NASA LEC Robotic Mining

1.0

Project Proposal

August 2020

By: Taylor Ertrachter, Bailey Hamant, James Spies

Table of Contents

1 Team Background

1.1 Names and Emails of Project Members

Name	Email	Position
Taylor Ertrachter	tertrachter2017@my.fit.edu	Programmer
Bailey Hamant	bhamant2017@my.fit.edu	Programmer
James Spies	jspies2017@my.fit.edu	Programmer

Name	Email	Position
Nathaniel Bouchie	nbouchie2017@my.fit.edu	Excavation/Structures
Michael Foster	mfoster2017@my.fit.edu	Excavation/Structures
Delanie Glock	dglock2017@my.fit.edu	Structures
Eric Hu	ehu2017@my.fit.edu	Software
Kevin Landers	klanders2017@my.fit.edu	Electronics
Steven Pazienza	spazienza2017@my.fit.edu	Software
Alex Perez	aperez2017@my.fit.edu	Electronics
Youpeng Xu	yxu2017@my.fit.edu	Electronics

1.1.1 Faculty Advisor

Name	Email
Dr. Marius Silaghi	msilaghi@fit.edu

2 Project Scope

2.1 Faculty Advisor

Our Faculty advisor will be Dr. Marius Silaghi.

2.2 Client

Our client is the Aerospace NASA Lunabotics Engineering Competition(LEC) team division. Our boss is Dr. Kimberly Demoret, the Aerospace team advisor.

2.3 Development Meeting Schedule

During the summer we met with everyone in the software team and Steven to discuss preparations for the fall semester. Our current fall semester has us meeting with just the full Aerospace Team at 5pm on Mondays. At 5:30pm, Dr. Demoret joins our meetings. Currently, the only Computer Science majors are Taylor, Bailey, and James; however, Steven is the leader of the Software Subteam for the NASA LEC project. He is an Aerospace major, but he is helping to manage our goals and progression with the software we work on.

2.4 Goals and Motivations

Every year, Florida Tech's entries for the NASA Lunabotics Engineering Competition have been met with varying degrees of success. Upon reflection, we have come to the conclusion that while the hardware has always been a strong part of the project, the software is usually lacking due to the team's vague knowledge of basic programming taught in their curriculum. It is our goal as computer scientists to apply the level of excellence found in the hardware design of the robot to its software.

The motivation behind this project is to further improve upon the work done by the people before us. This year, we all believe that we can create an autonomous robot that can perform the tasks laid out in the NASA LEC guidelines.

2.5 Approach

Our system will implement these three key features into the robot:

1. Manually controllable robot actions for all major motors (navigation, claw, conveyor belt) through Xbox wireless controller. While the final goal of the project is to make the robot fully autonomous, robots can still qualify and win the competition being controlled. It will be best to implement this manual control in case we are not able to fully finish the automation code. Using Bluetooth, we are able to connect the Xbox controller to the onboard computer, a Nvidia Jetson.

2. Object detection for future autonomous navigation functionality. This will be key to having a well working autonomous robot. During the competition, there are many objects laid out to try and trick the robot. Using the Zed dual lensed camera, we will be able to not only recognize obstacles on the playing field, but acquire additional information about the object such as distance from the robot.

3. Fully autonomous sample collection, navigation, and deposition system using manual controls implemented in feature one as backup. This will be the hardest feature to implement. An automation algorithm will be constructed that takes multiple samples from

the many onboard sensors that will be equipped to the robot. The most important will be the Zed camera discussed in feature two.

We also plan on creating a simulation to test the code in so that if we are working on a specific piece of the robot that has not been finished by the hardware team, we can still run and test our code. Having a virtual simulation will allow us to test code that we are wary of, and see what the code does so as to not break the actual robot running untested code. This was also suggested to us so that in case of something happening in the future where the robot is not built, or is not accessible due to online only learning, we can still showcase that we have written code to operate a robot on a similar level.

2.6 Novel Features & Functionalities

A novelty feature in this project would definitely be the autonomy of the robot. Even though we are still far away from achieving that goal, I believe that having an autonomous robot will be a huge breakthrough for this team. There is a massive difference between having a controllable robot that simply takes input from a controller, and having a robot that does all the work for you when you turn it on. There has yet to be a fully autonomous robot that can do all of the tasks on its own, so getting anywhere near full automation will be a huge achievement.

2.7 Technical Challenges

The client has requested that for readability purposes, the entire code be written in as much Python as possible. This means that all of our controls for the robot will be done using new libraries that we are unfamiliar with. Along with this, we are using new APIs, that we have no experience with, to interface with the camera for object detection and motor controls. We also have very limited experience with object detection and automation, so we will need to learn how to do these processes while applying them to our project.

Another challenge for us will be online learning. Currently, all of the group members for the Software Subteam are present in Melbourne, but that status may change as time goes on. If an order comes from the school to move all classes to online again, then we may have to abandon working on the physical robot and purely maintain work on our virtual environment robot.

3 Project Milestone Outlines

3.1 Milestone 1 Outline

Create a virtual environment in which we can test robot controls on a simulated level. This is so that we can continue to work on code for specific functions even if the hardware has not been finished for it yet. It is important that we have the virtual environment set up as soon as possible so that once we start writing and testing code, we know that there is a way to actually see what it will do.

We plan on having very similar, or the same, code be run in the virtual environment as the real robot. This is to ensure that the testing we do inside the virtual environment reflects what will happen to the physical robot.

3.1.1 Task Matrix for Milestone 1

Tasks	Bailey	James	Taylor
Compare and select Technical Tools	Unity	Blender	Python
"hello world" demos	Low level robot with the same features as real robot	The robot’s wheels can go forward, backward, and turn	The robot’s arm and conveyer belt are operable
Resolve Technical Challenges	3D model of robot is accurate to real life specifications	Creating controller for wheels to allow for locomotion	Robot arm and conveyer belt can move in all directions
Compare and select Collaboration Tools	Google Docs, Google Slides	Whatsapp	Discord, Github
Requirement Document	write 50%	write 25%	write 25%
Design Document	write 25%	write 25%	write 50%
Test Plan	write 25%	write 50%	write 25%

3.2 Milestone 2 Outline

Having a fully operable robot that can be controlled by an Xbox controller. Using Python to take inputs from the controller and map those inputs to specific actions on the robot will be key. A fully functioning robot includes full 360° movement, an operable arm, and a moving conveyor belt. We plan on testing the functionality first in the virtual simulation, and if possible testing some or all of these features on the physical robot.

3.2.1 Tasks for Milestone 2

- Implementing Python controller for robot. Testing to see if the inputs from the Xbox controller correspond to a robot action. Demo the virtual robot moving.
- Implementing movement based on controller input. Testing the robot's wheel movement working the way we want. Demo the virtual robot driving around.
- Implementing an operable arm controller based on controller input. Testing to see if the arm can be controlled by the Xbox controller. Demo arm moving around.
- Implementing a moving conveyor belt. Testing to make sure the belt can be turned on and off on command. Demo turning the belt on/off.

3.3 Milestone 3 Outline

Using a dual lens Zed camera, the robot can detect objects. The ultimate goal here is to detect whether there is an obstacle in the way for the robot that it would need to move to avoid. This milestone will be used as a stepping stone for creating an autonomous robot that could go through an obstacle course on its own. For now, we just want to see that the robot sees an object in front of it, recognizes its existence, and can take action according to what the object is.

3.3.1 Tasks for Milestone 3

- Implementing the camera's vision into the code. Testing to see how the camera can be used to detect objects. Demo the camera's visuals.
- Implementing the camera code into the robot's actions. Testing to see if the robot can use the camera visuals to move. Demo object detection and reaction to it.
- Implementing different actions based on different objects. Testing to make sure the robot knows what each object is that it detects. Demo robot acting according to different objects.

Faculty Advisor Approval

"I have discussed with the team and I approve this project plan. I will evaluate the progress and assign a grade for each of the three milestones."

DocuSigned by:

Signature: Dr. Marius Silaghi Date: 8/30/2020

4C78522EF27840A...