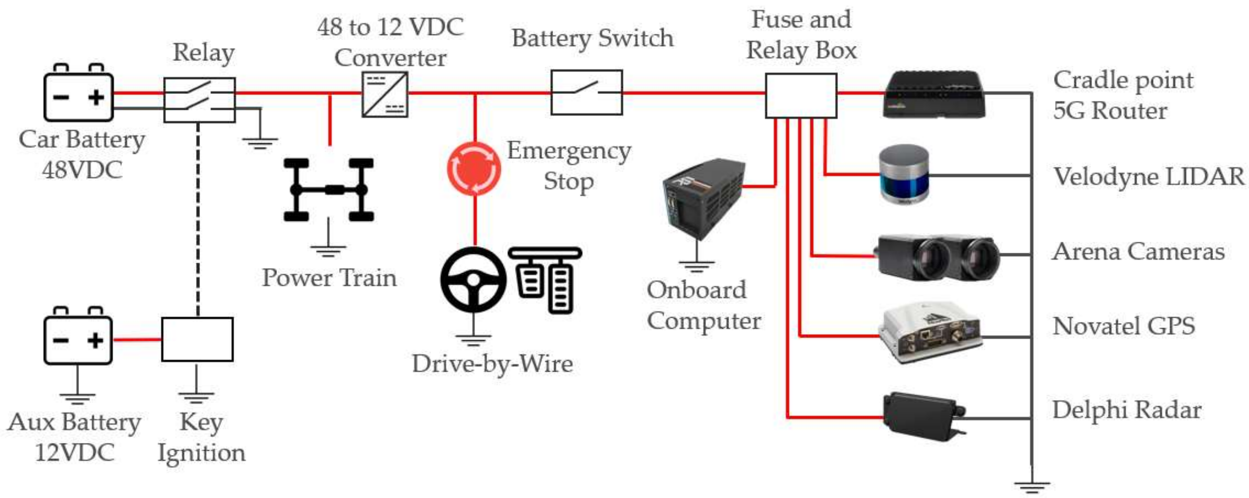


Figure 3. Power distribution wiring diagram of the vehicles.

disengage when the driver applies the pedal or moves the steering wheel. For brake control, an additional motor was installed, having a cable attaching the motor pulley to the brake pedal. The brake pedal is also fitted with a switch that allows the driver to disengage the drive-by-wire system when necessary. Similarly, the steering wheel is controlled using a power steering motor that receives a control signal from the PACMOD turning the steering wheel into a desired direction, while sensing whether any torque is applied to the steering wheel. The control signals are scaled linearly based on the minimum and maximum ranges applicable to acceleration percentage, brake pedal position, and steering wheel position. Subsequently, these scaled signals are transmitted to the ECU to be executed through the actuators for the braking and steering.

4.2 The Software Architecture of Autonomous Vehicles

The software architecture of the AVs in our CAV testbed offers a tangible representation of the software workflow and organization. The Robot Operating System 2 (ROS2) was employed as a flexible framework for developing the software for the AVs.

ROS2 is an open-source software platform that offers a robust and modular framework for the development of robotics applications [15]. The communication architecture of ROS2 organizes applications as *nodes* that can execute specific tasks and communicate with other nodes. The communicated information includes *topics*, *services*, and *actions*. Nodes use the *publish-subscribe* approach to communicate over asynchronous message-passing frameworks, known as topics. On the other hand, nodes use services to adopt the request-response style approach, which enables easy data association between a request and its response pair. Lastly, actions are goal-oriented and asynchronous communication interfaces that support requests, responses, periodic feedback, and cancellations. The architecture of ROS2, particularly the ROS2 MiddleWare (RMW), is designed with abstraction layers that facilitate the interchangeability of modular components. The RMW serves as an interface for communication by employing a Data Distribution Service (DDS) framework that can be implemented with various other middleware [16].

The developed autonomous vehicles are equipped with Autoware, an open-source autonomous driving software stack based on the ROS2 framework. Autoware encompasses a comprehensive suite of self-driving modules including localization, detection, prediction, planning, and control [17]. The software architecture of the developed autonomous vehicles is shown in Figure 4.

The sensors (described in Section 4.1) generate raw data inputs containing information about the driving environment. This information is then passed through the pre-preprocessing within the sensing layer. Subsequently, the perception and localization layers process the information to gain a comprehensive understanding of the surrounding environment and accurately determine the vehicle status. Next, the data obtained from the perception layer is further analyzed by the planning layer to devise an optimal path from the current position to the specified destination. Finally, the planned actions are tracked by the control layer which determines the desired actuation by the vehicle actuators and sends the commands to the PACMOD unit. Each one of the software layers of the autonomy stack is explained next.

4.2.1 Perception System

CAVs require a sufficiently in-depth understanding and analysis of the environment to make decisions about navigation. The perception layer plays a role by transforming data from the sensors layer into cognitive information. Core to the perception layer is the object detection module, which includes detection, tracking, and prediction algorithms.

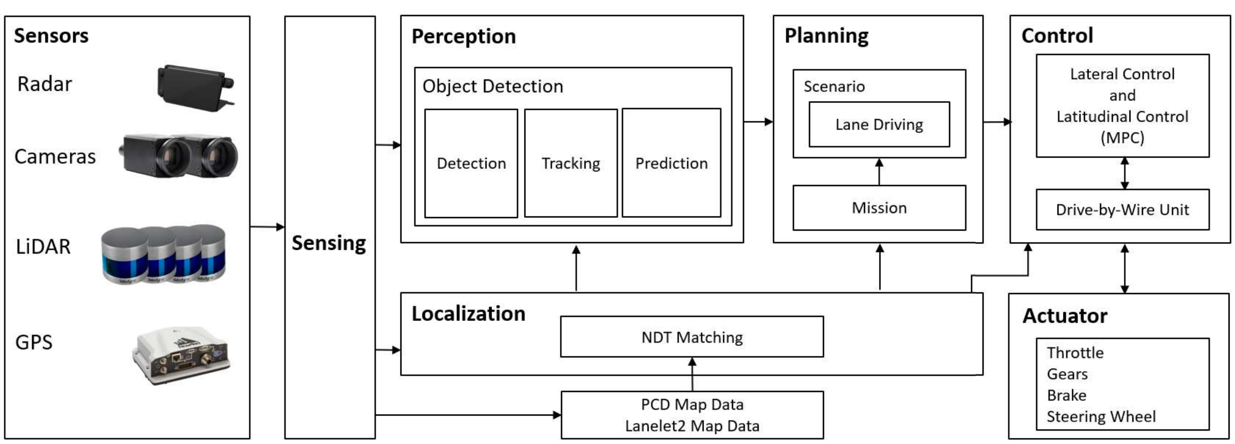


Figure 4. The autonomy software architecture of the shuttles of the developed CAV testbed.

The perception layer processes information gathered from the camera, LiDAR, and RADAR. This data is then subjected to deep learning techniques, enabling the system to identify and recognize different entities within the perception range.

Camera-based object detection utilizes 2D image sources that are partitioned into grids and subsequently inputted into Artificial Intelligence (AI) models for feature extraction, such as the You Only Look Once (YOLOv5) [18] and Single Shot Detector (SDD) algorithms [19]. These AI models can be used for detecting lane boundaries [20], traffic signs [21], pedestrians [22], or other objects in the driving scene. On the other hand, LiDAR-based object detection operates by analyzing reflected point clouds, which accurately depict the shape and location of objects within a 3D reference frame. Meanwhile, radar-based object detection provides information on the relative speed and spatial positioning of objects within a spherical coordinate system. Fusion of AI models [22] and sensing modes, e.g., camera and LiDAR [23] can enhance the object detection performance.

4.2.2 Localization System

The localization layer determines the position and orientation of the vehicle within the driving environment. The developed testbed initially employs GNSS-based localization, being capable of enabling Real-Time Kinematic (RTK) services to improve the position accuracy to up to 2 cm. The position initialization (or correction) can be triggered by the user using the ROS graphical interface (i.e., RVIZ). The localization algorithm then utilizes matching between 3D LiDAR scans and a Point Cloud Data (PCD) map available, employing the Normal Distribution Transform (NDT), which transforms the LiDAR point cloud into a piecewise continuous and differentiable probability density [24]. The map is built in advance using a real-time LiDAR-inertial odometry package [25]. The fused data between GNSS and LiDAR enables the vehicles to drive in real-world scenarios where GNSS is deprived, as the vehicle can maintain its position in the relative localization from the reference PCD map [26].

To create a highly accurate 3D point cloud map, a VLP-32C LiDAR is used to collect data on the environment surrounding the vehicle. The map presents not only the point cloud position but also the intensity level of the point cloud reflectivity. The distinct reflectivity between the lane markings and road pavement is exploited to create a PCD map for navigation. Beyond localization purposes, LiDAR sensors are used for sensing point cloud clusters of various entities, such as vehicles, pedestrians, road signs, and obstacles. By recognizing these entities from the uniform shape of the point-cloud cluster, the system facilitates information for making informed decisions to ensure safe navigation.

4.2.3 Planning System

The planning layer generates the trajectory for the vehicle to follow. The global path is converted into a sequence of trajectory points from the current position to a specified destination. These trajectory points can be modified based on the inputs from object detection and recognition of the perception system under different scenarios, such as proceeding to the stop line or collision avoidance. The local trajectory is generated considering the lanes obtained from the Lanelet2 map, a high-definition map framework for automated driving [27]. The implementation of Lanelet2 not only allows composing complex road situations but also incorporates tactical information for maneuver generation, such as rule-compliant traversal of intersections with traffic lights. When a PCD map is not available, the fusion of GPS and camera information can be used for localization and path planning [20].

4.2.4 Control System

The control layer enables the CAV testbeds to safely follow a trajectory generated by the planning layer to reach a specified destination. The trajectory follower module designed by Autoware generates both lateral and longitudinal control commands tracking the trajectory reference, considering the parameters of the vehicle while incorporating safety measures.

We use either Pure Pursuit or Model Predictive Control (MPC) methods to regulate the lateral movement of the vehicle.

In the Pure Pursuit method, the reference trajectory is divided into discrete waypoints, identifying the nearest neighbor waypoint to the vehicle position. By interpolating between these waypoints, the algorithm uses a look-ahead distance [28]. Finally, the steering angle command for trajectory tracking is defined considering the current radius of curvature of the path based on the target waypoint computed previously.

In the MPC method, the control of the vehicle is formulated as an optimization problem, the solution of which optimizes future states considering the kinematic or dynamic model of the vehicle [29]. More specifically, the current states of the vehicle including the position, velocity, and steering wheel angle, are employed to predict the future behavior over a finite time horizon.

In longitudinal control, a Proportional Integral Derivative (PID) approach is used to regulate the longitudinal velocity. This is achieved by calculating the compensation that is needed to reduce the errors between the current and desired velocity.

4.3 Connectivity of the CAV Testbed

The integration of wireless technology for vehicle connectivity, generally known as Vehicle-To-Everything (V2X) communication, allows vehicles to exchange real-time information with various elements of the Intelligent Transportation System (ITS). This includes but is not limited to, Vehicle-To-Vehicle (V2V), Vehicle-To-Infrastructure (V2I), Vehicle-To-Cloud (V2C), and Vehicle-To-Pedestrian (V2P). In this section, the connectivity framework for 5G cellular network communication developed for our CAV testbed is discussed.

4.3.1 Communication Technologies for CAVs

Over the past two decades, the Federal Communications Commission (FCC) has allocated Dedicated Short-Range Communications (DSRC) using a 75 MHz spectrum in the 5.9 GHz band on intelligent transportation programs [30, 31].

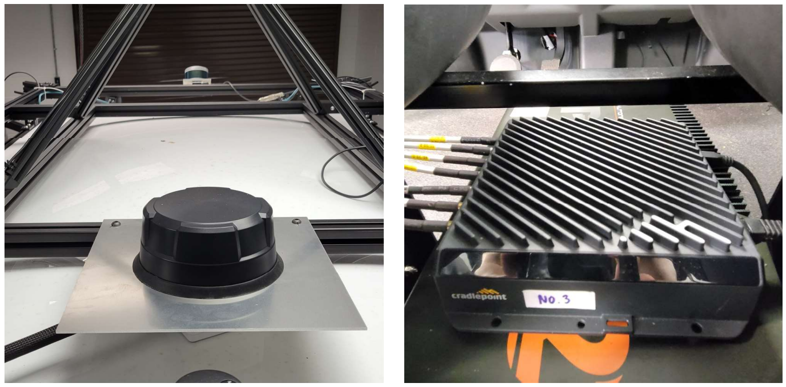
DSRC systems allow vehicles and roadside elements to wirelessly exchange information to elevate safety and efficiency while alerting drivers of potential collisions. In recent years, the suitability of DSRC-based communication technology has been challenged due to its limitations in providing sufficient data packets. In [32], a simulation framework was created to accurately simulate two widely recognized simulation platforms (NS3 for communication modeling and SUMO for mobility modeling), thereby capturing the impact of communication on safety applications. In the high packet traffic load and vehicle density experiment scenario, compared to DSRC, 5G resulted in fewer accidents owing to its capability to handle high background traffic rates.

A transition to Cellular-Vehicle-to-Things (C-V2X) based technology has been proposed to substitute the DSRC technology. C-V2X experiments have been conducted along with the development of cellular communication, including 4G and 5G, to examine their efficacy over the DSRC technology. In [33], DSRC and 4G-LTE systems were evaluated to perform V2V communication under several typical weather and road conditions. Even though DSRC provides lower latency compared to 4G-LTE, the latter provides better and more stable performance in terms of package delivery ratio.

Given that cellular communications have proven to outperform DSRC in managing high packet traffic loads, especially suitable for V2X applications, the decision was made to employ a 5G cellular system for equipping the CAV testbed with robust vehicular communication capabilities.

4.3.2 V2X Hardware Architecture

CAVs require a robust antenna and router to maintain connectivity for high data transfer rates over a 5G cellular network. In our CAV testbed, a MAKO 5G Dome Antenna was installed above the vehicle, as shown in Figure 5.a. The antenna was equipped with a 4×4 multi-input-multi-output,



(a) *(b)*

Figure 5. Installation of (a) 5G Antenna and (b) 5G Router.

receiving cellular frequency bands from 617-960 MHz (low-band) to 1710-6000 MHz (mid-band), capable of also receiving GNSS signals and providing dual-band Wi-Fi. Additionally, a CradlePoint R1900 router was installed, as shown in Figure 5.b. This router supports connections over 5G in the low- and mid-bands, 4G, 4 Gigabit LAN with 60 W PoE ports, 2.4 GHz and 5 GHz 802.11ax Wi-Fi 6, and GNSS. The antenna and router were integrated into the computing architecture described in Section 4.1.2, using an Ethernet cable and powered with 12 VDC from the controlled relay box.

4.3.3 V2X Software Architecture

The ROS2 framework provides high-performance real-time communication and control, which are critical for enabling the development of CAV systems with strict timing requirements. The ROS2 application layer is built on top of a DDS middleware layer that provides communication infrastructure. DDS is a standardized middleware protocol that facilitates efficient and scalable data distribution systems. Employing a publish-subscribe communication model, DDS enables data publishers to disseminate information and data subscribers to consume specific data types or topics and receive fitting information. DDS addresses the demanding requirements of real-time and mission-critical applications by decoupling publishers and subscribers, thereby ensuring flexible and scalable data communication [34].

The Eprosima Fast RTPS (Real Time Publish-Subscribe) vendor was implemented as a DDS service to enable connectivity between the vehicles in our CAV testbed. The RTPS architecture that provides access to all participating applications (i.e., domain participants) is shown in Figure 6.

The applications can become publishers to contribute information, or subscribers to access portions of the data space shared by publishers. Each time a publisher posts new data, the middleware propagates the information to all interested subscribers.

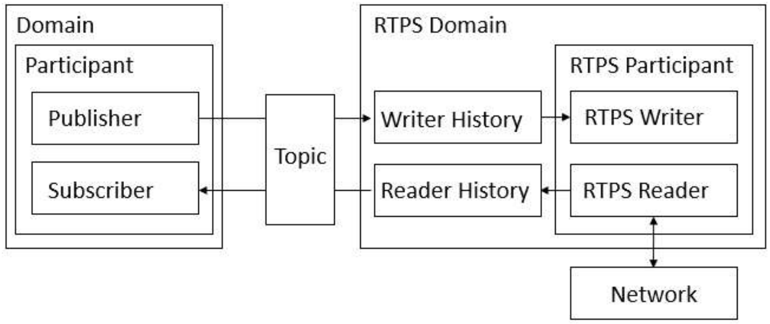
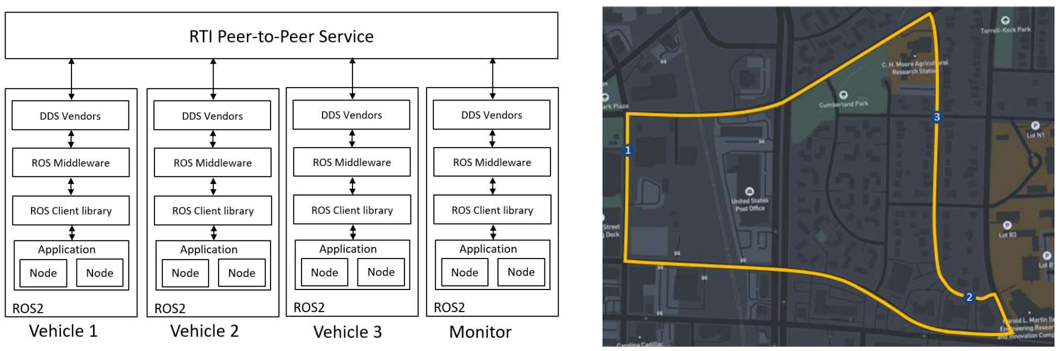


Figure 6. The implemented RTPS Architecture.



(a) (b)

Figure 7. (a) The V2V communications Architecture. The Application layer contains the modules shown in Figure 4 and (b) the Aggie Auto Web Application.

To establish connectivity between vehicles, Husarnet was implemented as a discovery service to provide a connection between RTPS domains. Husarnet is a global peer-to-peer VPN layer that supports multi-machine connectivity using UDP and IPv6 protocols. This service allows ROS2 computers to access the DDS data traffic. Specifically, using this service, vehicles, and clouds can be connected by subscribing to and publishing data on other topics.

V2V Communications V2V communication enables CAVs to share information about the position, velocity, and perceived entities in the surroundings. As mentioned in the previous section, the software stack of the vehicles is developed based on the ROS2 architecture which provides a communication framework for nodes to transfer messages to other nodes through topics. Applications, including sensor drivers and software stack modules, run on nodes and continuously publish-subscribe messages. This communication feature is exploited by collecting data from other nodes and broadcasting it to other vehicles in the network using the 5G router depicted in Figure 7.a.

The 5G bandwidth supports high-speed and low-latency data exchanges between vehicles. However, 5G high-bandwidth efficiency encounters limited coverage areas, especially in suburban areas, and attenuates over-obstructed buildings. Considering the data rate generated from the sensors (i.e., GNSS), the communication output (i.e., the position data messages) is downsampled and published at a limited frequency. This communication node subscribes to messages published by the same node from other vehicles to receive data from other vehicles including their location and speed.

V2C Communications Cloud servers offer a convenient means of accessing and monitoring vehicles through web services. In particular, to enable the streaming of geolocation through the cloud, a developed communication node leverages the publish-subscribe pattern. This node retrieves messages from the GNSS node and broadcasts the position data of all vehicles. The streamed data can be visualized on an interactive map. This cloud-based communication enables several features, including:

- Remote Monitoring and Control: Cloud servers provide access to operators to monitor the status of the vehicles over web services, and if needed, send control commands (e.g., new destinations) remotely. In this work, a communication node was developed to exploit the publish-subscribe pattern, which accesses the messages from the sensors (e.g., GNSS), and broadcasts the status of the vehicles to the cloud server.
- Software Web Application for End Users: The web application developed allows end users to track the availability and location of vehicles. By utilizing the vehicle GNSS location data from the cloud server, the web application displays real-time locations of currently operating vehicles along the operational route on the map shown in Figure 7.b. This information can be also used for booking rides enabling CAV-based ride-sharing.

5 Preliminary Deployment Results

5.1 Testing the Components of the AVs

The developed CAV testbed undergoes testing at the N.C. A&T University's Gateway Research Park North, located in Greensboro, NC. This test track serves as a controlled environment to evaluate and validate emerging CAV technologies before their safe deployment on public roads. The vehicles in our CAV testbed and the test track are depicted in Figures 8.a-b, respectively.

The test track features a 2-mile loop track that offers a diverse range of road characteristics, which are designed to simulate driving challenges and scenarios that CAVs may encounter during real-world driving. The track incorporates essential features such as a roundabout, intersections, pedestrian crossings, road signs, and road markings, that replicate common rural, suburban, and urban driving situations. In particular, the test track includes elements that mimic a rural terrain

(a) (b)

Figure 8. (a) Developed CAV Testbed and (b) The rural AV test track in N.C. A&T University's Gateway Research Park North map.

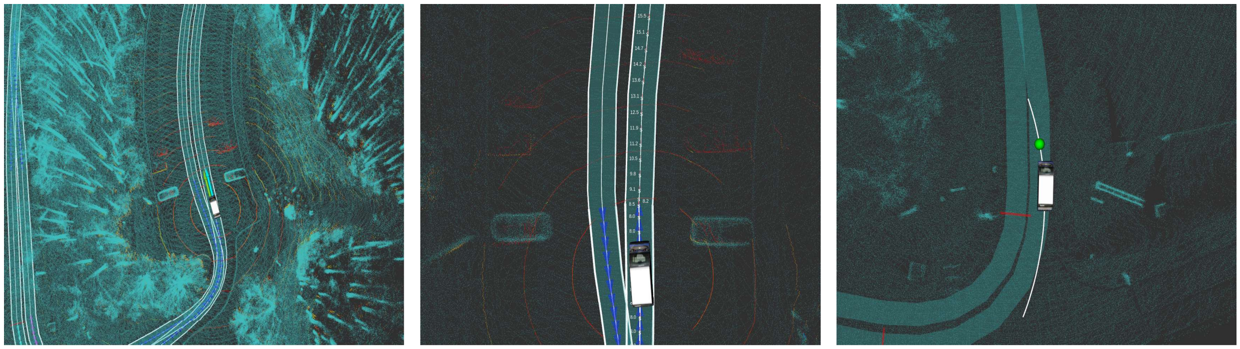
(a) *(b)* *(c)*

Figure 9. Perception of the driving scene using different sensors including (a) LiDAR, (b) Camera, and (c) Radar.

environment, including variations in elevation, turns, and dense vegetation. The developed CAV deployment is verified by testing the components of the CAV software stack shown in Figure 4, including perception, localization, planning, control, and communication systems.

5.1.1 Perception System

The LiDAR is one of the major sensors that is used in our AC testbed for object detection within its environment, as depicted in Figure 9a, which showcases the analysis of point cloud clusters to identify objects. In real-time, the system visually presents the detected objects along with their respective 3D models and pertinent information such as identity, distance, velocity, and tracking 1D. This object detection functionality is crucial for implementing obstacle collision avoidance; upon detecting a potential collision, the vehicle's system initiates appropriate actions such as deceleration or coming to a complete stop to avert the collision. In addition, camera sensor is used for object detection based on 2D images, as illustrated in Figure 9b, where objects are classified and bounding boxes are provided. The radar sensor output, depicted in Figure 9c, provides additional



(a) *(b)* *(c)*

Figure 10. Vehicles' localization, planning, and control: (a) Localization involves the utilization of GNSS (Green arrow) alongside NDT matching over the PCD map (Blue arrow). (b) Planning is facilitated through the depiction of waypoints for trajectory tracking, represented as white dots positioned at the center of the Lanelet. (c) Control is demonstrated by the implementation of pure pursuit trajectory tracking, which follows a look-ahead reference path highlighted in green.

information about the environment, including distances from different objects surrounding the vehicle.

5.1.2 Localization System

The position of the vehicle is initially obtained through the GNSS information, whereby the accuracy of the GNSS is contingent upon the presence and strength of the constellation and ground base signals. To precisely establish the position, an alternative method can be used by manually defining the initial coordinates through the user interface and aligning the LiDAR point cloud with a preexisting PCD map.

The result of the localization process is illustrated in Figure 10a, in which the positions obtained through GNSS and LiDAR methods are demonstrated, where the GNSS localization is depicted by the green arrow, and the NDT localization is represented by the blue arrow.

5.1.3 Planning System

The deployment route starts from the road located in front of the laboratory building, traverses the roundabout at the end of the track, and ultimately returns to the starting point following the same path as shown in Figure 8.b. By facilitating the Lanelet2 map, a trajectory is generated from the initial vehicle position to the designated destination at the roundabout.

The result of the planning process is shown in Figure 10b. This trajectory is broken down into multiple waypoints, encompassing both velocity and position information. The velocity profile is adjusted, taking into account factors such as sharp turns and narrow roads encountered along the route. Furthermore, the presence of stop lines, marked in the Lanelet2 map, affects the velocity profile, necessitating the vehicle to decelerate and come to a complete stop before the intersection.

5.1.4 Control System

The process of waypoint tracking using the Pure Pursuit and PID control methods is depicted in Figure 10c, in which the green point represents the look-ahead distance that determines the point the vehicle aims to reach desired waypoints. This look-ahead distance is used to generate the radius of curvature for lateral control of the vehicle as represented in the white line in Figure 10c.

5.2 Testing the Connectivity of AVs

To validate V2V connectivity, the CAV testbed is deployed on the test facility. The testbed establishes a communication node that facilitates the exchange of downsampling GNSS messages with other vehicles in the network. The transmission of crucial GNSS data, encompassing latitude and longitude information, is accomplished through a 5G bandwidth at 2 samples per second. These concise data messages are promptly received by other vehicles within the network without any loss of information. Moreover, an evaluation of communication efficiency is conducted by transmitting image data from a camera node. The camera generates image data at a rate of 200 MB/s, surpassing the transmission capacity of a 5G bandwidth, which is limited to 30 MB/s. To manage the data transfer and guarantee timely message delivery, a connectivity node is implemented to downsample the image messages to 4 MB/s. A visual representation and comparison of the communication data are shown in Figure 11.

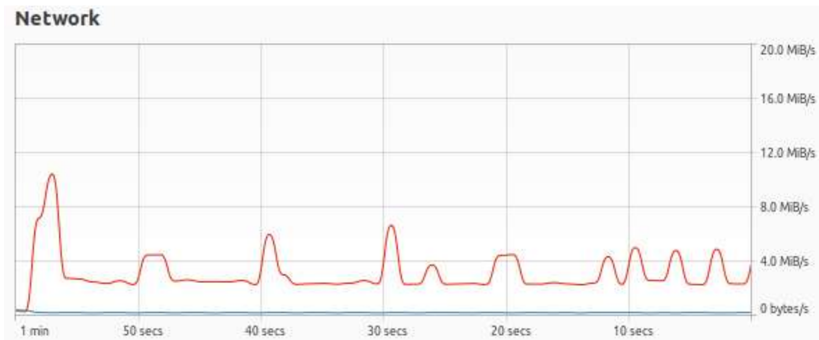
6 Conclusion

This project demonstrated the development of a CAV testbed, an examination of the components of AVs, and their 5G connectivity capabilities. The research detailed the required hardware on Gem E6 Polaris electric vehicles and software development utilizing the Autoware and ROS2 frameworks, creating an integrated architecture helping the validation and verification of various automated driving technology concepts. The operation of the developed CAV testbed was successfully executed in a controlled environment. Moreover, the performance of the commercial 5G communication systems demonstrated the capability of transferring critical data between the vehicles. The developed CAV testbed serves as a dependable platform for connectivity and automated driving research, offering valuable insights into V2V and V2C connectivity through practical use cases.

7 Future Research Needs

Future work include enhancing the hardware and software architectures of the vehicles to increase their computational power, enhance perception accuracy and robustness, and extend connectivity

(a)



(b)

Figure 11. Transferring (a) raw image data and (b) downsampled image data.

capabilities to enable V2I and V2P communication for comprehensive V2X solutions. This will significantly contribute to the ongoing advancement of autonomous driving technology and lays a solid foundation for future developments in CAV technologies.

References

[1] M. Bisht, J. Abbott, and A. Gaffar, "Social dilemma of autonomous cars a critical analysis," in *2017 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computed, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*. IEEE, 2017, pp. 1–3.

[2] I. W. Damaj, J. K. Yousafzai, and H. T. Mouftah, "Future trends in connected and autonomous vehicles: Enabling communications and processing technologies," *IEEE Access*, vol. 10, pp. 42 334–42 345, 2022.

[3] W. Huang, K. Wang, Y. Lv, and F. Zhu, "Autonomous vehicles testing methods review," in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2016, pp. 163–168.

[4] Ericsson, "5g security - enabling a trustworthy 5g system," https://www.ericsson.com/en/reports-and-papers/white-papers/5g-security---enabling-a-trustworthy-5g-system, 2020, accessed: YYYY-MM-DD.

[5] D. R. D'Costa and D. R. Abbas, "5g enabled mobile edge computing security for autonomous vehicles," *arXiv preprint arXiv:2202.00005*, 2022.

[6] Z. Wang, H. Wei, J. Wang, X. Zeng, and Y. Chang, "Security issues and solutions for connected and autonomous vehicles in a sustainable city: A survey," *Sustainability*, vol. 14, no. 19, p. 12409, 2022.

[7] E. Zhang and N. Masoud, "V2xsim: A v2x simulator for connected and automated vehicle environment simulation," in *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2020, pp. 1–6.

[8] R. K. Bhadani, J. Sprinkle, and M. Bunting, "The cat vehicle testbed: A simulator with hardware in the loop for autonomous vehicle applications," *arXiv preprint arXiv:1804.04347*, 2018.

[9] M. Kutila, P. Pyykönen, H. Holzhüter, M. Colomb, and P. Duthon, "Automotive lidar performance verification in fog and rain," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, 2018, pp. 1695–1701.

[10] M.-H. Ha, C.-H. Kim, and T.-H. Park, "Object recognition for autonomous driving in adverse weather condition using polarized camera," in *2022 10th International Conference on Control, Mechatronics and Automation (ICCMA)*, 2022, pp. 42–46.

[11] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, pp. 1330–1334, 2000.

[12] opencv dev team, "Camera calibration and 3d reconstruction," Dec 2019. [Online]. Available: $https://docs.opencv.org/2.4/modules/calib3d/doc$

[13] F. Aghili and A. Salerno, "Driftless 3-d attitude determination and positioning of mobile robots by integration of imu with two rtk gpss," *IEEE/ASME Transactions on Mechatronics*, vol. 18, no. 1, pp. 21–31, 2013.

[14] Y. Zhang, A. Carballo, H. Yang, and K. Takeda, "Perception and sensing for autonomous vehicles under adverse weather conditions: A survey," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 196, pp. 146–177, 2023.

[15] S. Macenski, T. Foote, B. Gerkey, C. Lalancette, and W. Woodall, "Robot operating system 2: Design, architecture, and uses in the wild," *Science Robotics*, vol. 7, no. 66, p. eabm6074, 2022. [Online]. Available: https://www.science.org/doi/abs/10.1126/scirobotics.abm6074

[16] G. Pardo-Castellote, "Omg data-distribution service: architectural overview," in *23rd International Conference on Distributed Computing Systems Workshops, 2003. Proceedings.*, 2003, pp. 200–206.

[17] S. Kato, S. Tokunaga, Y. Maruyama, S. Maeda, M. Hirabayashi, Y. Kitsukawa, A. Monrroy, T. Ando, Y. Fujii, and T. Azumi, "Autoware on board: Enabling autonomous vehicles with embedded systems," in *2018 ACM/IEEE 9th International Conference on Cyber-Physical Systems (ICCPS)*, 2018, pp. 287–296.

[18] G. Jocher, A. Chaurasia, A. Stoken, J. Borovec, NanoCode012, Y. Kwon, K. Michael, TaoXie, J. Fang, Imyhxy, and et al., "Ultralytics/yolov5: V7.0 - yolov5 sota realtime instance segmentation," Nov 2022. [Online]. Available: https://doi.org/10.5281/zenodo.3908559

[19] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *Computer Vision – ECCV 2016*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds. Cham: Springer International Publishing, 2016, pp. 21–37.

[20] T. Getahun and A. Karimoddini, "An integrated vision-based perception and control for lane keeping of autonomous vehicles," *IEEE Transactions on Intelligent Transportation Systems*, 2024.

[21] E. Sarku, J. Steele, T. Ruffin, B. Gokaraju, and A. Karimodini, "Reducing data costs-transfer learning based traffic sign classification approach," in *SoutheastCon 2021*. IEEE, 2021, pp. 1–5.

[22] M. M. Islam, A. Karimoddini *et al.*, "Pedestrian detection for autonomous cars: inference fusion of deep neural networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 12, pp. 23 358–23 368, 2022.

[23] M. M. Islam, A. A. R. Newaz, and A. Karimoddini, "A pedestrian detection and tracking framework for autonomous cars: Efficient fusion of camera and lidar data," in *2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, 2021, pp. 1287–1292.

[24] S. Azam, F. Munir, A. M. Sheri, J. Kim, and M. Jeon, "System, design and experimental validation of autonomous vehicle in an unconstrained environment," *Sensors*, vol. 20, no. 21, 2020. [Online]. Available: https://www.mdpi.com/1424-8220/20/21/5999

[25] T. Shan, B. Englot, D. Meyers, W. Wang, C. Ratti, and R. Daniela, "Lio-sam: Tightly-coupled lidar inertial odometry via smoothing and mapping," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 5135–5142.

[26] J. Matute, M. Rodriguez-Arozamena, J. Perez, and A. Karimoddini, "Sensor fusion-based localization framework for autonomous vehicles in rural forested environments," in *2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2023, pp. 1007–1013.

[27] F. Poggenhans, J.-H. Pauls, J. Janosovits, S. Orf, M. Naumann, F. Kuhnt, and M. Mayr, "Lanelet2: A high-definition map framework for the future of automated driving," in *2018 21st international conference on intelligent transportation systems (ITSC)*. IEEE, 2018, pp. 1672–1679.

[28] J. Ahn, S. Shin, M. Kim, and J. Park, "Accurate path tracking by adjusting look-ahead point in pure pursuit method," *International journal of automotive technology*, vol. 22, pp. 119–129, 2021.

[29] S. Yu, S. Hirche, Y. Huang, and et al., "Model predictive control for autonomous ground vehicles: a review," *Autonomous Intelligent Systems*, vol. 1, no. 4, 2021. [Online]. Available: https://doi.org/10.1007/s43684-021-00005-z

[30] Federal Communications Commission, "Intelligent transportation services," Engineering and Technology, Washington, DC, USA, FCC-99-305, 1999.

[31] ——, "FCC Modernizes 5.9 GHz Band to Improve Wi-Fi and Automotive Safety," Engineering and Technology, Washington, DC, USA, FCC-20-164, 2020.

[32] A. Elbery, S. Sorour, H. Hassanein, A. B. Sediq, and H. Abou-zeid, "To dsrc or 5g? a safety analysis for connected and autonomous vehicles," in *2021 IEEE Global Communications Conference (GLOBECOM)*, 2021, pp. 1–6.

[33] S. Liu, W. Xiang, and M. X. Punithan, "An empirical study on performance of dsrc and lte-4g for vehicular communications," in *2018 IEEE 88th Vehicular Technology Conference (VTC-Fall)*, 2018, pp. 1–5.

[34] M.-W. Park, S.-W. Lee, and W.-Y. Han, "Development of lateral control system for autonomous vehicle based on adaptive pure pursuit algorithm," in *2014 14th International Conference on Control, Automation and Systems (ICCAS 2014)*, 2014, pp. 1443–1447.