1-2022

Development and On-Road Applications of a 1/10-Scale Autonomous Vehicle

Laura Cornejo Paulino
Montclair State University

Abstract

Autonomous vehicles (AVs) are a key component in the creation of the new transportation infrastructure. Over the last several decades, nations across the world have experienced an increase in traffic congestion, environmental deprecation due to greenhouse gas emissions and an increase in time loss and productivity. A key factor in these components is the increase in numbers of vehicles on the road, a number that continues to increase gradually every year. In addition, the continued increase in vehicles on the road poses a threat to human and environmental safety. There is strong evidence to support that accidental vehicular deaths and injuries are highly correlated to driver impairment, further supporting the need to address human errors and the "human problem" in the existing transportation model.

Autonomous vehicles provide opportunities to mitigate the inefficiencies of the current transportation model by aiding, and in future cases, removing human agents as well as possibly reducing the number of vehicles on the road. Both enterprises and governmental entities have begun the process of exploring the various ways that they can benefit from adapting autonomous vehicles. As of today, several companies have autonomous driving fleets in US cities making promising strides towards a driverless future. The Victoria Transport Policy Institute estimates that the integration of AVs should occur within the next three decades, with a peak in public acceptance in the 2040s to 2060s.

This thesis explores the current state of development of AVs integration in a political, technological, and social context. It explores the potential benefits and challenges of self-driving vehicles, the typical construction of autonomous vehicles, and the typical safety features integrated in every AV. This thesis also provides an informative approach to developing an educational 1/10th-scale autonomous vehicle with three of the most important features: PID-based collision avoidance features, a lane-tracking system, and a stop sign recognition feature. The quantitative

and qualitative results of the implementation of the 1/10th-scale AV are verified and analyzed, and a plan of future improvements is presented.

MONTCLAIR STATE UNIVERSITY

Development and On-Road Applications of a 1/10-Scale Autonomous Vehicle

by

Laura Cornejo Paulino

A Master's Thesis Submitted to the Faculty of

Montclair State University

In Partial Fulfillment of the Requirements

For the Degree of

Master of Science

January 2022

College of Science and Mathematics

Department of Computer Science

Thesis Committee:

[redacted]

Dr. Weitian Wang

Thesis Sponsor

[redacted]

Dr. Michelle Zhu

Committee Member

[redacted]

Dr. Jiayin Wang

Committee Member

DEVELOPMENT AND ON-ROAD APPLICATIONS OF A 1/10-SCALE AUTONOMOUS VEHICLE

A THESIS

Submitted in partial fulfillment of the requirements
For the degree of Master of Science

by

Laura Cornejo Paulino

Montclair State University

Montclair, NJ

2022

Acknowledgements

I would like to express my sincerest gratitude to my thesis advisor, Dr. Weitian Wang. You made a significant impact in my life and career through your never-ending support, guidance, and encouragement. Thank you for welcoming me in your lab and sharing your expertise and excitement for the field of robotics. My pursuit of this degree took a significant positive turn after I began working with you, especially after discovering we shared the same passion for autonomous vehicles. This would not have been made possible without you being there every step of the way. Thank you!

I would also like to thank Dr. Michelle Zhu for being a source of support and being committed to the success of this project and letting us take over your lab. Thank you to Mihir Rao, Victor Robila, and Iris Li for their work and contributions to this project. It was a pleasure working with you all. Dr. Jiayin Wang, thank you for your support as a committee member for my thesis. I would like to thank the department of Computer Science at Montclair State University. A special thank you to Dr. Aparna Varde and Dr. Hongbo Zhou for your encouragement in the early stages of my studies. I would also like to thank my NASA mentor, Dr. Nicholas Cramer, for providing the opportunity to learn under his expertise and making learning difficult things much more pleasant.

On a personal level, I would like to thank my parents, Gladys Maria Paulino and Jose Raul Cornejo, who came to this country with nothing so that I may build on their foundation. Your courage and sacrifices are my driving force. Thank you for teaching and encouraging me to pursue my happiness. I love you both beyond words. Finally, I would like to thank my partner, Miles, for his love and support throughout this journey. The hard days of this process would have been impossible without you. I love you.

Table of Contents

CHAPTER I INTRODUCTION 1

CHAPTER II LITERATURE REVIEW 6

2.1 Autonomous Driving Technology 6

2.2 Typical Autonomous Vehicles 11

2.3 Safety Systems in Autonomous Vehicle 17

CHAPTER III OVERVIEW OF THE AUTONOMOUS VEHICLE 21

CHAPTER IV HARDWARE SYSTEM 22

4.1 Overview of the Hardware System 22

4.2 The 1/10th Vehicle 22

4.3 Computing System 23

4.4 Vision System 23

4.5 Pre-Collision System 23

4.6 Arduino MEGA 2560 Board 24

CHAPTER V SOFTWARE SYSTEM 25

5.1 Architecture of the Software System 25

5.2 Robot Operating System (ROS) 25

5.3 OpenCV 25

5.4 Arduino IDE 26

5.6 Visual Studio Code 26

5.8 Ubuntu 26

CHAPTER VI PID-BASED PRE-COLLISION SYSTEM 27

6.1 PID Control Algorithm 27

6.2 Collision Avoidance via PID Control 29

CHAPTER VII LANE TRACKING SYSTEM 31
7.1 Introduction to Lane Tracking 31
7.2 Lane Tracking Approach 31
CHAPTER VIII STOP SIGN RECOGNITION SYSTEM 35
8.1 Stop Rules 35
8.2 Stop Sign Recognition Approach 36
CHAPTER IX ON-ROAD APPLICATION RESULTS AND ANALYSIS 39
9.1 Experimental Setup 39
9.2 Collision Avoidance Results and Analysis 40
9.3 Lane Tracking Results and Analysis 41
9.4 Vehicle Stop-Sign Results and Analysis 42
CHAPTER X CONCLUSIONS AND FUTURE WORK 45
10.1 Conclusions 45
10.2 Future Work 46
REFERENCES 48

List of Figures

Figure 1 3

Figure 2 12

Figure 3 15

Figure 4 15

Figure 5 21

Figure 6 22

Figure 7 23

Figure 8 27

Figure 9 29

Figure 10 29

Figure 11 30

Figure 12 31

Figure 13 32

Figure 14 33

Figure 15 34

Figure 16 35

Figure 17 37

Figure 18 38

Figure 19 38

Figure 20 39

Figure 21 40

Figure 22 41

Figure 23 42

Figure 24 43

Figure 25 43

Figure 26 43

CHAPTER I INTRODUCTION

Motion is a fundamental need for human beings. In most parts of the world, we have evolved from traveling thousands of miles on foot, to relying on animals, and in the latest course of development, relying on machinery. The human need to move has shaped history and it continues to be a pressing matter and priority in our quality of life and development as a species. Since the first gasoline-powered automobile by Karl Friedrich Benz in 1885, our transportation model has had little to no evolution, with strong reconsiderations and investments happening in this sector in recent years [1].

Our current transportation model still relies on Benz's technology of using gasoline, which, along with the assembly line manufacturing model invented by Ford, have made it possible for most people in the developed world to have access to a vehicle [2]. While this system has helped drive humanity forward, the side effects of our current transportation model is posing a threat to humanity and the planet, threatening global health and safety, and the global economy. Some of the biggest challenges posed by the oil industry is its negative effects on the environment and public health. The current transportation model relies on petroleum drilling to power vehicles. The environmental effects of drilling for petroleum and burning of oil to power machinery is beginning to affect and threaten life as we know it. Newer methods in the extraction of oil seem to make matters worse. According to the United States Energy Information Administration, hydraulic fracturing or "fracking" is a technique used to drill for oil on land that requires large amounts of water to separate rock from the oil. Due to extensive water requirements, fracking can severely reduce water access to communities nearby [3]. Furthermore, water leaks from the fracking process can contaminate wells and drinking water which have been linked to the development of diseases in humans such as cancer, liver damage, neurological disorders, and immunodeficiency [4]. In addition, oil drilling through oceanic floors has led to oil spills such as the 1989 Exxon Valdez spill in Alaska and the 2010 Deepwater Horizon spill in the Gulf of Mexico, to name a few. These incidents compromise the health of marine life and aquatic creatures, further implicating the health of our oceans and its important role in the overall health of the planet [5].

After extraction, the negative effects of oil continue throughout its use cycle. The 2020 PBL Netherlands Environmental Assessment Agency report found that 2019 was the second warmest year, with "ocean surface and global land temperatures being +0.95 ° C higher than average" [6]. Additionally, in 2019, fuel combustion accounted for 88.6% of carbon dioxide (CO_2), the main gas found in greenhouse gas emissions, released in the atmosphere [7]. One of the biggest contributors to greenhouse gas emissions is traffic congestion (TC) which contributes roughly 60% of air pollutants, 90-95% of which are caused by privately owned vehicles [8]. Traffic congestion is defined as the over saturation of vehicles in comparison to available driving areas and it is a growing problem in both developed and developing nations [9]. In addition to damaging the environment, TC is responsible for an increase in commuting time, increase in vehicle land demands to create roads and highways, as well as an increased risk of injury and death. In both developed and developing nations, TC accounts for individuals spending an average of two to three hours a day in their vehicles to conduct their daily tasks [10]. Moreover, as of 2017, the cost of TC in the United States' urban areas was $179 billion dollars and an average of 8.8 billion hours, and 3.3 billion gallons of fuels are wasted each year due to TC [11]. To make matters worse, commuting time has a pattern of increasing by 15% every five years, a continued increase that could result in catastrophic environmental conditions and quality of life [11]. The research suggests that the impact of petroleum as the main source of energy for transportation is unsustainable if we aim to improve quality of life, human development, and safety. The threats that oil drilling poses on the environment will severely outweigh its benefits in the near future.

Conversely, another opportunity for improvement in the current transportation system is the need for human agents to operate vehicles. Since the inception of the commercially accessible vehicle, predominantly following the establishment of Ford General Motors in 1903, safety concerns pertaining to human drivers have been in the spotlight. By the 1920's, about 200,000 Americans were killed by human drivers, resulting in more casualties caused by vehicular crashes than those in the First World War. The dangers to public safety led the public and policymakers to reconsider the effectiveness of the human-operated vehicle [12]. This public safety threat along with the invention of the autopilot system in the 1910's by the United States military allowed the public to start fantasizing about cars driven without human agents in the early years of their adoption, over a hundred years ago. By the 1950's the idea of autonomous vehicles was commonplace, attached to the societal values of family time and safety [12].

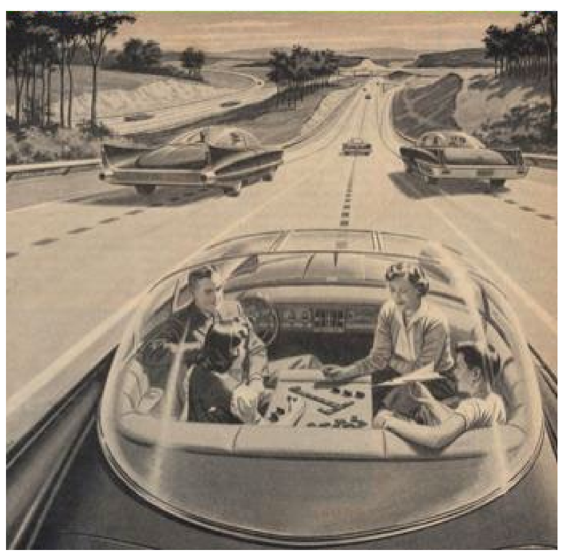

Figure 1 Electric company advertisement for autonomous vehicles - Magazine vol.40, Nr. 5, 30. January 1956, p.8. Image rights: copyright by the Author [13].

Strong contributions to the recent developments of autonomous driving came back into focus in the 1980s when further advances to computer and machine vision were made in association with organizations such as the European Traffic of Highest Efficiency and Unprecedented Safety, Mercedes Benz, and NavLab of Carnegie Mellon University [12]. Both in a cultural and technological perspective, the integration of autonomous vehicles has developed at a slower pace than the first enthusiasts of the 1920s had hoped. Nonetheless, the dream for an alternate transportation system remained, as vehicular crashes continue to be a public safety concern today.

As of 2020 over 39,000 people lost their lives due to vehicular crashes in the United States contributing to the 1.35 million vehicle-related yearly deaths globally [14, 15]. Human errors and driving behaviors including speeding, not wearing seatbelts, driving while distracted and driving while intoxicated are the leading factors involved in vehicular crashes [15, 16]. Human errors coupled with increased traffic congestion, and worsening environmental conditions pose a big threat to the development of our species and our need to move. As such, it is imperative to restructure and adapt the transportation model to better suit the needs of citizens everywhere in the world and develop technologies that contribute to a healthier environment and way of life [8, 11, 17] . Both policymakers and enterprises alike are beginning to discuss and incorporate autonomous vehicles in their plans, paving the way to a much-needed update to the way that we move. At last, it appears the time has come to design, develop, and evolve our transportation system.

There is strong evidence (further presented in the next portions of the thesis) to indicate that the widespread adoption of electric autonomous vehicles is in the near future. Because of this, it is imperative to engage students across STEM fields in learning about autonomous vehicles to better prepare them for the workforce. Therefore, we develop a 1/10th-scale autonomous vehicle featuring three of the key safety features for operation. In this work (1) a PID-based control system is developed for the acceleration and braking force with the capacity to initiate braking response when detecting obstacles; (2) a lane tracking system is implemented to monitor and regulate the vehicle's operating space; (3) a vision system is designed to recognize road signs in the environment that can also communicate with the PID control to trigger a brake response.

The potential benefits of this small-scale system include its ability to provide tangible practice in the development of AVs at a much lower cost than a full-size vehicle. We believe that the introduction to this technology at a collegiate level, not only will help address the public's acceptance of AVs but it will demystify the tools and components required to build them while providing a foundation for future employees of the automotive industry and the multidisciplinary requirements.

CHAPTER II LITERATURE REVIEW

2.1 Autonomous Driving Technology

The single most important goal in successful driving is to avoid collision while going from point *a* to point *b*. For a variety of reasons, including complex dynamic environments and human errors, many people throughout the world struggle to achieve driving success. Throughout the years, interventions have been incorporated in driving environments and vehicles to help individuals achieve this goal. However, the challenges persist as human errors and driving behaviors have proven to be difficult to change. An autonomous vehicle or self-driving car is a vehicle that can perform some or all functions of driving. Over the last two decades developments in software and hardware technologies have allowed for improvements to key components required in a successful autonomous vehicle. Advancements in sensors, machine learning algorithms, computer vision tools, and communication technologies work together to get AVs closer to the vision of the 1950s [18]. In a traditional vehicle, humans are responsible for acceleration, braking, and steering. The goal of a fully autonomous vehicle is to relieve the human of all or some of these responsibilities while driving. There are currently six different levels of autonomy described in the table below [19]:

Table 1 Levels of Operation.

Level 0: No Automation	Traditional vehicle where the human driver is responsible for all driving operations.
Level 1: Driver Assistance	The system is responsible for one of the functions of driving. A good example of driver assistance is cruise control where the vehicle is automated to control the break and acceleration pedal.
Level 2: Partial Automation	In this level acceleration and steering fall under the responsibility of the system. However, the human agent is still considered the driver and s/he is responsible for monitoring and responding to the environment accordingly.

Level 3: Conditional Automation	In this level the system performs most or all driving operations, but it may illicit support from the human agent thus the human driver must stay alert.
Level 4: High Automation	The system can drive itself as long as its required environmental and operational conditions are met.
Level 5: Full Automation	The autonomous system has full control and capacity of driving in any place, at any time.

The features of Levels 0 and 1 are common in today's marketplace where most drivers have cruise control accessibility. Level 2, partial automation, has launched in recent years, most popularly, in lane assist and emergency braking features commonly found in vehicles released within the last five years. Tesla's autopilot feature is a prime example of Level 2 automation [20]. In the US, Ford will be testing Level 3 autonomous driving system (ADS) fleets in cities like Miami, Austin, Pittsburgh, and Washington DC. In Arizona, Waymo has about 600 active Level 4 AV fleets in circulation providing car share rides to individuals with plans of expanding its model to the delivery of goods through a partnership with UPS in Phoenix and Tempe. Major rideshare company Lyft, in collaboration with autonomous driving software company Aptiv, has provided over 75,000 Level 5 driverless rides in Las Vegas. Enterprises such as Coca-Cola, Walmart and Domino’s Pizza began testing Level 4 autonomous vehicles in their delivery services with companies like NURO and Swedish company Einride. Additionally, Uber, Tesla, Volvo, Toyota, and China’s Baidu have partnered with various US enterprises to test the ways autonomous vehicles can be integrated in their business models [21]. Recent political focus and business investments in Level 4 and Level 5 autonomous vehicles are a strong indicator that the adoption of AVs is approaching.

Although investments and technological advancements point in the right direction, AVs must overcome many challenges before its full integration into society. One of the biggest challenges is the legislative infrastructure. In the congressional report "Issues with Autonomous

Vehicle Testing and Deployment," author Bill Canis explains some of the key concerns that need to be addressed by the United States Congress to enact legislation pertaining to AVs. First, Congress must agree on the distribution of regulation and responsibilities of AVs. In our current model, the federal government is responsible for vehicle's safety regulations while each state regulates the driver through processes such as registrations and licenses. Congress struggles to agree on the distinction between the two in autonomous vehicles since the lines are blurred as the level of automation increases. Second, the National Highway Traffic Safety Administration (NHTSA) must determine the number of AVs that it will allow to test on the road while being exempt from the safety standards and it must specify the safety standards which the AVs will be exempt from. Third, congress must agree on the level of privacy and cybersecurity requirements in AVs. The key question in this matter is the extent to which companies should inform and protect buyers from cybersecurity threats such as hacking of critical vehicular systems. Lastly, congress must agree on the limitation of data collection, selling and buying of vehicular data, including who should be able to buy and sell [22].

Another, perhaps, more pressing challenge is public acceptance. Since the inception of autonomous vehicles, the way in which they've been represented in the media through various decades has affected public opinion. In his article "Automated Driving in Its Social, Historical and Cultural Contexts", Author Fabian Kröger lists the examples of public perception about autonomous vehicles over the century. In the mid to late 1930s, for example, the excitement for AVs quickly diminished after some negative narratives were presented in stories like The Living Machine by David H. Keller where self-driving cars were depicted as out of control autonomous systems on a hunt for humans. In the 1950s, however, magazines like Popular Science Magazine GM's Futurama shows at the World's Fair contributed to the Jetsons-esque excitement and support for an automated transportation system where AVs were highly regarded [12]. Today, public acceptance extends beyond the retro-futuristic values of comfort, luxury, and family time. The public's willingness to transition from human operated vehicles to autonomous vehicles is predominantly measured by safety. There have been several incidents in recent years that have made the public skeptic about the safety value of AVs. More famously, in 2018, an Uber operated AV struck and killed Elaine Herzberg, a pedestrian in Arizona who became the first person to be killed by an AV in history [23]. A young woman, Naibel Benavides, and several others were killed to errors which the public attributed to Tesla's autopilot feature failing [24]. Tesla, which perhaps has the worst safety reputation in this sector, has faced several lawsuits since its release of autopilot in 2015. In several instances involving autopilot, however, some of the drivers were not using the feature properly, such is the case with two Texas men who were involved in a fatal crash in April of 2021, where physical evidence determined that there was no one in the driver's seat, failing to comply with the instructions provided by the company for proper operation [25]. In the case of Tesla's autopilot, some critics claim that the name of the feature is misleading, causing enthusiasts to believe that the feature is more advanced than it is [26]. To their point, integration of features that will eventually lead to level 5 autonomy will require clear and concise messaging regarding the different autonomy levels. Autopilot is a level 2 automation, which should have never led anyone to believe that the vehicle can operate without assistance as the name suggests.

Waymo (Level 4 in automation) on the other hand, has driven over 6 million autonomous miles on public roads since its inception in 2009 and according to their data have only been involved in accidents resulting in minor injuries [27]. Nuro's R2, which the company considers Level 4, is another example of current integration at higher levels of automation. Unlike Waymo, Nuro's mission is to transport goods rather than people, bringing a safer edge to the AV which allowed it to become the first vehicle exempted from the NHTSA [28]. Unlike Tesla, Waymo's and Nuro's models are based on using their own operated vehicles rather than a consumer operated system. Waymo is a rideshare company that operates and monitors their vehicles while Nuro operates their vehicles to transfer goods. With less room for human errors, both companies report having less or nonfatal accidents while operating at a higher level of automation. Given that Tesla's autopilot is a Level 2 of automation and that many of the accidents reported with their vehicles involved human error or misuse of the technology, it is too soon to make the case against the safety of highly automated autonomous vehicles and the potential benefits they may bring. Nonetheless, these accidents have caused the public to have a negative outlook on AV technology and integration. A 2018 survey conducted by AAA found that 73% of survey takers reported feeling too afraid to ride in an AV, a 10% increase from 2017. Similar surveys conducted by other organizations like Pew Research Center, Deloitte, MIT AgeLab and New England Motor Press Association showed a decrease in trust in autonomous vehicles in comparison to the 2016 results [29]. Nonetheless, key holders, affiliates, enterprises, and supporters of a transportation update have every inclination in at least exploring the logistics of what integrating AVs in the roads would look like. Improvements to the tools used in AVs continue to happen at a faster rate, improving their performance standards and affordability. Optimists of AVs predict that autonomous fleets will grow in normalization in the 2030s and 2040s despite the challenges they need to overcome [9, 29-31]. First adapters are already normalizing electric vehicles and automated functionalities, second adapters will expand on the normalization of robotic assistance and alternative sources of energy and the rest will be soon to follow as electronic autonomous vehicles increase in affordability [32]. In this century, the question pertaining to autonomous vehicle technology is not whether it can exist, but rather can it stay? Most entities involved in answering this question are doing everything in their power to ensure that the answer is yes.

2.2 Typical Autonomous Vehicles

Sight is the most important sense while driving, thus computer vision is the key to the success of autonomous vehicles. Currently, computers are not nearly as good as humans at seeing and processing data, but engineers are discovering different ways to supplement this disparity. AVs rely on a combination of different sensors to capture their environments, software to process the data gathered, and actuators to respond based on the given output. The requirement for each AV is determined by the Operation Design Domain (ODD) which should specify the intention and the conditions for safe use of the AV. Regardless of ODD, existing AVs rely on the use of sensors to create Simultaneous Localization and Mapping (SLAM) [19, 33]. SLAM is the process of creating a map of its local surroundings based on where the vehicle is located and the direction in which it is headed. SLAM includes localizing both static, and moving objects, road and street signs, lanes, and traffic lights to prevent collision as well as to predict behavior and or actions [34]. For example, if a traffic light is yellow, the vehicle must predict that it will turn red and thus anticipate a full stop. Furthermore, driving involves navigating around many moving obstacles and responding to unpredictable circumstances. A pedestrian crossing without the right of passage, a vehicle running a light or taking a risky turn, an animal chasing food or difficult weather conditions are all real-life driving situations that humans have grown relatively good at navigating and AVs need to be good at navigating as well.

Another built-in human superpower considered in the successful implementation of autonomous vehicles is geographical awareness. As in, an AV should have a sense of placement, not only within its local environment but also on a global scale to have a sense of direction from it starting point. Simply put, AVs must know where point a is to get to point b. To do this AVs rely on Global Navigation Satellite Systems (GNSS) which we will discuss in a further section [35]. Next, AVs must be able to process information in order to make decisions. Data gathered from sensors to create SLAM and geographical location should provide the vehicle with a list of conditions and decisions that it needs to make for path planning and its execution. A vehicle, for example, might need to make the next right turn based on the navigation satellite's information but before doing so must also determine whether it is possible, safe, and clear at every point of turn. We use data and software to create algorithms to help with the decision-making process. Thus, machine learning plays a pivotal role in the successful implementation of autonomous driving. Communication is trivial in a fleet of fully integrated AVs. The safest way to incorporate automation on the road on a grand scale is to ensure that vehicles can communicate with each other, its infrastructure, and non-automated agents like pedestrians while navigation thus a communication system is also integrated in typical builds [19]. Lastly, actuators perform the essential functionalities such as accelerating, braking, and steering.

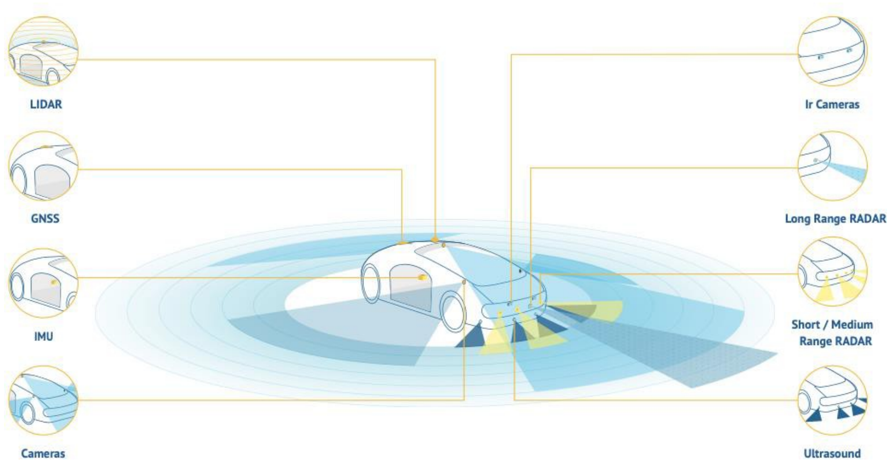

Figure 2 Example of autonomous vehicle sensor design - 2020 Autonomous Vehicle Technology Report, Wevolver, p. 18 [36].

Sensors

Existing autonomous vehicles rely on sensors to create SLAM. Passive sensors, such as cameras, capture information from the environment based on how energy interacts with objects within the environment. For example, cameras use light to capture shadows within a given moment, measuring the way objects interact with light. Passive sensors are standard in AV's vision system and are consistently improved for the purposes of emulating human vision. Some of the main benefits of cameras include their cost and their ability to build on usability in combination with other systems. Some of the limitations include limited visibility when environmental conditions make it difficult to capture light and the space requirements for the data collected [18, 19].

On the other hand, active sensors interact with the environment by transmitting a signal and waiting for the reflection of that signal to determine information, such as the distance between objects detected and its location. There are various types of active sensors, all of which offer benefits and limitations. Sound navigation ranging (SONAR), or ultrasonic sensors use soundwaves to detect the distance of the nearest object. These sensors are a cost effective and efficient solution to features where short distance detection is a priority such as parking assist. SONAR sensors have stood the test of time and are still used in the latest implementation of autonomous vehicles. However, their short-range limitations require that life-sized AVs incorporate them in collaboration with other sensors that can detect obstacles at longer distances [18, 19, 37]. Figure 2 provides an example of ways that different sensors work together. Ultrasonic sensors are used in the development of our 1/10-scale autonomous vehicle and will be explored in more depth in further sections.

Radio detection and ranging (RADAR) sensors work like SONAR but use radio waves instead of soundwaves. Traveling at the speed of light, they provide a better option for medium range distance detection since they have a longer wavelength thus can extend the signals further than SONAR. The strengths in RADAR predominantly rely on their ability to be unaffected by environmental conditions unlike cameras which lose their visibility in rain or fog. However, limitations in detectability due to interference and inability to provide spatial information about objects keep RADAR from being the sole tool for creating SLAM. Still, high resolution imaging RADAR has a longer degree of field with the ability to create a 4D mapping of the environment thus they are commonly incorporated in AVs for long-distance obstacle detection [18, 19, 37].

LIDAR or light detection and ranging sensors can create 3D mapping of static and moving objects as far as 250m in range. LIDAR sensors have a wide field of vision, allowing for object detection within a 360-degree range around a vehicle. While their long-distance range is lower than RADAR, LIDAR provides favorable contributions to the creation of SLAM needed for the vehicle to become "aware" of its surroundings. In different configurations of an AV, passive sensors (cameras) are combined with active sensors (SONAR, RADAR, LIDAR) to create a vision system for the vehicle [18, 19, 37]. The cameras behave like the eyes of the vehicle while the sensors allow the vehicle to "feel" its surroundings for ultimate safety. The configurations of passive and active sensors vary per build with some manufacturers using some or all different kinds of sensors. All autonomous vehicle configurations require passive sensors and at least one kind of active sensors depending on their ODD. Figure 3 compares the way Volvo-Uber, Waymo and Tesla incorporates these sensors into their build provided by the 2020 Autonomous Vehicle Technology Report from Wevolver [19].

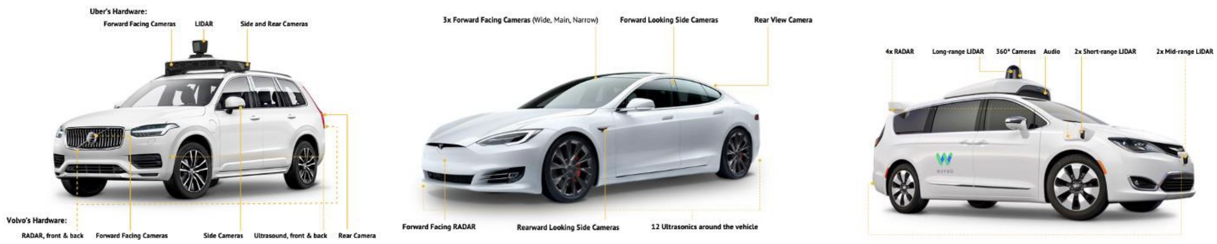

Figure 3 Volvo, Tesla, Waymo autonomous design - 2020 Autonomous Vehicle Technology Report, Wevolver, p. 30 [36].

Global Navigation Satellite Systems (GNSS)

Global navigation satellite systems such as the global positioning system (GPS) consists of a series of satellites, a ground station, and receivers. The satellites orbit the earth's atmosphere and send constant signals of their location from space. On earth, ground stations use RADAR to monitor the satellites' positioning and health. Receivers use signals from multiple satellites to calculate the distance from the receiver's position on earth [38]. Most technologies built in the last few decades have receivers that can use GPS, most commonly cell phones. Autonomous vehicles are equipped with receivers for GPS to help with navigation and path planning. Continued improvement to GNSS technologies continues to improve navigation which in turn will improve the accuracy in which AVs move.

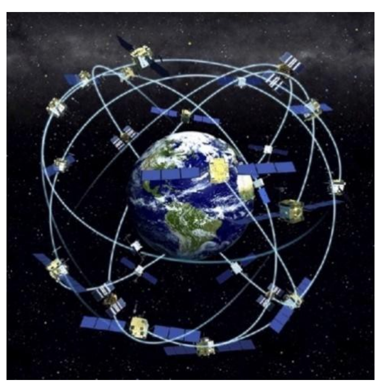

Figure 4 Depiction of navigation satellites orbiting the earth - NASA [39].

Machine Learning

Machine learning (ML) is the process of teaching computers to produce an output based on the data given. ML is at the root of computer "thinking". Autonomous vehicles process data and make decisions based on training they receive from machine learning algorithms. The most common ML algorithms used for AVs are convolutional neural networks (CNN), recurrent neural networks (RNN), and deep reinforcement learning (DRL). Convolutional Neural Networks (CNN) are commonly used with image classification problems. CNNs work by convoluting through an image pixel by pixel to understand how they relate to identify a general image. Their approach involves multiplying an input layer comprised of a matrix and a filter layer, or kernel, to create an output layer with the sum of all values. CNNs continues the sum of multiplications of the matrices until every pixel in an image is covered [40]. Recurrent Neural Networks (RNN) are a powerful tool for sequential data, where capturing time intervals is important to the output. One of its most efficient traits is their internal memory, which allows RNN to remember relevant information from each layer while exploring the current layer, thus making it better at predictions. RNN are commonly used in natural language processing [40-42]. Lastly, Deep Reinforcement Learning (DRL) are being introduced as a new tool in this domain. DRL combines deep learning and reinforcement learning by utilizing reward functions to attain the objectives [19]. Though not limited to the presented tools, AVs use a combination of several algorithms to achieve their objectives sometimes training their models to predict multiple outputs. ML algorithms in AVs are used to predict events in the environment, like the actions of other vehicles or pedestrians, identify and classify road signs and other objects, and process moving signals like traffic lights.

Communication

As more AVs are integrated into human environments, they need to communicate with each other, other non-automated vehicles, human drivers, and outside agents. Vehicle-to-vehicle (V2V) communication is important for interoperability between a swarm of autonomous vehicles on the road to exchange important information about the environment and system operations. In current systems, dedicated short-range communication (DSRC) aid in detecting and sharing information like traffic patterns, accidents ahead, construction and road closures to provide different routes for efficient and safe operations [19, 43]. Other communication technologies like 5g wireless communication and the cloud architecture developed and matured by Amazon, Google and Microsoft are also used in multi-vehicle communication working alongside cameras and other technologies like GPS to increase the quality of information shared [19, 22].

Actuators

The algorithmic output based on the data collected from the environment is used to decide the vehicle's main functions: steering, braking, and acceleration. In autonomous vehicles these functions consist of the collaboration between software and an electronic control unit (ECU). ECUs manage the vehicle’s control system through wired connections to the hardware components responsible for each function. In existing AVs, multiple ECUs could be used for different driving functions, or a single ECU could be used to share the control of multiple features' outputs. The distribution of responsibilities and number of ECUs vary per each vehicle's ODD [19, 44]. While no configuration of an AV is exact, they all must include actuators to perform the standard driving functions performed by humans.

2.3 Safety Systems in Autonomous Vehicle

The National Highway Traffic Safety Administration defines automated safety systems as driver-assistance technologies that are helping save lives and prevent injuries [45]. In response to the growing demands for safety standards involving autonomous driving systems (ADS), the NHTSA has published a Voluntary Guidance for Automated Driving Systems with recommendations for safety handling and implementation for developers and manufacturers of autonomous vehicles [46]. In this report, NHTSA has identified twelve components of safety standards:

System safety states that system designs, testing and validation should account for events that could jeopardize safety, including system failures. Within each system design, developers are encouraged to include approaches to handling events like system communication failure, traction or stability loss, and deviations from normal expected driving practices, for example, with high regards and prioritization of safety.

Operational design domain articulates that manufacturers should specify the intended use case of the vehicle, with clear indications of its purpose and limitations. Key information such as speed limitations, geographic areas, and weather conditions that the vehicle can safely operate in should be clearly stated as well as the intended roadway type that the vehicle can safety operate in according to its design and testing outcomes.

Object detection and event detection and response defines the need for AVS to accurately detect and respond to stimuli in dynamic environments which can vary based on the ODD but can include the detection of pedestrians, animals, other vehicles, and objects that could potentially pose a threat to safety. It also states that AVS must have an incorporated crash avoidance response to these factors.

Fallback (minimal risk condition) recommends that AVs must have fallback plan of reaction to prevent crash, injury, and death if the vehicle's main system fails or encounters a problem.

Validation methods encourages developers to test and validate their approach to system design, operational design domain, object detection and event detection and response, and fallback to understand the expected behavior of the AVS and its limitations. It is recommended that testing methods should include simulation, test environmental and real-life driving environments.

Human machine interface focuses on the development and implementation of human interfaces, communication tools and methods that will allow the automated driving system to communicate information to the human driver both inside of the vehicle and outside of the vehicle, for example, alerting a pedestrian of its approach, etc.

Vehicle cybersecurity recommends manufactures to consider, address and test cybersecurity threats posed to the ADS and to document the processes in which it plans to address such threats. In addition, developers are strongly encouraged to collaborate on such processes to improve the overall safety and security of the general autonomous driving environment.

Crashworthiness pertains to the protection of the vehicle's occupants if another vehicle crashes the ADS. Manufacturers should specify the maximum occupancy of the vehicle and conduct testing on the potential impact to the occupants posed by this scenario.

Post-crash ADS behavior involves the implementation and documentation of the vehicle's actions after a crash has occurred to prioritize safety. Such actions can include but are not limited to idling the vehicle in a safe location while waiting for resolutions, shutting off the gas pump, and disengage electrical powers. In addition, manufacturer should address whether the vehicle would communicate with responding agencies such as first responders post-crash and have clear documentation on the assessment of vehicle's system prior to releasing the vehicle on the road post-crash.

Data recording states that ADS entities are encouraged to record, analyze, and share data recording involving safety features and accident response to improve such systems. Data collection should incorporate data from various scenarios including those resulting in minor injuries, major injuries, and death to better address ways that such scenarios could be prevented in the future.

Consumer education and training recommends that developers and manufacturers should provide continuous training and education to all entities involved in the ADS process, including but not limited to employees, dealers, and consumers.

Federal, state, and local laws indicate that involved agencies should account for the compliance of federal, state, and local laws when establishing the operational design domain and should consider its plan to comply with changing laws.

For autonomous vehicles at varying levels of automation some of the most known safety features include cruise control, blind spot detection, lane departure warning, forward collision warning, automatic braking systems on obstacle detection, lane keeping assist, adaptive cruise control and self-park. Research and development continue to improve these features allowing for more driving responsibilities to be delegated to higher level autonomous systems. After 2025, it is expected that a successful highway autopilot system will be most likely in the market [45]. Thus, the recommended guidelines to the development and improvement of safety features play a pivotal role in the normalization, acceptance, and trust in autonomous vehicles as the technology continues to expand in the next coming decades.

CHAPTER III OVERVIEW OF THE AUTONOMOUS VEHICLE

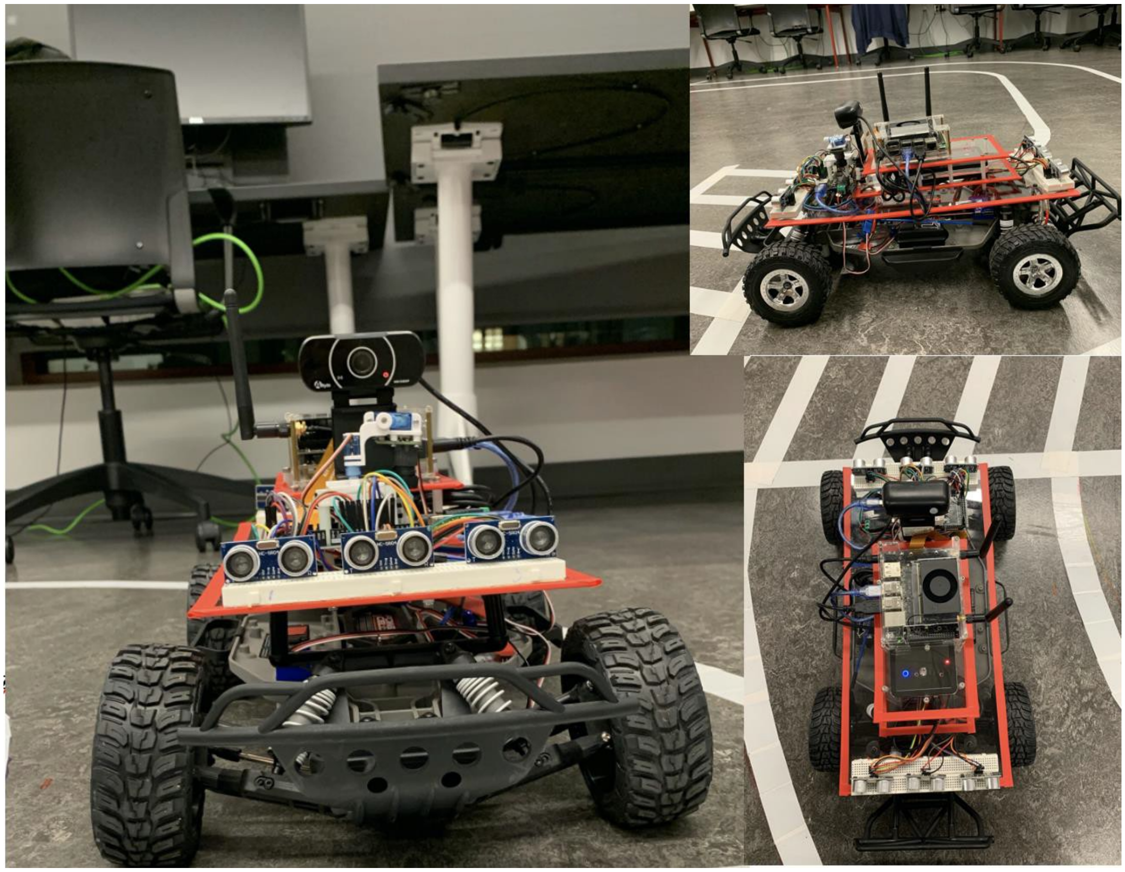

Figure 5 1/10-scale Electric Autonomous Vehicle.

The 1/10-scale autonomous vehicle is an educational small-scale autonomous vehicle model intended to imitate life-sized vehicles' functionalities and features [47-49]. In this approach, priority functions and safety features have been identified, developed, and incorporated into the model to gain knowledge and exposure of the tools, technologies, and composition of AVs. The features identified in this model include road sign recognition, collision avoidance, and lane tracking systems. As explored in previous sections, safety is the birth and the purpose of autonomous vehicles, therefore these were identified as the most important to understand. In this vehicle, we used modular approach consisting of three interpolating nodes focused on individual tasks that connect through Robot Operating System (ROS).

CHAPTER IV HARDWARE SYSTEM

4.1 Overview of the Hardware System

The hardware system contains the components responsible for performing the essential driving operations of the 1/10-scale autonomous vehicle and the hardware required to house the software. It includes the actuators that control acceleration, braking and steering systems. The vehicle consists of a short course racing truck, an NVIDIA Xavier board, ultrasonic sensors, a high-resolution camera, an Arduino board, and a scaled driving environment to test our implementation. Figure 6 demonstrates the relationship between the different physical components.

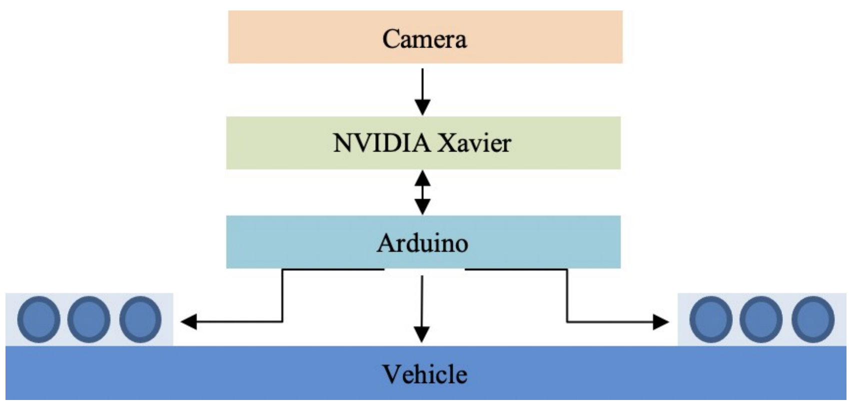

Figure 6 Hardware architecture.

4.2 The 1/10th Vehicle

For the vehicle body, we used a short-course racing truck with a Magnum 272™ transmission with 2-Wheel Drive (2WD). This model is equipped with a torque-control™ slipper clutch, a waterproof Titan® 12-Turn 550 motor with XL-5 electronic speed control and comprehensive steering capabilities [50]. This model's compact size, and configuration allowed us to customize it with the tools needed to develop our autonomous features. This vehicle is a good model to imitate the real-world driving experience and obstacles at a fraction of the cost of a life-sized autonomous vehicle.

4.3 Computing System

On the chassis of the vehicle sits an NVIDIA Jetson Xavier board, a small supercomputer specifically designed for the field of AI and robotics [51]. The Xavier runs Ubuntu 18.04 operating system.

4.4 Vision System

Acting as a passive sensor, a high-resolution camera also sits on top of the chassis capturing data in real time. The camera is used to capture the environment to respond to visual cues like road signs and driving lanes. The camera shares this data with the Xavier, which is responsible for processing the data based on algorithms which are discussed later. The algorithms generate the response to the visual stimuli and then communicate with the Arduino board which controls the vehicle's actuators.

4.5 Pre-Collision System

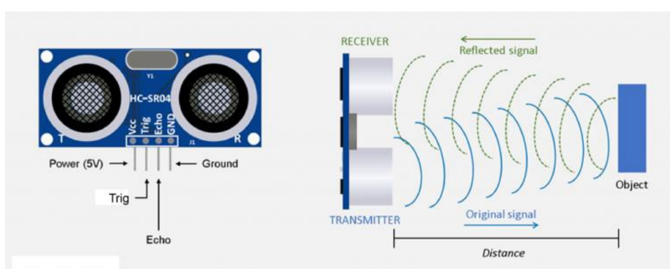

Figure 7 Illustration of how HC-SR04 Ultrasonic Sensors use soundwaves to detect objects [52].

A total of 6 HC-SR04 ultrasonic sensors sit on top of the chassis. Three on both the front and the rear of the vehicle. The sensors make up the hardware component of the collision avoidance feature. The close-range sensors can detect objects within the range of 0.02 m and 4 m. For our 1/10-scale vehicle these sensors are enough to identify close and "mid-range" objects based on scale. Ultrasonic sensors can detect obstruction and determine the distances using a transmitter and receiver. The transmitter releases ultrasonic waves into the environment, upon detecting an obstruction the waves reflect to the receiver (Figure 7). We can determine the distance from the obstruction by the measuring the time of return for the ultrasonic wave using the formula:

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

In the formula $343\ m/s$ denotes speed while T references the travel time of sound. Calculating distance allows us to determine a set point of 40.00 cm which is used to signal the vehicle to stop to avoid collision in our PID system further described below.

4.6 Arduino MEGA 2560 Board

Lastly, an Arduino MEGA 2560 board is used to control the actuators. The board is equipped with 54 digital input/output pins, 16 analog inputs and 4 serial ports. The board sends commands to the motors to control acceleration and braking response.

CHAPTER V SOFTWARE SYSTEM

5.1 Architecture of the Software System

The software system is the processing unit of the autonomous vehicle. Software tools work with the hardware to process the data collected and output the responses to the actuators that control the vehicle. The software system is made up of various tools which consists of Ubuntu 18.04 as operating system, Python, Robot Operating System (ROS), Open-Source Computer Vision Library (OpenCV), Arduino IDE, and Visual Studio Code (VSCode).

5.2 Robot Operating System (ROS)

The Robot Operating System (ROS) is an open-source collection of software and libraries for the implementation of robotics and artificial intelligence projects [53]. It is commonly incorporated in various projects because of its versatility and adaptability. The ROS platform helps with the integration of various features packaged as nodes that communicate with each other to create their infrastructure. ROS is a data exchange platform that also provides support for the development of features including pose estimation, robot geometry library, navigation, diagnostics and mapping and more [54]. In our 1/10-scale autonomous vehicle, ROS will be the middleware between the lane-tracking, collision avoidance and stop sign recognition features allowing for the exchange of inputs and outputs needed for driving.

5.3 OpenCV

The Open-Source Computer Vision Library (OpenCV) comprises various libraries and tools to aid in the goal of helping machines see. OpenCV works alongside the high-definition camera to identify, label, track and categorize objects in sight using specialized algorithms. OpenCV is a popular library that is frequently utilized in Artificial Intelligence and Machine Learning projects that require classification and regression problems. Another strength of OpenCV is its ability to work in various programming languages. In our project OpenCV is used with python for the lane-tracking feature and the stop sign recognition feature [55, 56].

5.4 Arduino IDE

The Arduino platform consists of the Arduino board as the hardware component and the Arduino Integrated Development Environment (IDE) as the software [57, 58]. The Arduino platform is used for Internet of Things (IoT) applications that expands to wearable technologies, mobile technologies and most applications that incorporate electronics with programmable software [57]. Similar to OpenCV, Arduino has many libraries available to Windows, Linux and Mac users and provides a relatively easy to use interface with versatile and strong capabilities allowing users of all skill levels to succeed [57]. In our project, Arduino IDE receives input from the MEGA 2560 Arduino board and ultrasonic sensors to communicate the vehicle's control values.

5.6 Visual Studio Code

Microsoft's open-source Visual Studio Code is an integrated development environment (IDE) that is compatible and provides support to most programming languages in macOS, Linux and Windows operating systems [59]. In our project, we use Python to develop the lane tracking and stop sign recognition features in VS code.

5.8 Ubuntu

Ubuntu is an open-source operating system commonly used in robotics projects. It is the third most used operating system worldwide for its expandable qualities and compatibility. Ubuntu's reputation is consistent with projects involving machine learning, autonomous vehicles, IoT, data science projects and many others due to its abilities to incorporate different components and software systems in a variety of projects.

CHAPTER VI PID-BASED PRE-COLLISION SYSTEM

6.1 PID Control Algorithm

The Proportional-Integral-Derivative algorithm is the control unit that translates data collected through passive and active sensors into a set of numerical values used to control the vehicle's steering and speed [60]. It is an essential component of the 1/10-scale autonomous vehicle as its responsible for executing the corrections produced by the lane-tracking and stop-sign recognition nodes. In our vehicle, the PID performs three main functions:

a. Prevent collision through obstacle detection
b. Control steering based on output from lane-tracking system
c. Respond to environment based on stop-sign recognition module

The PID algorithm works in conjunction with ROS to connect the nodes through a series of publishing and subscribing channels. Figure 8 demonstrates the communication structure of the nodes with PID and ROS.

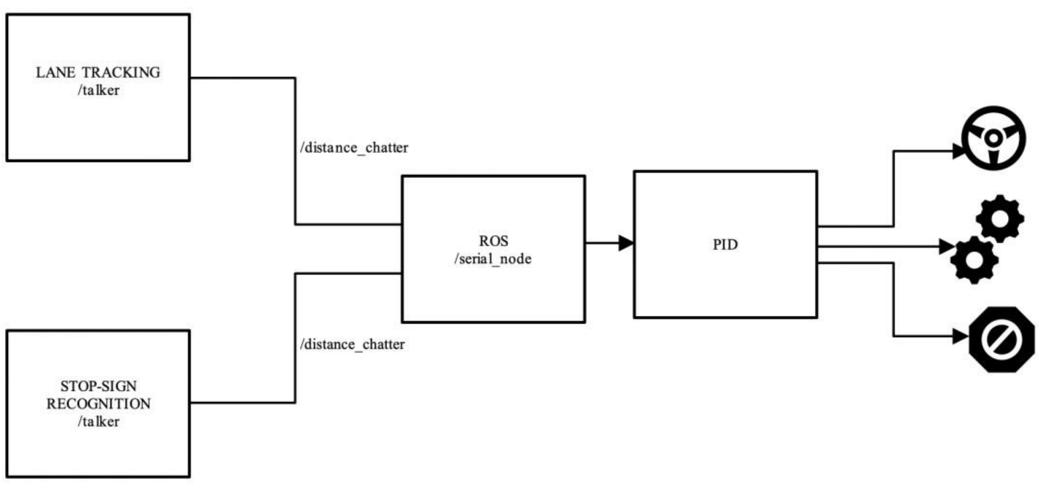

Figure 8 Integration architecture of lane tracking and stop sign recognition nodes and PID.

Generally, PID algorithms are used to create a wide range of responses for any system which consists of a desired outcome, an input, and an intervention in response to the difference between the input and desired outcome. In other words, it aids in triggering an adequate response to create the desired outcome if needed. In our vehicle, the PID controller is used to determine the acceleration and braking pressure needed to maintain a safe distance between the vehicle and obstructions to prevent collision, keep steering control using lane-tracking feature, and stop upon stop-sign identification. PID is not only at the center of a vehicle's mechanical control but also at the center of all essential safety features in our autonomous vehicle.

The PID algorithm responds to changes based on the setpoint established by the algorithms. For each feature the PID's job is to normalize the changes in real time and to adapt the values back to their desired ones. In the lane-tracking system, the setpoint is the steering value necessary to stay within the lane. The values are based on numeric representation of left and right turns in the ranges of -180 to 180 degrees. The negative values denote left turns, positive values denote right turns and zero denotes no turns thus continuing straight. The lane-tracking system uses the high-definition camera to track the lane and generate the deviation from the path in the form of angle integer values thus producing a continuous setpoint for steering correction. The value is shared from the lane-tracking node to the PID algorithm running in the Arduino IDE through Ros' publisher and subscriber tools which then outputs the mechanical steering control values to the Arduino board attached to the vehicle controlling steering.

For the stop-sign recognition feature, the PID algorithm communicates through ROS to stop when the calculated distance between an observed stop sign is less than or equal to 40 cm. Further explanation of the implementation of this feature found in later sessions. Figure 8 provides a high-level representation of the PID algorithm's role with the implementation of each feature. The PID interacts with all components of the vehicles from passive and active sensors, software, and hardware.

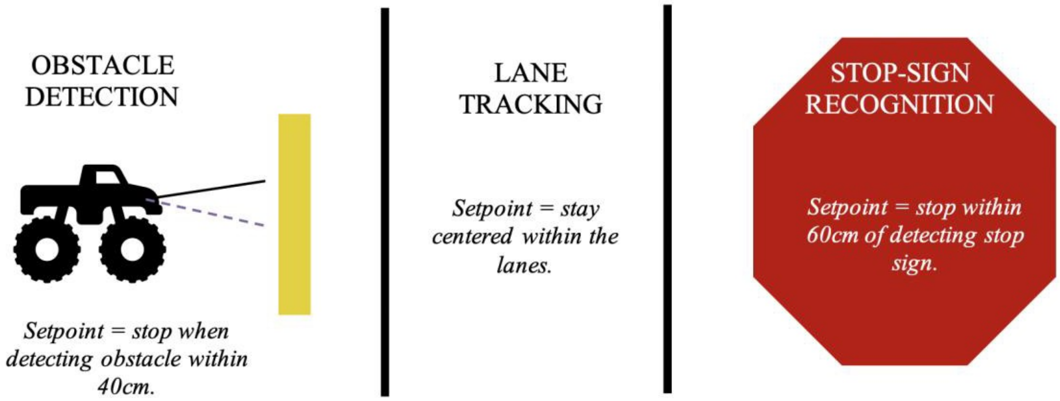

Figure 9 PID's central role in the implementation of each feature.

6.2 Collision Avoidance via PID Control

The collision avoidance feature is specific to the Arduino node and does not require ROS for the implementation. Instead, it is directly connected to its data source by way of the ultrasonic sensors, the Arduino MEGA 2560 board and Arduino IDE. The 1/10-scale AV is equipped with six ultrasonic sensors, three in the front and three in the back. For this implementation of the vehicle, only the front three sensors are used for forward handling. For accuracy, the distances observed by each sensor is computed into an average. Our setpoint for this implementation is 40cm therefore if the computed average of the calculated distance between all three sensors is 40cm or less the vehicle's PID function will activate bringing it to a stop. Once an obstruction is removed from the vehicle's front path, the PID algorithm is cleared of errors and resume driving.

```
Inputs: frontdistance //calculated from ultrasonic sensors
        if frontdistance ≤ 40 //if obstruction detected within setpoint stop vehicle
              velocity = 90
        if ≥40 frontdistance ≤ 70 //if obstruction detected within 40-70 cm slow velocity
              velocity = 120
        if frontdistance > 70 //if no obstruction within 70cm of vehicle velocity is normal
              velocity =180
```

Figure 10 PID collision avoidance pseudo code.

The velocity values range from 90-180 with 90 representing the vehicle at a complete stop and 180 at the maximum allowed velocity within our domain. Success of the implementation would result in the vehicle stopping upon detecting an obstacle. Testing involved placing an object in front of the vehicle and observing the mechanical response and data output. In Figure 11 below, the blue line captures the distance, and the red dashed line captures motor power. The data captured was consistent with the expectations. As the distance of an observed obstacle decreased the motor power decreased as well indicating a stop point.

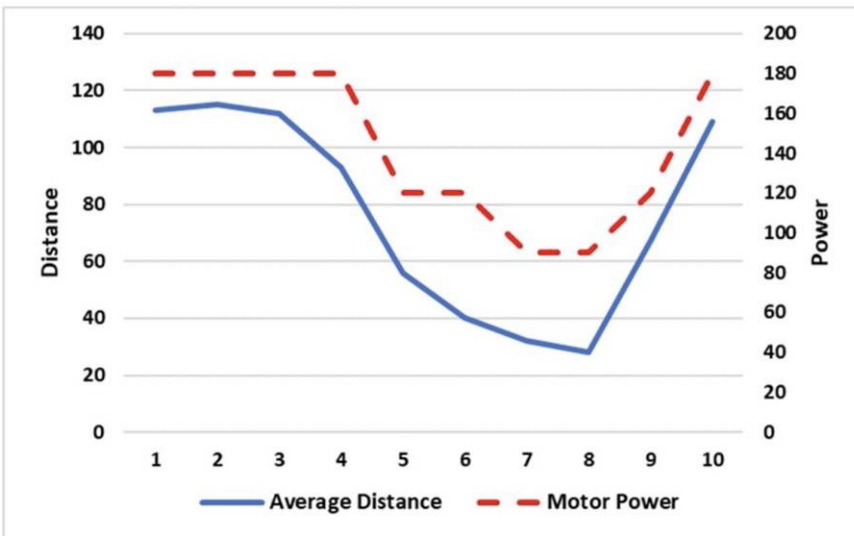

Figure 11 1/10-scale vehicle's motor power at varying distances from an obstruction from [61].

The implementation and testing of the PID obstacle detection for collision avoidance feature was proven successful in data analysis and observation. In any vehicle the highest priority is to avoid collision. The PID control response efficiently avoided collision in our experiments. The PID also supports the alternate features required to build a life-size comparable vehicle discussed in the next phases. Further improvements on this feature are also discussed in the future works section.

CHAPTER VII LANE TRACKING SYSTEM

7.1 Introduction to Lane Tracking

The lane-tracking feature is a primary safety intervention for autonomous vehicles. Lanes define a working space of operation and help humans keep a safe parallel distance while also aiding in steering. In the human brain this function happens autonomously and mostly effortlessly. In this portion we created a sub system using a high-definition camera as passive sensors mimicking human eyes incorporated with algorithms to extract key visual information for safe steering and lane keeping. The components required for the implementation are as follows. A scale-down driving environment was constructed using white tape to create a single lane oval track. A Logitech high-definition camera mounted on top of the vehicle facing forward captures a video feed of the lanes. The data is captured and processed through a series of transforming algorithms to aid in computing the numerical output value representing a correction angle used in steering. The algorithms were implemented on Visual Studio Code with Python 3.8. A lane-tracker publishing node was shared via ROS with Arduino's PID algorithm to control the vehicle's steering in real time. This process loops for a continuous lane-tracking system.

7.2 Lane Tracking Approach

```
Inputs: steering_value (received by ROS subscriber)
Updates: newsteering
    newsteering = (int)steering_value*(trans_value) +90
    if newsteering < 0
    // minimum steering value is 0 newsteering = 0
    else if newsteering > 180
    // maximum steering value is 180 newsteering = 180
```

Figure 12 Steering update for lane tracking, and PID pseudo code.

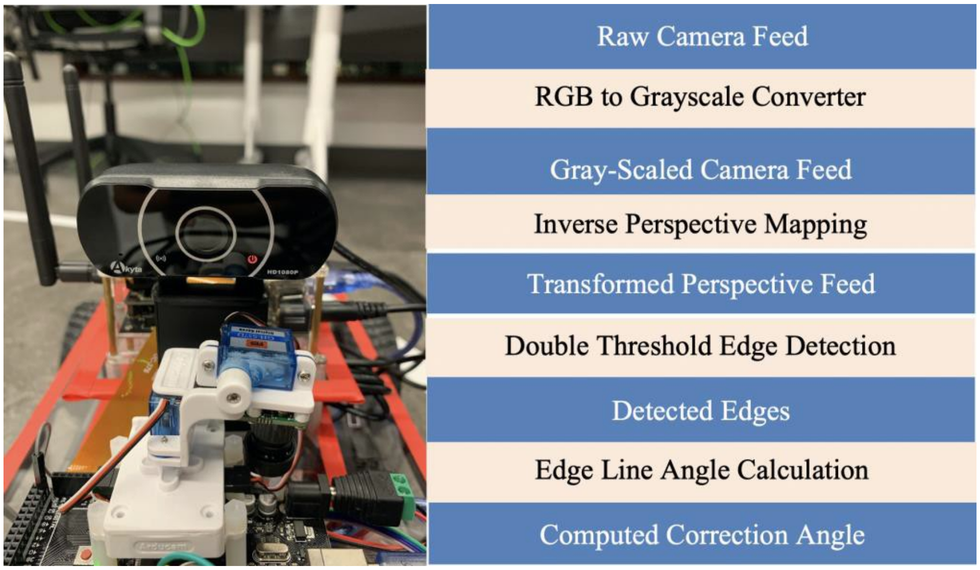

Figure 13 Vision system and lane-tracking implementation structure.

The first step in the implemented approach was to use a gray-scale version of the video feed and translate it using Inverse Perspective Mapping (IPM) in conjunction with Vanilla Perspective Mapping (VPM) to change the feed's perspective to a bird's eye high-level view for clear identification of the lane markings. The region of interest (ROI) was marked by four corner points set by internal code parameters to adjust for the angle and height of the feed captured. In Figure 14 the ROI identification area is selected based on the camera position. The transformed gray-scale feed with the ROI is used as input for OpenCV's *getPerspectiveTransform* function to determine a transformation matrix based on the ROI. In addition, OpenCV's *warpPerspective* is used to generate a perspective frame by multiplying vector coordinates in the original frame by the calculated transformation matrix resulting in the output of an inverse perspective mapping with a bird's eye view.

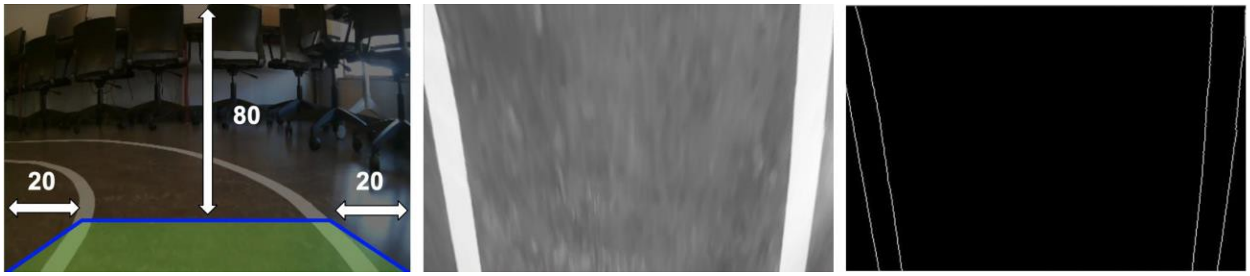

Figure 14 1/10-scale vehicle image transformation perspective.

After identifying the edges of the lanes, a Hysteresis threshold (double threshold) was used with OpenCV's Canny class to convert the IPM into a binary pixel map where edge pixels were given a value of 0 and edge pixels a value of 255. Threshold values from 100 to 200 were selected for OpenCV's Canny class implementation. Pixels with values higher than the 200 were automatically considered an edge value, and those with lower than 100 were deemed non-edge pixel. Values outputting within the range of 100-200 were classified on whether they were connected to edge pixels. Figure 14 shows output after pixel extraction for lane-edge detection. The goal of the Double Threshold Edge Detection sub-system is to create a two-dimensional pixel array created by a binary mapping of pixel connectivity that creates lines representing lane-edges. To do this, OpenCV's *HoughLinesP* function was used to implement a Probabilistic Hough Transform (PHTs). PHTs select random points to identify "real lines" in a feed and thus are favorable over standard Hough transform which considers all points as it alleviates some of the computational expense needed for real-time immediate classification domains like autonomous driving [62]. The angles for the returned line outputted by the PHT were determined by using the identified horizontal bottom of the frame as reference, where $\theta = 0$ represented the right and $\theta =$ 180 the left. To identify the vehicle's direction, we compared its position to the parallel line edges (θ_m) by taking the mean value of all returned Hough Lines (n) from θ_1 to θ_n thus using formula:

$$\theta_m = (\sum_{i=}^{n} \theta_{n}) /n$$

In this approach an output angle of $\theta_m = 90$ indicates that the vehicle is parallel to the lane edges or "straight". Any output value where $\theta_m \neq 90$ would call for an adjustment to be reflected in "steering". To find the adjustment angle (θ_a), the deviation θ_m value was subtracted from 90 in $\theta_a = \theta_m - 90$. Thus, adjusting the lane values where $\theta_a > 0$ indicates a left direction change in the lane, $\theta_a < 0$ indicates the lane turning right, and $\theta_a = 0$ indicates that the angle is parallel to the lane-edges or "straight". Thus, θ_a indicates the angle of the lane with the vehicle facing forward, where ($\theta = 90$). The vehicle's steering adjustment angles consist of steering movements of θ_a and $-\theta_a$ with a positive steering angle towards the right and a negative angle turning the vehicle left. The final output for the computed steering angle defined by is shared to the Arduino node via ROS where it interacts with the PID algorithm to adjust the vehicle's steering controls. Additional adjustments were made to the algorithm to account for a "pedestrian crosswalk" in the driving environment that featured short horizontal lines and caused an off course steering response. It was determined that the crosswalk lines produced a value of $|\theta_a| \approx 5$ hence a threshold value of $|\theta_a| = 7$ was set, where $|\theta_a| < 7$ would be published as $\theta_a = 0$ to the ROS lane tracking node and $|\theta_a| \geq 7$ would be published $-\theta_a$ like normal.

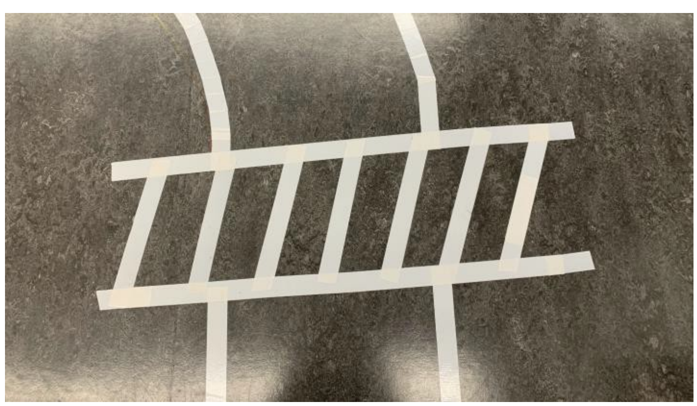

Figure 15 Pedestrian crosswalk.

CHAPTER VIII STOP SIGN RECOGNITION SYSTEM

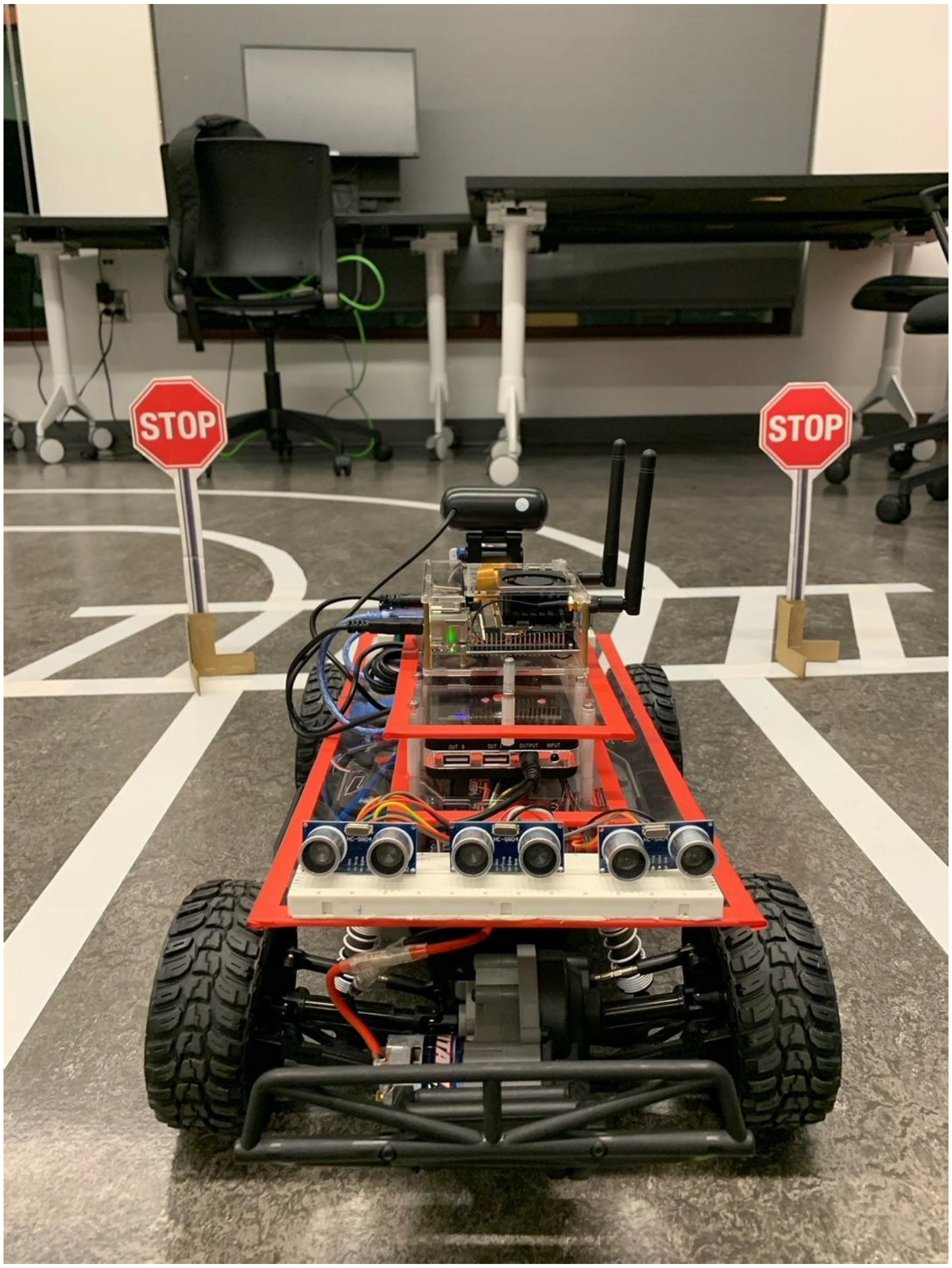

Figure 16 Stop sign in the driving environment.

8.1 Stop Rules

The birth of modern-day autonomous driving technology began with developments in machine vision in the late 1980s and 1990s [12]. Sight is the most important safety system in human drivers, thus naturally intelligent autonomous mobile systems cannot operate without this sense. Developments in the field of machine learning have made computer vision attainable for those interested in incorporating them in their projects. In the 1/10-scale AV we use a supervised machine learning approach to train a model to identify objects within an image and then testing its knowledge by feeding it images it has not seen before to create classifications. The implementation of this feature in this project is written in Python after receiving live-video feed from the high-definition camera mounted on the vehicle's chassis. Architecturally, the module then publishes its output as a ROS node to the PID control algorithm resulting in motor controls of the vehicle. The goal of this feature is to identify stop signs made to scale and integrated in the driving environment within 100cm away from the vehicle and executing a stop response when the stop sign is 60cm away. The PID's response to this feature also accounts for a 20 second delay in acceleration to imitate life-size vehicular stop-sign behavior. In this regard, if the vehicle stops due to a stop-sign identification and there are no obstructions detected by the sensors the vehicle would resume acceleration after 20 seconds. If the 20 seconds have gone and the sensors detect an obstruction, the PID's primary function would keep the vehicle from acceleration and as long as an obstacle is within the 40cm of the vehicle the vehicle will stop. The PID and stop-sign recognition safety features work together in the goal of avoiding collision. The output values published by this node are a percentage value which estimates the likelihood of a stop sign being recognized within the frame of the camera feed and its estimated distance from the vehicle.

8.2 Stop Sign Recognition Approach

The first portion of this feature is to recognize a stop sign. For this implementation we used MobileNet single shot object detection (SSD) model, a Deep Neural Network (DNN) approach. This approach was deemed to be the most accurate and cost effective for detecting objects in real-time, sparing the vehicle's computing power. Mobile SSD is capable of detecting multiple objects within a bounding box in a single shot resulting in a series of boxes containing the objects and class names (labels) that it was trained on, as well as a confidence percentage value for each classification [63]. The model was trained using the MS COCO dataset containing 330,000 images, 1.5 million object instances and 80 categories which include a stop sign class [64]. The algorithm was built to exclude objects that were not in the stop-sign class and was given a threshold confidence value of p = 0.5 to increase accuracy by identifying false positives. Inevitably, the algorithm will mislabel some classes, especially those farther in distance. While this can present some challenges, the objective of classifying stop signs is still achieved within the distance requirements of the 1/10-scale vehicle.

The second portion of this feature is to output an estimated distance between the SSD recognized stop sign and the vehicle to safely stop. We base the approach on the pinhole camera model to detect stop signs in front of the vehicle. The pinhole model is a common and simple approach to describe the mathematical relationship between a set of points of a 3D environment in an image plane [65].

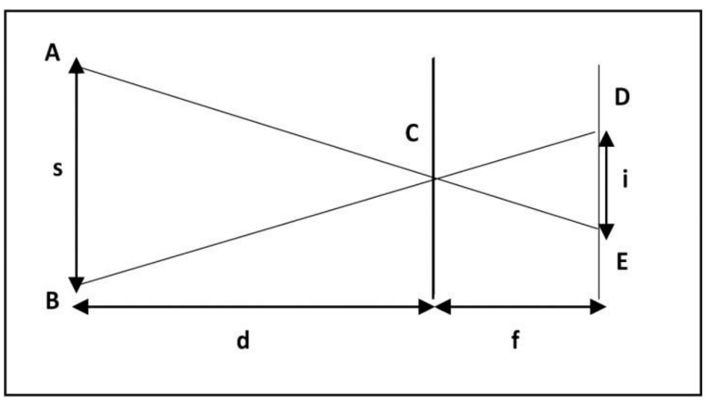

Figure 17 Geometry of pinhole camera model.

In Figure 17, point C defines the position of the camera, d represents the distance between the stop sign and the camera, *s* represents the size of the one-dimensional stop sign, *f* is the focal distance, *i* is the one-dimensional size of the image. Lastly *ABC* and *CDE* represent two triangles creating a three-dimensional space and establishing a relationship represented in equation Figure 18 *(a)* between objects in the environment. In formula *(b),* the product of s and f are replaced with a parameter α with $x = \frac{1}{i}$ as the predictor variable. Using the bounding box from SSD, we can determine the size of the image denoted by *i* by calculating the length, width, and diagonal size of the bounding box.

Then a linear model is created resulting in equation *(c)*. Lastly, a corrected term is added to the linear model in *(d)*. In the training phase, sample observations are used to fit the model where parameter α is calibrated and corrected with β. The calibration results in parameters $\hat{\alpha}$ and $\hat{\beta}$ which are used to predict the distances. In the prediction stage, we use real-time camera footage to gather more sets of observations *(e)* using the calibrated $\hat{\alpha}$ parameter to define residuals ε_i in *(f)* and (*g*), for example.

(a) $\frac{d}{s} = \frac{f}{i}$ *(b)* $d = \frac{s \cdot f}{i} = \alpha(\frac{1}{i})$ *(c)* $d = \alpha \cdot x$ *(d)* $d = \alpha \cdot x + \beta$

(e) $(x_0, d_0), (x_1, d_1), \ldots, (x_n, d_n)$ *(f)* $\varepsilon_i = \hat{\alpha} \cdot x_i - d_i$ *(g)* $\varepsilon_i = \hat{\alpha} \cdot x_i + \hat{\beta} - d_i$

Figure 18 Formulas used for pinhole model and size estimation.

Figure 19 below denotes the kind of errors SSD can make when making the bounding box at further distances on the two right images. The bounding box can shrink the images along the x and y axis, thus making it difficult to identify the class. To correct these errors, we opted to instead use the diagonal *i* for as shown in the picture on the left for better classification, size estimation and distance prediction.

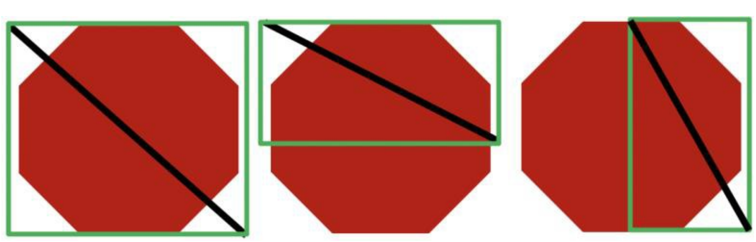

Figure 19 Bounding box errors.

CHAPTER IX ON-ROAD APPLICATION RESULTS AND ANALYSIS

Figure 20 Driving environment.

9.1 Experimental Setup

We created a scaled down driving environment to test the implementation of the integrated features. The driving environment consisted of a single lane oval loop with two pedestrian walk lanes and stop signs. The goal of this environment was to be able to test the PID algorithm, by placing an obstacle in front of the vehicle as the vehicle was approaching. The creation of a single double-line driving loop was set to test the lane-tracking algorithm's performance. Lastly, stop signs made to scale were made to test the implementation of the stop-sign recognition feature, however, due to integration issues pertaining to the compatibilities between our project architecture and the dependencies of the approach, particularly the dnn module, the stop sign recognition algorithm could not be integrated in the 1/10-scale autonomous vehicle. Nonetheless, the testing results of the stop-sign recognition performance on a virtual testing environment are discussed in the results and analysis section.

9.2 Collision Avoidance Results and Analysis

The method to test the PID collision avoidance was to compute the average stop distance between the vehicle and an obstacle. As mentioned in the PID section, the setpoint used as the maximum distance required to stop was 40cm. The purpose behind selecting 40cm was to account for the ≈10cm distance between the sensors and the front hood of the AV which requires the incorporation of "padding" when selecting a safe stopping point (Figure 21). Another factor to consider with the set point is the vehicle's velocity and any other environmental factors that could affect the vehicle's ability to stop within a reasonable distance from the obstruction. As such, it was important to measure the actual distance between the obstacle and the vehicle once it stopped to adjust the setpoint accordingly. To conduct the experiment, the vehicle was placed at the end of the track and an object was placed in front of the crosswalk prompting the vehicle to stop. Then the stop distance between the obstacle and the vehicle was measured and documented to calculate the average. The tests included performing this task ten times for three different set points, the results of which are summarized in Table 2.

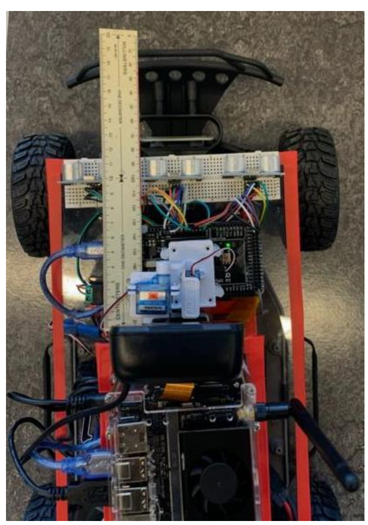

Figure 21 "Padding" distance for set point value selection.

Table 2 PID Collision Avoidance Experiments

Set point	Average distance
40 cm	7.5 cm
45 cm	12.9 cm
50 cm	20 cm

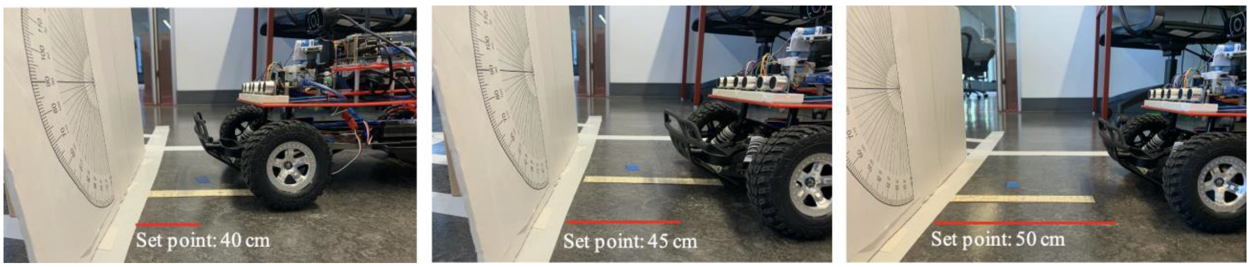

Figure 22 Results from PID collision avoidance tests.

Figure 22 denotes the differences in distance according to the set point values established in the PID algorithm. Performing these tests allowed us to recognize that a higher set point of 50 cm provided the best distance to account for possible factors that could decrease the average stop distance between the vehicle and an obstruction.

9.3 Lane Tracking Results and Analysis

An initial and qualitative important observation from the testing phase of the lane-tracking algorithm was that it had difficulties staying inside the two lanes as it tended to hug one lane over the other. This could have been caused by the footage captured in the IPM-transformed feed during vehicle turns which influenced the values of θ_m and $\theta_{a,}$ respectively, leading the vehicle to follow one lane more closely than the other. This observation provides future opportunities for improvements to the vehicle discussed in the future works section. Because of this observation, the lane tracking feature was evaluated on the AVs ability to recognize and stay within the driving environment. The testing of this feature consisted of three separate tests to measure *Stability*, *Accuracy*, and *Correction* percentage. To conduct the *Stability* experiment, the vehicle was placed on the center of the lane and was left to run 30 loops on its own. *Stability* was measured based on the vehicle's ability to make consistent turns in the driving environment. To measure the result, each loop was mapped and compiled in Figure 23, with an outline of the mean which resulted in a standard deviation of 8.22 degrees. In addition to this test, data was collected to test accuracy. *Accuracy* was measured by the vehicle's ability to complete loops without straying off course using formula accuracy = success/total loops. Based on the data collected of 30 iterations the accuracy percentage was 93%. Furthermore, in a separate test we measured *Correctness* percentage by placing the vehicle in random places inside and outside of the track to determine the vehicle's ability to recognize a lane and get on track to complete a loop. When the vehicle was placed outside of the lane on ten separate occasions, the vehicle had an out-of-lane correction percentage of 80%. When the vehicle was placed in random places inside of the lane, the vehicle was less likely to find and continue following the lane, resulting in a 60% inside-of-lane correction percentage.

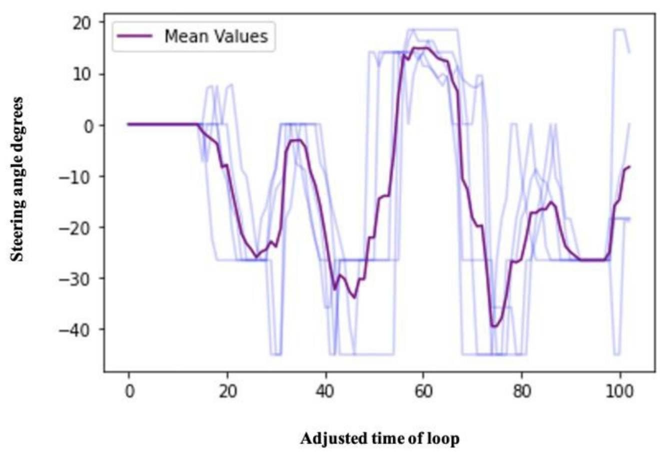

Figure 23 Lane-tracking stability tests results.

9.4 Vehicle Stop-Sign Results and Analysis

Due to incompatibilities in our model architecture and the implementation of this feature we could not test the integration. However, we opted for a simulated method to test the feature separately. To test the algorithm's success, pre-recorded video footage of someone driving was fed to the algorithm. The results of the algorithmic output are depicted in Figures 24 - 26 using different road views.

Figure 24 Stop sign recognition algorithm performance based on front-facing stop sign at closer distance. The algorithm performed the best under these conditions.

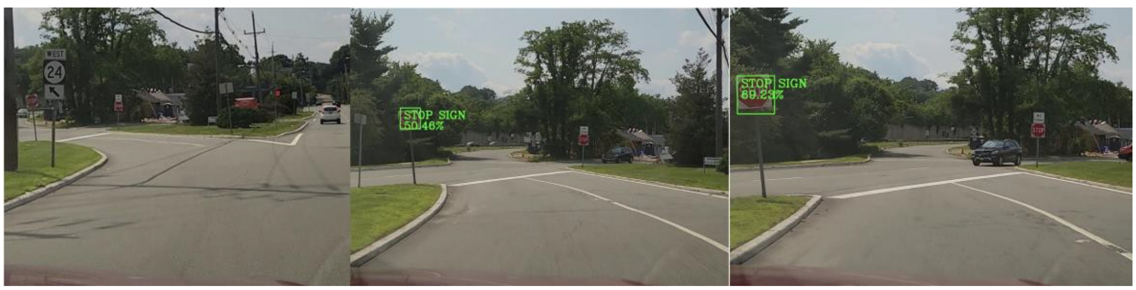

Figure 25 Stop sign recognition approaching stop sign on a curve road.

Figure 26 Stop sign recognition approaching a tilted sign facing a different direction at far distance. The algorithm performed the worst under these conditions.

The results are as suspected. SSD struggles to identify stop signs from farther distances but grows in accuracy the closer the vehicle is to the stop sign. The more visible, straightforward the stop is the higher the accuracy. In addition to testing the sign recognition, we tested the integration of its intended output with the PID algorithm. We used randomly generated data to test the PID's response to a distance value to be sent by the ROS publishers. The vehicle slowed down and stopped based on the appropriate values for the intended 20 seconds as expected.

CHAPTER X CONCLUSIONS AND FUTURE WORK

10.1 Conclusions

Autonomous vehicles have a long history in our transportation system. Their inception happened over a hundred years ago, almost immediately after the introduction of commercial automobiles when we learned the threat that human drivers posed to public safety. As of 2019, over 39,000 lives were taken by vehicular crashes, most of which are still caused by human error [14]. The threat posed to society then is still a matter of public safety now. Throughout the different development stages of our transportation system, AVs had been a dream deemed impossible to achieve. Improvements in computing power and the development of fields such as machine vision have turned the dreams of self-driving cars into a reality. Currently, the government and enterprises like Waymo, Tesla, Nuro and major car manufacturing giants like Toyota and Volvo are working together to develop vehicles of higher autonomy and to take the necessary precautions for their integration into dynamic human environments. While there is still some push back and a recent decrease in public acceptance, it appears that this generation will be the generation to live the reality of the AV enthusiasts of the early 1900s. Furthermore, climate change due to greenhouse gas emissions continues to worsen the environment and threaten the health and life of all living creatures. The biggest contributor to man-made greenhouse gas emissions is fuel combustion produced predominantly by private vehicles and traffic congestion [8]. Solar and electrical energy are proving to be a solution for our energy and environmental problems. Alternative energy sources are being integrated in both autonomous and human operated vehicles growing in commonality and affordability. All research, actions, collaborations, and contributions indicate that electric and autonomous vehicles are here, and their integration is moving forward for a long-waited update to our transportation system.

In this project, we present a 1/10-scale autonomous vehicle integrated with three key safety features: PID control system, lane-tracking system and stop sign recognition system. In our implementation we used software tools like Arduino, ROS, Python, OpenCV and hardware components like the ultrasonic sensors, Arduino board and NVIDIA Jetson Xavier board. We were able to achieve the integration of 2/3 features and identified opportunities for improvement for future implementations. Nonetheless, the purpose of this project was to test and learn about the different components of autonomous vehicles considering the important role that they will play in the future of transportation, the various software tools that can be used to build them and explore the different features that can be incorporated in them. This approach is an affordable and feasible way to experiment with AVs in comparison to life-sized vehicles. Similar projects at larger scale can be used to help the public "demystify" AVs and to expose students to the upcoming change in the workforce.

10.2 Future Work

There are many opportunities for improvements, however, we have identified the most relevant to our work. Future work in the PID control algorithm is to change the variables from being casted as *ints* to decimals for more precise steering and speed. For the lane tracking system, modifications to the algorithm to create a center lane focus needs to be implemented. In addition, incorporating machine learning methods like convolutional neural networks (CNNs) to produce a resilient and interference tolerant lane tracking system is another approach that we would like to incorporate in the future. For the stop sign recognition system, an alternative approach of reinforcement learning is proposed for future implementation to fix the dependency problem discovered with SSD. We can also incorporate more types of different traffic signs such as speed limit, school signs and animal signs to be recognized by the system using various image classification approaches with training datasets

The 1/10-scale AV was equipped with front and rear sensors but only the first sensors are used in this build of the vehicle. Future implementation in this regard is to use the back sensors and add more driving functionalities to the vehicle such as reversing, making intricate turns, and avoiding rear collision. Collaboration between two or even more separate models communicating on the road would also add another realistic functionality implemented in life-sized autonomous vehicles.

REFERENCES

[1] L. o. Congress. "Who invented the automobile?" https://www.loc.gov/everyday-mysteries/item/who-invented-the-automobile/ (accessed November 7, 2021).

[2] F. M. Company. "Our History." Ford. https://corporate.ford.com/about/history.html (accessed November 12 2021).

[3] (2021). *Oil and petroleum products explained*. [Online] Available: https://www.eia.gov/energyexplained/oil-and-petroleum-products/oil-and-the-environment.php

[4] E. L. Jill E. Johnstona, and Hannah Roha, "Impact of upstream oil extraction and environmental public health: a review of the evidence," *Department of health and human services,* vol. 657, pp. 187–199, March 20 2019, doi: 10.1016/j.scitotenv.2018.11.483.

[5] N. O. a. A. Administration. "Oil Spills." National Oceanic and Atmospheric Administration. https://www.noaa.gov/education/resource-collections/ocean-coasts/oil-spills (accessed November 17, 2021).

[6] J. G. J. O. a. J. A. H. W. Peters, "Trends in global CO2 and total greenhouse gas emissions: 2020 Report," PBL Netherlands Environmental Assessment Agency, The Hague, December 2020 2020. Accessed: November 2. [Online]. Available: https://www.pbl.nl/sites/default/files/downloads/pbl-2020-trends-in-global-co2-and_total-greenhouse-gas-emissions-2020-report_4331.pdf

[7] P. N. E. A. Agency, "Trends in global CO2 and total greenhouse gas emissions: 2020 Report," i. o. y. a. i. t. y. b. a. a. a. r. Table C.1 Average annual change of global CO2 emissions 1970-2018 in global recession years, Ed., ed. Trends in global CO2 and total greenhouse gas emissions: 2020 Report: The Hague, 2020, pp. Average annual change of global CO2 emissions 1970-2018 in global recession years, in other years and in the year before and after a recession.

[8] M. H. R. Hoong Chor Chin, "An Impact Evaluation of Traffic Congestion on Ecology " *Planning Studies & Practice* vol. 3, no. 1, pp. 32-44, 2011.

[9] 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.

[10] A. S. a. L. S. Vipin Jain, "Road traffic congestion in the developing world," presented at the Proceedings of the 2nd ACM Symp. on Computing for Development, Atlanta, Georgia, 2012. [Online]. Available: https://doi.org/10.1145/2160601.2160616.

[11] B. E. David Schrank, Tim Lomax, "2019 Urban mobility report," The Texas A&M University System, August 2019. [Online]. Available: https://static.tti.tamu.edu/tti.tamu.edu/documents/mobility-report-2019.pdf

[12] F. Kröger, "Automated Driving in Its Social, Historical and Cultural Contexts," 2016, pp. 41-68.

[13] F. Kroger, "Further developed panorama, USA 1953," ed. Mechanix Illustrated, 1953.

[14] N. H. T. S. Administration. "2020 fatality data show increased traffic fatalities during pandemic " United States Department of Transportation. https://www.nhtsa.gov/press-releases/2020-fatality-data-show-increased-traffic-fatalities-during-pandemic (accessed November 29, 2021).

[15] C. f. D. C. a. Prevention. "Global road safety." U.S. Department of health and human services https://www.cdc.gov/injury/features/global-road-safety/index.html (accessed 2021).

[16] U.S. Department of Health & Human Services. (2016). *Motor Vehicle Crash Deaths*. [Online] Available: https://www.cdc.gov/vitalsigns/motor-vehicle-safety/index.html#:~:text=Major%20risk%20factors%20for%20crash,more%20than%209%2C500%20crash%20deaths.

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

[18] S. A. Anand Bhat, Ragunathan Rajkumar, "Tools and Methodologies for Autonomous Driving Systems," *Proceedings of the IEEE,* vol. 99, pp. 1-17, July 2018, doi: 10.1109/JPROC.2018.2841339.

[19] Wevolver, "2020 autonomous vehicle technology report " Wevolver Nexperia 2020 2020. Accessed: November 7, 2021

[20] Tesla. "Future of Driving " Tesla. https://www.tesla.com/autopilot (accessed November 12, 2021).

[21] F. Agenda, "The future of autonomous vehicles," in "Global insights gained from multiple expert discussions," 2020. Accessed: April 20, 2021 [Online]. Available: https://www.futureautonomous.org/pdf/full/Future%20of%20Autonomous%20Vehicles%202020%20-%20Final%20LR.pdf

[22] Congressional Research Service (2021). *R45985, Issues in Autonomous Vehicle Testing and Deployment*. [Online] Available: https://crsreports.congress.gov/product/pdf/R/R45985

[23] D. Wakabayashi, "Self-Driving Uber Car Kills Pedestrian in Arizona, Where Robots Roam," in *The New York Times* ed. The New York Times The New York Times 2018.

[24] N. E. Boudette, "'It Happened So Fast': Inside a Fatal Tesla Autopilot Accident," in *The New York Times* ed. The New York Times: The New York Times 2021.

[25] T. Guardian, "Two die in Tesla car crash in Texas with 'no one' in driver's seat, police say," in *The Guardian* ed. The Guardian: The Guardian 2021.

[26] M. Taylor. (2020, 2021) German Court Bans Tesla 'Autopilot' Name For Misleading Customers. *Forbes* Available: https://www.forbes.com/sites/michaeltaylor/2020/07/14/german-court-bans-tesla-autopilot-name-for-misleading-customers/?sh=1817d9d4c7fb

[27] Waymo, "Waymo Safety Report " Alphabet Inc., Waymo, February 2021. Accessed: 2021 [Online]. Available: https://storage.googleapis.com/waymo-uploads/files/documents/safety/2021-08-waymo-safety-report.pdf

[28] NURO, "Delivering Safety," NURO FAQs, 2021. Accessed: 2021. [Online]. Available: https://nuro.sfo3.digitaloceanspaces.com/Nuro-VSSA-2021_Final.pdf?mtime=20210411085155&focal=none

[29] A. f. h. a. safety. "Public opinion polls show deep skepticism about autonomous vehicles." Advocates for highway & auto safety. https://saferoads.org/issues/driverless-cars/ (accessed November 18, 2021).

[30] J. T. C Bloom, J Ramjohn, and L Bauer, "Self-driving cars and data collection: Privacy perceptions of networked autonomous vehicles," presented at the Thirteenth Symp. on Usable Privacy and Security (SOUPS 2017), Santa Clara, CA, USA, July 12-14, 2017, 2017. [Online]. Available: https://www.ftc.gov/system/files/documents/public_comments/2017/11/00046-141905.pdf.

[31] 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

[32] J. Linkov. "Special Report: The Electric Car Comes of Age." https://www.consumerreports.org/hybrids-evs/electric-car-comes-of-age/ (accessed November 30, 2021).

[33] B. Berman. "The key to autonomous vehicle safety is ODD." https://www.sae.org/news/2019/11/odds-for-av-testing (accessed 2021).

[34] T. T. Omar Takleh, N. abu bakar, S. Rahman, R. Hamzah, and Z. Abd Aziz, "A Brief Survey on SLAM Methods in Autonomous Vehicle," *International Journal of Engineering and Technology(UAE),* vol. 7, pp. 38-43, 11/30 2018, doi: 10.14419/ijet.v7i4.27.22477.

[35] X. Liu, T. Ballal, and T. Al-Naffouri, *GNSS-based Localization for Autonomous Vehicles: Prospects and Challenges*. 2019.

[36] Wevolver, "Autonomous Vehicle Technology Report ", ed. Wevolver 2020.

[37] J. Kocic, N. Jovičić, and V. Drndarevic, *Sensors and Sensor Fusion in Autonomous Vehicles*. 2018.

[38] NASA. "What is GPS?" NASA TV. https://www.nasa.gov/directorates/heo/scan/communications/policy/what_is_gps (accessed November 20, 2021).

[39] N. S. Place, "How Does GPS Work? ," E. i. s. b. n. s. C. NOAA., Ed., ed. NASA Science NASA, 2021.

[40] A. Kipf, T. Kipf, B. Radke, V. Leis, P. Boncz, and A. Kemper, "Learned cardinalities: Estimating correlated joins with deep learning," *arXiv preprint arXiv:1809.00677,* 2018.

[41] X. Zhou, C. Chai, G. Li, and J. Sun, "Database Meets Artificial Intelligence: A Survey," *IEEE Transactions on Knowledge and Data Engineering,* vol. PP, pp. 1-1, 05/14 2020, doi: 10.1109/TKDE.2020.2994641.

[42] J. Sun and G. Li, "An end-to-end learning-based cost estimator," *Proc. VLDB Endow.,* vol. 13, no. 3, pp. 307–319, 2019, doi: 10.14778/3368289.3368296.

[43] N. H. T. S. Administration. "Vehicle-to-vehicle communication." NHTSA. https://www.nhtsa.gov/technology-innovation/vehicle-vehicle-communication (accessed November 30, 2021).

[44] M. S. U. Alam, "Securing Vehicle Electronic Control Unit (ECU) Communications and Stored Data," 2018.

[45] N. H. T. S. Administration. "The Evolution of Automated Safety Technologies." United States Department of Transportation. https://www.nhtsa.gov/technology-innovation/automated-vehicles-safety (accessed 2021).

[46] N. H. T. S. Administration, "Automated driving systems a vision for safety 2.0," National Highway Traffic Safety Administration, National Highway Traffic Safety Administration, 2017. Accessed: 2021. [Online]. Available: https://www.nhtsa.gov/sites/nhtsa.gov/files/documents/13069a-ads2.0_090617_v9a_tag.pdf

[47] 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, doi: https://doi.org/10.1016/j.jtrangeo.2018.03.010.

[48] S. International. https://www.sae.org/news/press-room (accessed April 21, 2021).

[49] B. M. D. Kaitlyn Atkins, Michelle S. Dromgold-Sermen, Hannah Potter, Viji Sathy & A. T. Panter "“Looking at Myself in the Future”: how mentoring shapes scientific identity for STEM students

from underrepresented groups," *International Journal of STEM Education,* vol. 7, April 7 2021 2020, Art no. 42, doi: 10.1186.

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

[51] NVIDIA. "Jetson Xavier NX Series " NVIDIA. https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/jetson-xavier-nx/ (accessed 2021).

[52] osoyoo, "About ultrasonic sensor HC-SR04," ed: osoyoo, 2018.

[53] ROS. "About ROS." https://www.ros.org/about-ros/ (accessed April 23, 2021).

[54] ROS. "Core Components." ROS.org https://www.ros.org/core-components/ (accessed April 24, 2021).

[55] V. A. Naveenkumar Mahamkali, "OpenCV for Computer Vision Applications," presented at the Proceedings of National Conference on Big Data and Cloud Computing (NCBDC'15), March 20, 2015. [Online]. Available: https://www.researchgate.net/publication/301590571_OpenCV_for_Computer_Vision_Applications.

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

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

[58] A. A. D. Mohamed Fezari, "Integrated Development Environment "IDE" For Arduino," Al Zaytoona University, A, October 2018. Accessed: April 25, 2021. [Online]. Available: https://www.researchgate.net/publication/328615543_Integrated_Development_Environment_IDE_For_Arduino

[59] V. S. Code. "Code editing. Redefined." https://code.visualstudio.com/ (accessed 2021).

[60] M. Maldini. "PID control explained." Medium. https://medium.com/@mattia512maldini/pid-control-explained-45b671f10bc7 (accessed November 28, 2020).

[61] L. P. Victor Robila, Mihir Rao, Iris Li, Michelle Zhu, Weitian Wang "Design and Implementation of PID-Based Steering Control for 1/10-Scale Autonomous Vehicle," in *2021 IEEE 12th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*, New York 2021, 2021.

[62] OpenCV. "Hough Line Transform." https://docs.opencv.org/3.4/d9/db0/tutorial_hough_lines.html (accessed 2021).

[63] OpenVINO. "mobilenet-ssd." https://docs.openvino.ai/2021.2/omz_models_public_mobilenet_ssd_mobilenet_ssd.html (accessed 2021).

[64] COCO. "Common Objects in Context " https://cocodataset.org/#home (accessed 2021).

[65] C. Tomasi. "A Simple Camera Model" https://courses.cs.duke.edu//fall16/compsci527/notes/camera-model.pdf (accessed 2021).