The Robotics Institute, Carnegie Mellon University

Concept Development Report
December 11, 2019

MRSD Project - Pit Navigator

Team G - The Pit Crew
Alex Withers
Awadhut Thube
Justin Morris

Sponsor - Dr. William Whittaker

Contents

1 Project Description **2**

2 Use Case **3**

3 System Level Requirements **5**
3.1 Mandatory Performance Requirements . . . 5
3.2 Mandatory Non-Functional Requirements . . . 5
3.3 Desirable Performance Requirements . . . 6
3.4 Desirable Non-Functional Requirements . . . 6

4 Functional Architecture **7**

5 System Level Trade Studies **8**
5.1 Rover Selection Trade Study . . . 8
5.2 Camera Motion Trade Study . . . 8
5.3 Navigation Method Trade Study . . . 9
5.4 Mapping Trade Study . . . 10

6 Cyberphysical Architecture **11**

7 Subsystem Descriptions **13**
7.1 Camera Operation . . . 13
7.2 Localization . . . 13
7.3 Risk Assessment and Planning . . . 14

8 Project Management **16**
8.1 Work Plan and Tasks . . . 16
8.2 Schedule . . . 18
8.3 System Validation Experiments . . . 20
8.3.1 Spring Validation Experiments . . . 20
8.3.2 Fall Validation Experiments . . . 21
8.4 Team Member Responsibilities . . . 22
8.5 Parts List and Budget . . . 22
8.6 Risk Management . . . 23

9 References **25**

1 Project Description

Studies of the moon's surface have led scientists to hypothesize the existence of a network of sublunarean tubes hundreds of feet across, extending for kilometers below the regolith. If these tubes exist, they could serve as the foundation of future human habitats on the moon. They would provide an enclosed space that would shield occupants from the radiation and extreme temperatures of the moon's surface, and could be sealed and pressurized with breathable air.

Although the existence and extent of these tubes remains a theory, there is direct evidence showing that the moon's surface is studded with large pits, some dozens of meters across. These pits sink deep enough into the moon that they could potentially connect to the tube network. Because of the exciting potential of the sublunarean tube structures, these pits have become a top priority for future research and exploration.

Astrobotic Technology, in cooperation with Carnegie Mellon, proposes to send a small, fast, autonomous rover to explore these moon pits and collect data about their composition and structure. This rover will have between one and two Earth weeks to travel to and collect images of a moon pit near its landing site. This process will take the robot beyond the range of wireless communication with the lander craft, which is its only connection back to Earth. Therefore, the rover must be able to operate autonomously while executing its mission, and return back to the landing site safely.

Any autonomous mission that must operate in the vicinity of a moon pit must have special routines for navigating around the pit edge in a safe and efficient manner. This project proposes to design and implement software that will identify a pre-selected pit from images collected by the rover's cameras, construct a model of the pit's dimensions and location from these images, and generate safe routes to a series of vantage points around the edge of the pit. From these vantage points, the rover will collect images of the opposite wall of the pit, aiming to photograph as much of the pit's circumference as possible over the course of its mission duration.

2 Use Case

A small autonomous rover is exploring the surface of the moon. It arrived aboard a lander, which set down near the known location of a pit. One of the rover's directives is to gather imagery of this pit and transmit that imagery to the lander, which will then forward the collected data on to Earth. The rover was deployed from the lander, and has since been executing various mission objectives, the most recent of which was to autonomously travel to a waypoint set for it by human operators on Earth. This waypoint is in close proximity to the pit, and is outside of the communication range of the lander, meaning that the rover must operate with complete autonomy during its trek to this waypoint.

The pit's location is defined by a map of the area around the landing site that was generated before the mission launched. Prior to the start of the rover's mission, the lander provided the rover with exact coordinates of the lander's location in this map. The rover localizes itself and the pit in relation to these coordinates for the duration of its mission.

Upon arrival at the waypoint, the rover software activates the Pit Navigator routine, which employs specialized behavior designed for successful navigation in the vicinity of the pit. First, the rover trains its cameras in the direction of the pit's expected location. It collects images of its surroundings, and attempts to identify the pit from those images. The rover calculates an estimate of the location and dimensions of the pit, which is incorporated into the rover's map of its surroundings.

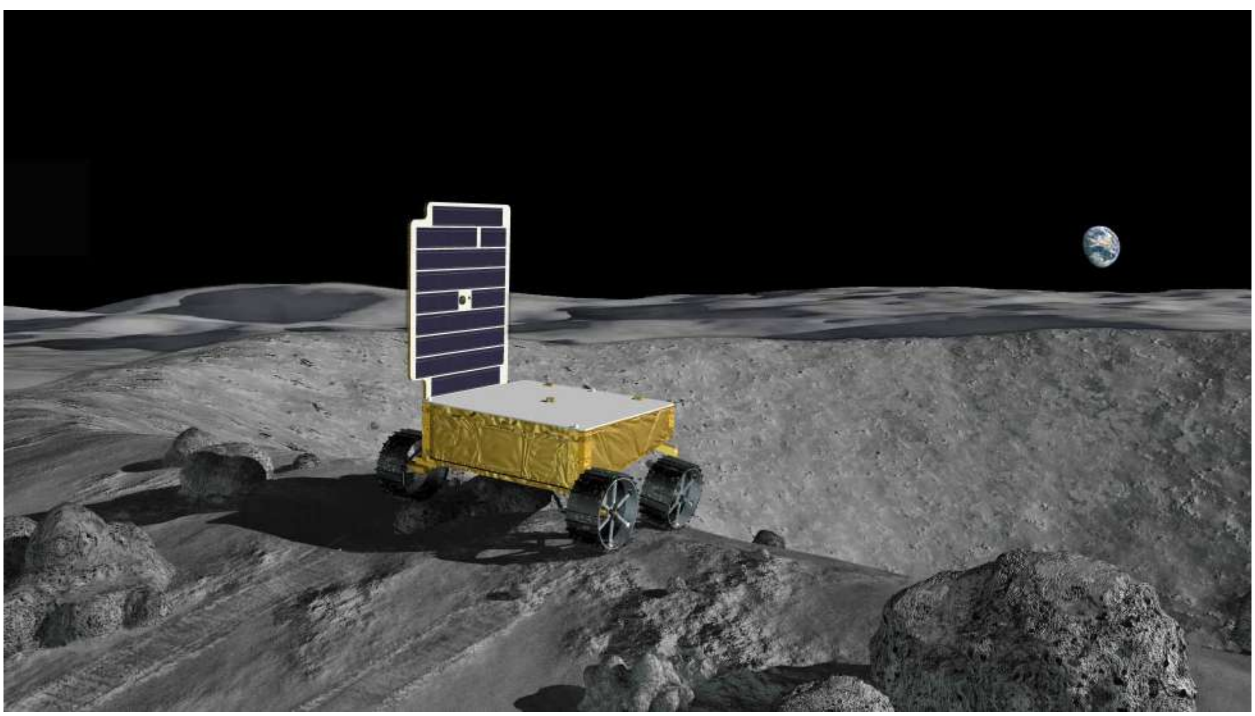


Figure 1: Rover at Pit Edge

As soon as the rover has modeled the pit in its global map, it autonomously selects a new waypoint at the edge of the pit. A local planning algorithm generates a route to this waypoint consisting of multiple small movement steps. The algorithm avoids any obstacles detected in the rover's vicinity and prevents the rover from moving too close to the pit edge, then selects the most optimal route to the destination position with these constraints applied.

The rover begins to execute the planned route to the destination waypoint. After each movement step, the rover uses its navigation cameras to collect information about the area in front of it, and uses its pit imagery camera to capture new images of the pit. These pit images are used to update and refine the rover's model of the pit. Both the pit images and the navigation images are used to update the planned route in order to account for unforeseen obstacles and dangers.

When the rover reaches its destination, it uses the pan and tilt actuators on its pit imagery camera to align the camera such that the camera's field of view is trained on the pit wall opposite the rover's location. The camera pans from side to side, taking images at a range of angles in order to capture an arc of the pit wall. Each of these images is incorporated into the rover's model of the pit. Once sufficient images have been captured, the rover uses its model of the pit to select a new waypoint at a different point on the pit's circumference, and plots a new path to reach that waypoint. This path takes the rover away from the pit edge before moving around the perimeter of the pit, to ensure that the rover will not fall into the pit while navigating to the next waypoint.

The rover will continue to select waypoints around the edge of the pit and take images at each point until the sun sets and the rover runs out of battery power, with the goal of collecting data on as much of the pit's wall area as possible. At all times during its mission, the rover is measuring how much data it has collected. In order to ensure that as much of this data as possible is transmitted safely to Earth, the rover will leave the vicinity of the pit at regular intervals in order to return to within communication range of the lander and deposit all collected data to the lander. When the rover leaves the vicinity of the pit, the Pit Navigator routine ends and control is returned to the standard moon operation subsystems. The data transmitted to the lander includes rover telemetry and navigation images in addition to the pit imagery collected at each of the waypoints. Once data is stored on the lander, the rover will return to the pit once again to collect more imagery for as long as it has sufficient power to continue operating.

3 System Level Requirements

The following requirements are derived from an objectives tree by taking in consideration the mission goals and sponsor expectations. The requirements are categorized as mandatory and desirable requirements. These categories are further divided and the requirements are classified as performance requirements which are functional requirements with an associated performance measure and non-functional requirements.

3.1 Mandatory Performance Requirements

The system will:

M.P.1 Capture **500 MB** image data on a surface similar to the moon terrain.

M.P.2 Capture **75 MB** image data over a single cycle in the mission.

M.P.3 Capture **15 MB** image data from specific co-ordinates on the surface of the moon.

M.P.4 Classify images of pits with classification accuracy $>$ **80%**.

M.P.5 Calculate the relative distance to the pit edge within **2% error**.

M.P.6 Calculate an optimal navigation plan within **20 seconds**.

M.P.7 Capture images covering **5° angle** of pit circumference from one position.

M.P.8 Detect pits from a max distance of **20 meters**.

M.P.9 Operate such that chance of occurance of a mission ending incident is less than **5:1**

3.2 Mandatory Non-Functional Requirements

The system shall:

M.N.1 Operate in the vicinity of a pit on the moon.

M.N.2 Operate using hardware that meets specifications of overall rover design and mission.

M.N.3 Operate within a Linux operating system environment

M.N.4 Be compatible with other software systems running on the rover.

M.N.5 Maintain a mission clock.

M.N.6 Operate when rover is not experiencing any major subsystem faults.

In addition to the mandatory performance and non-functional requirements, we have also identified certain desirable requirements. These additions are nice to have and extend the project scope in exchange for having a more robust, reliable and valuable system. The desirable requirements are formulated to extract the greatest amount of information even when operating under different conditions.

3.3 Desirable Performance Requirements

The system will:

D.P.1 Operate at a distance of **0.75 meters** from the pit edge 80% of the time.

D.P.2 Estimate the shape and size of the pit within **10% error**.

D.P.3 Capture high resolution images of the pit with each image being **18 MB** in size.

D.P.4 Capture data such that for **80%** of the images **60%** of the image will show the pit.

3.4 Desirable Non-Functional Requirements

The system shall:

D.N.1 Operate given pits of different sizes and shapes.

D.N.2 Take rover parameters and state into account during motion planning.

4 Functional Architecture

The architecture outlined below shows the functions that the system must execute to fulfill the previously mentioned requirements. The functions are derived assuming that the robot is already in the vicinity of the pit.

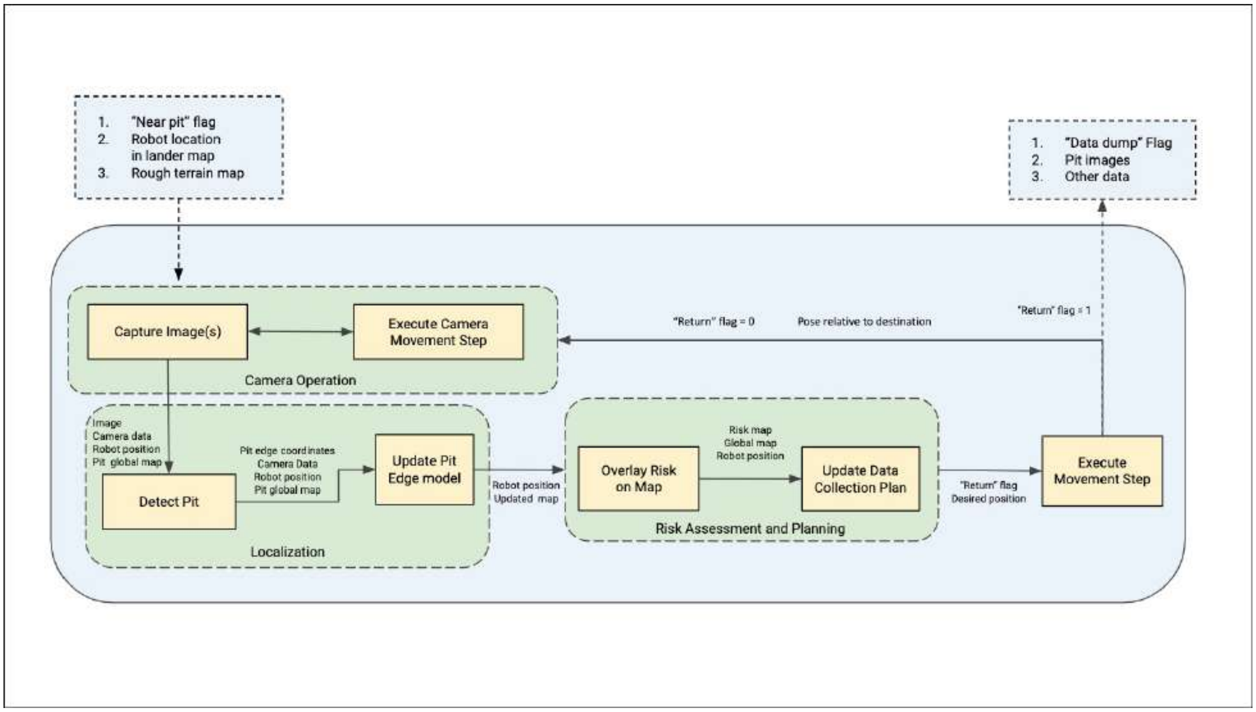


Figure 2: Functional Architecture

The pit exploration phase of the mission consists of the following essential functions:

- **Capture Images:** The robot captures images of its surroundings at regular intervals.
- **Execute Camera Movement Step:** This is where the rover will turn the camera to take pictures from different viewpoints/angles.
- **Detect Pit:** The system detects the pit in the captured images
- **Update Pit Edge Model:** After detecting the pit, the rover calculates it's position relative to the pit edge and also updates the position of the pit in the global map.
- **Overlay Risk in Map:** The rover estimates the risk of continuing the mission and losing all captured data.
- **Update Data Collection Plan:** Based on the estimated risk, the rover updates the previous data collection plan.
- **Execute Movement Step:** Once the decision is made, the rover either goes back to the starting point or continues to gather more data from different positions.
- **Interfacing Operations:** The entirety of the Pit Navigator project will constitute one software subsystem within the PitRanger system, which comprises the whole of the functionality for the moon rover. The Pit Navigator system will control the rover during the period of its mission where it is in the vicinity of the pit, during which time it will interact with other PitRanger subsystems.

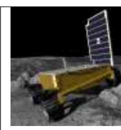
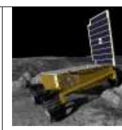
5 System Level Trade Studies

Trade studies for the Pit Navigator project are primarily divided into two categories. The first category of trade studies are for various hardware functions that are necessary for the rover to complete its pit mission. Because the approximate size and power capabilities of the rover are known, we can assess hardware components by their ability to meet these restrictions, as well as their functional suitability for the mission.

5.1 Rover Selection Trade Study

This trade study discusses the various options for using MoonRanger[1], altering MoonRanger, or building a new rover entirely. MoonRanger is the closest surrogate to the envisioned PitRanger rover. The major considerations for choice of the rover were, the types of sensors available, the amount of data that could be captured using the various sensors and how safely can the rover operate under the required conditions. This trade study most definitely concludes that some alterations would be required on the MoonRanger.

Table 1: Rover Trade Study

							
Criteria	**Weight**	**From Scratch**	**Mechanical Alterations**	**Electrical Alterations**	**Sensor Alterations**	**Any Alterations**	**Exact Copy**
Material Cost	10	0	2	3	4	3	5
Additional Software Effort	10	1	3	5	3	3	3
Additional Hardware Effort	10	0	2	5	5	2	5
Maneuverability	10	4	5	3	3	4	3
Sensors	15	4	2	2	4	4	2
Storage Capacity	5	4	3	5	3	4	3
Safety	20	3	2	2	4	4	2
Ease of Data Capture	5	4	2	2	4	4	2
Amount of Capturable Data	15	4	4	3	3	4	3
Total	**100**	**270**	**275**	**310**	**370**	**360**	**300**

5.2 Camera Motion Trade Study

Imaging the pit is an interesting challenge that can be solved in many ways. Using a skid steer rover these options compare and represent the best ways to image the pit. The alternatives for this trade study were identified through brainstorming the possible combinations for having camera motion. The panning of the camera would allow for covering a larger field of view from one particular location on the lunar surface. This means that the rover will be able to capture more information regarding the pit edge even when using a camera with a lower resolution and a lower aspect ratio. Thus, having camera motion indirectly affects the resolution requirement of the cameras. Stability of images is one of the most important considerations in this trade study along with the effect on resolution requirement. The conclusion of performing this trade study is that having dedicated motions for the cameras is best to satisfy the mission requirements.

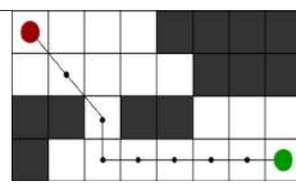
Table 2: Camera Movement Trade Study

					
Criteria	**Weight**	**Dedicated Pan + Tilt + Zoom**	**Robot Pan + Tilt + Zoom**	**Single Picture**	**Robot Pan + No Tilt**
Keeping Camera Warm Difficulty	10	2	2	5	5
Stability of Images	20	5	4	4	2
Weight	15	2	3	4	5
Power Consumption	10	3	2	4	3
Ease of Control	10	4	2	5	2
Cost	10	3	3	3	5
Effect on Resolution Requirement	20	5	5	1	3
Size	5	2	2	3	5
Total	**100**	**360**	**325**	**345**	**350**

The second category are existing algorithms that will be adapted to serve the various purposes of the software. Each of these trade studies must consider the limited computer power available on the moon rover, the need for high levels of robustness on systems that will be operating in the remote and inhospitable conditions of the moon, and the level of technical knowledge and raw effort required to implement each possible solution.

5.3 Navigation Method Trade Study

Table 3: Navigation Method Trade Study

				
Category	**Criteria**	**Weight**	**Heuristic + Field D***	**Markov Model[3]**
Theoretical	**Implementation Complexity**	20	2	3
	Adaptability (handle more situations)	15	3	5
	Availability of Prior Work	7.5	2	4
	Code Structure Availability	7.5	2	3
Performance	**Performance (accuracy to human path)**	20	TBD	TBD
	Computational Requirement	15	5	2
	Prior data requirement	5	4	4
	Reference map resolution	10	4	2
Total		**100**	**250**	**257.5**

As the rover should not fall into the pit, the act of autonomously moving around the pit can be dangerous. The options presented are compared as the first navigation strategies for the pit. Here, we perform an algorithmic trade study to identify the best choice that would satisfy the requirements of the risk assessment and planning subsystem. The requirement is to capture the risk associated with executing a certain process and to incorporate the calculated risk into future planning and navigation of the robot.

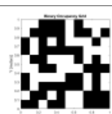
We have categorized the considerations for the algorithm selection into theoretical criteria and performance criteria. Some of the performance criteria weights and scores are based off the team's understanding of the two methods. Scores for the accuracy criteria are pending the actual implementation of the two models.

5.4 Mapping Trade Study

This trade compares the different maps and amounts of information that is useful and able to be processed by the rover. Balancing processing power needed and useful information is key. The different options included in this trade study were identified based on the most widely used techniques for mapping and the mission requirements. The criteria were decided based on the mission requirements and the amount of effort required to execute a mapping alogorithm to generate the required type of the map.

Most of the options listed can be implemented using open source frameworks like Robot Operating System (ROS). The current team working on the MoonRanger project plans to use the ROS implementation of Real Time Appearance Based Mapping[2] (RTAB Map) to generate an occupancy grid map from stereo images. The most definitive conclusion of performing this trade study is that there would be no requirement of generating 3D point clouds and the mission requirements could be satisfied by generating 2D maps of the environment.

Table 4: Mapping Trade Study

						
Criteria	**Weight**	**2D Binary Occupancy**	**2D Probabilistic Occupancy**	**3D Point Cloud**	**2.5D Map**	**Topological**
Sensor Requirement	20	4	4	5	4	4
Computation	15	5	3	2	3	5
Storage Space / Map Size	15	5	4	2	4	5
Input/Output Format	10	5	5	2	2	3
Amount of Information Captured	10	2	3	5	4	1
Reliability / Precision	20	4	5	4	5	1
Software Development Effort	10	5	5	3	4	5
Total	**100**	**430**	**415**	**340**	**385**	**340**

Apart from the above mentioned trade studies, our team plans to conduct more trade studies during the spring semester for different aspects of the project for which a solution has not been identified yet. We will be performing component level trade studies for specific parts as opposed to the process based trade studies done currently. We also plan perform a trade study for the process of pit detection.

6 Cyberphysical Architecture

Our cyberphysical architecture has direct correspondence to our functional architecture. It highlights the technology options which will be potentially used to execute the individual functions. The inputs and outputs of individual functions are also included.

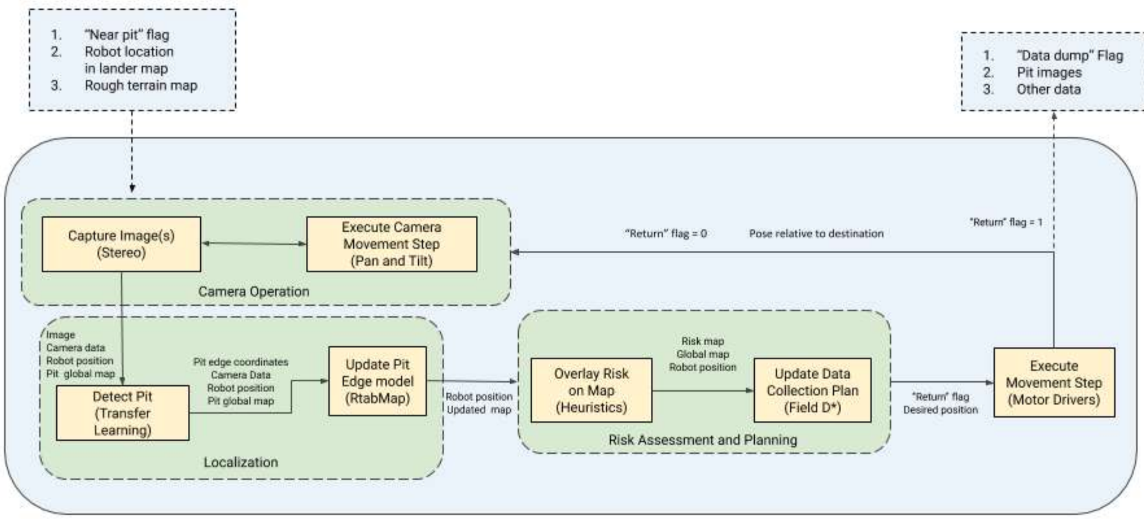


Figure 3: Cyberphysical Architecture

- **Capture Images:** A stereo camera will allow to estimate depth of points in an image.
- **Execute Camera Movement Step:** A camera mount having pan and tilt motions will be used.
- **Detect Pit:** This will be executed using transfer learning using pre-trained networks.
- **Update Pit Edge Model:** The ROS implementation of Real Time Appearance Based Mapping (RTAB Map) can be used to update the global map with pit data.
- **Overlay Risk in Map:** A heuristic solution will be employed for estimating the risk involved in continuing the mission.
- **Update Data Collection Plan:** Field D* will be used to calculate an optimal path that the rover traverses.
- **Execute Movement Step:** Once the decision is made, the rover either goes back to the starting point or continues to gather more data from different positions.
- **Interfacing Operations:** When the rover reaches the vicinity of the pit, the Pit Navigator system will activate, taking as an input the rough terrain map and robot pose established by the overarching PitRanger system. While the Pit Navigator system runs, it will continually pass collected data including images of the pit to other subsystems within PitRanger.

Our team has discussed potential options to be used as surrogate platforms. Currently our options include the robot Blue, the AutoKrawler and the CubeRover. However, we will most probably use the robot that will be developed by the current team which will be available in the spring semester. Until then, we will be using one of the above mentioned options.

Figure 4: AutoKrawler

Figure 5: Blue

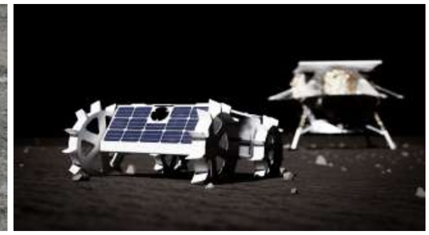


Figure 6: CubeRover

One loop through our cyberphysical architecture describes one cycle in the pit exploration mission. The robot continuously captures stereo images of its surrounding while navigating at a slow speed. It then classifies the images as being pit images or non-pit images. The robot also tries to detect (localize) the pit edge in pit images. Once the pit edge is detected in the image, the robot will calculate the distance between the pit and the camera frame. Thus, the robot localizes itself with respect to the pit edge. It also updates the global map with the newly computed coordinates of the pit edge.

After the map update is complete, the robot moves in close to the pit edge while continuously measuring its distance from the edge. During this process, the robot estimates the risk of moving close to the edge and decides whether to get closer to acquire better data. Upon getting sufficiently close, the robot captures multiple images covering a large area of the opposite wall of the pit. This will be achieved by having pan and tilt motions for the camera capturing images of the pit. When the required number of images have been captured the robot updates it current state and tracks the mission status. Based on these parameters, it decides whether to navigate to a new waypoint to capture more data or to go back to the starting location and cede control to the standard navigation system for return to the lander.

Table 5: Major Inputs and Outputs of Individual Functions

Subsystem	Function	Inputs	Outputs
Camera Operation	Execute Camera Movement Step	1. Pose Relative to Destination	1. Camera Pose
	Capture Images	1. Capture Flag	1. Images 2. Camera Data
Localization	Detect Pit	1. Captured Images 2. Camera Data	1. Pit Location in Images
	Update Pit Edge Model	1. Robot Pose in Global Map 2. Global Map	1. Robot Pose Relative to Pit 2. Updated Global Map
Risk Assessment and Planning	Overlay Risk on Map	1. Rover/Mission Parameters 2. Potential Waypoint	1. Updated Risk Map
	Update Data Collection Plan	1. Updated Risk Map	1. New Data Collection Plan 2. Optimal Path

7 Subsystem Descriptions

7.1 Camera Operation

The rover will have two sets of cameras. One set will be mounted statically on the front of the rover, and will be used for navigation during all phases of the mission. The second set will be mounted to pan and tilt motors, and will be used for capturing images of the pit. This extra actuation reduces the need for a high-resolution camera, since a single large image can be replaced by several lower-resolution images which capture only the most relevant data (i.e. the pit wall).

The camera operation subsystem is responsible for the operation of the physical hardware components of the cameras, and the initial handling of the data produced by the cameras. This system will drive the pan and tilt actuators in order to position the pit imaging camera, then actuate the shutter in order to capture images. This system will use established code from the MoonRanger product to control image capture from the navigation cameras. The camera operation subsystem will handle the processing of the raw image data from both sets of cameras into standard image files, then distribute those files to the localization subsystem.

Figure 7: Camera Operation Pan and Tilt

Figure 8: Two Cameras for Stereo Vision

An alternative method would be to statically mount the camera to the body of the rover, and drive the rover wheels in order to change what the camera captures. This would allow the rover to pan the camera, but rotating the rover near the pit edge would, at best, make the image capture sensitive to terrain conditions, and at worst could put the rover in danger of falling into the pit.

7.2 Localization

The localization subsystem analyzes the images collected by the rover's cameras in order to detect and model the pit. When new images are received from the camera operation subsystem, the localization subsystem first attempts to detect a pit in those images. All images are classified according to whether or not a pit is identifiable in the image. For all images containing a pit, the localization subsystem next estimates the position and dimensions of the pit. This is done by drawing a bounding box around the pit in the image.

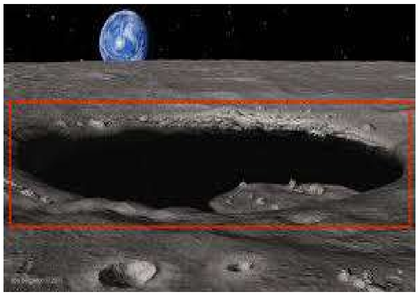


Figure 9: Detection Output

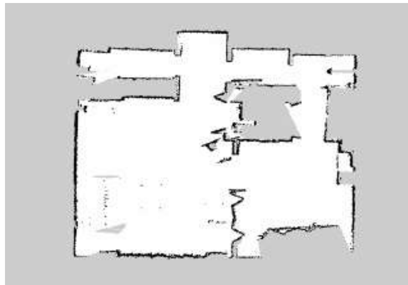


Figure 10: 2D Map Example

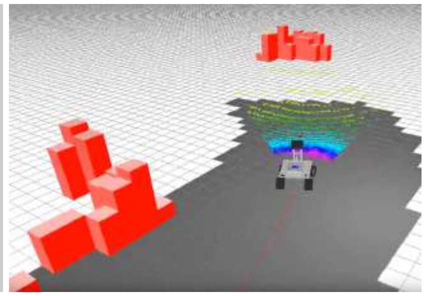


Figure 11: Localization Example

Research[4] is still being conducted on how best to detect pits and identify their characteristics from image data. The determination of which algorithm to use will be primarily based on robustness and ease of computation, since there is no room for failure in a moon mission.

In addition to calculating the size of the pit, the localization subsystem will also determine the position of the pit with respect to the camera frame. From this information it is possible to determine the distance between the rover and the pit edge, which is critical to safely navigating the rover around the pit.

The localization subsystem then translates the position of the pit with respect to the rover into coordinates in the global map, and updates the global map with this new calculation of the pit position. This global map is a 2D grid representing the environment through which the rover travels. By continually updating the map as the rover collects information about the pit we can ensure that the global map represents the area around the pit with the greatest possible fidelity.

While the exact method of map generation is still being discussed, the team has concluded that there is no need for the extra information contained within a 2.5D or 3D map. Because computational capacity and data storage are at a premium on the moon rover, it is ideal to enact a navigational strategy that uses a lightweight mapping algorithm and a compact global map.

7.3 Risk Assessment and Planning

At the core of this subsystem is a standard planning algorithm, which constructs a path for the rover to follow while avoiding obstacles that have been identified in the rover's map of its surroundings.

The rover generates a plan at two levels. The global plan is a series of locations in the global map, called waypoints, that the rover will travel to in sequence. These waypoints are determined based on the estimated size and location of the pit based on initial imagery, and are updated continuously as the mission progresses and the pit position is determined more exactly. The waypoints are points around the circumference of the pit, as close to the edge as the rover can safely access.

Each of these waypoints is a location from which the rover will take images of the pit wall, so the waypoints will be chosen to maximize the percentage of the pit's circumference that the rover will be able to capture.

The local plan is the route that the rover will take in order to travel from one waypoint to another. The local plan is based on the 2D map produced by the localization subsystem. Moving from cell to cell in the map grid incurs a cost, and the local path plan is generated by finding the connection from the rover's current position to the destination grid square that minimizes the cost incurred.

The planning algorithm bases the path solely on the map data, which itself only reflects the contours of the moon's surface. We aim to adjust this map based on our knowledge of the specific risks of a lunar pit and the conditions of our mission. The risk assessment package will use this information to adjust the map, thus changing the input to the planning algorithm.

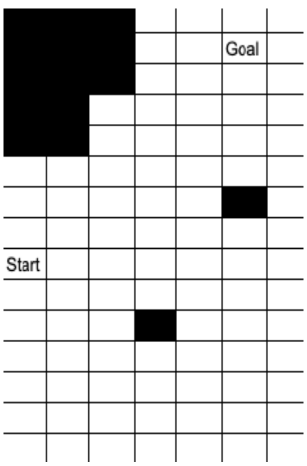


Figure 12: Costmap

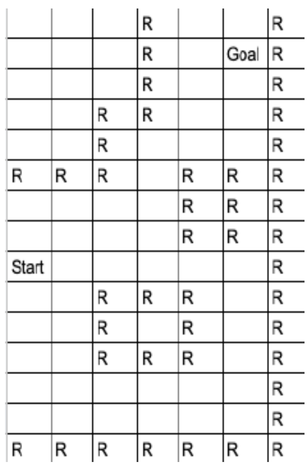


Figure 13: Risk Map

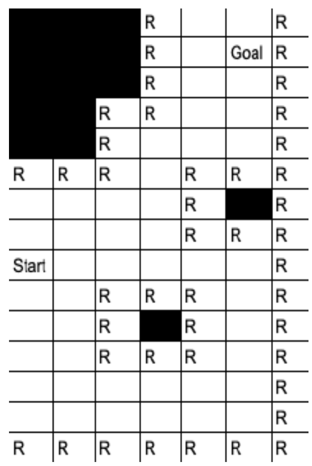


Figure 14: Updated Map

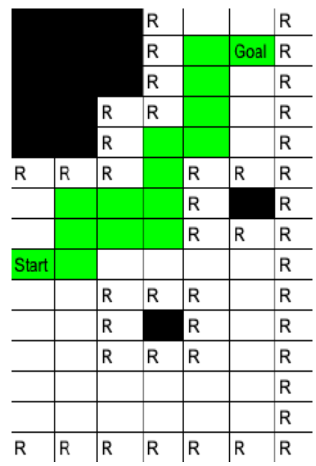


Figure 15: Final Path

At the global level, the risk assessment subsystem will consider the overall mission duration, the time required to travel between the pit and the lander, and the amount of data collected by the rover on each trip. The goal of this global risk assessment is to collect the maximum amount of pit data while ensuring that the data collected is regularly transferred to the lander, minimizing the data lost if the rover encounters a fatal error. The global risk will build up continuously as the rover spends time exploring the pit, so that the incentive to return to the lander becomes stronger until the global planner decides to return to the lander.

The local risk has the effect of adding additional untraversable areas to the map based on what we know about surface conditions near the pit. For example, the risk of going near the pit edge is high, so the local risk assessment should disincentivize the local planner from calculating routes that take the rover near the pit edge. This danger would not be reflected in the map data provided by the localization subsystem, so the risk assessment subsystem must adjust the map before the local plan is generated.

8 Project Management

8.1 Work Plan and Tasks

The majority of the work required for this project consists of software development. Our approach is to focus on building and testing each of the software packages described in our architecture separately during the spring, then spend the fall integrating them into a single functional system. This will allow us to verify the functionality of each individual subsystem and isolate any errors, making the eventual integration smoother. What hardware-related tasks are present relate to identifying hardware components that are necessary to enable our software functions, and acquiring those components as quickly as possible so that they can be used for testing. Lastly, our work plan reflects the need to practice project management throughout the duration of the project.

Table 6: Work Plan

1. HARDWARE		
1.1 Overall Rover	1.1 Camera	1. Perform Camera Trade Study 2. Acquire Camera 3. Test Camera
	1.2 Rover	1. Acquire Rover 2. Test Rover
2. SOFTWARE		
2.1 Camera Software	2.1.1 Camera Interface	1. Research Camera Interface 2. Acquire Camera Interface 3. Build Camera Interface 4. Test Camera Interface 5. Integrate Camera Interface
	2.1.2 Camera Movement	1. Research Camera Movement 2. Acquire Camera Movement Drivers 3. Build Camera Movement Package 4. Test Camera Movement 5. Integrate Camera Movement
2.2 Localization	2.2.1 Detection	1. Research Detection 2. Acquire Example Data Sets 3. Build Detection Package 4. Test Detection 5. Integrate Detection
	2.2.2 Edge Modeling	1. Research Edge Modeling 2. Acquire Edge Modeling Algorithm 3. Build Edge Modeling Package 4. Test Edge Modeling 5. Integrate Edge Modeling
2.3 Planning	2.3.1 Path Planning	1. Research Input 2. Acquire Input 3. Build Input 4. Test Input 5. Integrate Input
	2.3.2 Risk Assessment	1. Research Input 2. Acquire Input 3. Build Input 4. Test Input 5. Integrate Input

2.4 Interfacing Operation	2.4.1 Input	1. Research Input 2. Acquire Input 3. Build Input 4. Test Input 5. Integrate Input
	2.4.2 Output	1. Research Output 2. Acquire Output 3. Build Output 4. Test Output 5. Integrate Output
2.5 Movement Control	2.5.1 Movement Control	1. Research Movement Control 2. Acquire Movement Controller 3. Build Movement Control 4. Test Movement Control 5. Integrate Movement Control
3. MANAGEMENT		
3.1 Manage Work	3.2 Manage Schedule	
3.3 Manage Risks	3.4 Manage Finances	

The Pit Navigator Work Breakdown Structure has three sections. The first, Hardware, deals with tasks related to the physical components of the project; the cameras and the rover itself. The second section, Software, contains the majority of the tasks that comprise the Pit Navigator project. This section addresses multiple software subsystems, some containing multiple functions. Each of these subsystems must be researched, built into a functional program, tested rigorously, and integrated with other subsystems. The final section of the WBS covers the management tasks that are necessary for the project to proceed smoothly. These include adherence to schedule and budget, maintaining awareness of risks and plans for mitigation, and oversight of the work packages.

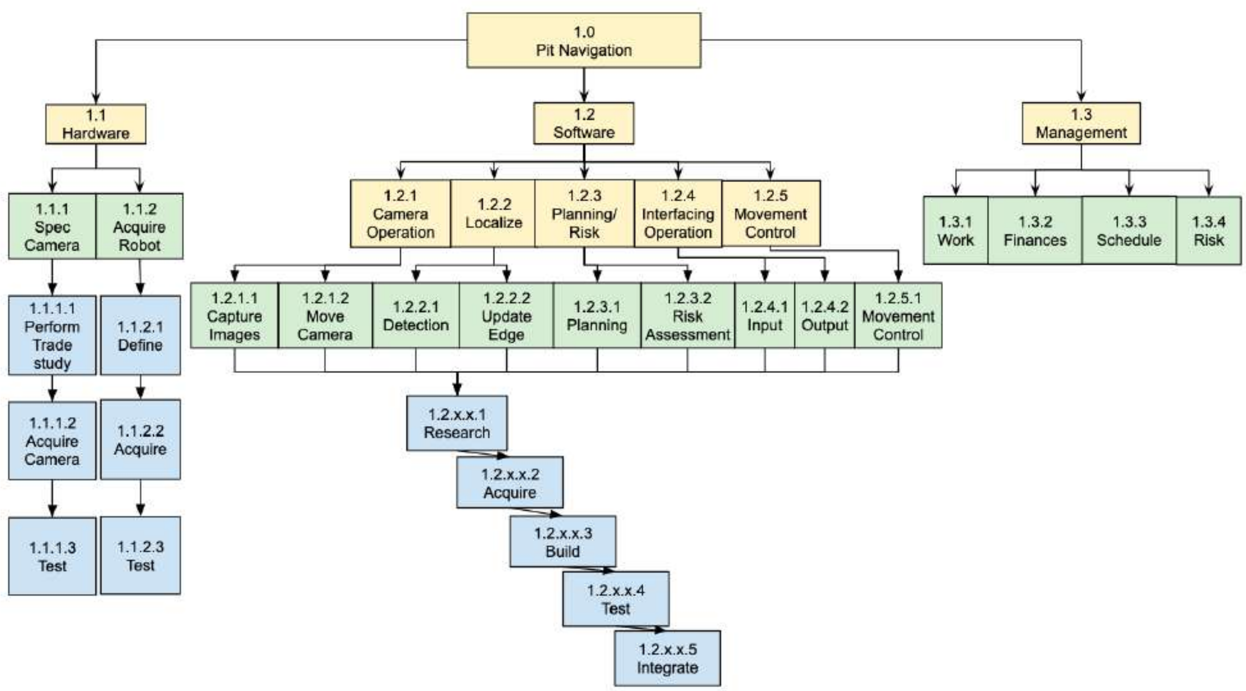


Figure 16: Work Breakdown Structure

8.2 Schedule

This schedule reflects our development priorities for the spring semester. We expect to continue research into various algorithms and hardware components in January, then make our selections in early February. Once each component has been chosen, we need to acquire whatever existing resources are available to serve as a starting point for our development. From that point, we can build any supporting code or additional functionality, then test the complete package. February will be largely occupied by this development process, with testing beginning around March. We anticipate that development and testing will occur in parallel, as we make revisions to our code in response to the results of our testing. Towards the end of the semester, as we prepare for our Spring Validation Demonstration, we will devote some effort to integrating the most closely related functions so that we can ensure that they interact in a logical way. We also anticipate that a rover surrogate will become available late in the semester, most likely in April. This assumption is based on the plan set forth by the MoonRanger project team for their future development. Once the rover surrogate becomes available, it will be a valuable tool for testing our code.

Table 7: List of Major Risks

ID	**Date**	**Task Completed**	**Task Started**
Progress Review 1	February 19, 2020	1.1.1.1. Perform Camera Trade Stud 1.1.1.2. Acquire Camera 1.2.1.1.1. Research Camera Interface 1.2.1.1.2. Acquire Camera Interface 1.2.1.2.1. Research Camera Movement 1.2.1.2.2. Acquire Camera Movement Drivers 1.2.2.1.1. Research Detection 1.2.2.1.2. Acquire Example Data Sets 1.2.2.2.1. Research Edge Modeling 1.2.2.2.2. Acquire Edge Modeling Algorithm 1.2.3.1.1. Research Path Planning 1.2.3.1.2. Acquire Path Planning Algorithm 1.2.3.2.1. Research Risk Assessment 1.2.3.2.2. Acquire Risk Assessment Algorithm 1.2.5.1.1. Research Movement Control 1.2.5.1.2. Acquire Movement Controller	1.1.1.3. Test Camera 1.2.1.1.3. Build Camera Interface 1.2.1.1.4. Test Camera Interface 1.2.1.2.3. Build Camera Movement Package 1.2.1.2.4. Test Camera Movement 1.2.2.1.3. Build Detection Package 1.2.2.2.3. Build Edge Modeling Package 1.2.3.1.3. Build Path Planning Package 1.2.3.2.3. Build Risk Assessment Package 1.2.4.1.1. Research Input 1.2.4.2.1. Research Output 1.2.5.1.3. Build Movement Control 1.2.5.1.4. Test Movement Control
Progress Review 2	March 4, 2020	1.1.1.3. Test Camera 1.2.1.1.3. Build Camera Interface 1.2.1.1.4. Test Camera Interface 1.2.1.2.3. Build Camera Movement Package 1.2.1.2.4. Test Camera Movement 1.2.4.1.1. Research Input 1.2.4.1.2. Acquire Input 1.2.4.2.1. Research Output 1.2.4.2.2. Acquire Output	1.2.2.1.4 Test Detection 1.2.2.2.4. Test Edge Modeling 1.2.3.1.4. Test Path Planning 1.2.3.2.4. Test Risk Assessment "
Preliminary Design Review	March 16, 2020	1.2.2.1.3. Build Detection Package	1.2.4.1.3. Build Input 1.2.4.1.4. Test Input 1.2.4.2.3. Build Output 1.2.4.2.4. Test Output "
Progress Review 3	March 25, 2020	1.2.2.2.3. Build Edge Modeling Package 1.2.4.1.3. Build Input 1.2.4.2.3. Build Output	1.2.2.1.4 Test Detection 1.2.2.2.4. Test Edge Modeling 1.2.3.1.4. Test Path Planning 1.2.3.2.4. Test Risk Assessment "
Progress Review 4	April 8, 2020	1.1.2.1. Acquire Rover 1.2.2.1.4. Test Detection 1.2.2.2.4. Test Edge Modeling 1.2.3.1.3. Build Path Planning Package 1.2.3.2.3. Build Risk Assessment Package 1.2.4.1.4. Test Input 1.2.4.2.4. Test Output	1.1.2.2. Test Rover 1.2.1.1.5. Integrate Camera Interface 1.2.1.2.5. Integrate Camera Movement 1.2.2.1.5. Integrate Detection 1.2.2.2.5. Integrate Edge Modeling 1.2.3.1.5. Integrate Path Planning 1.2.3.2.5. Integrate Risk Assessment 1.2.4.1.5. Integrate Input 1.2.4.2.5. Integrate Output

Spring Validation Demonstration (PR5)	April 22, 2020	1.1.2.2. Test Rover 1.2.1.1.5. Integrate Camera Interface 1.2.1.2.5. Integrate Camera Movement 1.2.3.1.4. Test Planning 1.2.3.2.4. Test Risk Assessment 1.2.5.1.3. Build Movement Control 1.2.5.1.4. Test Movement Control	1.2.5.1.5. Integrate Movement Control
SVD Encore (PR6)	April 29, 2020		
Critical Design Review	May 4, 2020	1.2.2.1.5. Integrate Detection 1.2.2.2.5. Integrate Edge Modeling 1.2.3.1.5. Integrate Path Planning 1.2.3.2.4. Integrate Risk Assessment 1.2.4.1.5. Integrate Input 1.2.4.2.5. Integrate Output 1.2.5.1.5. Integrate Movement Control	

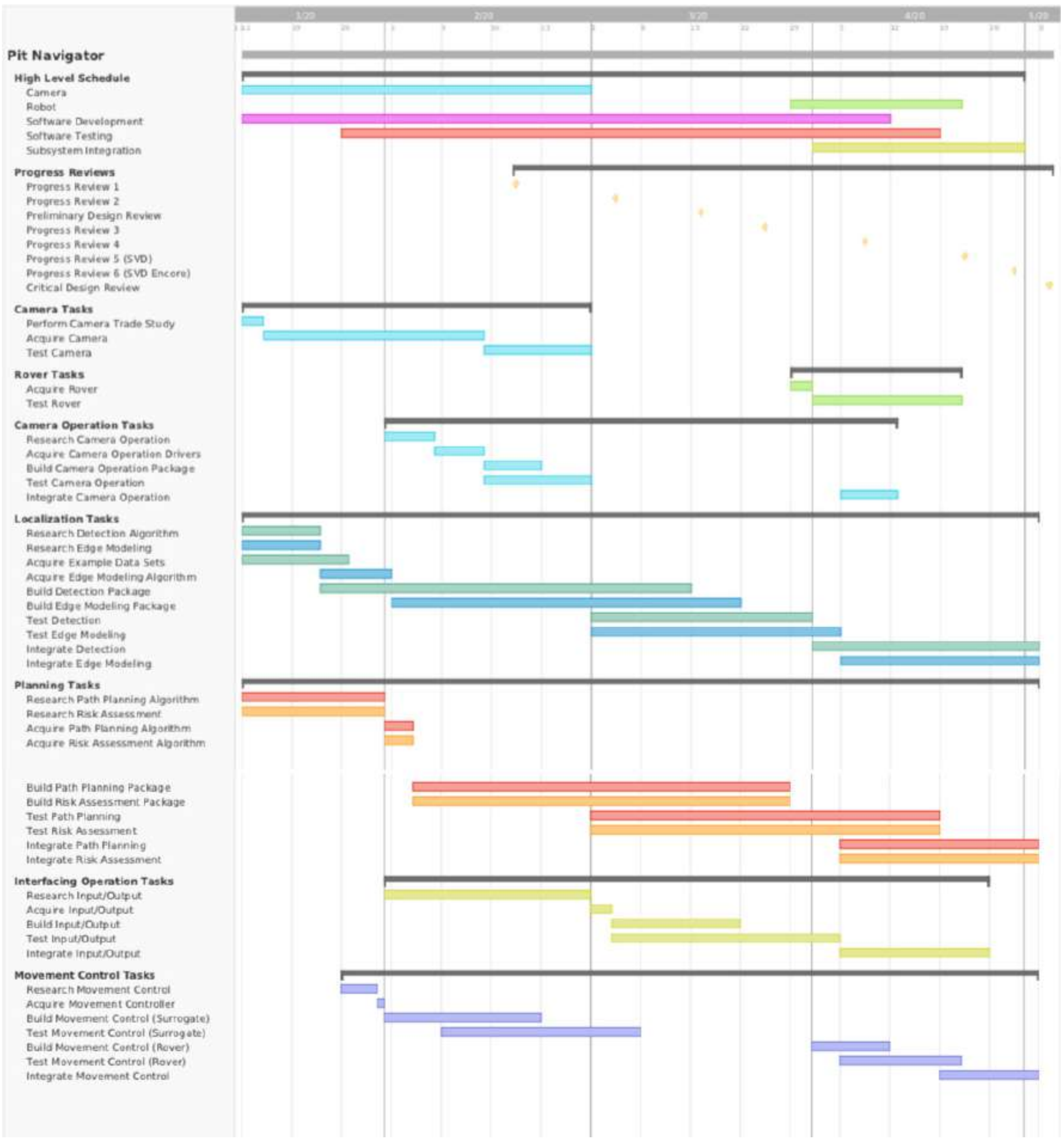


Figure 17: Spring Schedule

8.3 System Validation Experiments

8.3.1 Spring Validation Experiments

1. **Image Capture Test**
   (a) Objective
      i. Data captured per image (1 MB) — Mandatory Performance - 1
      ii. Data captured per location (15 MB) — Mandatory Performance - 3
      iii. Resolution of opposing wall at 100 m (1 pixel/sq.inch) — Desirable Performance - 3
      iv. 60% of image showing pit wall % of time — Desirable Performance - 4
      v. 5 degree angle of pit circumference covered per location — Mandatory Performance - 7
   (b) Conditions
      i. Location - Football field
      ii. Setup
         A. Place camera at specified location
         B. Place markers in 5 degree arc, approximately 100m from camera
         C. Markers have pattern to determine pixels per square inch
      iii. Equipment - Camera x2, Camera Mount, Markers
   (c) Steps
      i. Run "vantage point" code
         A. Camera aligns to first marker
         B. Camera takes images
         C. Repeat for all markers

2. **Pit Identification Test**
   (a) Objective
      i. Accuracy of recognition 80% — Mandatory Performance - 4
      ii. 20 meters max pit detection distance — Mandatory Performance - 8
   (b) Conditions
      i. Location - MRSD Lab
      ii. Setup - Acquired database of images
      iii. Equipment - Images of pit and Computer
   (c) Steps
      i. Instruct pit identification pipeline to act on images
      ii. Run pit identification pipeline
      iii. Show output of pit detection

3. **Pit Location Estimation Test**
   (a) Objective
      i. Distance to pit 10% error — Mandatory Performance - 5
      ii. Pit size and shape 25% error — Desirable Performance - 2
   (b) Conditions
      i. Location - MRSD Lab
      ii. Setup - Acquired database of pit images
      iii. Equipment - Pit images and Computer
   (c) Steps
      i. Instruct pit location estimation pipeline to act on images
      ii. Run pit location estimation pipeline
      iii. Show output of pit location estimation

4. **Risk-Adjusted Planning Test**
   (a) Objective
      i. 1 meter distance from edge achieved 80% of the time — Desirable Performance - 1
      ii. Less than 20 seconds plan calculation time — Mandatory Performance - 6
      iii. Max 5:1 risk of mission ending incident — Mandatory Performance - 9
   (b) Conditions
      i. Location - MRSD Lsb
      ii. Setup - Simulated pit environment and simulated rover
      iii. Equipment - Computer
   (c) Steps
      i. Define parameters for test (Distance to lander, Mission duration, Data storage available)
      ii. Begin simulation
      iii. Rover enacts calculated plan
      iv. Match rover actions to calculated plan
      v. Repeat with different parameters

8.3.2 Fall Validation Experiments

1. **Pit Edge Data Capture Test**
   (a) Objective
   (b) Conditions
      i. Location - LaFarge
      ii. Setup - Place rover near pit
      iii. Equipment - Rover surrogate
   (c) Steps
      i. Activate rover
      ii. Rover detects pit
      iii. Rover identifies vantage point on the pit edge
      iv. Rover drives to vantage point and captures images of the pit

2. **Simulated Mission Execution Test**
   (a) Objecttive
   (b) Conditions
      i. Location - MRSD Lab
      ii. Setup - Simulated pit environment and simulated rover
      iii. Equipment - Computer
   (c) Steps
      i. Begin simulation
      ii. Rover creates a global plan
      iii. Rover follows route to waypoints defined by global plan
      iv. Rover adjusts plan over time based on risks
      v. Rover returns to pit area
      vi. Rover continues executing global plan
      vii. Repeat until mission duration is complete

8.4 Team Member Responsibilities

The fact that our team consists of only three team members impacted the way in which we divided responsibilities. We concluded that assigning primary and secondary owners for every task would be overly redundant and would spread each team member's attention too thin. Therefore we only assigned secondary responsibility for those tasks that we concluded were the most significant, and assigned other tasks to single team members.

Table 8: Division of Responsibilities

Techinical responsibilties	Primary	Secondary
Spec Camera	Awadhut	Alex
Acquire Robot	Alex	Justin
Camera interface	Awadhut	
Move Camera	Awadhut	
Detection	Awadhut	Justin
Update Edge	Justin	Awadhut
Planning	Alex	Justin
Risk Assessment	Alex	
Input	Justin	
Output	Justin	
Movement Control	Alex	

Responsibility	Primary	Secondary
Structure Website	Awadhut	Alex
Content Website	Justin	
Budget Management	Awadhut	
Work Management	Justin	Alex
Schedule Management	Justin	Awadhut
Risk Management	Alex	Awadhut
MoonRanger Liason	Alex	
Member	**Total Primary**	**Total Secondary**
Alex	6	3
Awadhut	6	3
Justin	6	3

8.5 Parts List and Budget

Our team has two significant advantages in the budgeting of our project. The first is that our project is primarily software, which allows us to avoid the expense of purchasing physical components. Second, our project is contained within the umbrella of a larger development (the MoonRanger/PitRanger project) which has significant financial resources that we can leverage. Because we expect our costs to be minimal, and to be able to outsource many expenses to the wider organization, our expected budget is fairly low.

Table 9: Part List and Budget

Part	Type	Quantity	Cost	Total
Camera	Stereo Camera	3	177$	531$
Camera Mount	Custom	2	-	50$
Processor (Rover Computer)	Nvidia Jetson TX2[5] / Intel Nuc[6]	2	500$ / 880$	1760$
Connecting Cables	USB + MicroUSB	3	12$	36$
Microcontroller	Arduino For motor control	2	28$	56$
Battery	Lithium Polymer	3	160$	480$
Emergency Switch	Push Button	2	11$	22$
Total				**2935$**

8.6 Risk Management

The risks associated with the Pit Navigator project fall into four different categories relating to what needs to be managed. First are technical risks, this category represents risks to the technical performance of the rover. Similarly, if the risk were to become an incident then the technical performance of the rover would suffer as a result. An example of this is, the rover is unable to detect the lunar pit, so the rover might not stop to image the pit and instead fall in. The second category of risks are work-related risks. If these risks are not accounted for, the Pit Crew might not finish all the work that needs to be done, or re-do work that has already been done. The third category of risks are related to the budget. One can see that if the Pit Crew buys a camera that doesn't fit requirements, then we would need to buy another camera that does, thus wasting money. The final category is the largest and are schedule risks. As the Pit Navigator project has a timeline of a year, the timeline is full of risks that need special attention. Technical, Work, Budget, and Schedule Risks are the four categories of risk that we have organized and will mitigate throughout the project.

Table 10: List of Major Risks

ID	TYPE	DESCRIPTION	MITIGATION STRATEGIES	L	C	IMPACT
1	Technical	The hardware necessary to support the Pit Navigator system is beyond what the rover is able to carry.	**1**.Continuously follow up with the team developing the hardware architecture. **2**.Weekly updates for an hour discussing this and other topics will maintain the requirements of the customer.	4	4	HIGH
2	Schedule	Requirements change depending on the rover.	**1**.Develop code to be rover agnostic. **2**.Approach development in a way that makes minimal/no assumptions about specific hardware components, only approximate level of functionality available.	5	3	HIGH
3	Schedule	External resources are late or never become available.	**1**.Develop code to be run under simulation. Major changes to requirements and test beds may not be solvable with communication so the code must also run under simulation.	5	3	HIGH
4	Schedule	Schedule becomes untenable for a team of three.	**1**.Practice project management for schedule and cut scope when necessary. Project management for schedule includes: Create a schedule, hold weekly meetings to discuss progress, discuss schedule slip(if any), discuss long lead items/purchasing orders, personnel availability, facility availability, and how the schedule should change based on information.	5	4	HIGH
5	Work	Communication with the external community is inefficient and unclear.	**1**.Schedule regular meetings with the sponsor and the team working on the project. Provide them with clear statements of work. **2**.Establish specific personnel from external community as points of contact and responsible parties for deliverables.	5	3	HIGH
6	Technical	Pit Navigation software causes Pit Exploration mission to fail.	**1**.Perform verification of all Pit Navigation subsystems. **2**.Schedule in time to perfect the system and test for false positive test results. **3**.Prioritize verification of existing functionality over addition of additional functions.	4	5	HIGH

7	Technical	The rover is unable to identify the pit from camera data during operation.	**1**.Perform tests on the detection system at Lafarge with proper lighting conditions, perform pit tests at smaller scale, perform tests in simulation with the same algortithm.	3	5	HIGH
8	Technical	The rover takes too long to reach/navigate around/return from the pit and therefore captures minimal imagery of the pit.	**1**.Perform tests on the Planning/Risk Assessment systems in simulation. **2**.Make time consideration a priority of algorithm selection. **3**.Prepare alternative algorithm choices as backups.	3	4	MEDIUM
9	Budget	Acquired Camera does not satisfy requirements of system.	**1**.Buy a new camera. If budget is low ask sponsor for more money to acquire a camera that satisfies requirements.	3	3	MEDIUM
10	Schedule / Work	Important Risks are not tracked or identified.	**1**.Practice Project Management for Risks. This includes: Identify risks, estimate likelihood and consequence of risks, develop plan to mitigate risks where necessary. **2**.Meet each week and discuss tracked risks, attempt to identify new risks, and perform mitigation of tracked risks.	4	3	MEDIUM
11	Technical	The rover is unable to safely reach a point on the pit edge from which it can take usable pictures.	**1**.Perform tests on the Planning/Risk Assessment systems in simulation. Adjust risk weighting to ensure a solution is found. **2**.Prepare alternative algorithm choices as backups.	2	4	MEDIUM
12	Technical	Not having an appropriate test-bed	**1**.Have the ability to test on multiple different rovers and within different simuations. **2**.Maintain contact with external community for access to rovers and simulation enviornments.	4	3	MEDIUM

Figure 18: Likelihood vs Consequence Matrix for the Major Risks

9 References

[1] Whittaker, William. *MoonRanger, Flight-Forward Moon Rover with Exploration Autonomy.*. Google Drive, Robotics Institute, Carnegie Mellon University, 27 Feb. 2019,

[2] Real-Time Appearance-Based Mapping, 2019. http://introlab.github.io/rtabmap/

[3] Leonardo Araujo Santos. *Policy, leonardoaraujosantos.gitbooks, gitbook, 2017*.

[4] Heng-Tze Cheng, Feng-Tso Sun, Senaka Buthpitiya, Ying Zhang, Ara V. Nefian*Lunar Image Classification for Terrain Detection*. Department of Electrical and Computer Engineering, Carnegie Mellon University * Intelligent Robotics Group, NASA Ames Research Center

[5] Jetson TX2 Module, Nvidia, 2019. https://developer.nvidia.com/embedded/jetson-tx2

[6] Intel NUC, INTEL, 2019. https://www.intel.com/content/www/us/en/products/boards-kits/nuc.html

[7] Figure 6: https://www.post-gazette.com/business/tech-news/2019/06/06/CMU-moon-rover-astrobotic-red-whittaker-nasa-pittsburgh-space-china/stories/201906040108

[8] Figure 7: Lumens VC-B30U" Touchboards, Touchboards, 2019, https://www.touchboards.com/lumens-vc-b30u-pan-tilt-zoom-cameras/

[9] Figure 11: ZellTuebingen "Building Local Terrain Maps Using Spatio-Temporal Classification for Semantic Robot Localization", Youtube, Youtube, Apr 4, 2017, https://youtu.be/QjKvKrSgd4A?t=65