Autonomous Maze-Solving Robot: Explore and Solve Environments to Recreate 3D Spaces in 2D Using Ultrasonic Active-Imaging Sonar

Christian Leonard

Park University

Abstract:

This paper explores the creation of Walter, an autonomous, ultrasonic sensor-driven maze-solving robotic vehicle. Further detail is used to explain the many parts of Walter, including the Arduino microcontroller, ultrasonic sensors, servo micromotors, and the wheels and motors that operate Walter. The Left Hand Rule maze-solving algorithm is employed to allow Walter to navigate and record different environments. This paper also expresses the potential uses of robotics in cave exploration and rescue and boils the issue down to its basic components, utilizing a uniform maze as a starting point and building up from there. Finally, it will be discussed how future versions of Walter could improve upon some of the current iteration's limitations.

***Keywords*:** Autonomous, Ultrasonic Sensor, Maze-Solving, Arduino, Microcontroller, Left Hand Rule, Cave Exploration and Rescue

Intro:

In the middle months of 2018, 13 individuals were discovered to have been lost and trapped within a cave in Thailand. After becoming lost within the cave while exploring, in combination with an excess of rain, the 12 young boys and their soccer coach were stuck within the cave. It took many man-hours and the expertise of several skilled divers a week's worth of time to rescue the entire team of boys from the cave [1]. It was this exact incident that sparked the idea of an autonomous system that would have the means to navigate the cave and find a way to the exit. The goal of this research project was to create a relatively small, self-operating rover vehicle capable of solving complicated pathways through the use of sensor-driving data to, theoretically, be able to navigate and safely exit caves in order to fulfill this search and rescue need.

Caving, sometimes referred to as spelunking, is, "the recreational pastime of exploring wild cave systems." Caving is a growing activity in the United States. With an increase in the number of those exploring caves, naturally, it also comes along with an increased number of potential cavers who may fall off the path and become lost within the cave. Thus, the lost must be found [2].

Background:

Robotics has been used in various ways over the past few decades to solve many issues that mankind has to face. The ability for the field of robotics to grow and adapt means that the number of problems that robotics is able to overcome continues to expand. Robotics is an ever growing field of engineering and computer science that will proceed to snowball as the abilities of robots get cheaper, faster, and more capable.

Many forefront technology companies nowadays rely on robotics to accomplish many of humanity's hardest challenges. NASA currently employs a convoy of different robots to go where no man has ever gone before, such as satellites, probes, and now, rovers.. The Mars Rover named Curiosity traverses the surface of the Red Planet to test the soil of Mars, take photos, and explore Mars. Through the use of robotics, we now have photos of Mars that we, as humans, have never had nor would have to this day, without the Curiosity rover. To be exact, the Curiosity rover has traveled over 13.5 miles while taking an excess of 640,000 photographs during its 2,658 Sols, or Martian days spent away from Earth. NASA has collected photos and videos as well as discovered oases and seasonal variations across the planet with the use of Curiosity [3].

Tesla, the predominant electric car manufacturer in the world, has made many strides in the field of autonomous driving to operate their vehicles. The goal of their autonomous driving system is to be able to navigate their vehicles without the use of a human driver. Through the use of sensors, cameras, and specially designed software, the Tesla vehicles will be able to cross the United States without the intervention of anyone touching the steering wheel [4]. Tesla's combination of software and hardware have transformed their electric vehicles into car-sized computers capable of transporting their owners anywhere they wish.

NASA and Tesla are prime examples of how the use of hardware and software in combination results in cutting-edge robotics capable of overcoming the most challenging and sensitive obstacles mankind faces today. Both Tesla and NASA employ self-operating robotic vehicles to accomplish their goals. Both robotic systems rely on sensors to study their surroundings. These vehicles are then capable of analyzing the environments they are in, determine the course of action to take, and then successfully execute the necessary steps to ensure success in their different missions. Tesla and NASA are examples of how robots are capable of succeeding in their respective missions in similar sensor-driven, autonomous ways, though the goals of those missions vary greatly. Similarly, robotics has shown great promise in the field of cave search and rescue. Introduce Walter -- the Wall Tracking Extraction Rover.

When it comes to gauging the availability of Walter, both in terms of creation of Walter and increasing the availability of Walter to the cavers, the main aspect was keeping Walter simple. Simplicity refers to the function and price of Walter, or the ability to keep the price cheap while limiting what effectively operates Walter.

The starting point for Walter was to use the UCtronics K0072 -- an off-the-shelf toy [5]. Although highly customizable, the K0072 comes already programmed to perform basic tasks, such as being able to operate it remotely with a smartphone, tablet or Infrared remote. Additionally, the ability to follow lines along the ground through the use of a line-reader, and a very poorly functioning obstacle avoidance mode. The obstacle avoidance feature on the standard K0072 would routinely run into the objects it was designed to detect, usually ending in the vehicle driving into a wall, wheels continuing to turn while traveling nowhere. Thankfully, due to the self-teaching nature of the kit that is the K0072, nearly every aspect of the vehicle was able to be dismantled, rearranged, and reprogrammed.

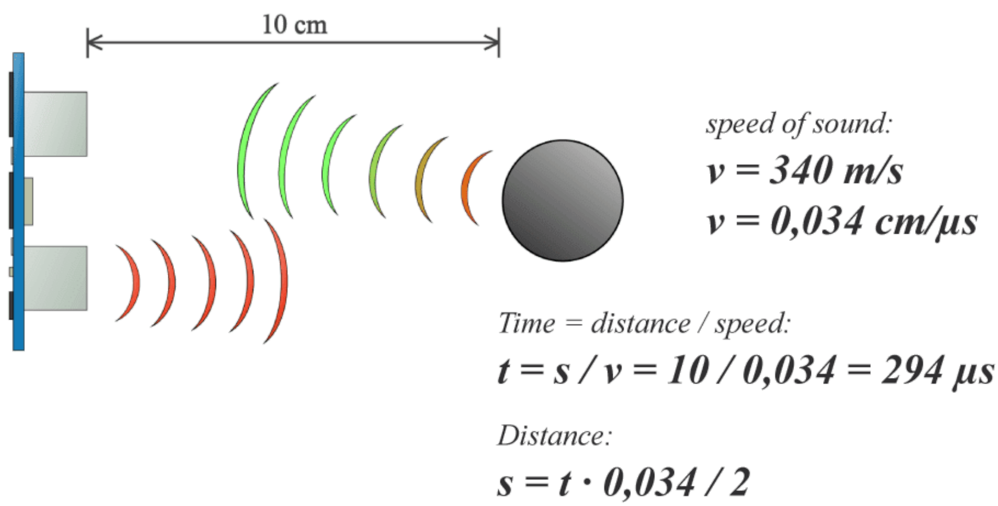
The small computer that operates the vehicle, what is essentially the brain of Walter, is an Arduino Uno unit, pictured below.

[6]

The Arduino is an Italian made microcontroller that is able to be plugged directly into a computer and updated as you so wish. The Arduino includes many additional ports that house and operate several sensors and accessories throughout the vehicle. The Arduino stores all the code that controls the functions of Walter and is ultimately responsible for the success of Walter’s movements.

In conjunction with the Arduino is the Drive Board, which can be thought of as the nervous system of Walter. Though the Arduino tells Walter what to do, the Drive Board is responsible for relaying these commands to the other portions of Walter, such as moving the sensors or driving the wheels. The drive board of Walter is itself not directly programmable, but instead, responds to the immediate demands of Walter's brain. With the drive board and the Arduino microcontroller at the helm of Walter, it has all of the logistical capabilities required to change this small toy generally made for children into a cave rescue robot.

It is impossible to look at Walter without noticing the large, ultrasonic sensor attached to the front of the body of the main board. The ultrasonic sensor functions as Walter's eyes. Besides the Arduino and drive board, the ultrasonic sensor is perhaps the most crucial part of Walter's anatomy. Similar to how a bat uses echolocation or a submarine uses sonar, the ultrasonic sensors send out a series of pings, or sound waves, and waits for any return signals to be received. The name 'Ultrasonic' refers to the type of waves that the sensor is sending out. The sound waves are above the audible hearing waves of a human, meaning that the sending and receiving of these waves aren't able to be heard by the human ear. The ultrasonic sensor Walter uses has two 'eyes' or sensors -- one for sending the waves and one for receiving the returning waves after they have reflected off of any nearby obstacles. The following diagram demonstrates how the ultrasonic sensor functions as well as the equation that Walter uses to determine the distance of obstacles encountered within the maze.



[7]

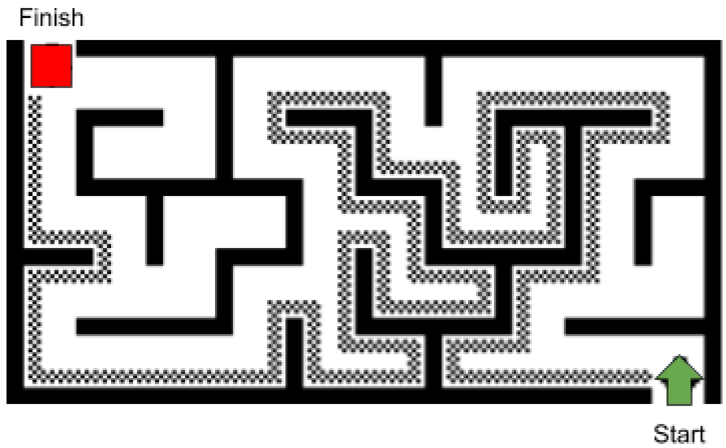
Although Walter's eyes start out looking straight forward, the ultrasonic sensor's view would be very limited if it weren't for the assistance of the Servo motor. The servo is essentially the neck of Walter, allowing for the ultrasonic sensor to turn left and right [8]. If the ultrasonic sensor facing forward is at 90 degrees, the servo allows for rotation between the range of 180 and 0 degrees. The servo motor moves with enough finesse to be able to accurately turn the ultrasonic sensors to each side for the sensor to take readings, and ultimately, detect the existence of obstacles within the entire front 180-degree field of view in front of Walter [9].

Walter features a set of four individually operated variable speed electrical motors used to drive the wheels. These motors allow for the traversal of different environments. The speed of these motors is adjustable within the program executed by Walter. However, since each motor is operated individually of the others, the speeds for each motor have to be individually dialed in to allow for similar rotational speeds from each wheel.

These motors are also powered by Walter's power supply, as is every piece of Walter. The power supply for Walter is a pair of 18650 Lithium-ion batteries. Though rechargeable, these batteries hold their charges for a considerable amount of time and don't often have to be charged in any capacity to appropriately power all components of Walter. This is ideal for the use of cave exploration and rescue, as having unreliable batteries powering your means of rescue would entirely negate the validity of the robot.

Walter is operated through a maze solving algorithm that is executed by the Arduino, or brain, and operates the other motors, sensors, and boards of Walter in order to form the complete picture that is this robot. The algorithm used can be broken down into two distinct parts. Firstly, the algorithm must be adaptable. Due to the ever changing and unpredictable nature of caves, Walter's algorithm must be able to analyze and respond to the environment nearby [10]. If there was a guarantee that each caver would get caught within the same cave, the underground labyrinth itself could be analyzed and programmed within Walter so that, instead of adapting, Walter knows that cave and that cave alone. However, the activity of spelunking dictates that cavers explore many different types of caves, each one different in its own way. When deep underground, the dark, cold walls of the cave can seem like a maze. That is exactly how Walter began to operate.

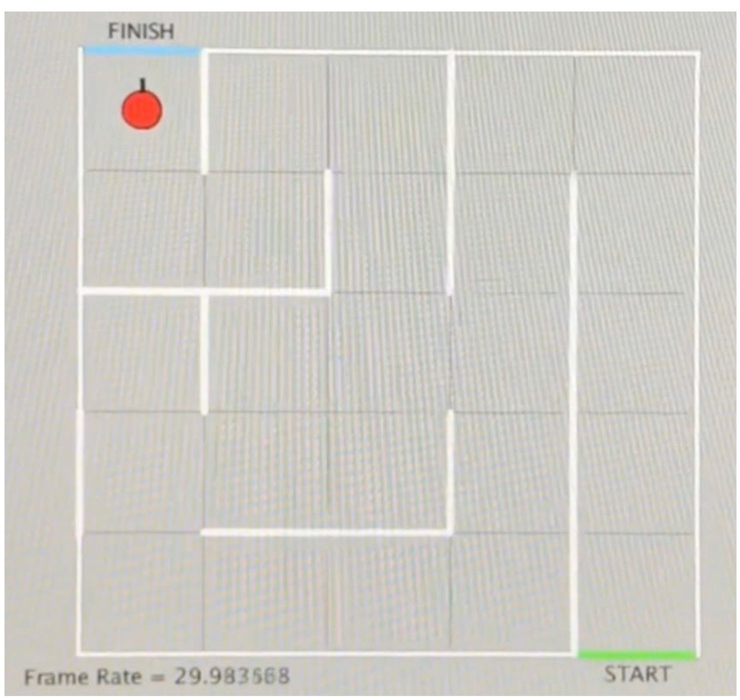
The first iteration of Walter uses what is referred to as the "Left-Hand Rule" to solve mazes and traverse environments. The Left-Hand Rule is a method of solving any maze by following the wall directly to the left of you throughout the maze until reaching the end.



[11]

By following the left wall throughout the process of solving the maze, Walter will be able to solve any maze of any size successfully.

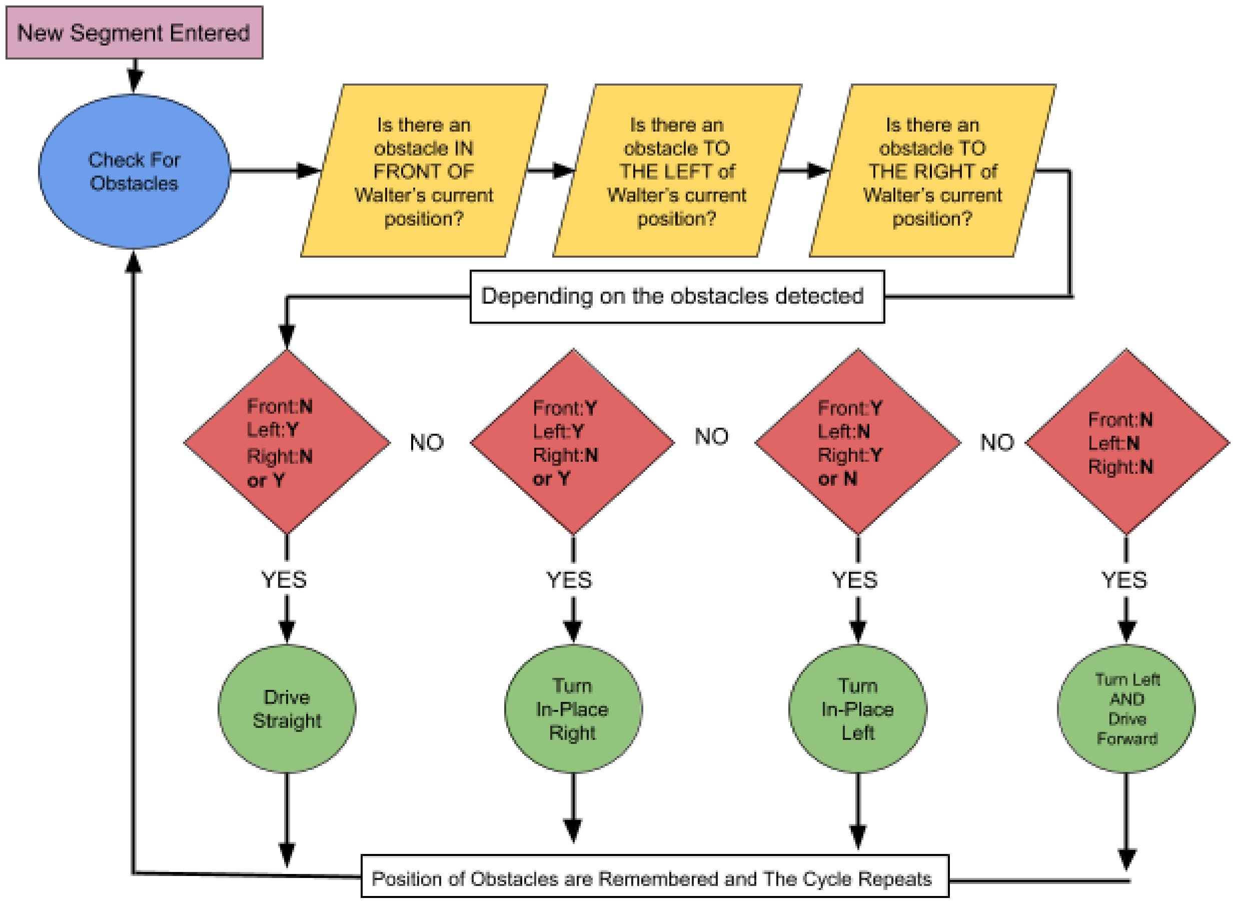
Finally, Walter reacts to the obstacles by recording a digital representation of the maze as it is solved. The digital representation can be best thought of as a drawing of what Walter saw while solving the physical maze that was navigated. Every distinct obstacle within the maze, every turn that Walter took while solving the maze, even the direction Walter was facing at a given instance of the maze is all logged and saved in Walter's digital storage. The following image is the digital recreation of the maze used for the testing of this iteration of Walter, represented as the red dot, with all obstacles represented as white lines.



This is all logged as Walter goes through the maze using a system called EEPROM [12]. EEPROM allows for Walter to recall the location of each wall and the directions followed for each maze until Walter's memory is reset through the use of a specially made program that can easily be uploaded to Walter between each trial.

Walter is able to interpret the different signals received by the ultrasonic sensor when encountering the walls within the maze as obstacles in his memory and where they are located compared to Walter at the instance they were encountered. Through this method, every time Walter explores an environment, it can be digitally reconstructed as an accurate 2D depiction to the cavers or rescuers before entering an unknown environment blind through the Arduino's built in Serial communication [13]. This feature combined with Walter's ability to navigate areas that people may not be able to due to his small size and then relay those findings accordingly.

Walter's algorithm works sequentially to solve the maze. Walter assesses each unit of the maze individually, then links those units together until the maze is solved. For instance, upon entering a new segment of the maze, Walter takes a reading of the surrounding area, checking firstly in front, then left, and finally to the right. Walter scans for obstacles nearby, and measures the distance to those obstacles to determine how much room is available to move in. If Walter determines that a turn is required to move forward in the maze, a correction of distancing occurs where Walter adjusts position to better execute the turn within the narrow pathways of the maze. This algorithm is executed numerous times throughout the maze, ensuring that Walter fully understands the position and condition of each segment of the maze until it is complete. It is here that Walter also learns the position of the obstacle within the maze -- remembering them to be recalled and drawn digitally at a later time through the use of Walter's onboard memory called EEPROM. The algorithm Walter uses to solve the mazes, however, is much more complex follows can be demonstrated in the following flowchart:



For a complete repertoire of Walter's coded algorithm, see Appendix A.

Procedure and Design:

The initial plan for creating Walter was to create a robot capable of navigating itself out of a cave should someone get lost spelunking or sending Walter into a cave that was previously unexplored, so cavers wouldn't be going into new caves completely blind. For the beginning stage of Walter, we focused on scalability. Reducing the project to its basic parts, the first stage was to create a robot that was capable of driving around and detecting obstacles [14]. The next stage was to create a basic environment to navigate that would represent the twist and turns of a cave.

During the first phase of testing, the measure for success was to answer the question "Can Walter solve a maze?" Walter's logic for the left-hand rule means that the left wall of the maze will always have priority. As previously mentioned, the goal wasn't to have Walter solve a single maze but to be able to solve any maze that

Walter was put up against. In order to handle every maze, Walter must first correctly solve all possible scenarios that can occur within any maze [15]. For testing, we simplified the cave down to a maze, and to further simplify it, we limited the maze to be a series of 5x5 cells. If Walter can solve a 5x5 maze, by the same logic, Walter can solve larger mazes. From there, Walter's abilities would be expanded to function within a cave environment.

Breaking the maze down to the individual cell, we can see how Walter handles each maze. Upon entering a new cell, Walter checks the path directly in front, to the left, and to the right of him and determines the correct decision to make. If there is an obstacle to the left of Walter but the front path is clear, Walter will continue forward while following the left-hand rule. If the front path is blocked but the left path is clear, Walter will turn left and drive forward. Similar logic is in place for every combination of obstacles, whether it be open paths or dead ends. Walter completes this cycle of logic every time a new cell is entered and continues until the maze is exited successfully. Walter has the ability to draw these cells as they are solved, being pieced together at the end to form the complete image of the 5x5 maze [16].

The first iteration of Walter is capable of accurately solving the 5x5 maze in under five minutes, with the fastest time of completion being 4 minutes 43 seconds. This time frame was previously much longer, as Walter is relatively large for the size of the physical maze created for this project. With Walter measuring just under 11 inches length-wise, and the cells of the maze being smaller than the intended 12 inch squares, some instances of Walter solving the maze leaves just 1-2 centimeters between the walls and Walter. This added a heavy focus to accuracy and exact positioning of Walter within the maze to ensure the success of specific turns within the maze. In a more size-appropriate setting, Walter would not have to self-adjust as frequently as is needed currently and would be able to solve larger mazes much quicker. However, with the given maze, it shows Walter's ability to work in confined spaces effectively while being able to align and maintain the needed positions to complete the maze from start to finish.

Starting from the original kit of the K0072, there was a need to define what Walter should be capable of doing and what features weren't needed for the purposes of this project. Along with the previously mentioned parts, the K0072 toy came with additional parts and features that were deemed accessory or unnecessary for Walter's success. One of these parts was a line reader, used for children to put tape on the ground and the robot to follow it without question. The issue here is that caves don't have any tape or other line features along the ground to follow.

To put something, such as tape, along the ground of a cave would imply the individual has already explored that part of the cave, or can even move in the first place, and would entirely defeat the purpose of Walter. Additionally, the K0072 also came with a television remote-style operating controller so it can be controlled with button presses from a distance [17]. This fails with regards to Walter's mission as well as the remote is infrared, and therefore, requires line-of-sight to be functional. This means Walter would only be functional for as long as the individual could point the remote directly at Walter. This is also skipping over that this remote operates on its own battery, which in testing needed to be replaced frequently. The omission of the line reader and the remote operation functionality both decreased the weight of Walter, preserving the onboard battery, and saved memory on the Arduino board to allow for Walter to both operate quicker and remember more information.

When creating Walter, it was important to understand each part that was deemed important enough to keep and each part's individual functions. While coding tests for each part, there were a few issues that became apparent in the initial phase. Firstly, the motors that control the wheels of Walter. As mentioned previously, each motor controls only one wheel and each motor operates at its own speed. That is to say, to keep Walter driving in a straight line with each wheel rotating at the same speed, each motor needs to be set to its own power level. This was a fairly arbitrary way to dial in each motor, as well, which turned out to be another issue when creating Walter. To determine the power of each motor, there was no method that worked as well as simply guessing and checking. This process had to be completed four times, once for each motor, before Walter would drive in a straight line instead of pulling off to one side or the other.

Besides the motors, Walter continued to struggle with driving due to the quality of the wheels that came included with the K0072 kit. Due to the poor tread of the tires, which are predominantly plastic, the wheels gain or lose traction depending on the surface Walter is driving on. Noting this, in conjunction with the individual motors per wheel, Walter often has to be reprogrammed to perform accurately on new surfaces. It was also recently discovered that Walter's performance solving the maze depends on the charge of the batteries operating the motors -- if the batteries drop too low, Walter will not turn the same distances as if the batteries were charged.

Testing Walter for the ability to detect obstacles discovered that ultrasonic sensors have as many limitations as it does benefits. The ultrasonic sensor that is currently used is an active imaging sonar with a very broad ability to define obstacles. The sensor can differentiate something being there versus something not, however, the ability for the sensor to define detail is limited. The sensor is capable of detecting things within a range of 300 centimeters, but the sensor can do little more than define the edges of obstacles at the moment.

Paired with the ultrasonic sensor, the servo motor had a major issue that limited the ability for Walter to turn the ultrasonic sensor. The servo motor is a series of gears that turn back and forth and rotate a shaft from within as the gears move. The servo motor fell apart from within during a test after the ultrasonic sensor got stuck while turning too far to one side. The servo would not turn at all after the gears became misaligned, and required it completely be taken apart, realigned, and put together again before turning at all. However, after the repair, the field of view that the servo could effectively rotate the ultrasonic sensor was nearly halved in each direction. The servo, therefore, may not be reliable enough in its current condition for the act of cave rescue. Though the configuration of Walter's components are now rearranged so that no one part interferes with the movement or function of another, the only way to get the servo back to full function was to completely replace it.

Interdisciplinary Aspects of the Project:

The creation of Walter employed knowledge from many aspects of several fields. Though similar vehicles have been built for search and rescue purposes, none specialize in caving. This required Walter to be specialized in the design and execution of Walter. The design of Walter involved a great deal of electrical engineering, having to operate and troubleshoot many sensors, including the ultrasonic sensor, and circuit boards throughout the project. Electrical engineering also helped explain Arduino's EEPROM volatile storage that is used to record and reconstruct Walter's memory after each solve. When parts were broken and had to be disassembled and reassembled, abilities in mechanical engineering were necessary to understand the forces and movement intended by each part, such as the case for the Servo motor.

Walter is very computer science intensive as well. Programming Walter required learning the Arduino embedded C++ coding language to program all of Walter's functions and to test each part individually, visual programming in the Java programming language to draw the information Walter stores during each solve to reconstruct the mazes, and maze-solving algorithms to determine how Walter would attempt to solve environments efficiently while still being mindful of the Arduino's limited storage capacity [18].

Math and physics allowed for the thorough comprehension of many specifics on Walter, namely how the ultrasonic sensors work and how to utilize Walter's four individually operated wheels to maintain effectiveness.

Through physics, we were able to reach a detailed understanding of the waveforms that the ultrasonic use to detect obstacles. Similarly, the varying rotation of each wheel had to be used to determine appropriate rotation for Walter to make for each turn, including angles and distances of obstacles in Walter's immediate area.

The interdisciplinary nature of Walter demanded a deeper understanding be had in each stage of the project. Through the use of electrical engineering, mechanical engineering, front-end and back-end programming, math and physics, the creation of Walter was able to teach not only how to solve issues within these fields, but an ability to look at a project, and that project's problems, from many different viewpoints. It was through such a diverse skill set as this that is required to undertake this project that truly allowed Walter to be as well-rounded as possible.

How the project could have been improved further:

The first iteration of Walter is a major improvement over the toy that it once began as and is a step in the right direction of cave exploration and rescue efforts. Though Walter is capable of solving the 5x5 maze built for this project in under five minutes, Walter has only solved the maze with no human interference six times. Walter is far from being at a point that a lost caver would be able to stake their life on. Though Walter is slowly improving with each software update and with further code refinement, there is only so much Walter can improve with the current hardware that is being utilized. Future versions of Walter would expand upon the current $50 base model of the robot and look to replace problematic parts.

Firstly, new wheels and motors are needed to refine Walter's movement abilities. The current plastic wheels that are currently being used provide little to no traction and are very unreliable. The ground that would need to be traversed within caves may be slick, rough, rocky, and anything in between. With the current wheels Walter has behaving differently between plywood and concrete floors, these wheels would be rendered useless within a cave. If these wheels were replaced with small rubber wheels or something resembling tank treads, the traction and reliability of Walter's movement would greatly increase.

Similarly, the motors Walter uses being independently operated with no way to determine exactly what power setting is required to get the desired results is wildly inefficient. If these electric motors could be replaced with stepper motors -- that is, motors that are highly accurate and programmable to the 'step' or single movement of the motor -- the synchronization of Walter's wheels would allow for more accurate movements from Walter. This would make turning easier and would provide less of an alignment issue when Walter moves.

Second, Walter requires at least an additional ultrasonic sensor mounted at the rear to better orientate. The single ultrasonic sensor at the front works great for a very limited environment, such as the 5x5 maze the initial tests occurred in. However, with the single sensor at the front, Walter can only determine the location of obstacles and the relative distance to those obstacles from the front of the robot. A second sensor would allow for similar readings to be made from the rear of the vehicle. The benefit of this is to be able to take more readings that would allow Walter to reposition the vehicle accurately and safely within the maze, as well as being able to take the necessary readings to solve each maze much quicker.

The final change for the next version of Walter would be to increase the on-board storage on Walter's Arduino unit. More storage increases the options Walter has when attempting to solve mazes, and ultimately, caves. Walter currently uses the Arduino Uno, a standard model of Arduino. However, if Walter instead were to operate on an Arduino Mega model, the storage would more than double and would allow for different algorithms to be used. With the limitations of the storage available on the Arduino Uno, Walter relies on the Left Hand Rule to solve mazes. The Left Hand Rule is far from the most efficient algorithm to solve mazes and instead, could be replaced with a better, larger algorithm if the storage were to be upgraded. Examples of more efficient maze-solving algorithms include A* or SLAM, both of which would fit well with Walter's current capabilities and goals [19].

Conclusion:

The current version of Walter is capable of navigating enclosed areas, recording the actions taken to exit and obstacles encountered along the way, and is finally able to digitally recreate a 2D overview of the area navigated. Though incredibly simplified, Walter 1.0 is, so far, successful in the mission of using robotics to explore and navigate cave systems. Walter has many flaws, including the parts required for movement and the use of a less than efficient algorithm, leaving Walter plenty of room for expansion and growth.

Though similar robots have been built for purposes such as helping with the aftermath of the September 11th attacks, Walter is the first to be focused on cave search and rescue [20]. The process of creating Walter has led me to a better understanding in robotics, electrical engineering, and programming of a physical system. Watching Walter come to life and improve with each update has only solidified my belief that robotics could be a solution to search, rescue, and exploration of caves.

Appendix A

GitHub Repository Featuring All Original Arduino and Processing Code Used For Functionality of Walter

https://github.com/MrChristianL/Arduino-Maze-Solving-Robot

Sources Cited: - UPDATE TO ORDER OF APPEARANCE IN PAPER

[1]. “Thailand Cave Rescue.” *CNN*, Cable News Network, 17 July 2018, www.cnn.com/specials/asia/thailand-cave-rescue.

[2] Maurette, M. “Mars Rover Autonomous Navigation.” SpringerLink, Humana Press, Mar. 2003, link.springer.com/article/10.1023/A:1022283719900.

[3]. “NASA Curiosity Rover.” *NASA Curiosity Rover*, mars.nasa.gov/msl/home/.

[4]. “Autopilot.” *Tesla, Inc*, www.tesla.com/autopilot.

[5]. “UCTRONICS Smart Bluetooth Robot Car Kit K0072- Charger Included, Battery Not Included.” *UCTRONICS*, www.uctronics.com/uctronics-smart-bluetooth-robot-car-kit-uno-r3-for-arduino-line-tracking-ultrasonic-sensor-hc-05-bluetooth-l239d-motor-shield-ir-remote-control-mobile-app-18650-battery-charger-included.html.

[6]. Figure 1. “Arduino Uno Rev3.” *Arduino Uno Rev3 | Arduino Official Store*, store.arduino.cc/usa/arduino-uno-rev3.

[7]. Figure 2. “Electronoobs.” *Electronoobs - DIY Projects and Tutorials*, en.xwebing.ro/tag/Arduino#.

[8]. “Arduino DC Motor Control Tutorial - L298N | H-Bridge | PWM | Robot Car.” *YouTube*, 8 Aug. 2017, www.youtube.com/watch?v=I7IFsQ4tQU8.

[9]. "Arduino Tutorial: Using a Servo SG90 with Arduino." *YouTube*, Jan. 2016, www.youtube.com/watch?v=SfmHNb5QAzc.

[10]. Arya, Ishan, et al. "Autonomously Solving Mazes with Robots." *Rutgers*, 21 July 2017, soe.rutgers.edu/sites/default/files/imce/gov2017/Autonomously%20Solving%20Mazes%20with%20Robots.pdf.

[11]. Figure 3. "Maze Solving Algorithm." *Wikipedia*, Wikimedia Foundation, 25 Feb. 2020, en.wikipedia.org/wiki/Maze_solving_algorithm.

[12]. Aasvik, Mads, and Ragnar Ranøyen Homb. "Using EEPROM to Store Data on the Arduino." *Norwegian Creations*, 17 Feb. 2017, www.norwegiancreations.com/2017/02/using-eeprom-to-store-data-on-the-arduino/.

[13]. "Serial." *Arduino Reference*, www.arduino.cc/reference/en/language/functions/communication/serial/.

Serial Input Basics, Dec. 2014, forum.arduino.cc/index.php?topic=288234.0.

[14]. Aggarwal, Param. "Mapping A Room With Ultrasonic Distance Sensors." *Hackster.io*, 7 Jan. 2018, www.hackster.io/paramaggarwal/mapping-a-room-wh-ultrasonic-distance-sensors-9725b7.

[15]. Toledo, F.J., et al. "Map Building with Ultrasonic Sensors of Indoor Environments Using Neural Networks." *Map Building with Ultrasonic Sensors of Indoor Environments Using Neural Networks - IEEE Conference Publication*, Oct. 2000, ieeexplore.ieee.org/document/885967.

[16]. YaddyVirusYaddyVirus 15311 gold badge22 silver badges1717 bronze badges, and Majenko♦Majenko 80.5k44 gold badges4646 silver badges9292 bronze badges. "Trying to Map a Room with Arduino Controlled Bot." *Arduino Stack Exchange*, 1 Mar. 1967, arduino.stackexchange.com/questions/39262/trying-to-map-a-room-with-arduino-controlled-bot.

[17]. Uctronics. "UCTRONICS/Smart-Robot-Car-Arduino." *GitHub*, 6 Nov. 2019, github.com/UCTRONICS/Smart-Robot-Car-Arduino.

[18]. Afrotechmods. “You Can Learn Arduino in 15 Minutes.” *YouTube*, Afrotechmods, www.youtube.com/watch?v=nL34zDTPkcs.

[19]. Durrant-Whyte, Hugh. “Simultaneous Localisation and Mapping (SLAM): Part I The Essential Algorithms.” *EECS Berkeley*, people.eecs.berkeley.edu/~pabbeel/cs287-fa09/readings/Durrant-Whyte_Bailey_SLAM-tutorial-I.pdf.

[20]. “Activities of the Rescue Robots at the World Trade Center from 11–21 September 2001.” *IEEE Robotics & Automation Magazine*, University of Waterloo, Sept. 2004, www.student.cs.uwaterloo.ca/~cs492/10public_html/papers/trial.pdf.