

Unmanned Ground Vehicle IED and Mine Detector

Javier F. Palomo, Ronald L. Hanifen,
Hernan D. Carvajal, Joshua A. Genao

Dept. of Electrical Engineering and Computer
Science, University of Central Florida, Orlando,
Florida, 32816-2450

Abstract — Current mine removal methods are risky and dangerous. It is significantly advantageous to have a robot perform the traversal of mine fields to not only find and record the location of mines, but to create a clear path that will allow traversal of a minefield with significantly less danger. That is the aim for this project, using the Robot Operating System (ROS) to create a semi-autonomous robot that will scour land for mines and create a clear path. Doing so will incorporate many aspects of Electrical and Computer Engineering, and will provide a beneficial robot for the detection of mines also known as the Unmanned Ground Vehicle Mine Detector.

Index Terms — *Servomotors, Embedded Software, Laser Radar, DC Motors, Wireless LAN, Robots, Robot vision systems, Simultaneous localization and Mapping, Telerobotics*

I. INTRODUCTION

Most people in the United States don't have to worry about stepping on a landmine or triggering an Improvised Explosive Device (IED) when they go about their personal lives; however, in many other countries around the world, this is a significant concern. Around the world, there are about 110,000,000 active landmines, waiting to be stepped on. According to the UN Mine Action Service, landmines kill 15,000-20,000 people every year (mostly children) and many countless more. Furthermore, the use of Improvised Explosive Devices has become a preferred method of engagement in guerrilla warfare.

At the current rate of clearing mines, it will take humans thousands of years to clear all active landmines in the world, and countless more detecting for buried Unexploited Ordnance (UXO) that can be used for IEDs. There is an overbearing concern for land mines in many other countries, and this system will be designed to combat these problems. That is where the motivation for our Senior Design project resides.

The goal for this project is to create an autonomous vehicle that will be able to start at an established starting

point and end up at an established end point, determining a clear path along the way.

II. REQUIREMENTS

To meet our goal and objectives set out at the beginning of senior design, we set a list of goals and objectives that we must meet to find our project suitable for its application. First, we will need a platform to operate semi-autonomously since the robot itself will be unmanned. Also, the platform will need to be powered by an on board system, since the application of this project is to be able to navigate freely to carry out its primary objective to detect any sort of metal or IED. Building on the power system, it is essential that we are able to operate the platform for more than an hour, if not the application will not be a viable solution and will not be able to produce an effective and consistent output.

III. DESIGN METHODS

The procedure in designing out this project is carried out by allocating related work to the members of the group. Since the project consists of mainly electrical hardware and software, we divided the work accordingly. Both computer engineers, Josh and Ron worked on the software implementation, and Javier and Hernan working on the electrical hardware.

This method proved to be efficient and allowed us to have an equal amount of responsibilities. Week in and week out, meetings were held to ensure that all sections were on pace to reach our prototype deadline.

Throughout this process test procedures and recorded data, provided the necessary information to help us debug and resolve issues. Overall the procedures carried out enabled the team to pave a path for success.

IV. SYSTEM DESIGN AND INTEGRATION

A. System Overview

The UGV Mine Detector contains two main components: the Main Control Unit and the mobile robot. The mobile robot contains multiple sensors that are used to move semi-autonomously through a given waypoint. All of the data received from the sensors are sent via wirelessly to the main control unit in order to be processed.

Initially the user is able to specify the waypoint in which it will travel by selecting the area on the map found on the Main Control Unit. When the waypoints are

selected, it will signal the mobile robot to move accordingly.

Data is sent via wireless communication in which is processed through the main control unit. The sensors included on the UGV are: IMU, LIDAR system, GPS, and a metal detector.

Once processed, the main control unit will determine the next action based off of the sensors and preprogrammed algorithms. This will repeat until a clear mine-free path is found and destination is reached.

B. BeagleBone Black

The BeagleBone Black will be primarily used as a sensor measurement device. In this case, the BeagleBone Black is running Ubuntu 14.04 Trusty armhf version. This version of Debian was chosen as it is the most widely supported in terms of ROS and the arm processor community in general. The BeagleBone Black will be directly connected to a wireless router via Ethernet connection that will allow the information from the sensors to be sent over the personal wireless network to a host computer that will be performing the bulk of the calculations. This information will then be output from the host computer in the form of movement messages. The BeagleBone Black will interpret these messages and then send instructions to the motor controller; and, consequently, the servomotors that will move the robot.

With limited computational ability, the BeagleBone Black should not be used as the primary source of navigation computation. This is why the utilization of ROS to create a modular ROS network is so beneficial. There is no need to use the BeagleBone Black as anything other than a sensor input/output device because of this. Otherwise, the amount of processing power would extremely limit the robots ability to make decisions quickly. This isn't necessarily something that is bad for all cases, but would definitely limit the robots ability to move fluidly. That fluid movement is extremely important in the implementation of this robot, because it is dealing with a highly time dependent scenario. If the robot does not stop moving when a mine is found, it triggers the mine. This defeats the entire purpose of this project.

To extend the usability of the BeagleBone Black and allow serial USB connection (this is not provided by the USB port on the BeagleBone Black by default), the use of a USB serial hub will be used. This hub is to be powered separately from the BeagleBone Black as it needs its own 5 V/2.5 A power source. This USB hub will be used to interface with the RS232-to-USB cord that goes to the SICK LIDAR device.

C. Micro Controller

In this project the selected microcontroller is the ATmega328. This microcontroller was chosen because it was the most compatible application when communicating with the Beagle Bone Black. Also, the system does not require much processing power, nor memory space, for the MCU and Beagle Bone Black will be computing all the top level algorithms and ROS environment. Another factor is that the ATmega328 is widely used microcontroller, so there are ample amounts of resources and sample codes, which provided assistance when issues with integration of the system.

Technically the ATmega328 allows the user to communicate to the Beagle Bone Black, through serial communication. The micro controller will not only be receiving signals from the Beagle Bone in terms of directional movement, but will also be transmitting serial information that is obtained through an encoder from the motors. The UART lines will take the information from the two DC Motors and metal detector, which are transmitted to the Beagle Bone to interpret and execute future instructions to be carried out by the micro controller.

Pulse width modulation (PWM) is another important attribute that the micro controller utilizes. The H-bridge will serve as the bi-directional controller based off of the digital signals being sent from the ATmega328, but the power and speed of the motors will be controlled by these PWM's. The microcontroller has six pins that can support pulse width modulation, which is more than enough to control the duty cycle of the two DC Motors, as well as the servo motor that will control the pan of the metal detection.

D. Motor Controller

For the project the platform is controlled through a custom designed motor controller, which, consists of an ATmega328 microcontroller, and an L298P H-bridge. By using the ATmega328 microcontroller the system is able to save processing power that would otherwise be allocated to the Beagle Bone Black. Therefore the only function that the Beagle Bone Black will have to execute, relative to the motor controls, is send a set of signals to the motor controller to carry out its PWM for each independent motor as displayed in Figure 1.

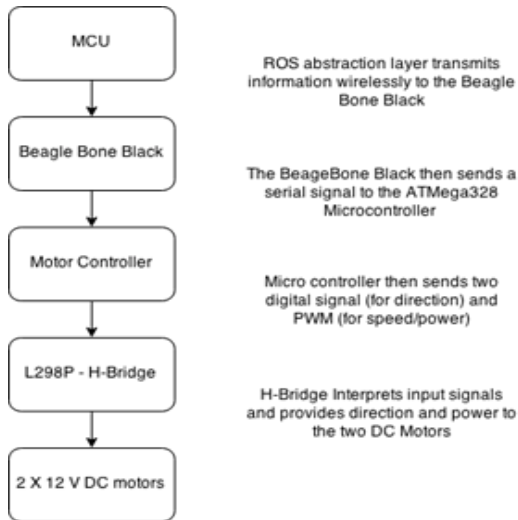


Figure 1

The L298P h-bridge is taking in three inputs from the micro controller for each motor. These inputs consist of three signals: one pulse width modulation line, and two digital signals that will control the directional movements (using high (1) and low (0) signals), as shown below in Table 1. The PWM signal is used as a speed controller using a duty cycle concept. Since the signal is 8-bit, the PWM values can range from 0 to 255, with 0 running at 0% and 255 running at 100%. For our application the motors will be running a relatively high duty cycle, not for speed, but because the platform requires a high amount of torque to push the heavy payload at approximately 20 Kgs.

Direction	Digital Bit 0	Digital Bit 1
Forward	1	0
Reverse	0	1
Stop	0	0

Table 1: H-Bridge Functions

The L298P h-bridge, has a power supply rating up to 50 V 2A (used to power the motors), and up to 7 V logic supply voltage (used to power the logic operations). This h-bridge is a dual full-bridge driver that allows the user to control both motors at once. The motors will run at 12 V and 1.4 A the motors maximum rating, which is well within the output limits for the h-bridge.

E. DC Motors and Specifications

In order for the platform to meet the large payload capacity a Faulhaber 12 VDC motor was selected. The motor runs at a load current of 1.4 A and has a 64:1 gearbox ratio that can support 7,000 rpm, and 16 mNm.

Another, reason for this motor selection is that it already has an encoder on the motor itself, which will provide the necessary motor movements that the ROS environment and IMU require to provide accurate results for out odometry.

F. Printed Circuit Boards

The project consists of two printed circuit boards (PCB): Motor Controller (PCB1), and the Power Regulation (PCB2). Both PCBs are two layer boards and were printed by OSH Park.

The first PCB is the Motor Controller and as shown in Figure 2 contains the ATmega328, and the L298P dual full-bridge drive. The input voltage for this PCB is 12 V and it has an on board 5V and 3.3V regulator that will allow it to power the IC's on the board. The system will utilize this 5V signal to power the servomotor that will serve as the pan for the metal detector.

As seen there are three sets of headers both female and male. These extra headers were added so that we can manipulate the input signals that will go from the microcontroller to the inputs of the h-bridge itself. Also, we needed to have an extra set of headers to transmit the encoder's information to be able to receive the information from the motor.

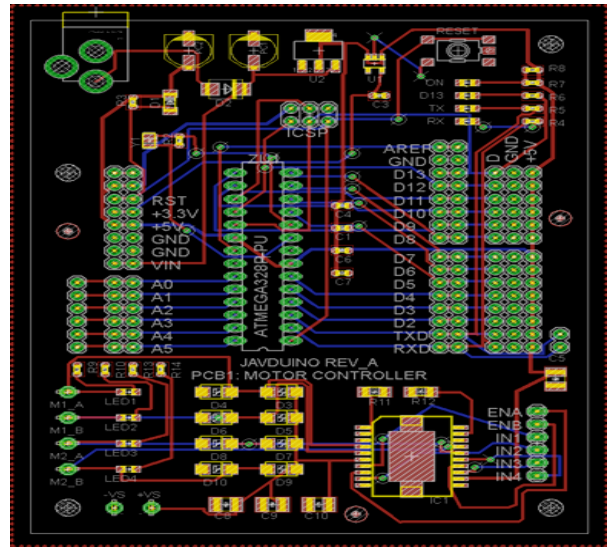


Figure 2 : Motor Controller

The second PCB is the power regulation circuit as seen in Figure 3. The power regulation circuit is powered through two sources: a LiPo 14.8 V 5,000 mAh, and a LiPo 14.8V 10,000 mAh (two 14.8 V 5,000 mAh in parallel). As seen in the power regulation break down (Figure 4), we are regulating the power to 14.8 V, 12 V, and 5 V. We decided to separate the sources since the 15

A current is too large to distribute on a PCB, unless an unreasonably large trace line is used.

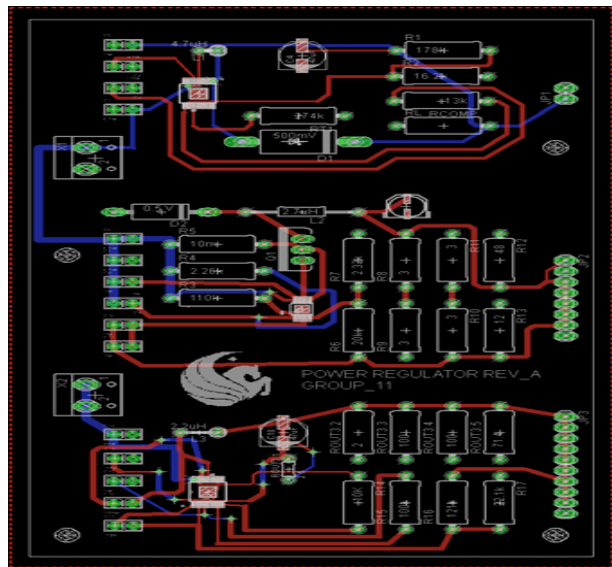


Figure 3: Power Regulation

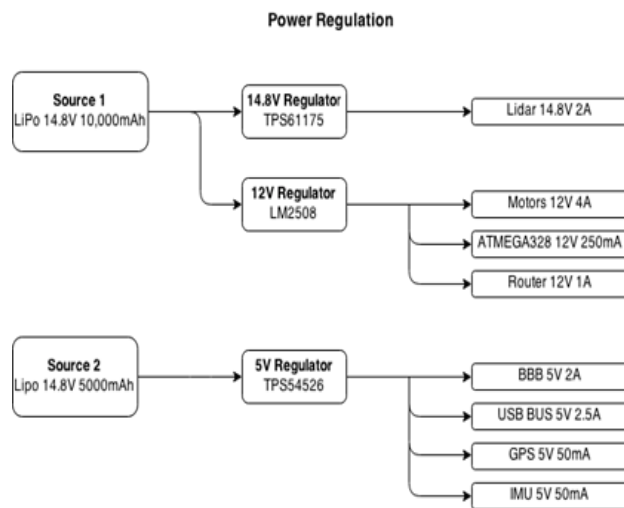


Figure 4 : Power Diagram

To implement an efficient and accurate power system, Texas Instruments Power Architect Web Bench was used for the selection of the most efficient and cost effective regulation circuit. The 14.8 V regulator used in the circuit is the TPS61175, a monolithic switching regulator that can provide an output current up to 3 A, and the desired output voltage of 14.8 V. The 12 V regulator used is the LM25085 PFET buck switching regulator. This regulator is the ideal selection for the 12 V regulation because we

have three loads that add up to approximately 5.250 A, which this regulator can support given the input voltage. Lastly, the 5 V regulator used is the TPS54526 5.5A synchronous step-down converter. Once again this regulator requires a high current load of approximately 4.5 A, with a 5 V output. Given the specifications of our power regulation, the system will be fully functional for approximately 1.5 hours at maximum output.

G. Inertial Measurement Unit (IMU)

A robot that can detect objects is fantastic, but knowing where the robot is in relation to the laser scan information is even more important. An IMU supplies a significant amount of data including but not limited to magnetometer information, gyroscopic information, and accelerometer information. These pieces of information can be fused together with fusion algorithms (RTQF in this case, from richards-tech) to create a position quaternion. This quaternion information is extremely useful as it determines position over time, essentially giving a 3D vector a fourth dimension. The IMU also provides helpful transform information that will allow any sensor information to be related to the base of the robot so the LIDAR and other sensors will work accurately together.

The BeagleBone Black receives this information directly to the I2C bus, because this is the type of serial communication that was chosen for the IMU. This information will be sent from the BeagleBone Black over the wireless network to the MCU as all other sensor information is sent in the form of a navigation odometry ROS message (or nav_msgs - Odometry.msg as ROS understands). From there, the navigation stack will determine position in terms of local and global maps which will help aide in the avoidance of obstacles and clear path creation.

H. GPS

The GPS device will primarily be used as a device to record and display the location of found mines to the user via the MCU. However, it will also aide in the functionality of navigation as well. The GPS has the ability to compute velocity, and will be able to aide in the calculation of an accurate velocity based on both the IMU and GPS

With the GPS velocity we will be able to take a sample of both the velocity readings from the IMU and the GPS and combine them with a weighted average, to get a more accurate representation of the current velocity. Accelerometers are notoriously inaccurate when calculating discrete velocity, because it is relatively difficult to sample acceleration properly and get accurate readings. That's why it makes sense to use mostly velocity

information from the GPS (at a weight of about 75%) and combine that with the discrete velocity calculation from the accelerometer (at a weight of about 25%).

I. Laser-Range Finder

Since the robot will be used outdoors, the SICK LMS-200 is used. Although this laser range finder is not ideal for outdoor use, this selection was considered since most LIDAR systems are costly and provided the best area monitoring than any of the laser measurements compared. With a field of vision of 180 degrees and a range of up to 30 m, the SICK LMS200 provides a good coverage for obstacle avoidance

A fan-shaped scan is made when the pulsed laser beams deflected by an internal rotating mirror. The outline of the target objects is determined from receiving the impulses of laser data. Measurement data is sent real-time using the data interface. The LMS-200 measures the time of flight of the laser pulse that is emitted from the system and returns back. The impulse of the laser and the time is directly proportional to the distance between the LMS-200 system and the object. Real-time measurement data is available for further evaluation. The measurement data corresponding to the scanned environment is outputted into binary format.

The BeagleBone Black located on the UGV receives the data through RS232 to USB in which is later sent through wireless communication to the main control unit

J. Plotting LIDAR Scan

The MCU will receive a constant feed of the SICK LMS-200 laser scans. In order to show the user operating the robot on the standalone computer of the constant feed of data, the LIDAR scan is displayed using ROS visualization tool (RVIZ). RVIZ is a 3D visualization package for displaying sensor data and state information from ROS. This visualization tool will show the live representation of sensor values from the ROS LIDAR topic. This will allow the user to determine if there are any obstacles in the UGV set path.

K. Simultaneous Localization and Mapping

With the SICK LMS-200 and the odometry data, the unmanned ground vehicle is able to provide 2D simultaneous localization and mapping. This allows the mobile robot to build a map of the environment and compute its location relative to the map. While in motion, the unmanned robot dynamically updates the position by comparing key features of the map.

The SLAM algorithm relies on Bayes filter to go through the prediction and the update phase of SLAM. The prediction phase estimates the state space from the

previous scan, while the update phase combines the observations from the sensors.

L. Navigation Stack

The navigation stack is a crucial element to the success of the robot. Without the navigation stack, the robot knows what's directly in front of it and how fast it is going in a local scale, but not in the global scale. This is to help with path planning and decision-making based on what the robot has already seen and not just what it is currently seeing. This is not the only operation of the navigation stack, however. It also provides costmap information that will help avoid obstacles. Costmaps are essentially a map of the area around the robot, and determines whether or not an area has been explored, hasn't been explored, or is to be avoided.

When deciding whether or not it is possible to move forward towards the goal, the robot must decide a few things. Without any information, the robot will move in a straight line towards its goal. That is, until it runs into something that it can't move past. There is a list of things that this could be, and are determined through one of two sources: LIDAR or metal detector.

When the LIDAR detects an obstacle, it will overlay that information to the obstacle avoidance costmap layer and immediately try to find an alternate path. When the metal detector detects a mine, it will insert that obstacle information into mine costmap layer to avoid the mine immediately.

The navigation stack is primarily how goals will be sent to the robot. There will be a visualization of the costmap through RVIZ that will show the entire unexplored map that the robot will traverse. Simply by knowing its starting position, and clicking on a position goal on RVIZ, the robot will happily strut through the motions avoiding obstacles in real time while trying its hardest to get to the final destination.

M. Kinematics

The UGV is equipped with a differentially steered driving system. The mobile robot is based on two separately driven tracks placed on either side of the robot. It is able to change its direction by varying the relative rate of rotation of its wheels and therefore not requiring any additional steering mechanisms. If one of the tracks rotates faster than the other, the UGV will follow a curved path moving inwards towards the slower track. In order to pivot in the same location, the robot is able to set one of the tracks in a forward direction while the other is set in the opposite direction. A diagram depicting the UGV movement is shown below in Figure 5.

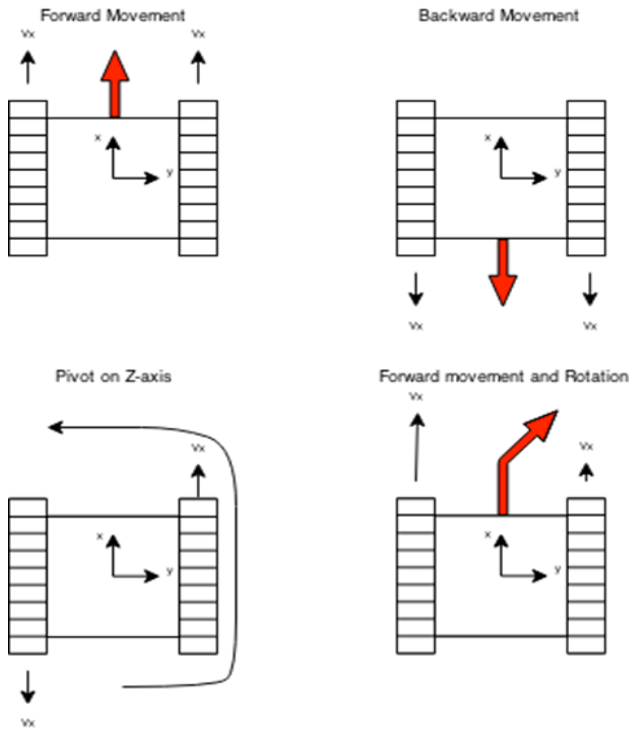


Figure 5: Example of the UGV Kinematics. The mobile robot receives linear and angular velocity to move from location to the other.

N. Robot Operating System Integration

ROS uses a variety of different packages to enable differentiable functionality, but there are a few packages that remain important regardless of what packages are being used/created. Behind every major package is the `roscpp` or `rospy` package that provides the backbone for any package. The two are different only in language. If a node is to be written in python, the `rospy` package will be in effect. The same effect will occur with C++ but with the `roscpp` package being the backbone instead.

Behind every package, a message can either be created or received. The management of these messages is done through a standard package called `rostopic`. This package views/publishes/receives information related to any topic posted through the master node. Each topic can be dynamically created and is created under the strict guidelines of `actionlib`. `Actionlib` is a package that manages the transportation of messages across a topic, and was built on top of ROS messages. `Actionlib` is the backbone behind the TCP/IP connection that is used in wireless communication.

O. Detection System

When deciding on the detection system for the vehicle, multiple systems such as Ground Penetrating Radar

(GPR), ultrasound sensors, infrared imaging, and metal detection, were considered, however, after a great deal of research, the team decided to go with the metal detection system, it is less expensive, obtains more consistent results in various types of weather and terrain, does not require human analysis of data, making it easier to implement in an autonomous system, and it is easier to build.

Our main concern for the metal detection circuit is to be able to send a digital signal to the main board which would be used as a flag to immediately stop the vehicle, and let it know that an IED was detected. Multiple prototypes of different types of metal detection were built. However, due to the complexity and accuracy of some of them, the team decided to go with a metal detector based on the Beat Frequency Oscillator (BFO).

A BFO metal detector works by using a larger coil in the detector head, and a smaller coil close to the circuit. Making a slight offset between the frequencies of the two coils, when a metal is present, will offset increases giving the user an audible signal. For our circuit the team decided to detect this changes in the frequency by using a microcontroller instead, giving us the ability to easily change the sensitivity of the metal detector, through the code. This also makes it easier to get a digital signal output to be sent as a flag to the main board.

This circuit works by using a simple oscillator circuit to send a pulse frequency through the coil, this pulse generates a magnetic field around the coils. If a metal is present, this magnetic field will create a very small current flow through the metal, also known as Eddy current. Consequently, this creates an opposing magnetic field, changing the voltage drop through the coil and slightly changing the frequency as well. This output from the coil is then sent to a microcontroller, in this case an Arduino Uno, which takes in an initial frequency without any metal present and then detects any change in this frequency. This is done by using a null switch connected to a digital pin, to manually set the base frequency once the circuit is first powered up, then the output signal is sent to PWM Pin 5 of the Arduino, which is internally connected to an external timer/counter. This counter will then compare any deviation from the base frequency to the output frequency from the coil, and in the case there is a change due to a metal present, an output is sent to a piezo sensor to make an audible notification of a metal present as shown in Figure 6 below.

low center of gravity, preventing it from flipping backwards due to the heavy weight on the top.

V. CONCLUSION

Overall the project as a whole proved to be a success, given the circumstances and constraints battled throughout the semester. From the metal detector to the main control unit user interface, the project challenged us due to its vast range of requirements and complexities.

The group was able to implement and successfully satisfy all requirements and objectives sought out from senior design 1. The metal detections reliability, the LIDAR's consistent ability to avoid obstructions, and the MCUs ability to integrate all sensors and systems together is more than enough evidence to deem this project a success. Not just in our academic careers, but as future engineers this project was a giant step toward success.

VII. ACKNOWLEDGEMENT

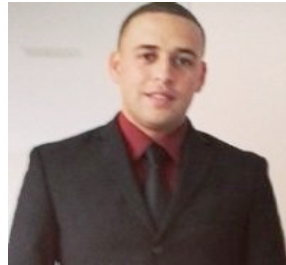
UCF EECS Spring 2015 Senior Design Group 11 would acknowledge and thank the University of Central Florida, E-Z Claims, Inc. for donating the platform, and Dr. Richie for providing guidance throughout the semester.

VIII. REFERENCES

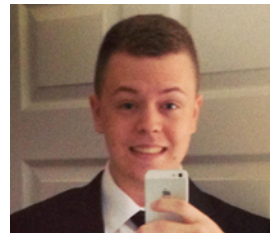
- [1] "LMS200/211/221/291 Laser Measurement Systems." (n.d.): n. pag. Dec. 2006. Web.
- [2] Siegwart, Roland, Illah Reza Nourbakhsh, and Davide Scaramuzza. Introduction to Autonomous Mobile Robots. Cambridge, MA: MIT, 2011. Print.
- [3] S. Huang, G. Dissanayake. Convergence and Consistency Analysis for Extended Kalman Filter Based SLAM, In IEEE Trans. on Robotics, 2(5), Oct. 2007.
- [4] M. Quigley et al., ROS: an open-source Robot Operating System. In IEEE International Conference on Robotics and Automation(ICRA), Workshop on Open Source Software, 2009.

IX. GROUP MEMBERS

Hernan Carvajal is graduating from the University of



Central Florida in August of 2015 with a Bachelors of Science in Electrical Engineering, upon graduation Hernan will be working full time at NAVSEA Norfolk VA.



Ronald Hanifen is currently a senior at the University of Central Florida and will receive his Bachelor's of Science in Computer Engineering in May of 2015. He is currently working at Lockheed Martin through the CWEP program at UCF. He has attended the University of Central Florida. for four years now and plans to work for Lockheed Martin after graduation.

Joshua Genao is graduating from the University of



Central Florida with a Bachelors of Science in Electrical and Computer Engineering. During his time at UCF, Joshua held an internship at L-3 Communications and Northrop Grumman. After graduation, Joshua will be working at Northrop

Grumman as a Systems Engineer.

Javier Palomo is graduating from the University of Central



Florida with a Bachelor of Science in Electrical Engineering. At the University of Central Florida Javier was a College Work Experience Participant and upon graduation will begin a full-time position at for Texas Instruments.