Miniature Mars Rover

PROJECT PLAN

Team Number: sdmay19-09

Client: Dr. Zambreno
Advisers: Craig Rupp

Team Members/Roles:

Calvin McBride:	Hardware/Software Developer, Communication, Status Reporter, Meeting Note Taker
Sam Westerlund:	Software Developer, Communication
Mitchell Freshour:	Hardware Lead
Douglas Kihlken:	Hardware Developer

Team Email: sdmay19-09@iastate.edu
Team Website: sdmay19-09.sd.ece.iastate.edu

Revised: 2018-10-26 Version 2

Table of Contents

List of Figures

List of Definitions

JPL: Jet Propulsion Labs, this is the organization that created the open source rover that we are basing our design off of.

Iowa State/ISU: Iowa State University

Websockets: A communication protocol between computers for sending data. Stream of data is constantly kept open.

LIDAR: Light Detection and Ranging, this is a remote sensing method that uses a pulsed laser to measure ranges to an object.

Raspi: Raspberry Pi electronic board

ABET: Accreditation Board for Engineering and Technology

IEEE: Institute of Electrical and Electronics Engineers

1 Introductory Material

1.1 Acknowledgement

Craig Rupp is our technical advisor. He assisted us with guidance and advise on what features would be interesting to develop for the rover. We discussed various possibilities for the advancements to be made to the rover. Craig also assisted in verifying the potential implementation ideas for our expansions to the rover.

1.2 Problem Statement

Iowa State is in need of a demonstration piece for groups and interested students. Dr. Zambreno would like to have a Mars Rover demo unit that can show off the abilities of the students at Iowa State. A mars rover demonstrates a knowledge of robotics, hardware, software, and engineering. This provides an a excellent overview of what is possible at Iowa state and will surely impress any guests.

The secondary goal of the rover is to provide a platform on which to develop further technologies. Future groups should be able to modify and improve the rover design. This means that documentation, standardization, and accessibility should all be top priorities for the rover. Ease of use is important as well since the demo unit will need to be operated quickly and with as little interaction as possible.

1.3 Operating Environment

The operating environment for the rover will likely be inside buildings on the Iowa State University campus. Outdoors operation will likely be possible but not recommended for frequent outside use. Hallways and rooms should have smooth floors and large areas for easy navigation.

1.4 Intended Users and Intended Uses

The intended users for the rover are Iowa State University students and visiting interest groups and potential students. ISU students will likely use the rover for exploring robotics, learning by extending the platform, and fun. Visiting groups will use the robot to stir interest in the ISU engineering program.

1.5 Assumptions and Limitations

Assumptions:

- Rover will be operated by a single user at any given time.
- The rover will be used mostly inside, except when moved between buildings.

- The rover will be extendable, so that future teams may build upon it.
- The rover will be able to stream video to a server.

Limitations:

- The rover will have a battery life of approximately 3-5 hours.
- The cost will not exceed the project's $2500 budget.
- Constructing the Mars Rover from JPL is expected to take 200 hours
- Constructing the rover requires soldering, milling, laser cutting, and 3d printing of parts.

1.6 Expected End Product and Other Deliverables

- Base Configuration - December 2019
    - At this point the Mars Rover will be partially constructed to the farthest point possible given time constraints. The software for the rover will be completed for manual control and ready for integration. A functional web page will be displayed with image streaming and the ability to place markers as a destination point for the rover. At this point the rover as a entire system may or may not be able to be controlled manually via the website.
- Prototype - March 2019
    - The prototype will include the functional webpage to at least manually control the rover, its machine learning components may or may not be at a demonstrable position for navigating. The webpage should be able to connect to the backend server on the rover, and send commands to it as well. The rover will have been tested with its machine learning components/computer vision and the viability of completing such a task.
- Final Product - May 2019
    - The finished rover, along with its source code, will be handed to the client at the end of the spring 2019 semester.
    - Finalized documentation will be provided on how to use the mars rover and how its was coded, and include any other additional information needed for future students to continue the project.

2 Proposed Approach and Statement of Work

2.1 Objective of the Task

The goal of the Mars Rover is to try and impress potentially enrolling high school students and other public members with our project. The end product, a functional Mars Rover, will be able to perform in front of a group of people with the end user being a non-programmer / non-tech-savvy person. The Mars Rover will then be able to navigate a path or go to a destination selected by said person. The end user will also interact with the rover with a website to send manual commands and view its current labels for its image classification. The hardware aspect will be a functional rover based off of the NASA JPL open-source Mars Rover project.

2.2 Functional Requirements

The end user must be able to view the current image that the Mars Rover is processing and its corresponding image labels/classifications. This information should be accessible within the browser and be in real time with less than 3 seconds of latency.

The end user must be able to manually control the Mars Rover. To manually control the rover the user will connect to the rover in their browser at the same page of the camera input. There will be simple buttons to click (forwards, backwards, left, right) that will move the rover in that direction with a lag of less than 1 second.

The Mars Rover will be able to move across various terrain ranging from gravel, road, dirt, and small rocks. It will also be able to handle inclines at 40 degrees, and uneven terrain where the disparity from the left to right wheels is less than 5 inches.

The Mars Rover must be autonomously capable of going from its current position to a destination selected by the end user. It must reach the destination in a reasonable amount of time compared to manually driving to the destination.

The end user will select a destination position on the same website as the other controls. The position will be displayed to the user in the form of a map and they can simply click a waypoint.

The rover shall automatically stop itself if it does not have any active connections over the websocket. This will prevent potential issues of the rover running away, and not being in range to manually control it.

The rover will map its surroundings using LIDAR and generate a layout of its surroundings using this system. This will allow users to view a map of the area during and after operation.

2.3 Constraints Considerations

The website displayed for the end user will have a React application as a frontend, and it will communicate via websocket to the python backend on the rover. The website will be accessible in the browser via http at port 80. Information sent over the websocket will include position commands, current camera image, list of objects in camera image (classification labels), data received by the LIDAR system, and other commands needed to interface with the rover. The React application and Python backend will both run on the rover on the Raspi with an Intel Compute Stick.

The Mars Rover will use all of the parts in the NASA JPL project and build off of it using its existing architecture.

Because of the complexity of an autonomously driving machine, there will be leniency on the accuracy of the rover and the time it takes to complete its path. The neural network may also not be very complex because of the limitations of the Raspi and the Intel Compute Stick that we will be using.

The project overall will be an ethical project, with it's goal to encourage students to pursue a degree in a STEM field. It will also provide a service to the community by being a platform for more expandability. It follows the IEEE code of ethics, by providing a service and being an honest project that includes everyone. It also follows the code by being open to criticism, giving credit to JPL for its efforts thus far, and allowing for improvements by other future developers. Like IEEE it also follows ABET criteria. Fulfilling criteria 4 for continuous improvement because we plan to hand off the project to other students. And many of the other criteria because of the documentation generated during this project and the involvement from the university.

2.4 Previous Work And Literature

We will be building our Mar's Rover based off of the NASA JPL's project[1]. They have schematics and instructions on how to build a fully functional rover that can be manually controlled. We will not be competing with anyone in building this project, only expanding the existing one to do more and accomplish our clients goal of getting future students attention.

2.5 Proposed Design

The design we implement will have software side the web server and image classifier/rover controller on the Raspi with the option to receive external manual commands by a user over websocket. The hardware component will be almost entirely the same as the JPL with a few modifications and addition of hardware. We may had additional hardware later or a future team might so we will want to keep modifiability in mind.

Alternative design ideas included not using a neural network for navigation. In this scenario we would program the rover with many lines of carefully planned out code and algorithms to help it navigate. The problem with this scenario is that it would take many hours to code, design, and test. Alternatively we could just simply create a neural network that would accommodate for many scenarios that we did not envision and also provide a quick buildup and overall a more robust navigation system. Not only would the neural network perform better, but it would also be much more interesting to prospective college students instead of a lot of manually created decision trees.

A separate alternative design we had was for the hardware. We discussed using an already pre-built robot that would simply take commands from our software to move. This would have provided a much faster buildup time and draw in focus to the software. However, we decided to go with building up our own hardware as the client desired it and future teams would have more options to build upon or modify the self-constructed rover.

2.6 Technology Considerations

The Raspi is what we will be using to control the rover because it is cheap and also what the JPL used. To add computer vision and neural network capability we will be using an Intel Compute Stick. This combination will not be able to provide the best neural network and may suffer from accuracy loss. The technology may also take a lot of time to assemble but will be very stable once completed.

2.7 Safety Considerations

There are safety concerns with running an autonomous vehicle. It may run into someone or something, or block a path. These concerns should be taken into consideration when running the rover. The rover should ideally be run in an open field or space, much like the environment it would encounter on Mars. There should be a person on the manual controls ready to stop or control the rover when necessary. Electrical components inside the rover will also need to be carefully tested before implementation to ensure that during operation there is not an electrical or mechanical malfunction leading to possible damage to the rover or the components themselves.

2.8 Task Approach

To tackle this problem we will be working on the software and hardware side separately. They will be somewhat unified for testing purposes, but for the most part they will not depend on each others progress. The manufacturing and construction of the rover's parts will need to be a combined effort due to the amount of man hours necessary as given by the JPL documentation for the rover.

2.9 Possible Risks And Risk Management

Creating a self-driving rover is difficult. We may not have the knowledge to complete the task let alone accurately. However, we are confident we will at least be able to create a manually driving rover with computer vision data. We are also keeping in mind that this project will most likely be built upon by other students, so we are giving them the option to complete what we may not have finished. The cost and materials are not of concern. All materials needed for the project will be purchased when necessary. The hardware construction is based upon a solid list of parts and well planned, there is very little chance of us needing to go over the budget for the base construction. Time and opportunity cost of ordering and machining parts will need to be taken into consideration.. There will likely be an issue with the time that it takes to assemble the rover. In the JPL documentation for the rover it said that it would take (200 hrs) longer than our current available time to just assemble their rover, however we think we are capable of finishing it in less time. Many of the rover's body panels and various mounts need to be machined using milling, laser cutting, or 3d printing. These are tasks that nobody in our group has prior experience with, and therefore we will need to learn how to accomplish these tasks ourselves.

2.10 Project Proposed Milestones and Evaluation Criteria

There will be several independent milestones for the project, based on the hardware and software side. The first key milestone hardware wise will be completing of electrical testing to ensure that our rover will be able to operate within safe limits when inside the rover. The next key milestone will be the implementation of the hardware components into the rover. For the software it will be having a working image classifier with the data being streamed live from the Raspi to a clients browser. After this the next big milestone will the the successful integration of manual rover navigation via the website, then autonomous navigation.

To test that these milestones work we will be performing user test cases, where the client will use the software we created to control the rover. We will also be doing internal code reviews to make sure everything is functional. Hardware testing will be completed in the Coover labs using the various testing equipment provided by the university. This will include testing of PCBs and electrically driven motors.

2.11 Project Tracking Procedures

We will be using Trello to track the progress of the project. Each task on Trello will be assigned to a member or members. We will also be communicating with each other about the details of the project using Slack so we know that everyone is up to date.

2.12 Expected Results and Validation

The desired outcome will be a user controllable mars rover. We will know if the solution is correctly implemented if it is able to navigate autonomously or at least manually with auxiliary data from an image classifier. Validation data for the neural network will be provided to make sure that the neural network part of the project is accurate. We require the end product to fulfill all of the client's requirements mentioned above. To test these requirements the rover will be put through multiple use case scenarios and will be expected to complete all the requirements in those.

2.13 Test Plan

We will be doing code reviews and internally testing each function to provide test coverage. Our plan is to debug our own code separately to the extent we believe it will run correct and then have code reviews once we think it is done. This will ensure that the code reviews are concise and done in a efficient manner. System tests will be performed throughout the duration of the project to ensure a working and demoable product. As the project progresses we will have a harder time testing it because machine learning in our case is essentially a black box and each individual component of its decision making cannot be tested. However, for all hard coded elements of the project we will implement unit tests or at least perform rigorous system testing.

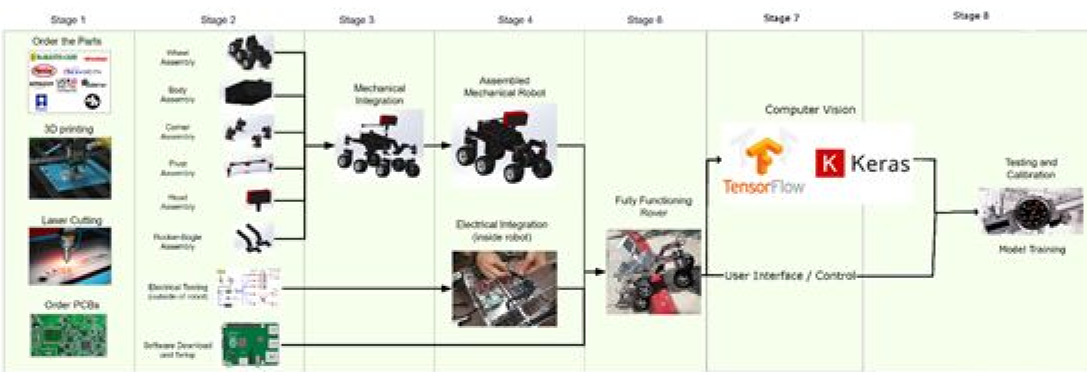
3 Project Timeline, Estimated Resources, and Challenges

3.1 Project Timeline

a. First, we identify design specifications and modify existing ones
b. Present our plan to the client. This plan should be open to changes and improvements
c. Modify plan once more, while continually building out architecture to better understand what is possible for a solution and better estimate what is feasible
d. Meet with client to discuss further improvements and goals and the final plan

e. Test electrical and mechanical parts to insure they are within safe parameters once implemented, and do not malfunction or cause excessive degradation to the components.
f. Build our prototype:

Figure 1.



g. Show off prototype to the client. Get his input and what he wants changed
h. Modify prototype to make sure it meets client's needs
i. Show off prototype once more. This should include very harsh input, this is the last chance the client has to change something
j. Make final changes. Make sure project is well documented for passing it on.
k. Demonstrate final product to the client.

Gantt Chart:

3.2 Feasibility Assessment

We project that we will have the JPL Open-Source Rover design at stage 3 (mechanical integration) by the end of the semester, with stage 4 (electrical integration) almost completed and preferably done early spring 2019. The basic software for the rover that is provided in the GitHub repository, as well as our personal code for camera operation and streaming, should also be complete by the end of the semester.

The biggest challenge for the project will likely be time. The JPL Open-Source Rover documentation predicts that the project takes a minimum of 200 man-hours for an experienced individual to complete. Since our team is more experienced with software rather than hardware, we expect that mechanical and electrical integration to take longer than expected, and therefore the 200 man-hour estimate is likely a very optimistic deadline for us. We don't expect the rover to be completed until next spring, excluding any custom extensions we make to the robot platform.

3.3 Personnel Effort Requirements

Include a detailed estimate in the form of a table accompanied by a textual reference and explanation. This estimate shall be done on a task-by-task basis and should be based on the projected effort required to perform the task correctly and not just "X" hours per week for the number of weeks that the task is active

3.4 Other Resource Requirements

Parts and time are two critical components of our project. The JPL Rover documentation has pre-compiled list of parts necessary to build the rover, all of which need to be ordered, tested, and integrated into the robot. This large list of parts will likely take significant time to test and integrate, and will use most of our $2,500 budget.

Besides that, since our project heavily involves hardware and mechanical construction, we need to mill, laser cut, and 3d print parts in order to properly construct the rover. There is also soldering necessary to connect our various electrical motors, PCBs, and the Raspi together.

3.5 Financial Requirements

The financial resources required by the JPL Rover project is listed to be just under $2,500 in the documentation. Our financial requirements will likely use most of our budget, and may go over if parts are damaged during construction or finishing. Along with these we also need to purchase any additional equipment to be used for additions to the rover, which may also push us over our projected budget.

4 Closure Materials

4.1 Conclusion

The Mars Rover project was initiated with the hopes of creating a controllable robot in order to impress potential students and visitors by demonstrating the skills and potential results that Iowa State University engineering students can produce. The JPL Open-Source Rover is perfect for this task because it requires skills in several engineering domains (electrical, software/computer, and mechanical) and is an easily identifiable culture reference. Many visitors who see the rover will likely recognize the look of the iconic Mars rover.

The Mars Rover will be designed with extensibility in mind so that future students/teams may build upon it for their own projects, as well as implement computer vision and autonomous navigation. Video streaming and remote control will allow visitors to play with the rover and get a feel for the engineering quality of the robot. We expect to have most of the rover built by the end of the semester, with the final project objectives completed or almost-completed by the end of the spring semester.

4.2 References

[1] Github (2018, Sept.). NASA JPL Open-Source Rover [Online]. Available: https://github.com/nasa-jpl/open-source-rover [Accessed Oct. 13, 2018]