Learning Autonomous Driving in Tangible Practice: Development and On-Road Applications of a 1/10-Scale Autonomous Vehicle

Laura Paulino
Department of Computer Science
Montclair State University
Montclair, NJ, USA
adamespaull1@montclair.edu

Michelle Zhu
Department of Computer Science
Montclair State University
Montclair, NJ, USA
zhumi@montclair.edu

Weitian Wang
Department of Computer Science
Montclair State University
Montclair, NJ, USA
wangw@montclair.edu

***Abstract*— This Innovative Practice Work-In-Progress Paper presents a case of learning autonomous driving in tangible practice. As technology sustainably enhances the quality of life, intelligent systems continue to contribute solutions to some of the biggest challenges faced by humans. Autonomous vehicles offer humans the opportunity to increase transportation safety by reducing human errors on the road, preventing accidents, improving human productivity by reducing commuting time, and possibly mitigating air pollution. There is a critical shortage of educational and training programs in autonomous vehicles due to the high cost of full-size vehicles, computing and sensor equipment, and big lab space needed. To address this problem, we develop a 1/10-scale autonomous vehicle powered by pre-collision detection, lane tracking, and road sign recognition systems. The pre-collision system is built using ultrasonic sensors, and the Proportional-Integral-Derivative (PID) control is implemented to manipulate the vehicle's safety response. The Open-Source Computer Vision Library (OpenCV) is exploited to detect and process real-time on-road streaming video to enable lane-tracking and road sign recognition. AI techniques are utilized for the model training. Preliminary results of this work are presented and analyzed. We also discuss the future directions of this study.**

Keywords—Autonomous vehicle, computer vision, PID control, hands-on engagement, STEM education

I. Introduction

Transportation plays a pivotal role in our daily activities and productivity. As more vehicles are integrated into society, we have experienced more congestion, deteriorating environmental conditions, more commuting time for employees, increased land needs for vehicles, and increased risk of injury and death. Traffic congestion (TC) is defined as the disbalance between the number of vehicles and available traveling areas [1]. TC in both developed and developing nations result in individuals spending an average of two to three hours each day in their vehicles to conduct their daily activities [2]. In 2017, TC in urban areas cost a total of $179 billion dollars in the United States. In addition, TC accounts for the loss of an estimated 8.8 billion hours and the waste of 3.3 billion gallons of fuel. Without inconsistencies, commuting time is expected to continue its pattern of increasing by 15% every five years [3]. The ecological damage caused by TC is another major cause of concern associated with the current transportation model. TC is the leading cause of greenhouse gas emissions and is responsible for about 60% of pollutants in the air, with 90-95% of these pollutants originating from private vehicles [4]. Furthermore, in 2018 there were over 36,000 fatal crashes in the United States alone, with human errors such as driving while intoxicated, not wearing seatbelts, speeding and driving while distracted being the leading factors involved in vehicular crashes [5, 6]. In a growing economy, it is imperative to restructure and adapt the transportation models to better suit the needs of citizens everywhere in the world and develop technologies that contribute to a healthier environment and way of life [3, 4].

Autonomous vehicles (AVs) aim to solve traffic congestion and its associated problems to improve quality of life. As such, various companies and organizations in the developed nations are focusing their investments and developments towards a future with fully integrated AVs. In the US, Ford will be launching autonomous driving system (ADS) fleets in Miami, Austin, and Washington DC. Waymo has about 600 active AVs fleets in Arizona and has partnered with UPS to use driverless delivery vans in Phoenix and Tempe. Lyft and Aptiv have collaborated to provide over 75,000 driverless rides in Las Vegas. Enterprises such as Walmart, Domino's Pizza, and Coca-Cola are testing the integration of autonomous vehicles in their delivery services with companies such as NURO and Swedish company Einride, respectively. Tesla, Volvo, and Toyota have begun training and testing driverless fleets throughout the US, with many creating partnerships to further advance the adaption of AVs in businesses [7].

While traffic conditions are expected to worsen during the beginnings of AVs integration, it is expected that the benefits of AVs will improve the general transportation model significantly over time [7]. It is also worth mentioning that social and behavioral factors may cause a delay in the adaption of AVs. However, since technical tools and autonomous vehicles continue to improve in performance, the most optimistic predictions state that autonomous fleets will grow in commonality, affordability, and normalization in the 2030s and 2040s [1, 8, 9]. Despite the challenges it may present, there is strong evidence to indicated that the future of transportation involves the widespread adoption of autonomous vehicles.

Due to the effects that autonomous vehicles will have in all of our lives in the near future and their multidisciplinary requirements, we believe it is imperative to engage students across STEM fields in learning about autonomous vehicles in anticipation of their adoption in society and job markets. We present a tangible approach of learning about AVs through the development of a 1/10-scale model on a scale down real-world environment using the project-based learning pedagogy.

II. Educational Benefits Of Autonomous Vehicles

A. Project-Based Learning Pedagogy

Project-Based Learning (PBL) pedagogy theory was first introduced in 1959 [10]. It is actively used as a teaching method for students because it provides many benefits. PBL fosters independent critical thinking skills, problem-solving skills, and research skills. In PBL, students are given a problem and the tools necessary to solve the problem with independence and support from the instructor allowing them to be creative in their decision making and to perform research on their interested problems. In solving problems using PBL, students can collaborate with each other, make decisions together, and distribute responsibilities to each group member's strength and interests. PBL allows students to improve their communication and teamwork skills thus better preparing them for the level of collaboration that exists in real work environments [11]. In addition, PBL allows students to build a strong learning foundation as they apply contextual learning onto projects, establishing a more profound, stronger understanding of the subject matter [10, 11]. The foundation of PBL utilizes four scientific learning ideas: 1) active construction 2) situated learning, 3) social interaction, and 4) cognitive tools [10].

Active construction aims to establish the connection between the subject matter and the world in a way that makes sense to the students. This connection is important for deep learning. PBL makes strong use of this learning idea as students interpret the problem while being given the independence to provide a solution as it best applies to them [10].

Situated learning builds on the use of active construction by further contributing to the student's connection of the subject matter to the real world. This learning idea relies on the application of scientific concepts, further strengthening learning and establishing the significance of the students' project. Situated learning allows students to see the direct impact they can make globally, thus establishing relevance and a deeper connection to their roles as scientists [12].

Social interaction provides personal and professional enrichment to the students by helping them develop communication and leadership skills. PBL uses social interaction as students are asked to collaborate and work together and defend their ideas both within their team and while they are presenting to others [10].

Cognitive tools integrate all learning ideas by calling onto students to present data in a visual matter. Students gain a deeper understanding of their findings while creating visualizations such as graphs and multimedia presentations that facilitate understanding to their audience. While using cognitive tools, students retrace their steps and outline the process of solving their problem [10].

In our 1/10-scale project, we make use of the benefits of the scientific learning ideas presented by using the PBL pedagogy. Students will learn valuable skills and widely used technologies with a clear explanation of their application and the impact that it will have in their lifetime.

B. Software Integration Across All Domains

The software has become an integral part of daily life. It plays a pivotal role in the advancement of science and technology as most, if not all scientists and engineers rely on software tools to create simulations and interpretations of their data in work environments and research. Within STEM, disciplines create or make use of existing software for their investigative needs, such as creating mathematical and computational models [13]. Beyond STEM, software applications usage extends to a multitude of disciplines such as Social Sciences and the Arts [14]. The software makes it possible to create, collaborate, and share inventions, knowledge, and tools. It is highly beneficial for students outside of the software engineering field of study to have a general introduction to the development of software and its applications. In the 1/10-scale autonomous vehicle project, students will be required to use code and algorithms to develop the AV's features. Apart from learning new skills, students will benefit from being exposed to code and software prior to entering their professional environments, as they will most likely utilize software tools throughout the course of their careers and education.

III. Hardware System

The 1/10-scale autonomous vehicle is intended to model the basic standard features required for autonomous driving [15]. The priority features that we've incorporated into our model include road sign recognition, collision avoidance, and lane tracking systems. These safety features take precedence because safety is the most important factor in the success of an AV. Performing the basic operations of our AV is the actual physical vehicle and the necessary hardware components. To efficiently build our model, we require a 1/10-scale racing truck, a mini-PC, a MEGA 2560 Arduino board, a high-resolution camera, and three HC-SR04 ultrasonic sensors.

A. Vehicle Model

We have selected a short-coursed racing truck as our model primarily because of its size, its comparable Magnum 272™ transmission with 2-Wheel Drive (2WD), and torque-control™ slipper clutch. The waterproof Titan® 12-Turn 550 motor is equipped with XL-5 electronic speed control and comprehensive steering capabilities. The built of this model provides some similarities with real-life driving experiences, and its watertight features will allow us to test the features in our autonomous project in different weather conditions [16].

B. Computer System

On top of the chassis of the vehicle, we have incorporated a mini-PC with an Intel Atom® x5-Z8350 Processor [17]. The mini-PC is configured to run Ubuntu 16.04 as its operating system to feasibly incorporate the software features [18]. This PC was chosen over other computing methods because its compact size and lightweight structure of 97 mm x 97 mm. Its 40 GB hard drive and 4 GB memory make it a strong and expandable addition to our small vehicle.

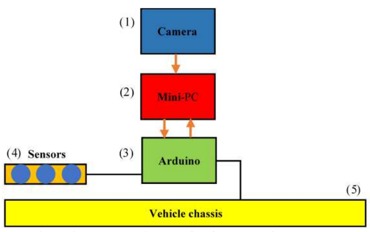

Fig. 1. Hardware implementation of 1/10-scale autonomous vehicle.

C. Arduino Platform

In addition to the mini-PC, we added a MEGA 2560 Arduino board which comes equipped with 54 digital input/output pins, 16 analog inputs and 4 serial ports. The

Arduino board plays a significant part in developing our autonomous features by collecting and managing the data used in our pre-collision system and sharing it with the mini-PC running the algorithms that determine the course of action for the vehicle based on this data. The Arduino then executes the autonomous commands directly to the vehicle via output pin connections, as demonstrated in Fig. 1.

D. Camera

A high-resolution camera is mounted on top of the chassis to collect real-time data needed for the vision system. The vision system is used to recognize road signs and lane markings that dictate whether it is appropriate for the vehicle to accelerate, decelerate, and steer, respectively. The camera communicates the real-time image data to the mini-PC while it executes the algorithmic response to the real-time data as the environment changes. Similarly, the mini-PC communicates the algorithmic commands to the Arduino board, which directly controls the vehicle's response. Fig. 1 provides a demonstration of the relationship between hardware components for the functionality of the autonomous vehicle. The orange arrows represent the collection and sharing of data between each hardware component to perform its function.

E. Pre-Collision System

Lastly, three HC-SR04 ultrasonic sensors are used to implement the collision avoidance feature. The sensors are able to detect objects or pedestrians in its path within the range of 0.02 m to 4 m. Their functionality works by releasing ultrasonic waves into the environment originating from the transmitter and measuring the distance of the reflection of the obstruction.

In this project, the distance is calculated using the formula:

$$distance = (343\, m/s\ * T)/2 \qquad (1)$$

The speed of sound wave is denoted by $343\, m/s$ while T references travel time of sound. The sensors are placed in front of the vehicle collecting data in real time to continuously check if there are obstacles in front of the vehicle and to measure the distance between the nearest obstacles and the vehicle. We have selected a setpoint of 20.00 cm as the minimum distance between the vehicle and obstruction for the PID algorithm (further explained in the software section) to be triggered. The sensors support the pre-collision system by ensuring that there is a safe distance between the moving vehicle and obstructions and pedestrians.

IV. Software and Algorithm

A. Ubuntu Operating System

Ubuntu is a Linux-based open-source operating system (OS). Ubuntu has established its strength as a preferred OS for emerging technologies, and as of 2018, it is the third most used OS in the world. The future of Ubuntu OS is promising as applications spanning various technical areas such as robotics, machine learning, autonomous vehicles, data science, internet of things (IoT), among others are built on the OS due to its versatility, security, and integrative abilities [19]. For our project, we installed Ubuntu 16.04 as the operating system running in the mini-PC.

B. Robot Operating System

The Robot Operating System (ROS) is an open-source collection of robot-specific software and libraries [20, 21]. The ROS platform is a middleware platform that helps in the integration of complex features. Some areas of focus include navigation, mapping, pose estimation, diagnostics, and robot geometry library. In our project, ROS will be used as the bridge between the data collected from the sensors attached to the Arduino board and the data collected from the high-resolution camera. Real-time data is crucial to the safety features integrated into our mobile robot. ROS is a strong tool that makes real-life data integration possible.

C. Open Source Computer Vision Library

The goal of computer vision is to emulate human vision [22]. The high-resolution camera is the hardware capturing the environment in front of our vehicle. However, we use Open Source Computer Vision Library (OpenCV) to perform the computational analysis for road sign recognition. OpenCV is a library of specialized computer vision algorithms. It contains powerful functions for image processing, image classification, object tracking, and motion detection [23, 24]. OpenCV is heavily integrated into the fields of Artificial Intelligence and Machine Learning projects that involve vision classification and regression problems. It is powerful and versatile with modules available in various programming languages [23]. We use OpenCV with Python in our project to develop the lane tracking and road sign recognition features.

D. Arduino Integrated Development Environment

Arduino boards and the Arduino Integrated Development Environment (IDE) are open-source platforms that incorporate hardware and software [25, 26]. Arduino boards are used to create technical IoT applications, wearable technology, mobile technologies and have many other uses and applications that combine electronics and programming. Arduino provides a number of libraries that can be used in Windows, Linux, and Mac operating systems allowing users from beginning to advanced stages a simple and easy-to-use interface and programming environment with strong versatility and capabilities [25]. Our project uses Arduino IDE to program the MEGA 2560 board to communicate and control with the hardware. The Arduino board receives input from the sensors and the mini-PC and outputs our vehicle's motor and servo systems.

E. Proportionate-Integral-Derivative Control

One of the triggers on the Arduino IDE is based on the distance output from the sensors, which are used for the pre-collision and collision avoidance system. The Proportionate-Integral-Derivative (PID) control is used in this study [27, 28].

The proportional (K_p) could evaluate and decrease errors that warrant a response [28]. Errors are the differences between the setpoint and process variable. In our autonomous project the setpoint is 20 cm and the process variable is the proximity of an obstacle read by the sensors [28]. If there are no obstacles within the setpoint distance of 20 cm then there are no errors. Integral (K_i) is a constant responsible for eliminating the errors identified by K_p . K_i fixes the errors by adding all of the errors from start to finish and basing the output force on the intensity of the errors [27, 28]. The higher the cumulative error, the stronger the K_i output. It continues to calculate the cumulative error and adjusts the brake pressure accordingly. The primary function of the K_i is to prevent oscillation by fixing the errors established by K_p. K_i is the backbone of PID and what makes the driving experience of our autonomous vehicle as seamless as possible. Lastly, the derivative (K_d) is responsible for tracking the response of the

K_i to ensure that the rate of change performed by the K_i is appropriate [27, 28].

In our 1/10-scale autonomous vehicle, PID works with the sensors and Arduino board as they track real-time distance data of obstructions in front of the vehicle. When the distance between the vehicle and an obstruction matches the setpoint of 20.00 cm or less, PID is triggered to output a response accordingly. PID is the acceleration and deceleration agent that controls the vehicle's brake force depending on the distance between the vehicle and obstruction. Thus, the closer the object is to the vehicle, the stronger the PID break response. After a strong break response from PID, the algorithm is able to stabilize and continue driving after the threat of collision has been removed. Similarly, the acceleration force is controlled to avoid oscillation.

V. Preliminary Results

A 1/10-scale autonomous vehicle has been preliminarily developed as a learning tool for students in STEM. The vehicle has standard key features that emulate some of the key features in life-sized AVs, including a pre-collision system, lane tracking, and road-sign recognition. We use Arduino hardware, software, and PID algorithm to develop these features. The road sign recognition is fortified using a high-definition camera as the hardware and OpenCV library algorithms for real-time image processing and classification for the road sign recognition feature. The lane tracking system also relies on the high-definition camera and the OpenCV library to process real-time video footage of the environment. This vehicle will act as a model to demonstrate the final results and features that the students will be working towards.

VI. Conclusions and Future Work

The 1/10-scale autonomous vehicle project is intended to introduce students to the fast-growing industry of autonomous vehicles. Students will learn about AVs through the PBL pedagogy. PBL allows students to strengthen critical, independent thinking, and creative skills as well as improve communication through collaboration with others. The project uses a variety of technologies throughout multiple disciplines like Arduino, OpenCV, and ROS, further enriching the students' educational experience. In this paper, we have explored the role of AVs in the future of society, the environmental benefits of AVs adoption as part of a new transportation model. We have broken down the different components required to successfully build the model. The three major features incorporated in our model include pre-collision, road sign recognition, and lane-tracking systems.

Future developments of this project involve expanding access to all students who are interested in learning robotics, autonomous vehicles, computer vision, and any of the aforementioned technologies. We believe that it will be very beneficial for students to partake in this hands-on project. We plan to conduct research and collect data both on the students' impression of the project and on the features they develop for future improvements of the model and learning experience.

Acknowledgment

This research is partially supported through a STEM +Computing grant from the Division of Research on Learning of the National Science Foundation (# 1742125). The authors would thank the great support from the Department of Computer Science at Montclair State University.

References

[1] A. Ongel, E. Loewer, F. Roemer, G. Sethuraman, F. Chang, and M. Lienkamp, "Economic Assessment of Autonomous Electric Microtransit Vehicles," *Sustainability,* vol. 11, p. 648, 01/26 2019, doi: 10.3390/su11030648.

[2] V. Jain, A. Sharma, and L. Subramanian, "Road traffic congestion in the developing world," *in Proceedings of the 2nd ACM Symposium on Computing for Development*, 2012, pp. 1-10.

[3] D. Schrank, T. Lomax, and B. Eisele, "2012 urban mobility report," *Texas Transportation Institute*. [Online]. Available: http://mobility .tamu.edu/ums/report, 2012.

[4] H. Chin and M. Rahman, "An impact evaluation of traffic congestion on ecology," *Planning Studies & Practice,* vol. 3, pp. 32-44, 10/18 2011.

[5] National Highway Traffic Safety Administration. (2018). *National Statistics.* [Online]. Available: https://www-fars.nhtsa.dot.gov/Main/ index.aspx.

[6] U.S. Department of Health & Human Services. (2016). *Motor Vehicle Crash Deaths*. [Online]. Available: https://www.cdc.gov/vitalsigns/ motor-vehicle-safety/index.html.

[7] F. Agenda, "The future of autonomous vehicles," in "Global insights gained from multiple expert discussions," April 20, 2021. [Online]. Available: https://www.futureautonomous.org/pdf/ full/Future%20of %20Autonomous%20Vehicles%202020%20-%20Final%20LR.pdf.

[8] C. Bloom, J. Tan, J. Ramjohn, and L. Bauer, "Self-driving cars and data collection: Privacy perceptions of networked autonomous vehicles," *in Thirteenth Symposium on Usable Privacy and Security*, 2017, pp. 357-375.

[9] T. Litman, "Autonomous vehicle implementation predictions: Implications for transport planning " Victoria Transport Policy Institute, March 5 2021. Accessed: April 20, 2021. [Online]. Available: https://www.vtpi.org/avip.pdf.

[10] J. S. Krajcik and P. C. Blumenfeld, *Project-based learning*. na, 2006.

[11] V. Rousová, "Project-based learning: Halloween party," Diploma thesis Faculty of Education, Brno, Masaryk University, 2008. [Online]. Available: https://is.muni.cz/th/qzgwd/?lang=en.

[12] W. Wang, C. Coutras, and M. Zhu, "Situated Learning-Based Robotics Education," *in 2020 IEEE Frontiers in Education Conference (FIE)*, 2020: IEEE, pp. 1-3.

[13] J. E. Hannay, C. MacLeod, J. Singer, H. P. Langtangen, D. Pfahl, and G. Wilson, "How do scientists develop and use scientific software?," *in 2009 ICSE Workshop on Software Engineering for Computational Science and Engineering*, 2009, pp. 1-8.

[14] R. Melnik, "Universality of Mathematical Models in Understanding Nature, Society, and Man-Made World," in *Mathematical and Computational Modeling*, 2015, pp. 1-16.

[15] M. K. Dimitris Milakis, Bert van Wee, "Implications of automated vehicles for accessibility and location choices: Evidence from an expert-based experiment," *Journal of Transport Geography,* vol. 68, pp. 142-148, 2018.

[16] Traxxas. https://traxxas.com/products/models/electric/58024slash (accessed April 21, 2021).

[17] Intel. "Intel Atom® x5-Z8350 Processor.".

[18] Ubuntu. https://releases.ubuntu.com/16.04.7/.

[19] S. Dickinson, "Six reasons why developers choose Ubuntu Desktop," [Online]. Available: https://ubuntu.com/blog/six-reasons-why-developers-choose-ubuntu-desktop.

[20] S. Bier, R. Li, and W. Wang, "A Full-Dimensional Robot Teleoperation Platform," *in 2020 IEEE International Conference on Mechanical and Aerospace Engineering*, 2020: IEEE, pp. 186-191.

[21] M. Quigley et al., "ROS: an open-source Robot Operating System," in ICRA workshop on open source software, 2009, vol. 3, no. 3.2: Kobe, Japan, pp. 1-6.

[22] J. F. Peters, *Foundations of Computer Vision* Switzerland: Springer International Publishing 2017.

[23] G. Bradski and A. Kaehler, Learning OpenCV: Computer vision with the OpenCV library. " O'Reilly Media, Inc.", 2008.

[24] OpenCV. "Modules." https://docs.opencv.org/master/modules.html (accessed April 22, 2021).

[25] Arduino. "What is Arduino?" https://www.arduino.cc/en/Guide /Introduction (accessed April 24, 2021).

[26] M. Fezari and A. Al Dahoud, "Integrated Development Environment "IDE" For Arduino," WSN applications, pp. 1-12, 2018.

[27] D. Semiconductor, *Position and speed control of a DC motor using analog PID Controller*: Dialog Semiconductor, 2018.

[28] M. A. Johnson and M. H. Moradi, PID control. Springer, 2005.