Development of an Autonomous Campus Vehicle Platooning System

Final Report for MAE 4620

Presented to the Faculty of the School of Engineering and Applied Science
University of Virginia · Charlottesville, Virginia

2023-2024 Autonomous Campus Vehicle Team:
Santiago Merida
Carolyn Pitorak
Ben Tharakan
Rachel Thirumalai
Riley Tufts
Victoria Vettoretti

May 10, 2024

On our honor as University students, we have neither given nor received unauthorized aid on this assignment as defined by the Honor Guidelines for Thesis-Related Assignments.

Advisor:
Tomonari Furukawa, Department of Mechanical and Aerospace Engineering

Introduction

Walking around the campus of the University of Virginia (UVA), it does not take long for one to notice the lack of accessible travel for students and staff with limited mobility. For some, commutes across campus can be over 2 miles long just to go from the parking lot to the classroom. For elderly professors and students with limited mobility, these routes can create severe barriers to education. UVA and other universities do employ transportation systems through the use of buses in order to provide some aid in this department, but it does not specifically target the needs of accessible travel from all points of campus. The use of campus vehicles could heavily decrease the burden placed upon faculty members and students who face these concerns. Specifically, a campus vehicle that can transport a large number of passengers and can bring them to the specific location desired. The UVA Autonomous Campus Vehicle (ACV) was designed for such purposes and works by using a platooning algorithm to allow additional carts to follow a human-driven lead cart. This report details the pre-existing technology along with the creation and implementation of the autonomous campus vehicle at the University of Virginia in the 2023-2024 academic year.

Literature Review

Campus Vehicles

College campuses provide a unique environment for the application of platooning systems. Due to the high variety of transportation modes, including bikes, cars, buses, and foot traffic, college towns are distinctly affected by traffic congestion and other transportation-related incidents (Haines et al., 1974). These issues can result in the stagnation of university functions as there are often a limited number of alternate paths to popular destinations. In addition to the previously discussed benefits of platooning in broader society, platooning could help to alleviate these transportation issues on college campuses. In one platooning system developed by researchers at Berkeley, the direct traffic congestion benefits were apparent. During testing of their system, there was a significant increase in intersection throughput using their platooning algorithm (Smith et al., 2020). With the bottlenecking caused by traffic signals, increased throughput through these obstacles will greatly relieve congestion on college campus roadways.

Platooning

In 2019 alone, there were 12.15 million vehicles involved in car accidents in the United States (Carlier, 2022). This, coupled with the emissions these vehicles release into the environment, shows how inefficient the current driving system is in the United States. While technology is rapidly changing and car companies are switching to electric and autonomous vehicles, society has yet to deal with some of the problems that create unsafe environments on the roads. Platooning systems have been researched since the 1980s in order to decrease fuel emissions and increase safety on roadways, but they still need to gain popularity in the transportation industry (Bhoopalam, Agatz, Zuidwijk, 2018). These systems work by using advanced sensing and communication between vehicles that are following each other (Puplaka, 2016). The safety among vehicles increases due to the awareness of where each vehicle is located around them, and the fuel emissions decrease because vehicles can drive closer to one another, decreasing their drag coefficient (ALEC, 2017).

In the past 30 years, a lot of research has gone into different platooning systems as they can save money on fuel emissions and wages for drivers as well as provide a safer driving system on roadways. Most of these technologies utilize the same basic structure, using sensors to relay information between vehicle-to-vehicle communication (V2V) (Shaver and Droege, 2021).

The difference between the research done at companies and universities lies in the algorithms they develop that take in sensor data and control the distance between vehicles (Volpe Center, 2017). Some of the groups that have ongoing research on this are Peloton Technology, Daimler, Isuzu and Hino, Volvo Trucks, and Partners for Advanced Transportation Technology (PATH) (Bhoopalam, Agatz, Zuidwijk, 2018). Each has their own algorithm of allowing the vehicles to platoon, and some groups have even been able to implement their system on public roads in other countries.

Platooning provides several advantages, notably in enhancing safety through advanced sensing and communication that increases awareness of vehicle positions and velocities in real-time. This significantly reduces the risk of accidents and contributes to a safer overall driving environment. With controlled vehicle spacing, a platooning system also lowers fuel emissions, aligning with the industry's aim for more sustainable transportation practices (Staudacher, 2024). However, platooning systems also come with several limitations, including the requirement of a human driver which does not enable this system to be fully autonomous. These systems are also dangerous because it is not guaranteed that the cars around the system are operating on the same automated system; therefore one cannot control the reactions or impulses of human drivers around the platooning system (Arnold, 2021). Although a platooning system comes with many autonomous limitations, the requirement of a manual driver eases complexities allowing for more versatile system designs and functionalities.

With these concepts in mind, the team this year has the overall goal of updating the existing golf cart platooning system at UVA to use a CACC platooning algorithm that uses feedback from LiDAR and Camera sensors. In order to achieve the main goal, the group has come up with three smaller objectives: establish a functioning platooning system, integrate sensors into the platooning system, and apply additional system upgrades. When completed, the carts should be able to complete a loop around Engineer's Way and the Observatory Mountain Engineering Research Facility (OMERF) using the completed system fit with the platooning algorithm and the sensors.

Essential Knowledge

For the past four years, the University of Virginia has been developing an Autonomous Campus Vehicle (ACV) system with the intention of using a platooning system to have three golf carts drive a route from Engineer's Way to OMERF. However, this project originated at Virginia Tech under Professor Tomonari Furukawa as the Self-Driving Vehicle Team (SDVT). This team's focus was to incorporate the Kairos Autonomi System on a single Club Car golf cart to achieve autonomy (Furukawa et al., 2018). Ultimately, the SDVT was not able to effectively use the Kairos Autonomi System and instead began work on a Robotic Operating System (ROS) instead. This was the extent of the work conducted by the SDVT as Professor Furukawa then transferred to UVA, bringing his autonomous vehicle endeavors with him. At UVA, the SDVT became the Campus Vehicle System (CVS): the first of these teams with the goal of developing a platooning system to provide transportation from Engineer's way to OMERF (Furukawa et al., 2021). Since its establishment at UVA, the CVS has undergone countless hardware changes and system redesigns, but the work conducted by the 2023 CVS team has laid a solid foundation for the 2024 ACV team to further develop the golf cart system.

One of the major improvements conducted by the 2023 CVS team was an overhaul of the CVS's Electronic Control Module (ECM). The ECMs were powered off of a single 12V supply, thus simplifying the previous wiring scheme (Furukawa et al., 2023). The 12V power source was then connected to a 4-way automotive fuse box, which provided power to the fan controller, a

12V-5V DC-DC converter to power the Raspberry Pi 4B, and the NETGEAR Nighthawk router. Next, all data was designed to be sent and received through a centralized control board which routes the data to the appropriate locations. Besides the speedometer input data, which first passed through a 5V-3.3V voltage divider, all other inputs and outputs were directly connected to the Raspberry Pi's general-purpose input/output (GPIO) pins using a ribbon cable (Furukawa et al., 2023). The control board also featured three C93416 relays and a MCP4151 digital potentiometer. The first relay was used to control the power going to the Nexsteer Electronic Power Steering (EPS) module, the other two relays were used to switch between manual and autonomous operation, and the digital potentiometer was used to control acceleration when driving autonomously. Finally, the ECM also featured a Peak PCAN-USB to allow communication between the Raspberry Pi and the Nexsteer EPS module.

There were also several hardware improvements performed by the 2023 CVS team. Each Club Car's steering was controlled by a Nexsteer EPS, but the second golf cart did not come with a mounting bracket to attach the EPS to the cart, so the CVS team designed and manufactured a new EPS mount to properly secure the second EPS (Furukawa et al., 2023). However, due to some fit issues with this part, the team made some adjustments using an angle grinder and scrap metal pieces before installing the bracket on its cart. The 2023 CVS team also worked to install the braking system on the second cart. Following the braking system layout on the first cart, the team designed and manufactured another bracket to attach a Clearpath MC Servo motor to the bottom of the cart to control braking (Furukawa et al., 2023). This bracket was designed to withstand the forces exerted during braking by the motor and also featured removable pulley wheel shafts for eased maintenance.

Along with this hardware work, the 2023 CVS team created the first iteration of the project which successfully implemented platooning. In this system, there was one leader cart that was controlled by a human and one follower cart that copied the actuation and braking of the leading cart. This allowed for the platooning system to follow a straight path but did not allow for the turns and bends that are along the one-mile route to Engineer's Way. The technology behind the platooning system used ROS to communicate between the two carts and had an algorithm that determined the distances between the carts using their velocity vectors.

Neither of the carts used sensors in their algorithms because the previous team found the LiDAR sensors and cameras in place were ineffective with route mapping (Furukawa et al., 2023). The computers used by the 2023 team did not contain an NVIDIA graphics card, so these computers were not compatible with camera sensors that were necessary to perform any tracking or object addition. Also, since the algorithm only copied the acceleration and velocity vectors of the lead cart, LiDAR sensors were also not implemented. These vehicles are shown below in Figure 1.

Fig. 1: Golf Carts Platooning on Straight Path

Thus far, the SDVT and CVS teams have failed to completely develop a platooning system that can travel the distance needed to go from OMERF to Engineer's Way and back. By following the work of the previous autonomous golf cart teams, the ACV team seeks to further this project's progress toward establishing a reliable platooning system that can effectively provide transportation from OMERF to Engineer's Way.

Design Process

The team followed the product design method laid out in "Product Design and Development," by Karl T. Ulrich and Steven D. Eppinger. This led us to investigate the customer needs, target specifications, concept generation, and concept selection.

In order to establish specific objectives for our project, the needs of our customers needed to be identified. For this project, our customers include MAE 4610 students, users of OMERF, Professor Brian Park, and Professor Tomonari Furukawa. First, a survey was conducted on a controlled group of MAE 4610 students. This survey focused on questions pertaining to both the students' interactions with UVA's University Transit Service and their experiences with travel to OMERF. The questions from this survey are displayed below in Table I.

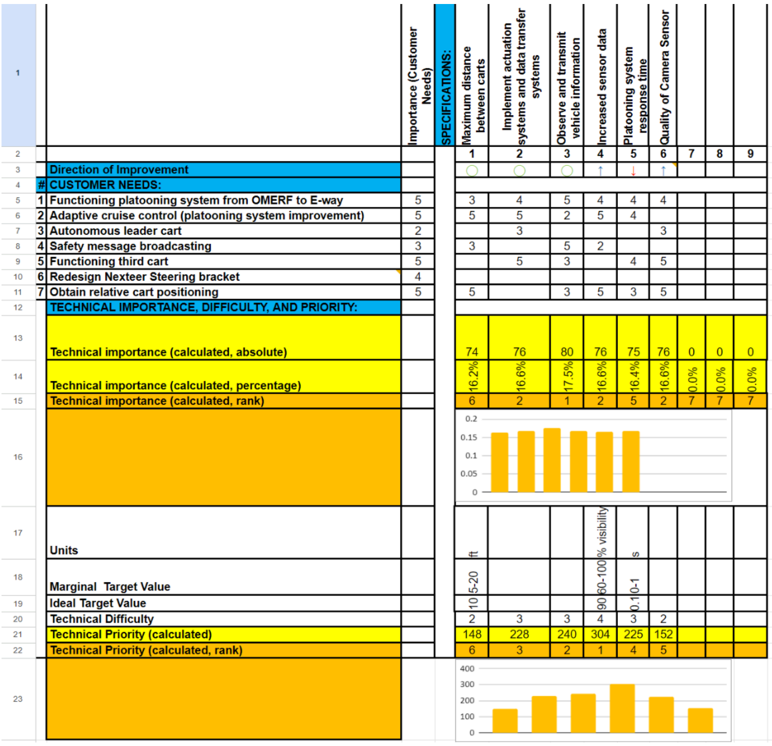
Table I: Customer Needs Survey Questions

UVA UTS Questions	
Question:	**Answer Format:**
How often do you use UVA transportation?	Ranking: 1 (Rarely) - 5 (Often)
Where do you typically travel using UVA transportation?	Short Answer
What factors encourage your use of UVA transportation?	Short Answer
What factors discourage your use of UVA transportation?	Short Answer
What improvements would you like to see from UVA transportation?	Short Answer

OMERF Questions	
Question:	**Answer Format:**
What is your primary mode of transportation to reach OMERF from Grounds?	Short Answer
How difficult would you rate your average journey to and from OMERF?	Ranking: 1 (Easy) - 5 (Difficult)
How likely would you be to use a fully-developed platooned golf cart system to travel from Engineer's Way to OMERF?	Ranking: 1 (Not Likely) - 5 (Likely)
What reason(s) would encourage your use of a platooned golf cart system to travel from Engineer's Way to OMERF?	Short Answer
What reason(s) would discourage your use of a platooned golf cart system to travel from Engineer's Way to OMERF?	Short Answer

Upon reviewing the responses to the survey, it was determined that these customers prioritized safety, consistent scheduling, weather protection, efficient travel, and no overcrowding. After discussions with Prof. Park and Prof. Furukawa, our customer needs were consolidated and refined. Our first customer need is to establish a functioning platooning system from OMERF to Engineer's Way. This need encompasses a couple of tasks. First, the platooning system from the 2022-2023 ACV team needed to be repaired to working condition. Next, the computers and sensors on both carts needed to be upgraded in order to support the new platooning algorithm being developed. This new algorithm operates using probabilistic positioning to determine the optimal travel paths for the follower carts by calculating the instantaneous radius of curvature. Our second customer need is to incorporate adaptive cruise control into our system. This addition will improve our carts' platooning abilities by basing the follower cart's speed on that of the leader cart. Our third customer need is to assemble a functioning third cart, which will increase the carrying capacity of our system. Our fourth customer need is to redesign the leader cart's steering column bracket. The current bracket is ineffective at securing the steering column to the cart's frame so a new bracket will improve the operation of the leader cart. Our fifth customer need is to obtain relative cart positioning. This addition to our system will allow identification of each follower cart once more are added. Our sixth customer need is to broadcast safety messages from the vehicles. Once our platooning system is operational, broadcasting safety messages will allow convenient monitoring of the carts from one hub. Finally, our seventh customer need is to develop an autonomous leader cart. Autonomous operation would dramatically increase the utility of our platooning system since it would relieve the need for manual operation of the leader cart.

Next, target specifications were created to measure the interpreted customer needs quantitatively. A quality function deployment spreadsheet, as seen in Figure 2, was then used to evaluate the importance of each technical specification regarding each of the previously established customer needs.



1	#		Importance (Customer Needs)	SPECIFICATIONS:	Maximum distance between carts	Implement actuation systems and data transfer systems	Observe and transmit vehicle information	Increased sensor data	Platooning system response time	Quality of Camera Sensor			
2					1	2	3	4	5	6	7	8	9
3		Direction of Improvement			○	○	○	↑	↓	↑			
4	#	CUSTOMER NEEDS:											
5	1	Functioning platooning system from OMERF to E-way	5		3	4	5	4	4	4			
6	2	Adaptive cruise control (platooning system improvement)	5		5	5	2	5	4				
7	3	Autonomous leader cart	2			3				3			
8	4	Safety message broadcasting	3		3		5	2					
9	5	Functioning third cart	5			5	3		4	5			
10	6	Redesign Nexteer Steering bracket	4										
11	7	Obtain relative cart positioning	5		5		3	5	3	5			
12		TECHNICAL IMPORTANCE, DIFFICULTY, AND PRIORITY:											
13		Technical importance (calculated, absolute)			74	76	80	76	75	76	0	0	0
14		Technical importance (calculated, percentage)			16.2%	16.6%	17.5%	16.6%	16.4%	16.6%	0.0%	0.0%	0.0%
15		Technical importance (calculated, rank)			6	2	1	2	5	2	7	7	7
16													
17		Units			ft			% visibility	s				
18		Marginal Target Value			5-20			60-100	0-1				
19		Ideal Target Value			10			90	0.1				
20		Technical Difficulty			2	3	3	4	3	2			
21		Technical Priority (calculated)			148	228	240	304	225	152			
22		Technical Priority (calculated, rank)			6	3	2	1	4	5			
23													

Fig. 2: Quality Function Deployment Spreadsheet

First, it was determined how each target specification either maximized (↑), minimized (↓), or represented a target (○) for each improvement (Row 3). Rows 5-11 of Figure 2 evaluated each technical specification on a scale of 1 (complete correlation) to 5 (minimum correlation) regarding how it correlated to each customer need. These correlation and importance values were then used to perform calculations to evaluate the technical importance of each target specification (Rows 13-15). Each specification was also evaluated based on technical difficulty on a scale of 1 (trivial) to 5 (very difficult) and given an ideal target value. Finally, the technical priority of each specification was determined by performing calculations using the importance and technical difficulty of each (Rows 21-22).

Table II shows the priority and quantified unit, where applicable, for each target specification, with Increased Sensor Data being the top priority for the ACV.

Table II: Technical Specifications Ranked by Technical Priority

Technical Priority	**Target Specifications**	**Unit**
1	Increased Sensor Data	% Visibility
2	Observe & Transmit Vehicle Information	
3	Implement Actuation Systems & Data Transfer Systems	
4	Platooning System Response Time	seconds
5	Quality of Sensor	$
6	Maximum Distance Between Cars	ft

Taking both the customer needs and target specifications into consideration, the next steps were to determine the desired functionality of the system to be developed. This was done through functional decomposition, as shown below in Figure 3.

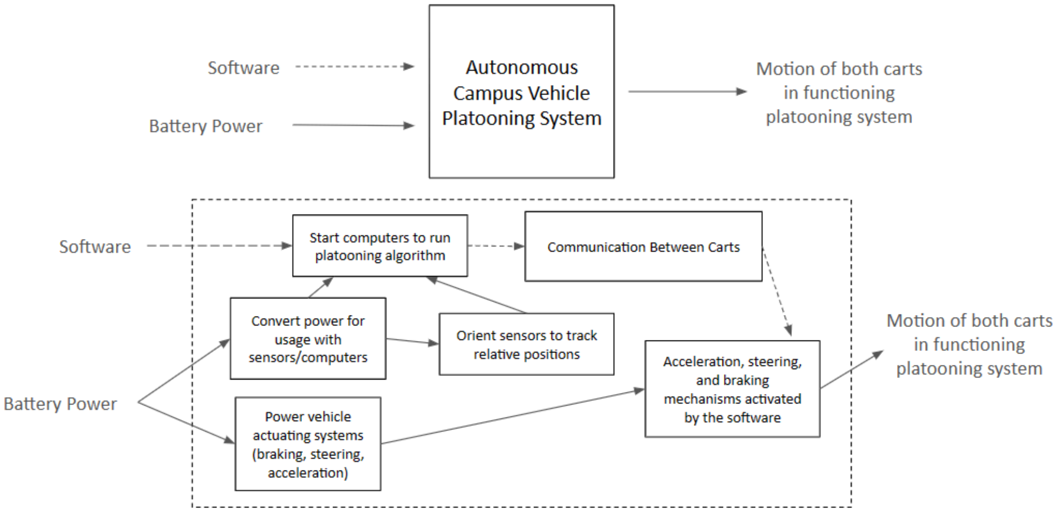


Fig. 3: Diagram of Functional Decomposition

As seen above, the desired functionality of the ACV system is the motion of both carts in a functioning platooning system, achieved by using power and input software. This software will be run on onboard computers, and with the visual information collected from implemented sensors in conjunction with the actuating systems already present on the carts, the final goal can be accomplished. From these determined subsystems, morphological analysis was completed to generate possible solutions. This analysis is summarized in Table III.

Table III: Morphological Analysis

Sub Functions	**Solutions**		
Start Computers	New Intel NUC with NVIDIA graphics card	Old Intel NUC with new NVIDIA graphics card	Build computer from scratch
Orient sensors to track relative positions	LiDAR Sensors	Camera Sensors	Infrared Sensors
Platooning Algorithm	Duplicate velocity/acc. vectors from leader car	Instantaneous Radius of Curvature	
Communication between carts	WiFi Router	Mesh Routers	Bluetooth

These potential solutions were considered during the concept selection process, with a focus on the computer system and desired sensors, as the platooning algorithm that implements the instantaneous radius of curvature is given as a customer need and mesh routers were suggested to be the most efficient communication system for the proposed system.

After generating concepts, two spreadsheets were used to compare and contrast each concept using selected criteria to determine the best choices going forward. The concepts evaluated were the possible options for sensors and computers for the ACVs. For the sensors, we considered camera sensors, LiDAR sensors, Infrared (IR) sensors, and a QR Code reader. For the computers, we considered building our own computer, purchasing a brand new Intel NUC with an NVIDIA graphics card, and implementing a new NVIDIA graphics card to the existing Intel NUC.

First, a screening matrix, as shown in Figures 4 and 5, was used to help narrow down the choices for sensor and computer solutions. For each selection criterion, this matrix used a plus (+) for a positive solution, a minus (-) for a negative solution, or a zero (0) for a neutral solution. Then, the values were summed to give each an overall score. If the score was greater than or equal to zero, it was advised to continue.

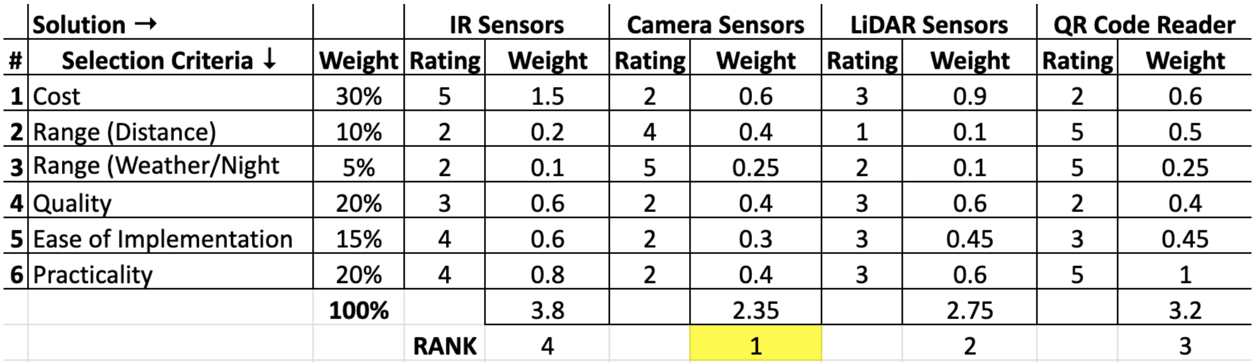
Potential Solution → Selection Criteria ↓	IR sensors	Video Camera Sensors	LiDAR Sensors	QR Code Reader
Cost	-	+	0	+
Range (Distance)	+	0	+	-
Range (Weather/Night Conditions)	+	-	+	-
Quality	+	0	+	0
Complexity	-	0	0	+
Practicality	0	0	+	-
Sum +'s	3	1	4	2
Sum 0's	1	4	2	1
Sum -'s	2	1	0	3
Net Score	1	0	4	-1
Rank	2	3	1	4
Continue?	Yes	Yes	Yes	No

Fig. 4: Screening Matrix for Sensors

Potential Solution → Selection Criteria ↓	Build Computer	New Intel NUC w/ NVIDIA Graphics Card	Old Intel NUC w/ new NVIDIA Graphics Card
Cost	0	-	+
Longevity	+	+	-
Size	-	0	0
Processing Power	0	+	+
Modularity	+	0	0
Ease of Implementation	-	0	+
Sum +'s	2	2	3
Sum 0's	2	3	2
Sum -'s	2	1	1
Net Score	0	1	2
Rank	3	2	1
Continue?	Yes	Yes	Yes

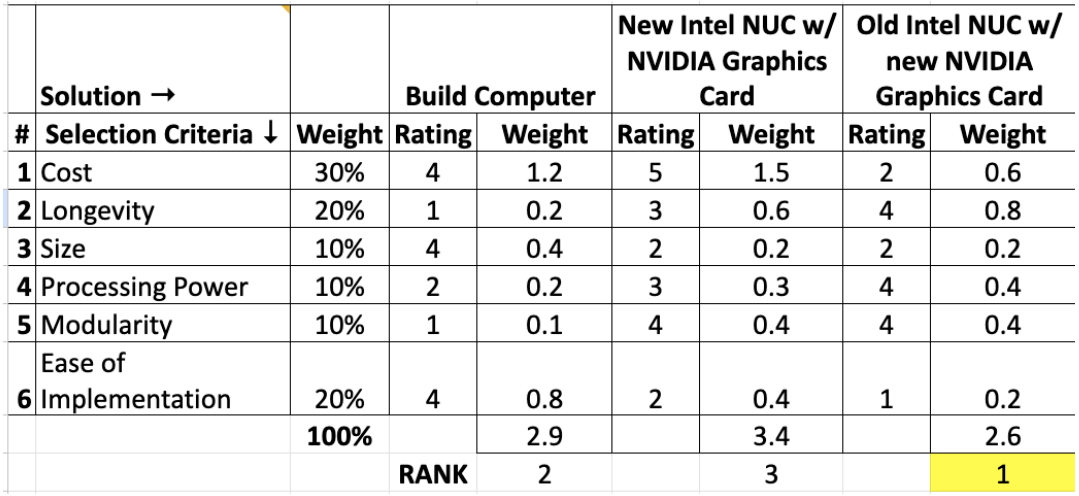
Fig. 5: Screening Matrix for Computers

Next, we utilized a scoring matrix, as shown in Figures 6 and 7, to determine which solution best satisfied our selection criteria. For this matrix, each selection criterion was given a weight based on the desired importance for our system. Each concept was then rated on a scale of 1 (best) to 5 (worst), and then the total score was calculated by summing every weight multiplied by the corresponding rating.



	Solution →		**IR Sensors**		**Camera Sensors**		**LiDAR Sensors**		**QR Code Reader**	
#	**Selection Criteria ↓**	**Weight**	**Rating**	**Weight**	**Rating**	**Weight**	**Rating**	**Weight**	**Rating**	**Weight**
1	Cost	30%	5	1.5	2	0.6	3	0.9	2	0.6
2	Range (Distance)	10%	2	0.2	4	0.4	1	0.1	5	0.5
3	Range (Weather/Night	5%	2	0.1	5	0.25	2	0.1	5	0.25
4	Quality	20%	3	0.6	2	0.4	3	0.6	2	0.4
5	Ease of Implementation	15%	4	0.6	2	0.3	3	0.45	3	0.45
6	Practicality	20%	4	0.8	2	0.4	3	0.6	5	1
		100%		3.8		2.35		2.75		3.2
			RANK	4		1		2		3

Fig. 6: Scoring Matrix for Sensors



	Solution →		**Build Computer**		**New Intel NUC w/ NVIDIA Graphics Card**		**Old Intel NUC w/ new NVIDIA Graphics Card**	
#	**Selection Criteria ↓**	**Weight**	**Rating**	**Weight**	**Rating**	**Weight**	**Rating**	**Weight**
1	Cost	30%	4	1.2	5	1.5	2	0.6
2	Longevity	20%	1	0.2	3	0.6	4	0.8
3	Size	10%	4	0.4	2	0.2	2	0.2
4	Processing Power	10%	2	0.2	3	0.3	4	0.4
5	Modularity	10%	1	0.1	4	0.4	4	0.4
6	Ease of Implementation	20%	4	0.8	2	0.4	1	0.2
		100%		2.9		3.4		2.6
			RANK	2		3		1

Fig. 7: Scoring Matrix for Computers

Based on the results from the screening and scoring matrices shown in Figures 6 and 7, the team decided to move forward with prioritizing the implementation of the camera sensors for the visibility of the system and a modified computer system that includes a new NVIDIA graphics card.

Final Design

Our final design includes the actuating systems present on the golf carts, contributions from previous teams, and the following components determined from the concept selection process: camera sensors for the visibility of the system, mesh routers to allow ease of communication between vehicles, and a modified computer system that includes a new NVIDIA graphics card. Figure 8 below shows what our new system diagram will look like with these new concepts implemented.

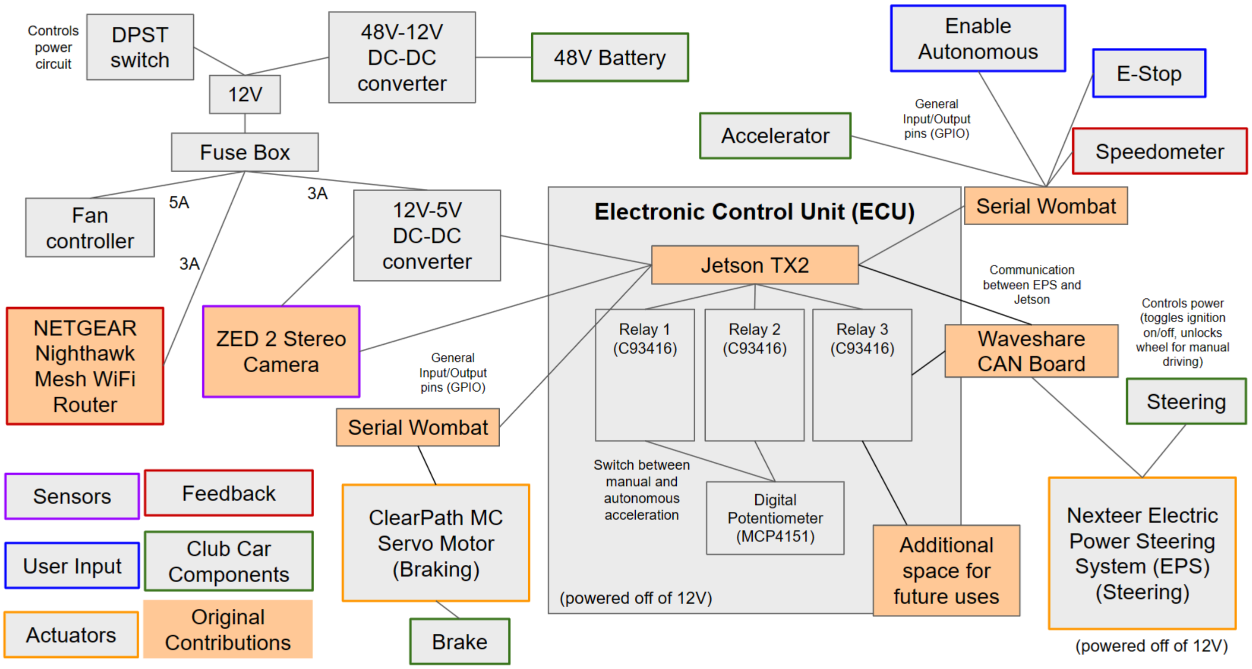


Fig. 8: Final Design System Diagram

The ZED2 camera sensor is used to track ArUco markers and evaluate angle and position data to ensure the carts are functioning properly in the platooning system. A CACC platooning algorithm uses the data given by these sensors to control the actuation and braking of the following car. It uses probabilistic positioning to determine the optimal travel paths for the follower carts by calculating the instantaneous radius of curvature. Mesh routers have been determined to be the most effective way of communication between the carts in the system, allowing for direct communication without interference from other sources. Finally, our chosen computer solution will be to use the previous NUC computer in the system with the modification of replacing the Raspberry Pis with Jetson TX2s, which are similar computing boards to the Raspberry Pis with the addition of an NVIDIA graphics card, making them compatible with the ZED2 camera sensors. Additionally, two microcontrollers were added to the system to handle additional communications. First, a Serial Wombat board was added to handle braking, accelerating, operation of the emergency stop, the speedometer, and the enable autonomous switch. Second, a Waveshare CAN board was added to the system to handle the Nexteer Electric Power Steering system.

Results

The ZED2 camera tracking code has been written using OpenCV and Python and has also been translated into ROS. The wifi routers have established a connection between the two carts. The new electrical boxes were developed with the new hardware, and the CAN system was set up in order to receive data from the steering wheel from each cart and send that information to the next cart. All these systems are working and will be ready for future teams to implement onto the carts.

ZED2 Camera

The ZED2 camera sensor has been implemented using ArUco trackers for the follower cart to track the angle, depth, height, and pose of the leader cart. Code was written using Python with OpenCV to track the two ArUco markers that will be attached to the back of the leader cart and it runs with ROS to publish the necessary data in order to be used in the platooning algorithm. The ArUco tracking code publishes the marker depth, angle, height, and tracking angle. The tracking angle is found using the following equation from Cox et al. 2019, shown in Figure 9.

$$a^2 = b^2 + c^2 - 2bc * \cos A$$

$$\alpha = 90 - \cos^{-1}\left[\frac{(\frac{1}{2}l)^2 - d_L^2 + d^2}{dl}\right]$$

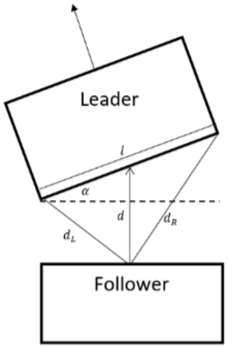
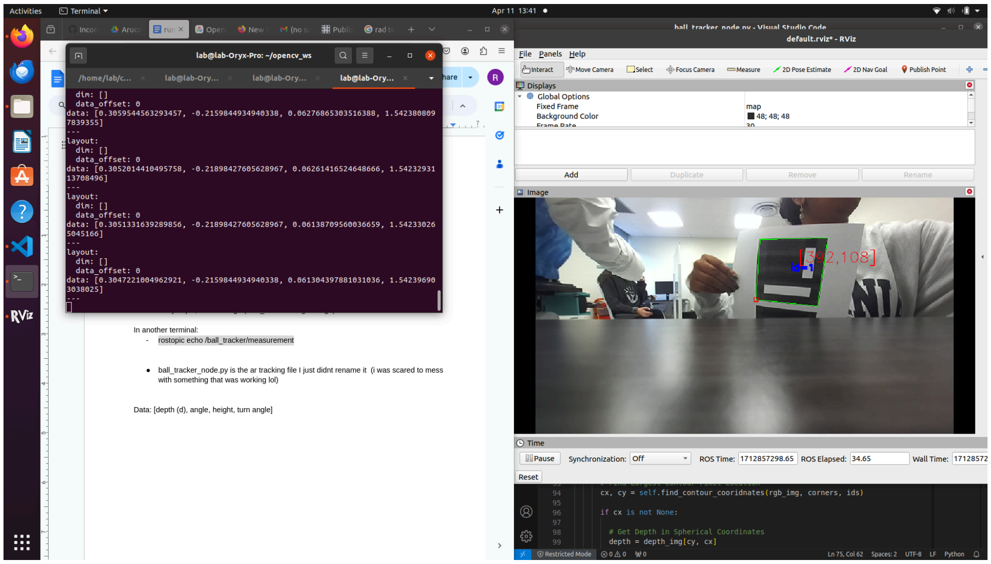


Fig. 9: Equation for turn angle of leader vehicle. Image from Cox et al. 2019.

This data works with the platooning algorithm for the follower cart to successfully follow and keep track of the leader cart. Figure 10 below shows the ArUco tracker and data published through ROS. Appendix A provides the steps to run the data publishing and open the camera feed as seen in Figure 10.



```
  dim: []
  data_offset: 0
data: [0.3059544563293457, -0.2159844934940338, 0.06276865303516388, 1.542380809
7839355]
---
layout:
  dim: []
  data_offset: 0
data: [0.3052014410495758, -0.2189842760562896 7, 0.06261416524648666, 1.54232931
13708496]
---
layout:
  dim: []
  data_offset: 0
data: [0.3051331639289856, -0.21898427605628967, 0.06138709560036659, 1.54233026
5045166]
---
layout:
  dim: []
  data_offset: 0
data: [0.3047221004962921, -0.2159844934940338, 0.061304397881031036, 1.54239690
3038025]
---
```

In another terminal:
- rostopic echo /ball_tracker/measurement

- ball_tracker_node.py is the ar tracking file I just didnt rename it (i was scared to mess with something that was working lol)

Data: [depth (d), angle, height, turn angle]

Fixed Frame: map
Background Color: 48; 48; 48
Synchronization: Off
ROS Time: 1712857298.65
ROS Elapsed: 34.65
Wall Time: 17128572

```
cx, cy = self.find_contour_cooridnates(rgb_img, corners, ids)

if cx is not None:

  # Get Depth in Spherical Coordinates
  depth = depth_img[cy, cx]
```

Fig. 10: ROS terminal on left publishing sensor data. The image on the right depicts the tracking of one ArUco marker.

WiFi Connections

The wifi routers have been set up to create a new wifi network called “acvwifi” and static IP addresses have been set up on each of the Jetson TX2s. The Jetson TX2s are connected to each mesh router by ethernet cable. When the Jetson TX2s turn on, they are automatically set to the correct IP address and connected to one another via the wifi network. The Jetson TX2s can now communicate via ROS to send information to one another. This is now automatically run as soon as the Jetson TX2s are turned on and rebooted to streamline the process of connecting the computers on each of the carts.

Electrical Boxes

The electrical boxes have been cut to allow for the installation of the laser-cut acrylic pieces. Most of the hardware that can be implemented in the boxes has been installed including the fans and vents, shown in Figures 11 and 12. The PCB, which is necessary to complete the new installation of the electrical boxes, was redesigned to accommodate the addition of the Serial Wombat, as seen in Figure 13. In the final state of the system, a perf board replaced the PCB, to ensure each connection was the correct connection. In the future a correct PCB can replace the perf board in the box.

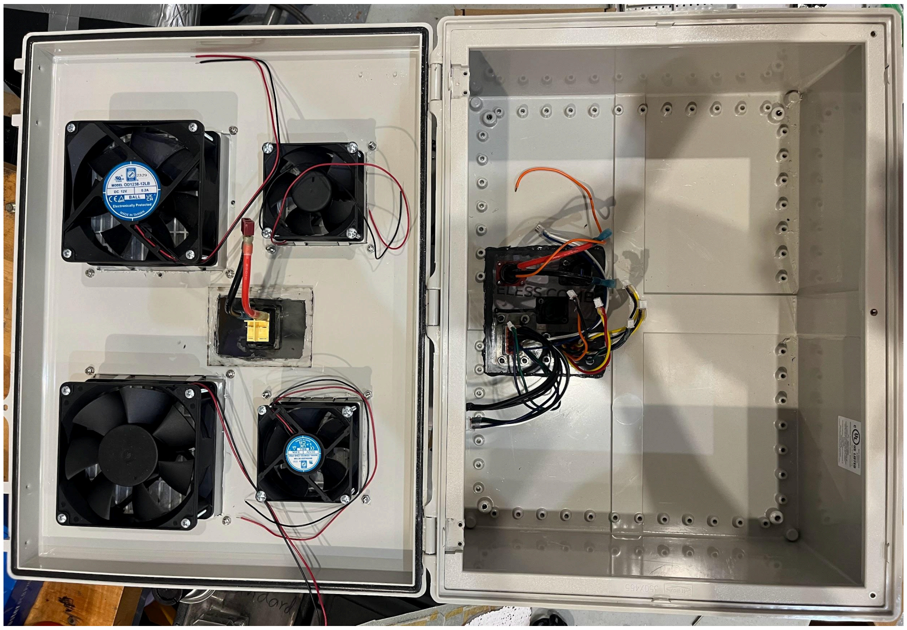


Fig. 11: Electronic box, inside with fans and cables installed

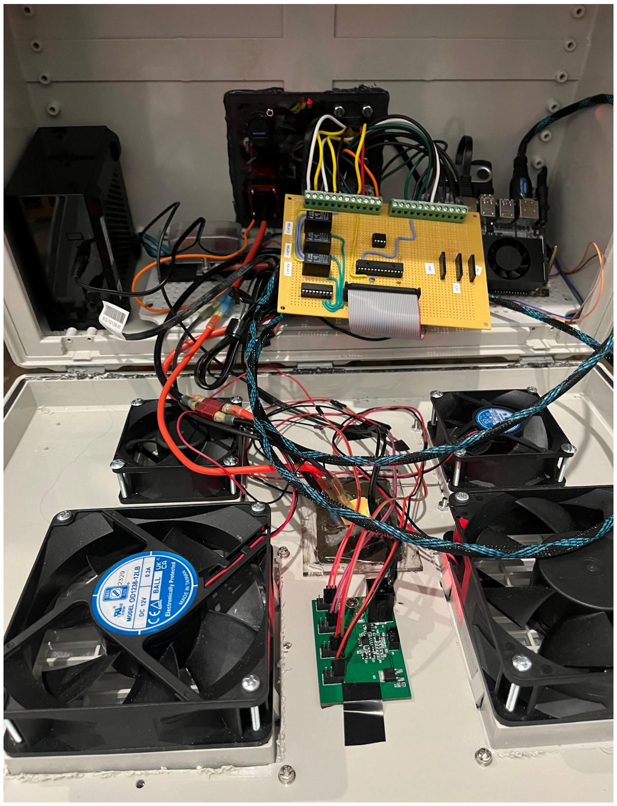


Fig. 12: Electronic box, inside with TX2, perf board, and router connected

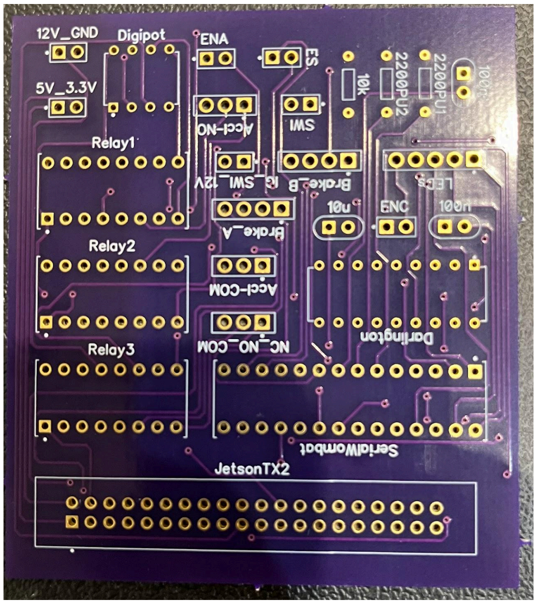


Fig. 13: PCB design, with pinouts for the Serial Wombat and the Jetson TX2

CAN System

The CAN bus system consists of a Waveshare SN65HVD230 transceiver and a Jetson TX2 for transmission of each golf cart's steering wheel data. To test code written in Python for receiving and transmitting data across the CAN system, a wired configuration is used to test communication between two separate Jetson TX2s shown in Figure 14.

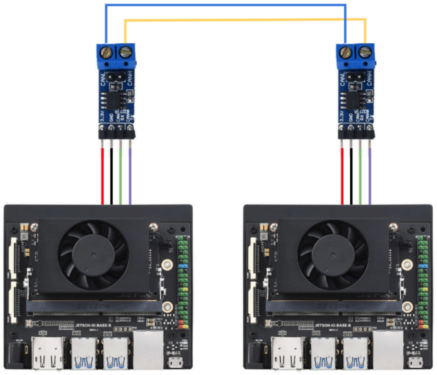


Fig. 14: CAN bus configuration for communication between Jetson TX2s

To transmit data between the CANs, the following code had to be run on each separate desktop to set up the bus system for communication:

```
sudo modprobe can
sudo modprobe can_raw
sudo modprobe mttcan
sudo ip link set can0 type can bitrate 500000 dbitrate 2000000 berr-reporting on fd on
sudo ip link set up can0
```

This code has been added into the bashrc file as a defined function to be called as 'SETUP_CAN' rather than requiring the user to type in each line after startup.

Validation

Qualitative Analysis

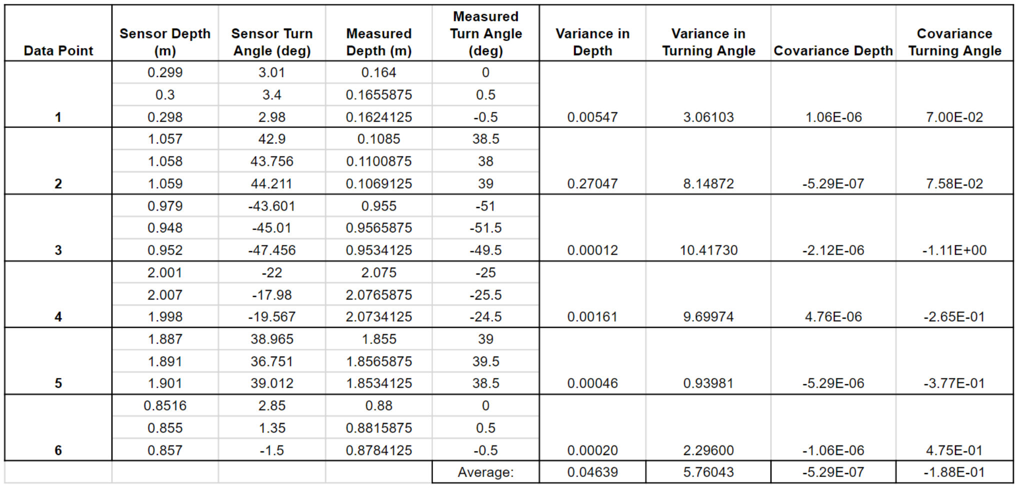
The system is currently not fully functional. The functional parts of the system are the following: One computer, the ECU, and the CAN systems are able to communicate with one another in order to transmit data. The ZED2 cameras are able to accurately track the various QR codes attached to the leader car. The system however has yet to be tested with the platooning algorithm as hardware updates still need to be made. Testing will occur when the electrical boxes are fully completed and the rest of the system hardware is connected to these components.

Quantitative Analysis

To validate the ZED2 camera sensor data, multiple data points were collected using the sensor to track the location (x and z coordinates) and the angle of the ArUco marker, and then these values were compared with measured values to determine the variances and covariances.

To conduct this analysis, data was collected at six varying positions, with three data points each to account for uncertainty in sensor data fluctuations. Measured values were collected using a tape measure with an uncertainty of ⅛ inch. Table IV below shows the recorded and calculated data from this sensor analysis.

Table IV: Recorded and Calculated Data from Sensor Analysis



Data Point	Sensor Depth (m)	Sensor Turn Angle (deg)	Measured Depth (m)	Measured Turn Angle (deg)	Variance in Depth	Variance in Turning Angle	Covariance Depth	Covariance Turning Angle
1	0.299	3.01	0.164	0	0.00547	3.06103	1.06E-06	7.00E-02
	0.3	3.4	0.1655875	0.5				
	0.298	2.98	0.1624125	-0.5				
2	1.057	42.9	0.1085	38.5	0.27047	8.14872	-5.29E-07	7.58E-02
	1.058	43.756	0.1100875	38				
	1.059	44.211	0.1069125	39				
3	0.979	-43.601	0.955	-51	0.00012	10.41730	-2.12E-06	-1.11E+00
	0.948	-45.01	0.9565875	-51.5				
	0.952	-47.456	0.9534125	-49.5				
4	2.001	-22	2.075	-25	0.00161	9.69974	4.76E-06	-2.65E-01
	2.007	-17.98	2.0765875	-25.5				
	1.998	-19.567	2.0734125	-24.5				
5	1.887	38.965	1.855	39	0.00046	0.93981	-5.29E-06	-3.77E-01
	1.891	36.751	1.8565875	39.5				
	1.901	39.012	1.8534125	38.5				
6	0.8516	2.85	0.88	0	0.00020	2.29600	-1.06E-06	4.75E-01
	0.855	1.35	0.8815875	0.5				
	0.857	-1.5	0.8784125	-0.5				
				Average:	0.04639	5.76043	-5.29E-07	-1.88E-01

Operations Manual

Components:

Fig. 15: Autonomous Campus Vehicles

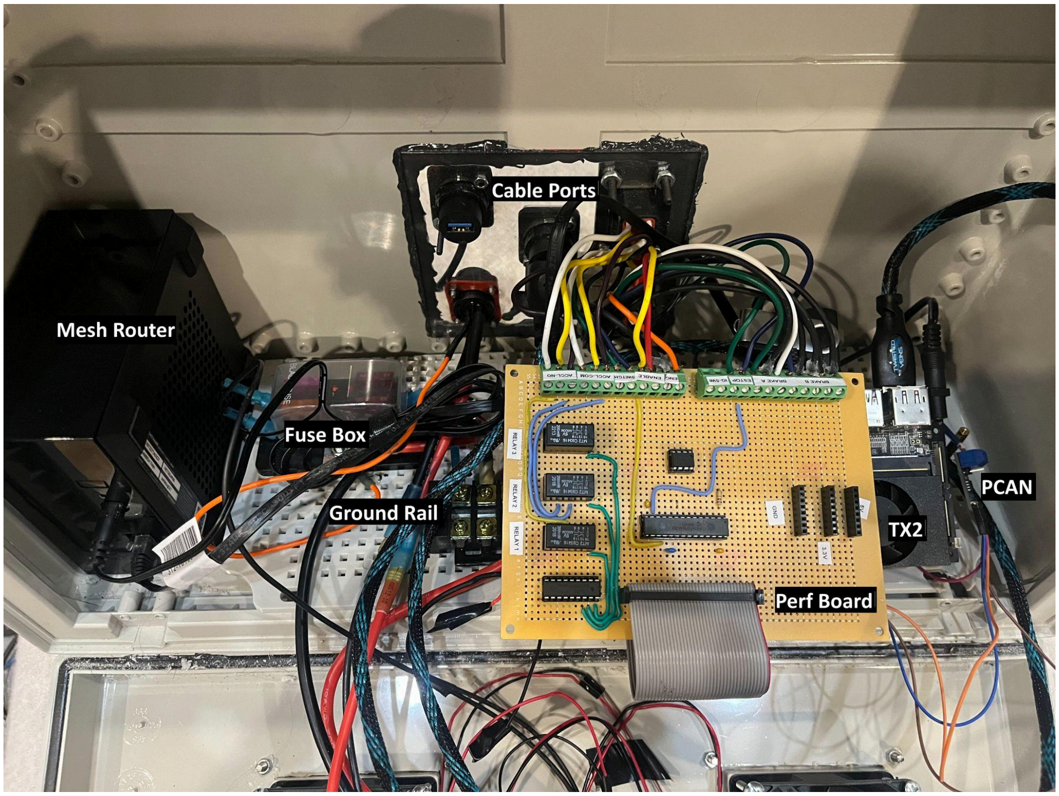


Fig. 16: ECU (Body)

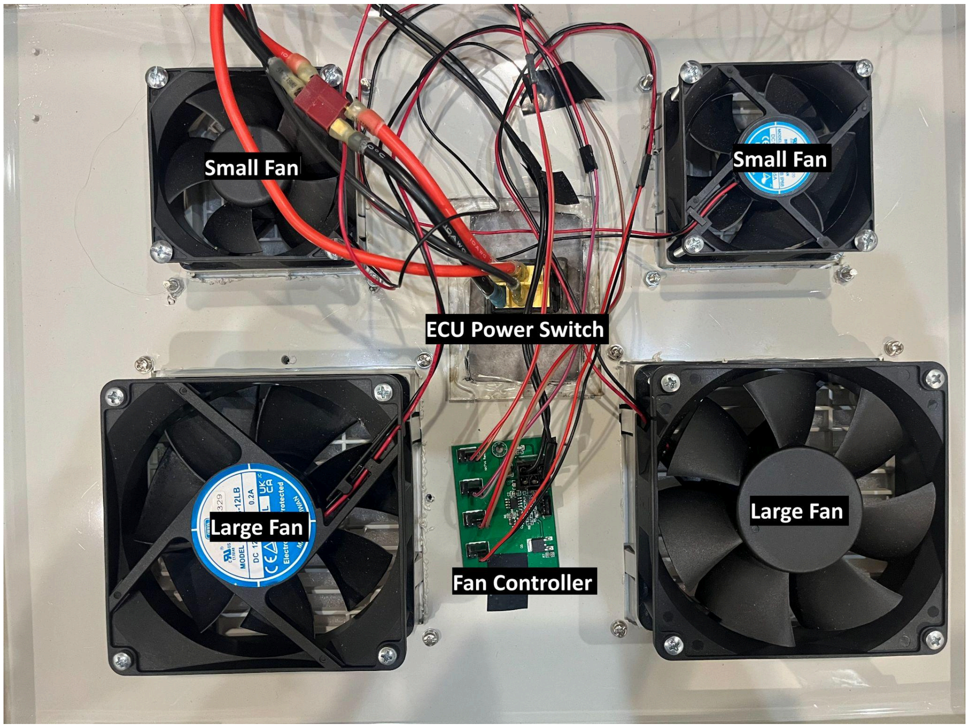


Fig. 17: ECU (Lid)

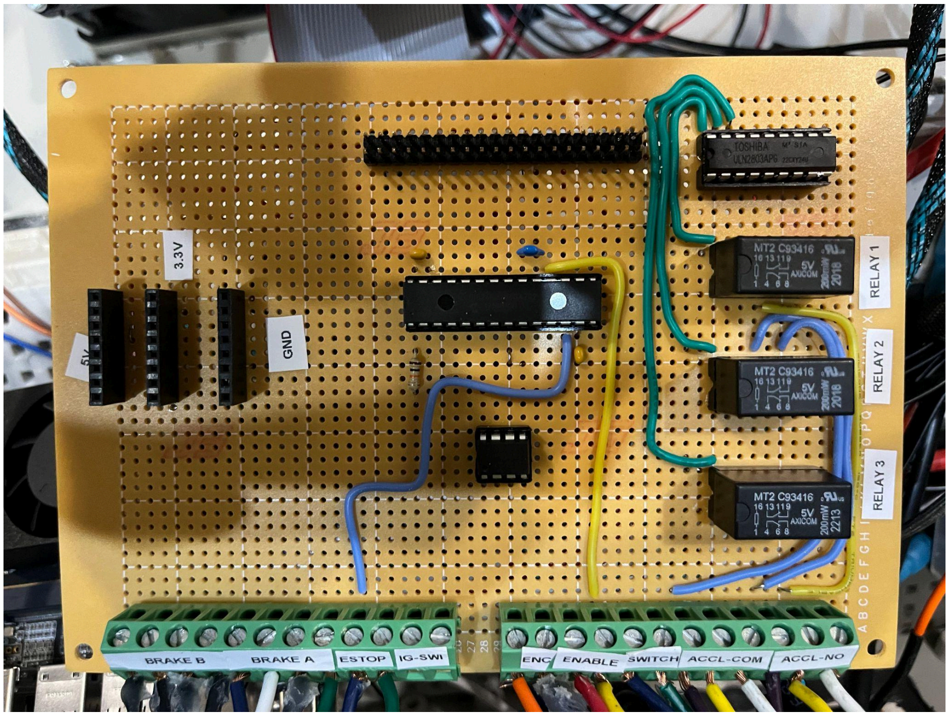


Fig. 18: Perf Board (Front)

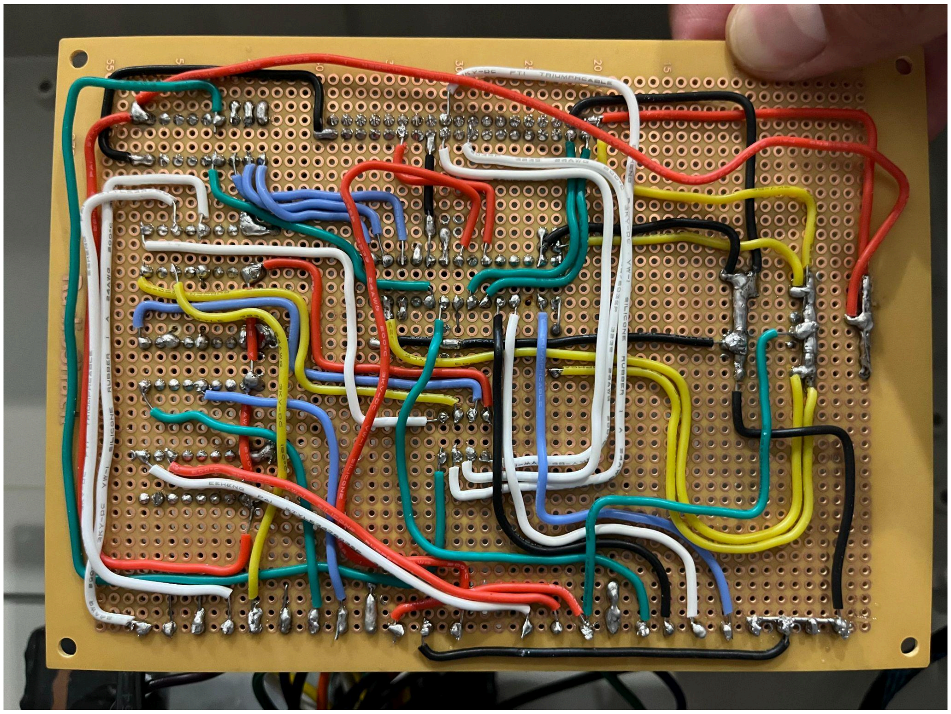


Fig. 19: Perf Board (Back)

Fig. 20: ZED2 Camera Sensor

Operations

In order to use the golf carts with the installed platooning algorithm, the user must first turn on the functionality of the motor, which can be done by flipping a switch located in the battery well of the vehicle beneath the front seats, as seen in Figure 21. The switch should be flipped from TOW to RUN. Additionally, to allow power to run from the cart batteries to the electronic box and to other components within the cart, the red switch located beneath the back seats must be turned until the switch clicks and the notch in the switch shows green. This is shown in Figure 22. Power is now being provided to the electronic box, steering system, braking system, and the monitor.

Fig. 21: Motor switch

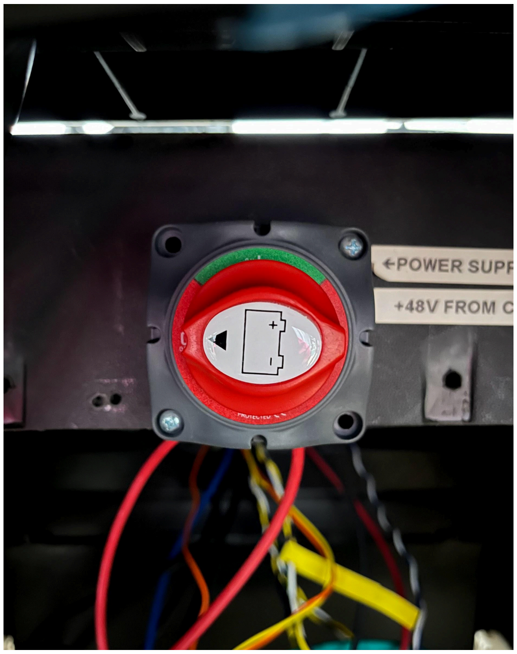


Fig. 22: Power switch

Then, the user must use the power switch (red light-up switch) on the electrical box to turn on the rest of the system including the monitor connected to the Jetson TX2s. This switch is shown in Figure 23. When the computer turns on, the user must sign in and the computers are automatically connected to one another through the mesh routers. The CAN system is also automatically initialized so the two computers can transmit information between the two of them. Finally, the ignition switch, as seen in Figure 24, must be in the ON position. If you want to run the autonomous platooning system, the enable switch, shown in Figure 25, must also be in the ON position. If it is turned OFF, the cart will function manually.

Fig. 23: Electronic box switch

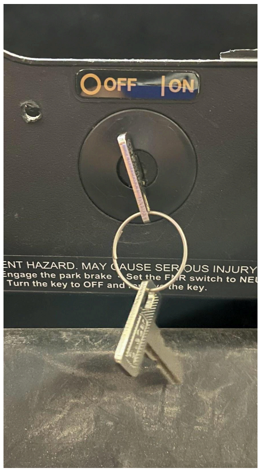


Fig. 24: Ignition switch

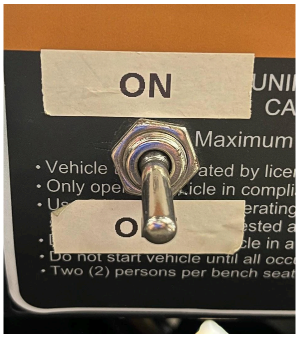


Fig. 25: Enable switch

The leader cart should be in front of the follower cart so that the QR codes are visible using the camera sensors. Next, the user can use ROS to initialize the platooning algorithm on the carts. This can be done by following a file on the Desktop of both vehicles titled “Setup File”. Within this file, the connection between both vehicles will be confirmed and the files running the steering, braking, acceleration, emergency stop, and enable systems, along with the platooning algorithm program, will run. The follower cart must be put in “forward” gear using the switch in front of the front seats. The leader cart can then be driven manually and the follower cart will follow behind it.

In case of an emergency, both vehicles are equipped with emergency stop buttons (yellow boxes with a red button), seen in Figure 26. Pressing this button will terminate the system from running by stopping the acceleration and applying the brakes. These buttons will then need to be twisted and released before running the system again.

Fig. 26: Emergency Stop Button (E-Stop)

Troubleshooting

To ensure electrical components are working as they should, Gil Johnson helped the team develop code to test different outputs for each system. Implementing pcb_driver_node.py, the user can comment out which system is needed to be tested, and watch the output. To run the program use the following code on the cart:

```
cd CUSTOM
sudo python3 pcb_driver_node.py
```

If error occurs use:

```
sudo modprobe spidev
```

Conclusions and Future Work

The Autonomous Campus Vehicle was not fully functioning at the end of the academic year; the Jetson TX2 used in the lead car was unretrievable after a short circuit in the debugging phase. The short circuit was caused by a floating pin in the PCB, causing it to be unusable. A perf board was made in place of the PCB with the correct connections, which is now implemented in the following car with the working Jetson TX2 and rest of the ECU. The ZED 2 camera sensor worked successfully with ArUco tags and published depth and turn angle data which is necessary for the platooning algorithm. The CAN System is able to read in steering wheel sensor data through the CAN system by locating the hex byte corresponding to steering wheel orientation. The platooning algorithm is ready to be used on the carts once necessary hardware is replaced, debugged, and implemented on the lead cart.

The team has also identified several areas for future ACV teams to work on. First, the necessary hardware should be added to the system in order to have the platooning algorithm function as planned. This entails purchasing a new TX2, wiring another perfboard or creating a PCB, connecting everything to the lead cart ECU, confirming all of the hardware works, and then sending orientation data to actuate the steering wheels with the CAN system. Once these updates occur, several other additions can be made to enhance the overall system. Future teams should construct and implement additional golf carts into the platooning system. This will expand the capacity of the system, which will thus increase the viability of its use as a transportation method from OMERF to Engineer's Way. The team was also unable to implement the LiDAR sensor on the following car in time to use it in the current version of the system. The LiDAR sensor would have been able to identify objects that obstructed the vision line between the ZED 2 and the QR codes and account for it in the platooning algorithm. Future teams should continue our work and establish a second way of validating the lead cart’s position with the LiDAR. Next, a safety message broadcasting unit should be added to the system. By having a central hub to display safety messages, the golf carts can always be monitored to ensure their operability. Along with this comes the addition of a GPS system. Knowing the exact locations of the golf carts in real-time will aid in increasing the system’s overall safety. Finally, instead of having the follower carts platooned to a manually operated leader cart, the leader cart can be designed to operate autonomously. This will eliminate the need for a driver of the leader cart, increasing the utility of the ACV system.

Bibliography

ALEC. American Legislative Exchange Council. (2017, June 2023). Vehicle Platooning for Safety and Efficiency Act. *American Legislative Exchange Council.* https://alec.org/model-policy/vehicle-platooning-for-safety-and-efficiency-act/#:~:text=A%20vehicle%20platooning%20system%20can,times%20to%20help%20avoid%20crashes.

Arnold, Michelle. (2021, January). Effective Indicators of Partially Automated Truck Platooning. *U.S. Department of Transportation Federal Highway Administration.* https://www.fhwa.dot.gov/publications/research/safety/21016/index.cfm?_gl=1*1wghns6*_ga*MTkyNjgzNDEzOS4xNjk3NTA1NjUx*_ga_VW1SFWJKBB*MTcxMzE5ODQ1NS4zLjEuMTcxMzE5ODQ3MC4wLjAuMA..

Bhoopalam, A. K., Agatz, N., & Zuidwijk, R. (2018). Planning of truck platoons: A literature review and directions for future research. *Transportation Research Part B: Methodological,* 107, 212-228.

Carlier, M. (2022, January 25). *Road accidents in the United States*. Statista. https://www.statista.com/topics/3708/road-accidents-in-the-us/#topic

Cox, J., et al. (2019, May 9). *Self-Driving Vehicle Team Fall Final Report.* Virginia Tech.

Furukawa, T., Arriaga, A., Bray, W., Chen, A., Chu, S., Cox, D., Cox, J., Ishanzai, A., Karas, M., Lee, D., & Voelker, A. (2018). Spring Final Report [Unpublished undergraduate thesis]. Virginia Polytechnic Institute and State University.

Furukawa, T., Chiaramonte, C., Dunnington, P., Johnson, G., Sofinski, N., & Wilson, A. (2023). Development of an Autonomous Platooning Campus Vehicle System [Unpublished undergraduate thesis]. University of Virginia.

Furukawa, T., Fontelera, A., Shin, J., Smith, W., Wellman, P., & Yocom, J. (2021). Modification of a Golf Cart to Autonomous Campus Vehicle [Unpublished undergraduate thesis]. University of Virginia.

Haines, G., Kochevar, R., & Surti, V. H. (1974). Analysis of Campus Traffic Problems. *Highway Research Board*, *498*, 1–12.

Puplaka, Y. (2016). *Implementation of platooning concept from business perspective.* Lund University.

Shaver, G., & Droege, M. (2021). Develop and deploy a safe truck platoon testing protocol for the Purdue ARPA-E project in Indiana (Joint Transportation Research Program Publication No. FHWA/IN/JTRP-2021/12). West Lafayette, IN: Purdue University. https://doi.org/10.5703/1288284317314

Smith, S. W., Kim, Y., Guanetti, J., Li, R., Firoozi, R., Wootton, B., Kurzhanskiy, A. A., Borrelli, F., Horowitz, R., & Arcak, M. (2020). Improving Urban Traffic Throughput With Vehicle Platooning: Theory and Experiments. *IEEE Access*, *8*, 1–14. https://doi.org/10.48550/arXiv.2006.10272

Staudacher, David (2024, April 8). New Truck Platooning System Can Benefit Everyone. *The Board of Trustees of the University of Illinois.* https://cme.uic.edu/news-stories/new-truck-platooning-system-can-benefit-everyone/

Syed, M. H., & Abadin, Z. U. (2020, July). A pattern for autonomous vehicle platoon. *In Proceedings of the European Conference on Pattern Languages of Programs 2020 (EuroPLoP '20). Association for Computing Machinery, New York, NY, USA, Article 26, 1–7. https://doi.org/10.1145/3424771.3424813*

Volpe Center. US Department of Transportation. (2017, April 27). Experts Reveal Safety Impact of Following Distances between Heavy Trucks in Automated Platoons. *U.S. Department of Transportation.* https://www.volpe.dot.gov/news/researchers-reveal-safety-impact-following-distances-between-heavy-trucks-automated-platoons

Yang, Shiyan & Shladover, Steven E. & Lu, Xiao-Yun & Spring, John & Nelson, David & Ramezani, Hani, 2018. "A First Investigation of Truck Drivers' On-the-Road Experience Using Cooperative Adaptive Cruise Control," Institute of Transportation Studies, Research Reports, Working Papers, Proceedings qt92359572, Institute of Transportation Studies, UC Berkeley.

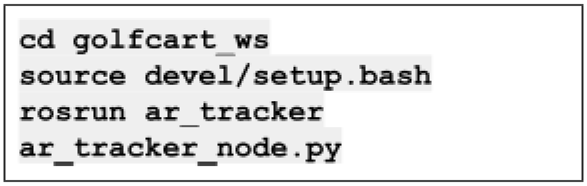
Appendix A: Running ar_tracker_node.py with ZED2 Camera in ROS

- Type ROS_LOCAL before commands if connected to ethernet
- Open a new terminal for each set of commands
- ROS will publish the following data: [avg depth in meters (d), turn angle in degrees (alpha)]
    - See Figure 9 for turn angle equation
- Change l in code depending on distance between tags (e.g. width of cart)

Terminal 1:

```
cd golfcart_ws (opencv_ws)
source devel/setup.bash
roslaunch zed_wrapper zed2.launch
```

Terminal 2:



```
cd golfcart_ws
source devel/setup.bash
rosrun ar_tracker
ar_tracker_node.py
```

Terminal 3:

```
rviz
```

In rviz, click:

- Click **Add** (bottom left)
- In **By Topic**, select **Image (/ar_tracker/image/Image)**

This opens up the camera feed on the computer.

Terminal 4:

```
rostopic echo /ar_tracker/measurement
```