Survivor Identification and Retrieval Robot Project Proposal

Karun Koppula   Zachary Wasserman   Zhijie Jin

February 8, 2018

1   Introduction

1.1   Objective

After the Fukushima Daiichi didaster in after a 2011 tsunami, where robotics proved incapable of responding in disaster scenarios massive amounts of funding has been allocated to solving the problems that prevent robots from operating in unconstrained and unknown environments. Many teams developed different platforms which have since been used in disaster response situations [6] DARPA, specifically, has a Robotics challenge that requires robots to perform tasks that might be required in a disaster scenario that include interacting with objects, maneuvering in uneven and unknown terrain, and even driving a car. Unfortunately progress in the these directions have not made significant enough progress to be able to reliably use in disaster scenarios [1]. Disaster response robots of a different kind are currently being used, however, as documented and run by the Center for Robotic-Assisted Search and Rescue. These robots are for the most part human operated and are limited by human restrictions and the ability to communicate with the operator. Since for these robots are not being used for heavy manipulation, their primary task is to identify humans in danger as well as dangerous areas. In disaster scenarios, it is often the case where finding survivors is a time critical operation given their possible medical states. Having an autonomous robotic platform that is onsite and can immediately begin exploration of the environment would increase the overall response time dramatically. It would cut down on the time that it takes for first responders to arrive on the scene, set up their robotics systems, and establish safety parameters for operating in the environment. If an autonomous system can do a search and survey and present accurate information to the first responders, they can act immediately and potentially safe lives.

We envision an implementation of a robotic system that uses reinforcement learning to train how best to navigate in a maze environment while identifying the positions of "survivors" and performing a simple retrieval task. We will develop a differential drive robot with visual and depth sensors, that is given an occupancy grid of its environment. It will move through the maze to unknown survivor positions and move a simple object to and goal position from arbitrary starting positions. Since we are limited in scope for this project, we wanted to develop a proof of concept that can be exteneded in future work to a robotic platform that can operate in real building environments.

1.2   Background

In recent years, work has begun on using machine and reinforcement learning techniques to develop robotic systems that can navigate through a simulated environment using visual and distance data, that eliminates the need for classical navigation tasks such as localization and trajectory planning. At the 2017 NIPS conference Pieter Abeel gave a keynote presentation about the use of "meta-learning" which is an algorithm that can learn a policy for a reinforcement learning task, referencing work done by Mnih [5] This method greatly cuts down on the number of training episodes needed to

converge to an optimal behavior for each new environment. It allows the system to generate a general policy for exploration that doesn't overfit to a particular environment. This presentation inspired this project, but is beyond the necessary requirements for this project. With classical reinforcement learning, the training will allow the robot to function very well in a specific environment or on a specific task. If a robot is trained on a particular building before a disaster, it will be optimally equipped for the search as opposed to a generalized search robot.

Work has also been done with directly training motor outputs from image data using deep convolutional neural networks in manipulation tasks [3]. This can be applied in the future to working to directly help survivors out, but requires large amounts of processing power that we will not have on our platform. We will use simple control and verification techniques to interact with the object. Actor-critic training algorithms are also at the leading edge of progress in this direction. They have been used for manipulations tasks [7] as well as navigating 3-D mazes in simulation using only visual and depth measurements [4]. Since operating a robot in real space produces much noise that is not apparent in simulation, we simplify the inherent localization task by providing a representation of the environment, that would be interefered with by motor and environment noise.

1.3 High-Level Requirements

- The robot should be able to navigate through a three dimensional maze that is free of obstacles.
- The robot's performance should improve over the course of training on the same maze.
- The robot should prioritize finding the survivors and retreiving the goal object.

2 Design

2.1 Physical Design

The physical design of this robotic platform was chosen to minimize the dynamic constraints and associated challenges. Differential drive robots have a relatively simple set of dynamics and are inherently stable with the two unactuated wheels. Because of this, we decided to use a tiered approach to component distribution, which minimizes the physical footprint and allows for a greater safety margin while operating in the maze environment. Within our performance parameters, we do not expect to run into any situations that would make the robot uncontrollable. We have decided to place the actuator and environment sensors pointing forward in the centerline of the robot to align, as close as we can, the coordinate frames of each unit. This reduces the computational complexity, which is important, given the resources necessary for the sensor data processing.

Figure 2.1 shows a probable physical design. The motors and wheels will be chosen to suit a range of performance parameters. Because we do not know what the optimal responsiveness of this system in this environment is, we have chosen an arbitrary desired speed and acceleration at the high end of what we estimate to be the operable range. The two unlabelled circles in the "Bottom View" image are the two unactuated ball bearings that act as rolling supports for the robot. The gripper at the front of the platform will be actuated by two servos, one used to open and close the claw, and the other used to tilt the claw up and down in order to lift the goal object.

Figure 2.1 shows the tiering aspect of the design. This streamlines the connectivity of the platform by moving the Raspberry Pi closer to the sensors and out of the way of the major power connections between the motors and controller. Not included in the diagrams are possible wiring holes and access points that could be included to clean up the appearance of the platform as well as make wiring easier for the group. We will not permanently mount any of the components into the platform frame, to give us the freedom to interchange componenets.
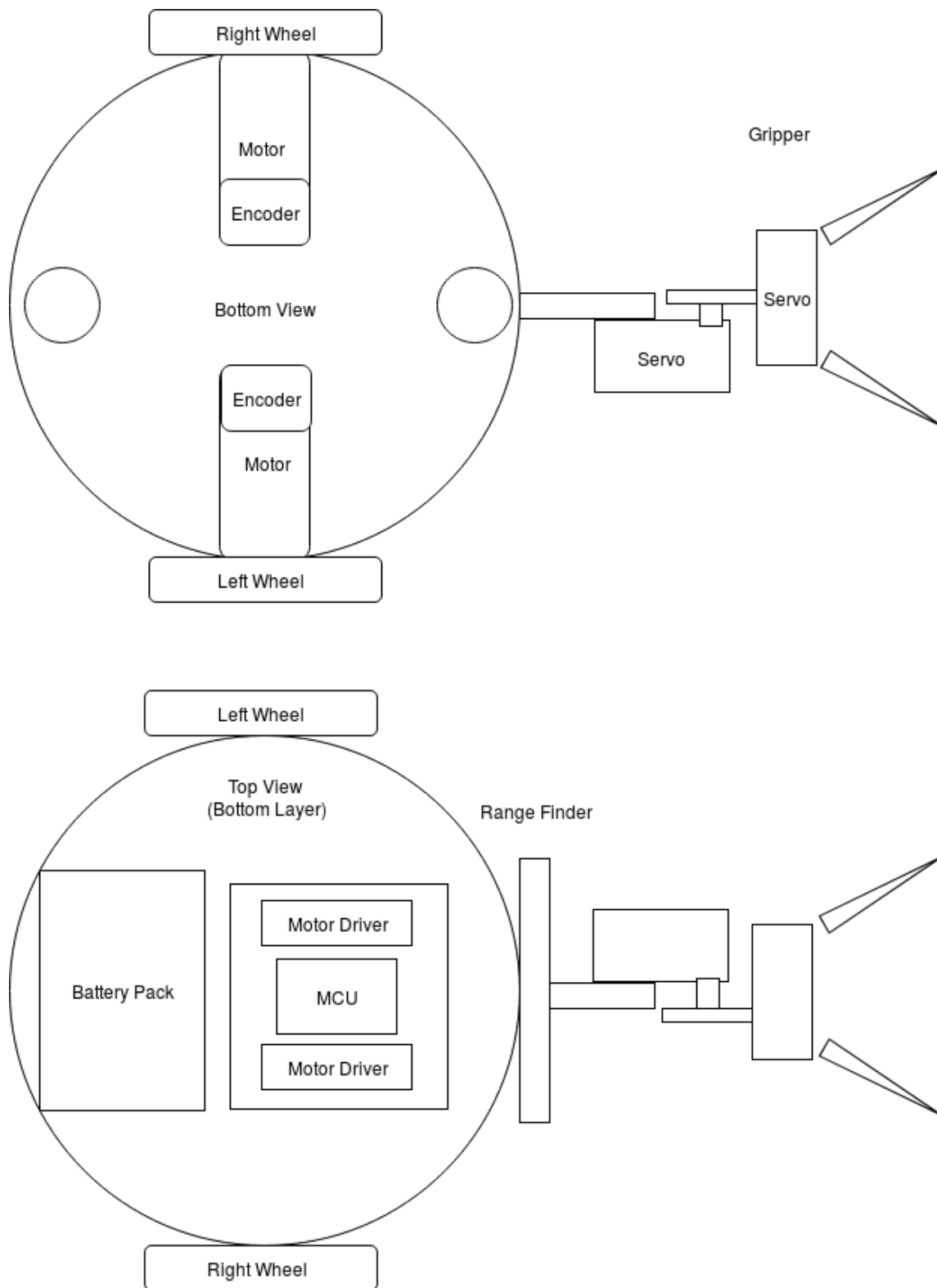
Right Wheel

Gripper

Motor

Encoder

Bottom View

Servo

Servo

Left Wheel

Left Wheel

Top View
(Bottom Layer)

Range Finder

Motor Driver

Battery Pack

MCU

Motor Driver

Right Wheel

Figure 1: Prospective Physical layout of robotic platform
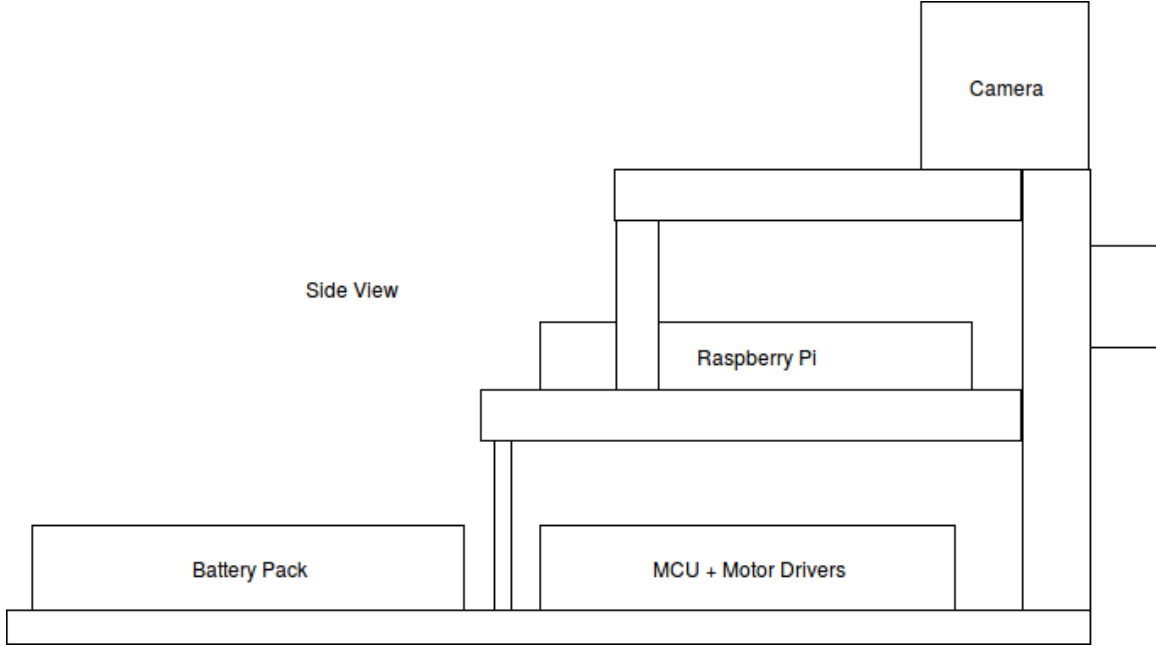
Figure 2: Side view to show tiered layout of components

2.2 Block Diagram

Figure 2.2 gives a layout of the different modules that are necessary for the robotic platform to function properly.

2.3 Block Requirements

2.3.1 Control Block

The control block encapsulates two processors, that will control the entire operation of the robot. The Raspberry Pi will be used to process the sensor data and store the RL policy and the occupancy grid. It will make decisions in real time about the optimal steps to take and output desired locations to the motor control processor which will then drive the robot to the desired position. The motor control MCU will implement a standard PID controller to drive the robot to the desired position. The Raspberry Pi block requirements are as follows,

1. The RL policy must allow the robot to drive through the environment without colliding with the walls.

2. The RL policy must direct the robot to reach the goal position.

3. The Raspberry Pi must interpret the camera data to identify the object and pick it up.

4. The Raspberry Pi must be able to communicate with the MCU through SPI/I2C protocol.

The Motor Control MCU requirements are as follows,

1. The PID controller should provide accurate position control to within $1cm$ of the desired position for the robot reference frame.

2. The MCU must be able to communicate through SPI/I2C protocol.

3. The MCU must be able to interpret and execute control signals from the Raspberry Pi.
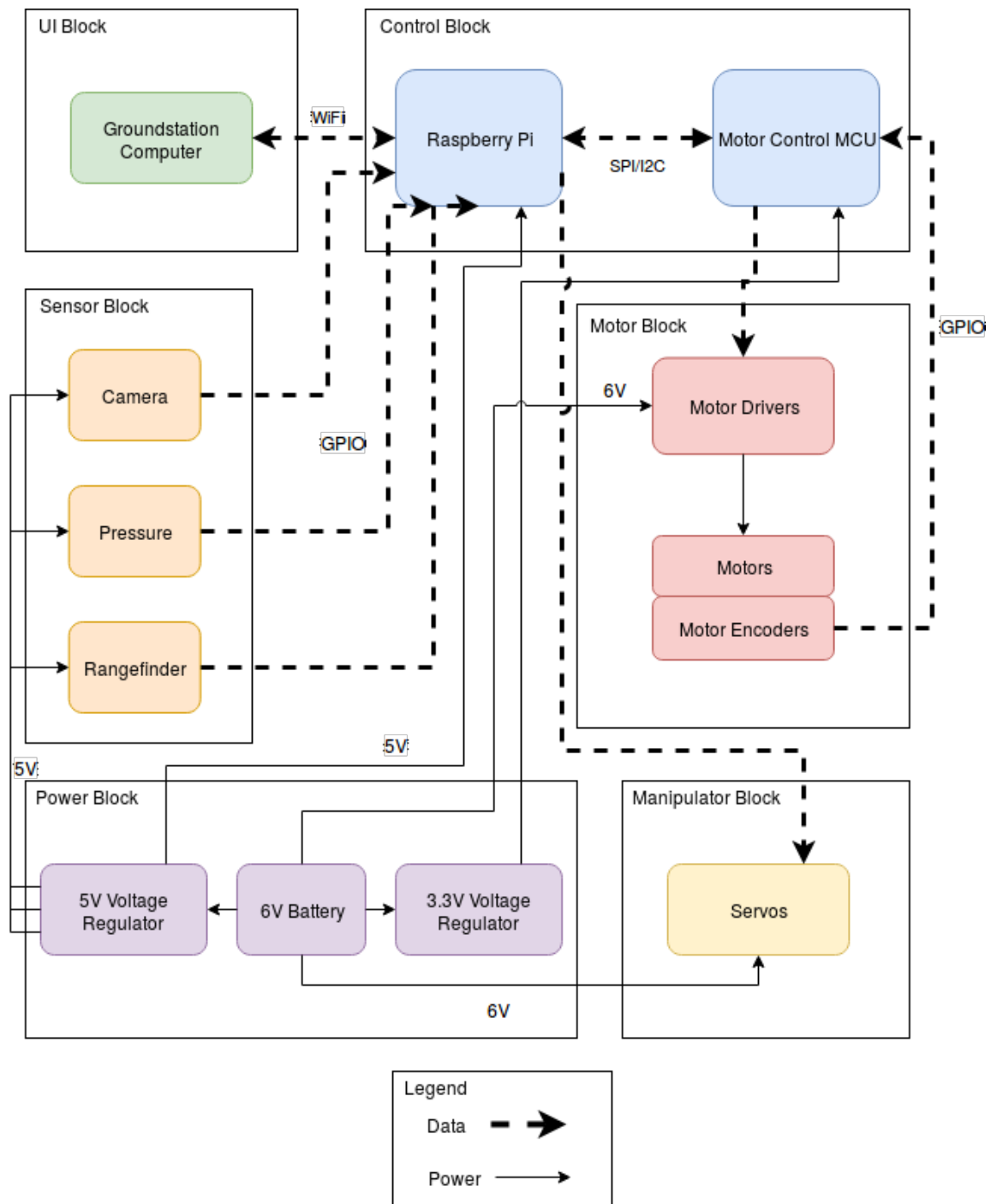
Figure 3: Robot Block Diagram with Legend

2.3.2 Sensor Block

The sensor block contains all of the sensors that will be used to derive a probabilistic representation of the current state of the robot in the maze. We will be using a camera and rangefinder/depth sensors as the primary information sources. The image data will be used to identify features of the maze that can be correlated to previously knowledge of the environment. It will also be used to identify survivor markings, goal positions, and the object to be manipulated. The depth measurements provide help with localization and help to inform the decisions about driving the robot. The robot will also be equipped with pressure sensors about the body to detect collisions with the maze environment. The camera block requirements are as follows,

1. The camera should provide RGB color channels.

2. The camera should have a field of view greater than forty-five degrees.

3. The camera should be able to communicate with the Raspberry Pi.

The rangefinder block requirements are as follows,

1. The rangefinder must provide measurements up to two meters away, with accuracy of two millimeters.

2. The rangefinder must operate at 5V.

The pressure sensor block requirements are as follows,

1. The pressure sensors must accurately detect impact with the environment with 95% accuracy.

2.3.3 Motor Block

The motor block consists of the motors themselves, along with the encoders and drivers. This is the entirety of the locomotive capabilities of the robot. It will be fed control signal from the control block, but will translate that information into the physical movement necessary to navigate the maze environment. The motor drivers must provide the correct voltage across the motors and protect them from overdrawing current and the encoders must provide the controller with accurate information about the position and speed of the motor.
The motor block requirements are as follows,

1. The motors should be able to drive the robot at $0.3m/s \pm 10\%$.

2. The motors should be protected from over-drawing current.

3. The motor encoders should be able to provide at least $0.5° \pm 10\%$ precision.

4. The motors must be able to be driven in both directions.

2.3.4 Power Block

The power block consists of the battery and voltage regulators necessary to power all the components onboard the robot.
The power block requirements are as follows,

1. The 6V battery should provide at least 15 minutes of power at with all peripherals on and motors operating at constant speed of $0.15m/s$.

2. The 3.3 V voltage regulator should provide 3.3 V $\pm5\%$ from a 6 V power source.

3. The 5 V voltage regulator should provide 5 V $\pm5\%$ from a 6 V power source.

2.3.5 Manipulator Block

The manipulator block consists of two servos that will be used to grasp and lift the goal object in the maze. Their actuation will be controlled by another block and their power is also provided by another block.

The manipulator block requirements are as follows,

1. The grasping servo must provide enough torque to grasp the goal object.

2. The grasping servo must operate at 6V input voltage.

3. The lifting servo must be able to provide a constant $1.2N \cdot m$ of torque to lift a $0.1kg$ object to the goal position.

4. The lifting servo must operate at 6V input voltage.

2.3.6 UI Block

In order to visualize the state of the robot and its decision making process, we have decided to send back state information that can be presented on a groundstation computer. This state and decision information can be presented alongside useful visuals such as optimal navigation decisions at each point of the maze for different objectives and overhead views of the maze environment.

Since this block only a groundstation computer, the block requirements are simplified to needing a Wi-Fi enabled computer than can recieve and display information from the Raspberry Pi and activate the robot's maze exploration over SSH protocol.

2.4 Risk Analysis

The block that poses the largest challenge to the successful completion of the robot is going to be the Control Block. The challenge arises because of the algorithmic complexity involved with creating the reinforcement learning system on the Raspberry Pi. We will need to identify an optimal policy that can be used to successfully navigate real mazes and train the system over thousands of iterations to train it. We will have to train the system in simulation in order to have any chance of successfully training the robot. We will not be able to sit through thousands of physical trial in real-time. There are also uncertainties that are involved with the parameters learned in simulation that may cause different functionality than what is required in the physical space. We will also need to determine the density of visual information that is required for the robot to be able to place itself within the environment. There exists a large amount of uncertainty in the optimal strategy for solving this problem that will require us to attempt many solutions in order to find the one that works best.

3 Ethics and Safety

3.1 Ethics

We believe that our project is aligned with the first tenet of the IEEE Code of Ethics, "to hold paramount the safety, health, and welfare of the public," [2] because our project is designed to help move robotic understanding of real world systems towards the ability to save lives. We strive to use the understanding of intelligent systems to benefit the public good. This leads to the importance of #5 of the Code, "to improve the understanding by individuals and society of the capabilities and societal implications of conventional and emerging technologies, including intelligent systems" [2] in that it will be our duty to inform the public about the beneficial uses of the technology that we are working with and how they can be further used to help society.

Since the success of the project is directly dependent on the functionality of the reinforcement learning algorithm, it is very important that we accurately report our results, regardless of the

outcome. Inconsistent data and unreliable reporting would violate #3 of the Code [2] and would negatively impact the field of robotics research and our character as engineers. In the same vein, it is very important that we give appropriate credit for the previous works that we use and build on to develop our system. It would be unethical to take credit for the work of others in accordance with #7 of the Code [2]. We will be using and learning from many different research sources as well as from our peers and faculty members as we progress through this project and need to accurately present the chain of knowledge and development.

3.2   Safety

The major safety consideration of this project resides in the safe operation and storage of the battery. NiMH batteries are sold as AA type batteries and so can easily be removed from the robot, stored, and charged. They do generate heat during high-load discharge, which we must monitor throughout robot operation. We can use a standard reusable battery wall charger to charge the batteries that removes the need for a specialized circuit to charge the battery. We will, however, still be careful to not work alone while batteries are in operation. Since our group has little experience with building and designing circuits, we will have to be especially careful when designing and testing our custom printed PCB that contains the MCU, motor drivers, and voltage regulators. Voltage regulators can dissapate a lot of heat as well, so we must ensure that appropriate heat dissapation is provided to the circuits and other components. Short circuits, fires, and electrication are all possible safety hazards when working with these materials, so we will take standard lab safety precautions, as well as asking for input from the course staff and other experienced personnel.

References

[1]   Leslie D'Monte. *5 Disaster Robots That May Rescue You From Natural Disasters. The Defense Advanced Research Projects Agency is conducting a global competition to design robots that can perform dangerous rescue work.* 2015. URL: `http://www.govtech.com/em/safety/5-Robots-That-May-Rescue-You-From-Natural-Disasters.html`.

[2]   IEEE. *IEEE Code of Ethics.* 2017. URL: `https://www.ieee.org/about/corporate/governance/p7-8.html`.

[3]   Sergey Levine et al. "End-to-End Training of Deep Visuomotor Policies". In: *CoRR* abs/1504.00702 (2015). arXiv: `1504.00702`. URL: `http://arxiv.org/abs/1504.00702`.

[4]   Piotr Mirowski et al. "Learning to Navigate in Complex Environments". In: *CoRR* abs/1611.03673 (2016). arXiv: `1611.03673`. URL: `http://arxiv.org/abs/1611.03673`.

[5]   Volodymyr Mnih et al. "Asynchronous Methods for Deep Reinforcement Learning". In: *CoRR* abs/1602.01783 (2016). arXiv: `1602.01783`. URL: `http://arxiv.org/abs/1602.01783`.

[6]   Dan Nosowitz. *Meet Japan's Earthquake Search-and-Rescue Robots. The combination of vulnerability to earthquakes and a natural affinity for robotics has led to a surplus of Japanese rescue robots.* 2011. URL: `https://www.popsci.com/technology/article/2011-03/six-robots-could-shape-future-earthquake-search-and-rescue`.

[7]   Lerrel Pinto et al. "Asymmetric Actor Critic for Image-Based Robot Learning". In: *CoRR* abs/1710.06542 (2017). arXiv: `1710.06542`. URL: `http://arxiv.org/abs/1710.06542`.