Designing and Prototyping an Accessible Kit for Middle School Students as an Introduction to Robotics

Alayna Erickson, Angel Sylvester
Introduction to Intelligent Robotic Systems
University of Minnesota
Minneapolis, Minnesota
eric3331, sylve057

***Abstract*— Educational robots offer an exciting opportunity for students to engage with the field of robotics. However, with the lack of accessibility of model kits for schools in lower income school districts in particular, the opportunities to receive hands-on experience are limited. This can potentially deprive students of the chance to be creative and learn in a non-traditional, perhaps more engaging, setting. In order to combat problems associated with the high cost of kits, we hope to introduce a robotics model kit that utilizes cheaper materials while still offering a sleek and fun interface for students in middle school. Inspiration will be drawn primarily from past precedents such as the eROSI and Lego Mindstorms kits in order to develop improved, more cost-effective alternatives. Further, a general overview of the landscape of educational robotics in the classroom will be undertaken in order to address other possible gaps in performance and effectiveness that other models possess.**

I. Introduction

Currently, existing robotics model kits can cost up to hundreds of dollars. For instance, the Lego Mindstorms Robotics and MakeBlock Kits can cost up to $350 each [7, 8]. This could impose a major financial burden on schools, especially for schools with limited funding. However, children can derive immense educational satisfaction through the hands-on, team-oriented approaches that are necessary in robotics-oriented coursework. Therefore, this project attempts to address the financial roadblocks to robotics course implementation by introducing a prototype of a model kit that is suitable and reasonably priced for students in middle school. Ideally, the kit would consist of a simple, easy-to-assemble model that students would be able to experiment with alongside guided lectures. Further, higher level coding activities would also promote hands-on exposure to relevant logic and syntax associated with programming robots.

The ultimate design and functionalities of the robot will culminate into a manual with curriculum centered activities to promote understanding in key robotics concepts such as kinematics, debugging, and team-work. The initial primary sensors that are being assembled will be LIDAR and RGB cameras to gain visuals on the environment, especially obstacle/target detection. In addition, the current visual design is very simplistic, with a box-like exterior through which the more intricate hardware components will be stowed away. Through a gentle introduction to the process of design and programming and the usage of an affordable robot kit, the hope is that more engagement is sparked for robotics in general. With the basic elements of the robot complete, versatility and expandability are other considerations to promote the longevity of this robotics kit for future generations.

II. Problem Description

Educational Robotics is a growing field with the potential to enhance engagement and excitement in students. It can serve as a promising tool that is versatile and effective at enforcing computational thinking [1, 2, 5, 9, 10, 11, 16, 17]. However, the costs associated with current robotics kits could make them inaccessible in low income school districts, as in [7, 8, 14]. In order to address this problem, a general review of the landscape of education robotics kits as well as a general examination of the parts cost breakdown will be necessary to acknowledge areas in which expenses can be minimized while still retaining a moderate level of functionality. In order to properly assess what fundamental components are needed for a basic robot, this project will also consist of a developed robot made from scratch to ensure a solid foundation for further development and optimization of price and functionality through future iterations of this project.

For this project, we will be designing and prototyping a robot to make into a kit that could be sent out to schools in the Twin Cities area. This kit would include an instruction manual detailing how to build the robot, all the parts needed, and the required code for programming. The robot overall would be easy and relatively cheap to produce, using widely available parts and materials. However, the design will be expandable and flexible so that it could be altered for different grade levels. The initial design must be simple enough for younger students to understand, so that they have an enjoyable time building the robot [11]. The target age for this initial robot is 11-13 years old, or students who are in 6th or 7th grade. This would allow students to be introduced at an early enough age that they could begin to understand robotics, and start to gain an interest.

To keep the kit build simple, we will likely decide to pre-build some of the components. This might include simplifying the wiring so the students only have to plug in a few cables. In that sense, we imagine the kit would be similar to a Lego Mindstorms kit [8]. Inspiration for the design of the robot will also be drawn from other implementations such as the eROSI that was done by the University of Minnesota [6].

III. Related Work

A. *Design Inspiration for the Current Robot*

One of the related works that we found while researching is *The Design and Evolution of the eROSI Robot* [6]. They designed and produced a small robot that had lower costs compared to other robots, and is compact in size, like our robot is going to be. There were a few generations listed in [6], which show some of the things we may run into with designing our own robot. These include design issues such as over-saturating photo-voltaic sensors, the speed of simple information retrieval, and battery charging. We can use this information of problems they faced in furthering the design of our own robot, in hopes that we can avoid some of these problems.

Another related work is *A Feasibility Study of Arducation Bot: An Educational Robotics and Mobile Application Kit for Computational Thinking Skills* [18]. In this paper, the authors explain some of what the robot does, like take in instruction to follow lines and solve a puzzle. This helps show some of the simple aspects that can be included in educational robotics; we used some of these simple functionalities to determine what we wanted our robot to do. They also detailed some issues that were present in the initial design of the robot. These include motor imbalances, Bluetooth connection issues, and a motor driver which did not provide enough drive. These instances will allow us to take into consideration some of these possible issues when we actually build the robot, and work on debugging it.

Other model kits that have been proposed in the literature include Flutter [3] and a robotics kit developed for the Boy Scouts of America [13]. Both have attempted to address the cost barrier that limits students from fully utilizing these tools in the classroom/extracurriculars. Flutter utilizes a block based programming interface to connect the robot to the user while the latter uses the Energia Development environment. Both sources provide a table of parts utilized to minimize cost while still maximizing experience. For instance, a microcontroller with bluetooth, programmable LEDs, sensor ports as well as general motors for navigation appeared to be the essence of a robot that would be engaging while not contributing to exorbitant costs.

Finally, the book *Lego Mindstorms NXT 2.0: The King's Treasure* [19] is a great example of an instruction manual for building robots. Lego Mindstorms are the most competitive kits available given their high degree of versatility, customizability, and price. Though it is specifically for Lego Mindstorms, it not only describes how to build the robot, but also provides a story behind why you need to build the robot that way. It makes building robots interesting for students, and can get them excited about learning robotics. To improve our own instruction manual, we may take a few pointers from this book, especially the diagrams showing how the robot needs to be put together.

B. An Overview of Curriculum Objectives

In order to cater to the needs of educators and the students, a general review of the learning objectives for students in primary and secondary schools was done through surveys sent to educators in the Twin Cities and perusal of literature that explore different curriculum objectives. A sweeping consensus is that curriculums should engage students under the guidelines imposed under the constructivist theory of learning [3, 12]. This means that through hands-on experience, educators can serve as a guide to learning through new experiences. Through the usage of open-ended problem solving tasks and construction of the robot, the students can gain more critical thinking, problem solving, and meta-cognitive skills. The primary learning objectives that a curriculum would be inspired by [12] where skills such as problem solving, self-efficacy, computational thinking, creativity, motivation, and collaboration are the central goals during the whole process of learning for students.

Within the robotics context, more specific goals include successful assembly of a robot with teacher guidance and general comprehension of basic computational concepts such as sequences, conditionals, and loops. From the literature, it appears that in order to streamline this process, the guide for assembly and further programming should be as intuitive as possible, allowing educators to assist even if they have little to no experience programming outside of this context. As a result, implementations have included a large amount of block coding based interfaces [4, 18, 19] given it’s intuitive user interface and minimized possibility of syntax errors.

IV. The Physical Robot

A. The Initial Design

As a starting point, different designs were explored such that the important functionalities (ie. sensors, motor, battery) can be incorporated while still being appealing for students as an introduction to robotics. For our initial design [*refer to Figure 1*], the behaviors that were chosen as promising were related to how the robot could navigate through some environment and interact with objects: it would detect an object that it needs to pick up, drive over to that object, extend an arm that reveals a container to hold the object, lower itself to the ground so that the arm was surrounding the object, retract the arm, and scoop up the object. Once it securely has that object, it will raise itself up, drive to a designated location, extend its arm, lower itself to the ground, and the back panel of the holding container will extend forward to push the object out.

The robot will then raise back up, retract its arm, and drive around in search for the next object to pick up. We designed a brief animation of the robot in action using Autodesk Fusion 360; this included the general parts of the robot, including the shell, arm, wheels, and holding container.

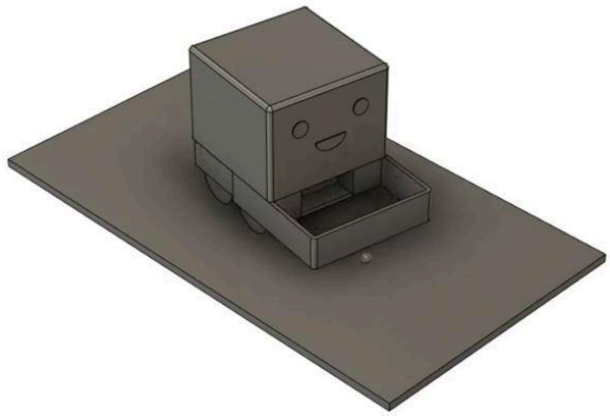

Figure 1: The proposed design of robot created and animated using AutoDesk Fusion 360.

Once we narrowed down our ideas, we began researching parts that we would need to build the robot. This includes: a single board computer, some vision component so that the robot can see, wheels and motors, a battery pack, etc. Once we did some research, we began drafting up a parts list [*refer to Table 1*]. The Raspberry Pi would serve as the onboard computer, since we can run it using Python as the primary programming language. For vision, we are going to use LIDAR to determine the boundaries of the room it's in, and also detect objects that might be in its path. In addition, we are going to use a small camera to be able to detect the object that it is going to pick up. For mobility, we are going to use omni-directional wheels. This is because omni-directional wheels could potentially add more versatility in terms of the forms of motion the robot might take, and we think it would add an extra interest level for students. Normal motors will be used for the wheels and arm, and a servo will likely be used for the back panel of the holding container. A 12V rechargeable battery pack will be used to power the robot. Finally, maroon- and gold-colored filaments will be used to 3D-print the shell of the robot, and the support structures within.

The finalized parts, which are all from Amazon, were added to a list with a description and link, and ordered. The budget for our project is $500, and the finalized parts amount to approximately $438.

TABLE I. List and Prices of Parts

Part	**Purpose**	**Price**
Raspberry Pi	On-board computer	$99.99
LIDAR	Obstacle detection	$99.99
Motors	Moving parts	$11.99
Mecanum wheels	Moving the robot	$31.99
Gear pack	Moving the arm	$7.99

Motor controller	Controlling the wheels	$19.99
Jumper wires	Connecting parts	$6.98
Servos	Moving parts	$8.99
Battery pack	Powering the robot	$34.99
Camera	Object detection	$9.99
Camera ribbon cables	Connecting the camera	$9.99
Standoffs	Mounting hardware	$12.99
Maroon PLA filament	3D printing the body	$18.99
Gold PLA filament	3D printing the body	$21.99
Micro USB cables	Connect LIDAR to the Pi	$7.63
USB C cables	Connect battery pack to the Pi	$12.99
2-strand wire	For connecting motors	$19.89

B. The Current Design

Once the parts arrived, we started building the robot. The current design of the robot turned out differently than the intended design. The omni-directional wheels came with a set of acrylic plates and a few standoffs to build the base of a smart robot car kit. We decided to start building with this because the design of the shell and support structures were not completed yet.

Leads were soldered to each of the motors, then the motors were attached in between these base plates. The Raspberry Pi and motor controller were put together and attached on the top plate, along with the battery pack. A secondary stand was attached above the battery with more standoffs to hold the LIDAR above everything else. The motor leads were connected to the motor controller, and the LIDAR and battery pack were plugged into the Raspberry Pi. Finally, the camera was plugged into the Raspberry Pi, and mounted at the front of the robot. An image of the current robot can be seen in Fig. 2. As this is also an outreach project for the Minnesota Robotics Institute, further work will be done to complete the design of the robot, including encasing it in a shell and giving it a way to pick things up. These will be discussed in a later part of the paper.

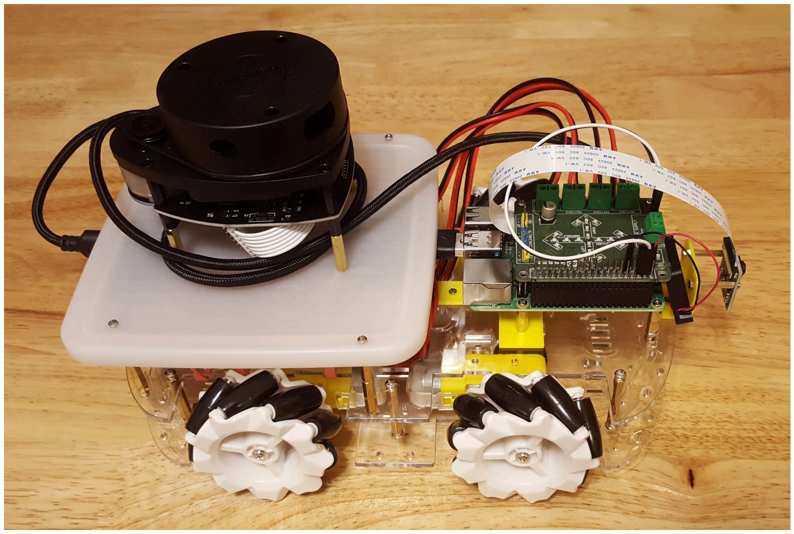

Figure 2: The current design of the prototyped education robot.

V. The Virtual Simulations

In order to properly assess the performance of our anticipated robot design, robot simulations using Gazebo were employed. The initial prototype of the completed robot was created using Autodesk Fusion 360. This mesh served as the basis for further intricate designs and capabilities once the baseline functionality of the robot is achieved. This design would ultimately incorporate the essential sensors, wheels, and motors so that a variety of different objectives can be simulated. Given that this robot would ideally be used in middle school settings, we hope to ensure that the robot is versatile in terms of what functionalities will currently be implemented as well as what capabilities can be added.

Currently, the base functionalities of movement and sensing are the primary focus. Once that is ensured in the actual robot, the projected next phase would consist of collaborating with educators in the Minneapolis/St. Paul area in order to gain a better sense of what the learning objectives are so that the functionalities added on to this robot align well to those goals. The current work done is very preliminary (establishing an optimal test environment and incorporating sensors/motors to model simulations/activities).

Using Gazebo, basic simulations were conducted to assess the different functionalities of the robot as it currently stands. With the embedded motors, camera, and LIDAR sensor, activities include simple navigation through maze-like environments [*refer to Figure 3*] or utilizing the teleop and camera to visualize what visual feedback the robot is currently receiving [*refer to Figure 4*]. The general purpose of these simulations were to formulate some activities that are visually intuitive for first-time users in robotics. Further, the primary focus for these simulations was to determine what programming skills would be necessary in order to be successful on the basic tasks, like navigation and obstacle avoidance, under various scenarios (ie. is a maze like environment vs in a arena style environment). Given that the ultimate goal includes a tangible robot, more exploration of the robot's interface was done using code that was directly added to the Raspberry Pi.

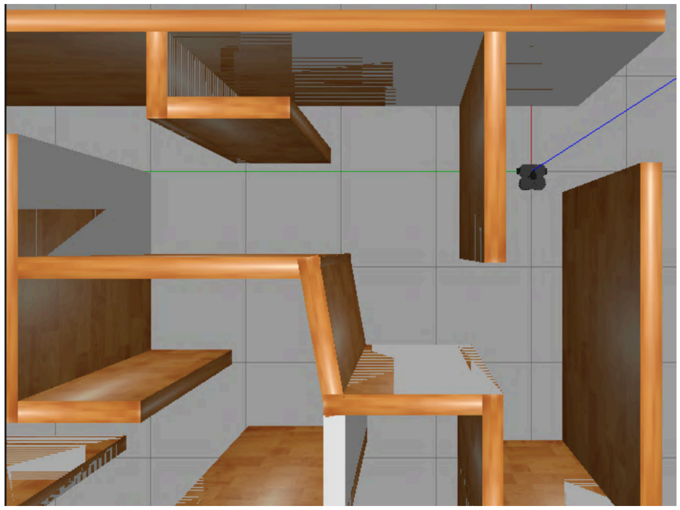

Figure 3: Gazebo simulation of the robot roaming through a maze-like environment

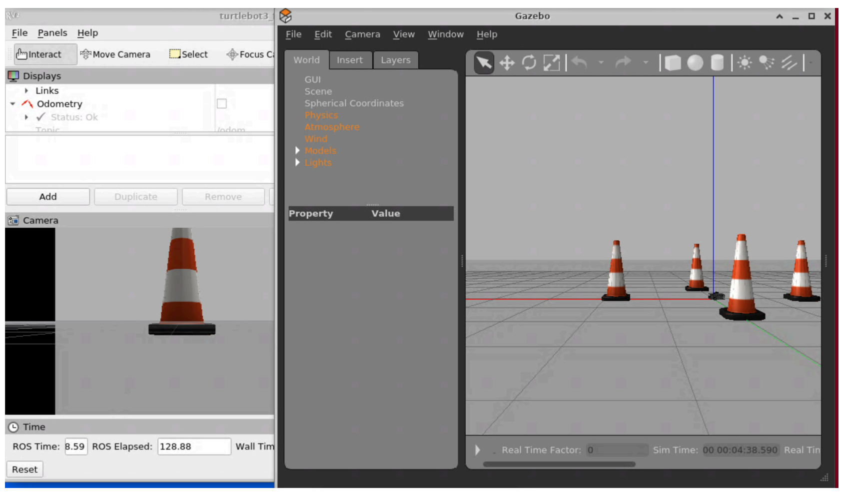

Figure 4: Utilizing the robot's color camera during navigation through basic environment

VI. The Code

The programming for the robot was written in Python, since that is the primary language that is used with Raspberry Pi's. The programs are run using the Robot Operating System (ROS), which is installed on the Raspberry Pi. The goal of the initial code is to ensure that the robot is operating as expected. This initial code is a teleoperation mode that is used to control the robot using a keyboard. Code controlling the motors was written and tested first, then a camera preview and LIDAR test were added. The code that controls the motors was based on example code and uses the PiMotor Motor Shield library, both of which were developed by SB Components, the maker of the motor controller shield being used [20]. This is because the motors are controlled through the motor controller using PWM and GPIO pins on the Raspberry Pi. This means that we cannot just publish commands to /cmd_vel using ROS to move the motors, like we can in the simulations. The library sets up the GPIO pins, and defines the functions forward(self, speed), reverse(self, speed), and stop(self) for a few different classes of motors. Using this library, we can directly tell each of the motors to go forwards, backwards, or stop. By subscribing to /cmd_vel and using this library, we can use the Twist message in a callback function to determine how to control the motors.

Next, we added the camera preview to the teleoperation code. Using the PiCamera library, we can enable a display of the camera feed at the start of the program, and disable it once the program is closed. The camera could also be used to take pictures, and use computer vision algorithms to determine if obstacles are in its path, or if the object it's looking for is in the picture. However, this would be future work done for the project. Finally, we added the LIDAR to the program as well. It is being used to avoid obstacles in its path. If the LIDAR detects that there is an object getting too close to the robot as it is moving, the robot will then be able to move in the opposite direction, likely with some rotation, to safely avoid colliding with that object.

As of right now, the code that is integrated into the robot is still in a rather elementary phase. The robot is able to utilize the teleop and camera to navigate through an environment, but the existing scripts for Gazebo simulations would eventually be translated so that they are compatible with the robot's interface. Ideally, a number of different programs would be developed such that the curriculum of the robot aligns with at least some of the key elements of robotics (ie. kinematics, debugging, coding). However, we also want to ensure that the curriculum has an aspect of fun and creativity. Therefore, this portion of the project has been a source of experimentation with the code and making sure that the different functions of the robot work correctly, with the exception of the odometer, which will be discussed later in the paper.

A. Survey Results

During the first week of November, we sent out surveys to the CSCI 5551 class as well as to school districts throughout the Twin Cities area. As recommended by Professor Papanikolopolous, we sought out other people's feedback to gain a better sense of the market that we are attempting to address. From our own experiences, we've seen courses offer exposure to robotics kits such as the Lego Mindstorms. However, we acknowledge that these kits may not necessarily encompass the necessary foundations to be successful in future advanced robotics courses. As a result, survey questions were created through Google Forms so that a greater understanding of the efficacy of current programs is achieved.

For the CSCI 5551 class, questions revolved around previous experience in robotics classes and recommendations to improve future curriculums. For the school districts, the survey questions centered around queries of how robotics curriculums are currently arranged as well as general examination of how educators anticipate this improving or being an inhibitor to student engagement and learning. Currently, we have only heard back from a few educators, but the feedback was insightful in that the lack of motivation for the implementation of a robotics curriculum currently being grounded in the lack of "time, money, upstart, and ongoing support", which explains why robotics kits aren't as prevalent despite their availability in the market. As for the feedback from the CSCI 5551 class, our survey results indicate that group engagement is a fundamental component to ensuring optimizing learning when working with robotic systems. In addition, some basic prerequisites that appear necessary in conjunction with a robotics curriculum include statistics, algebra, and geometry. These insights are helpful when trying to incorporate this course into the curriculum in middle school during future iterations of this project.

VII. Discussion

A. The Physical Robot

Despite being different from the initial design, the current design of the robot seems to be working well. It has a nice stable base, with room to expand upwards if we decide to add more components to the robot. However, since the base was designed more for a specific kit, the smart robot car, we found that there were a few issues with trying to determine where to put everything, and have a stable place to mount it. Where the Raspberry Pi was placed on the front of the robot, only the front two standoffs were initially connected, because there were two large holes where the back two standoffs were supposed to mount. To accommodate for this, we 3D printed a thin, but stable, T-shaped plate to mount to the base of the robot and would cover those holes. We added a couple of small screw holes to either end of the plate to still be able to securely mount the Raspberry Pi.

In addition, the placement of the camera needed to be at the very front of the robot, but there weren't any holes that were wide enough to be able to mount the camera securely. Therefore, we designed and 3D printed an L-shaped bracket with holes on either side to mount to the base, then to the camera using small standoffs. These brackets allow for us to securely mount some of the more crucial parts of the robot, without having to worry about them potentially falling off. Because of these couples of issues that we ran into with hole placement on the base, it would likely be beneficial for us to determine the final placement of each of the components, and design and print our own base. This would allow us to place the holes exactly where we need them, and would give us the possibility to expand on the design.

B. The Virtual Simulations

With the current robot design and functionalities decided upon, further development was to be done to design the optimal testing environment as well as implement the behaviors that the robot will possess. The primary goal was to integrate basic coding practices into the potential activities that will be embedded into the robotics curriculum. For instance, students should be able to have a firm grasp on computational thinking concepts such as breaking a large problem into manageable parts/modularity, conditionals, loops, and defining the key elements or variables for some given problem. Using the turtlebot waffle pi to serve as a stand in for the basic concept design of the robot, different scenarios were produced to further assess which environments will best convey essential robotics concepts at a very high level.

The example simulation depicted in Figure 3 was intended to determine the level of competency in the key areas of computational thinking with the ultimate goal of maze traversal. The maze activity would most likely be a culmination activity after students are comfortable using loops and conditionals to determine when a robot should be in a specific state, navigating through an environment, reorienting itself, or avoiding some obstacle. Other environment types such as an arena might also be incorporated so different obstacle behaviors (i.e utilizing some form of a bug algorithm) could be potentially explored. Further, these types of problems require a fair amount of planning so that the large task of maze navigation is more manageable. As a result, this curriculum would probably be a guided learning experience where each step of the planning and execution are clearly explained in the build instruction guide (with heavy emphasis on teamwork) that will also be integrated into this curriculum.

As for the simulation depicted in Figure 4, the intended purpose was to determine the functionality of the topics that are embedded in the turtlebot. Robots are inherently not perfect and certain components could be prone to bugs or hardware decline. In order to simplify the process of its usage in the classroom, educators should have access to methods to fix issues so that the robot itself is more accessible, without the need for a background in programming, engineering, or computer science. Therefore, it is important to undergo some basic tests of the essential sensors and motors the robot possesses prior to its employment in the classroom to ensure that the activity objectives are not muddled with problems that are not relevant to the student just yet. This simple demonstration was intended to depict how common problems in hardware can be anticipated and addressed in the user manual. The manual would be developed once the robot's basic components have been successfully assembled through basic scripts. The scripts would test whether the sensors/motors are attached correctly and the user is able to communicate basic commands to the robot prior to the robot's usage with middle school students. This appeared to be a problem that the Lego Mindstorms kit does not address, but has been noted as a potential barrier for instructors [15].

C. The Code

In addition to the teleoperation program, we wrote an autonomous navigation program that would move the robot to a specified point while avoiding obstacles. This program worked well in the Gazebo simulations, however, it did not work correctly when implemented with the robot. We discovered that this is due to the odometry, and that /odom is not receiving any messages. It is likely due to the motors and motor controller; similarly to how we cannot control the motors using a command published to /cmd_vel, the odometry of the system cannot detect the speed of the wheels. However, we might be able to determine the odometry of the robot using the robot localization package developed by Charles River Analytics, Inc. [21]. This package would use a number of different sensors and messages to estimate the state that the robot is in. Using the new package, we would be able to determine the odometry of the robot, and therefore get the autonomous mode working better than it currently is.

VIII. Future Directions

Ultimately, a functioning robot was created from scratch equipped with a LIDAR sensor, a camera, and Raspberry Pi single-board computer. This firm foundation will allow for more complex adaptations of our existing model as a more optimized (and affordable) version of a robot is sought out. However, with the successful integration of the LIDAR and camera into the robot, work still remains to find creative ways to incorporate these features into coding activities so that the user's understanding of the purposes of these sensors are more clear. Potential routes that could be taken include physical refinements such as the addition of a gripper arm for acquisition of objects in the environment and the incorporation of an exterior shell encompassing the hardware of the robot. This would especially engage students in that the functionality of the robot would expand beyond basic navigation tasks and different activities related to the gripper mechanic. In addition, the design of an outer shell will allow for a more appealing design of the robot to be adopted, concealing the complex hardware within the robot.

In addition, the primary coding language used for the sake of this preliminary design was Python. However, for those who are not familiar with programming, [4] has indicated that block based programming (preferably with some iPad-based interface) would be the most suitable interface to introduce core functionalities of code (ie. sequencing and conditionals) when attempting to program certain behaviors into a robot. Furthermore, to elevate the user friendly nature of this model kit, further work would be done to finalize an instruction manual equipped with intuitive diagrams and activity layouts to introduce basic coding concepts and critical thinking skills inspired by [17]. The layout of activities would range in different levels of difficulty so that growth and learning in each objective can be effectively gauged.

Finally, given that this is the initial stage of an outreach project with the Minnesota Robotics Institute with high hopes of potential deployment for schools in the Twin Cities area, more efforts will be made to collaborate with educators and school administrators to potentially add this curriculum to classrooms to demo how the robotics kit fares under real classroom conditions with instructors with little/some programming and robotics experience. A more ambitious goal would also be to potentially extend the functionalities of the robot to also address high school students as the programming language transitions towards Python and the assembly guides are more challenging.

IX. Team Management / COVID Plan

With the advent of COVID-19 over the course of this project, to keep our team safe, we conducted all of our work online and through simulations in Gazebo. The research, coding, and robot work could all be done remotely. At least one Zoom meeting per week was planned, where we discussed how the project was going and what needed to happen next. For the prototyping of the robot, we had one person design, order parts, and actually build the robot, while the other worked on the simulations and literary review. This way, we did not have to worry about sanitizing the parts and determining how to safely transfer the robot to the other member of the group.

X. Conclusion

Throughout the whole process of the robot kit design, different functionalities and visuals that the robot might adopt were explored before ultimately constructing the robot depicted in Figure 2. The sole basis of this exploration of robotics kits and the design of a robot from scratch was to inspire future iterations of robot kits that could offer engaging activities and functionalities at an affordable price. Construction decisions utilized visualization/simulation softwares like AutoDesk Fusion 360 and Gazebo to determine the specific functionalities that the robot will initially possess as well as the core computational concepts that could be programmed into the preliminary prototype. With a functioning product with basic capabilities such as obstacle detection via LIDAR sensors and camera, the foundation for advancement in functionality and affordability for widespread usage in the classroom can be developed as this project is pursued further.

This project as a whole was a great learning experience for the both of us. A few things that we learned through the duration of the project include how to use ROS in Gazebo simulations and in an actual robot, and how to use a Raspberry Pi, Pi camera, and LIDAR. It was very informative and helpful to work hands-on with a robot, and visually see a simulated robot work in Gazebo. We also realized that the first proposed model of a product isn't always going to be the final design, as can be seen with our various iterations of potential robot designs. It is unfortunate that we were unable to complete everything that we had hoped to, especially after we lost a member of our group, but overall, we are excited that we still were able to design and create a functioning robot.

A. *A side note on including Margaret Burkart in our presentation and paper*

We had initially started our group with Maggie Burkart included. At the beginning of the semester, she helped by chiming in during our discussions about what we wanted to do for the project, working on the first paper, and helping send out the surveys to schools in the Twin Cities area. The last time that we know she helped with something on the project was November 4th. She attended the following meeting on November 11, but then we lost contact with her after that. Seeing that, from what we know, she hadn't contributed a lot to the project, and not really being able to get in contact with her after November 11, we have decided to remove her name from the presentation and this paper. We think that it is unfortunate that this happened, but we really hope that she is okay.

References

[1] C. Angeli and N. Valanides, "Developing young children's computational thinking with educational robotics: An interaction effect between gender and scaffolding strategy," *Computers in Human Behavior*, vol. 105, p. 105954, 2020.

[2] C. Reardon, H. Zhang, R. Wright, and L. E. Parker, "Robots Can Teach Students With Intellectual Disabilities: Educational Benefits of Using Robotic and Augmented Reality Applications," *IEEE Robotics & Automation Magazine*, vol. 26, no. 2, pp. 79–93, 2019.

[3] E. Hamner, L. Zito, J. L. Cross, M. Tasota, P. Dille, S. Fulton, M. Johnson, I. Nourbakhsh, and J. Schapiro, "Development and results from user testing of a novel robotics kit supporting systems engineering for elementary-aged students," *2017 IEEE Frontiers in Education Conference (FIE)*, 2017.

[4] I. M. L. Souza, W. L. Andrade, L. M. R. Sampaio, and A. L. S. O. Araujo, "A Systematic Review on the use of LEGO® Robotics in Education," *2018 IEEE Frontiers in Education Conference (FIE)*, 2018.

[5] I. Nourbakhsh, E. Hamner, K. Crowley, and K. Wilkinson, "Formal measures of learning in a secondary school mobile robotics course," *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004*, 2004.

[6] M. Walter, M. Anderson, I. Burt, and N. Papanikolopoulos, "The Design and Evolution of the eROSI Robot," *Proceedings 2007 IEEE International Conference on Robotics and Automation*, 2007.

[7] "Makeblock DIY Ultimate Robot Kit - Premium Quality - 10-in-1 Robot - STEM Education - Arduino - Scratch 2.0 - Programmable Robot Kit for Kids to Learn Coding, Robotics and Electronics", *Amazon.com*, 2020. [Online]. Available: https://www.amazon.com/Makeblock-DIY-Ultimate-Robot-Kit/dp/B00W6Y0Z4E. [Accessed: 05- Oct- 2020].

[8] "MINDSTORMS® | Themes | Official LEGO® Shop US", *Lego.com*, 2020. [Online]. Available: https://www.lego.com/en-us/themes/mindstorms. [Accessed: 05- Oct- 2020].

[9] Q. Hu, J. Qi, F. Pan, and N. Wu, "Online Learning Algorithm and Framework for Educational Robots," *2019 Chinese Automation Congress (CAC)*, 2019.

[10] R. Rosenberg-Kima, Y. Koren, M. Yachini, and G. Gordon, "Human-Robot-Collaboration (HRC): Social Robots as Teaching Assistants for Training Activities in Small Groups," *2019 14th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 2019.

[11] S. Atmatzidou and S. Demetriadis, “Advancing students’ computational thinking skills through educational robotics: A study on age and gender relevant differences,” *Robotics and Autonomous Systems*, vol. 75, pp. 661–670, 2016.

[12] S. Evripidou, K. Georgiou, L. Doitsidis, A. A. Amanatiadis, Z. Zinonos, and S. A. Chatzichristofis, “Educational Robotics: Platforms, Competitions and Expected Learning Outcomes,” *IEEE Access*, pp. 1–1, 2020.

[13] T. C. Major, B. B. Rhoades, and J. M. Conrad, “A novel and affordable robot kit for completion of the requirements for the Boy Scouts of America robotics merit badge,” *SoutheastCon 2015*, 2015.

[14] "Thames & Kosmos Robotics Workshop Model Building & Science Experiment Kit | Build & Program 10 Robots with Ultrasonic Sensors | Program & Control with App for iOS & Android", *Amazon.com*, 2020. [Online]. Available: https://www.amazon.com/Thames-Kosmos-Robotics-Workshop-Kit/dp/B01CNSPGU0/. [Accessed: 05- Oct- 2020].

[15] T. M. Iwano, J. S. D. S. Vieira, D. M. D. Oliveira, and D. Scherer, “A Teacher Experience Report with LEGO Robotics,” *2019 IEEE 19th International Conference on Advanced Learning Technologies (ICALT)*, 2019.

[16] W.-C. V. Wu, R.-J. Wang, and Y.-A. E. Jou, “Application of Educational Robots in the Elderly English Vocabulary Learning,” *2020 IEEE 20th International Conference on Advanced Learning Technologies (ICALT)*, 2020.

[17] Y. Diyas, D. Brakk, Y. Aimambetov, and A. Sandygulova, “Evaluating peer versus teacher robot within educational scenario of programming learning,” *2016 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 2016.

[18] T. Phetsrikran, W. Massagram, T. Phoka and A. Harfield, "A Feasibility Study of Arducation Bot : An Educational Robotics and Mobile Application Kit for Computational Thinking Skills," *2018 22nd International Computer Science and Engineering Conference (ICSEC)*, Chiang Mai, Thailand, 2018, pp. 1-6, doi: 10.1109/ICSEC.2018.8712671.

[19] J. F. Kelly, LEGO MINDSTORMS NXT 2.0. Apress, 2009 [Online]. Available: http://dx.doi.org/10.1007/978-1-4302-2492-1

[20] SB Components, “sbcshop/MotorShield,” *GitHub*. [Online]. Available: https://github.com/sbcshop/MotorShield

[21] Charles River Analytics, Inc, “cra-ros-pkg/robot_localization,” *GitHub*. [Online]. Available: https://github.com/cra-ros-pkg/robot_localization