Robotic Tour Guide Assistant

Amy Elmer, Tyler Gutridge, and Gabe Kruse
Ohio Northern University, Ada, Ohio 45810
Email: a-elmer@onu.edu, t-gutridge@onu.edu, g-kruse@onu.edu

I. Abstract:
The Ohio Northern University T.J. Smull College of Engineering gives a large number of tours to prospective students each year. To assist in these tours, as well as spark interest in prospective students, a request was made by the college for an autonomous tour guide assistant. The overall project goal, started in 2011, was to develop a fully autonomous robot that would provide support in the many tours given each year as both a visual aid and material storage unit. In the past year an autonomous robot was built that had the potential to track a tour guide, wearing a LED pendant, while avoiding obstacles. The current goal for this year is to finish the software algorithm development of this LED Ensuing Robot that is Obstacle Yielding (LEROY).

The learning experience the autonomous robot provides to all it encounters is invaluable. The teams that have been part of the development have gained work force experience and the knowledge to learn with minimal guidance. Ohio Northern University's Engineering College gains value from LEROY by gaining the ability to supplement the information provided during the tour. LEROY also provides value to prospective students by demonstrating the range of opportunities available to students of Ohio Northern University's T.J. Smull College of Engineering.

II. Introduction:
The Ohio Northern University T.J. Smull College of Engineering made a request in 2011 to a senior design team to develop a fully autonomous robot to accompany a tour guide. The autonomous robot is desired to assist in tours and spark interest in prospective students by being a visual aid and material storage unit. The assignment for the 2012-2013 team was to develop the algorithms needed for the robot's navigation and to incorporate these algorithms into a previously built robotic system.

An autonomous robot must be able to perform three fundamental abilities. The robot must have a method of movement, the logic to solve the location problem and the capability to avoid obstacles. The location problem can be divided into two separate levels of tracking, global and local tracking. The concept of global tracking is to know where the robot is located within a predetermined mapped area. Localized tracking uses the found global position and updates this position over time through feedback systems. This however is difficult to do because of imperfections in the other fundamental abilities of the system such as motor slippage and unforeseen obstacles in the path. The robotic tour guide assistant uses simple and cost effective hardware to solve these problems.

The robotic tour guide was defined to be an assistant on the many tours given at Ohio Northern University meaning that the robot would follow the tour guide throughout the entire tour. In order to do this the robot needs the ability to distinguish the tour guide from those being given the tour and identify his/her position between five and ten feet. To make this position determination as well as separation from the group a pendant will be worn by the tour guide and tracked by a camera mounted on the robot. The pendant will also provide a simple solution to the location problem by eliminating the need for global tracking. The robot will move on a motorized base and use sonar sensors to identify unexpected obstacles. This LED Ensuing Robot that is Obstacle Yielding will henceforth be called LEROY.

The contributions made to LEROY during the 2012-2013 school year include design improvements, algorithm developments and system integration. The remainder of the paper will discuss in detail the system description of LEROY including the decisions behind the design and the advancements made in the robot's development to date and will conclude with the experience gained during the work done on this project.

III. Literature Survey:

Through the research it was found that there were various groups working with autonomous robots. The first ones were fully autonomous robots, meaning they didn't need the assistance of a human to guide or interact with people. One robot researched was TPR-Robina who has been exhibited since 2007 at the Toyota Kaikan Exhibition Hall in Toyota City [1]. Robina uses ultrasonic sensors and a laser range finder to detect and avoid obstacles. She is able to dynamically recreate her path to continue leading tours despite unexpected obstacles. Another robot, Minerva, who was exhibited at the Smithsonian's National Museum of American History, was also able to create maps dynamically through the use of occupancy and textual maps. The occupancy maps were used to determine and predict probable obstacles and how best to avoid those obstacles and the textual maps were used to determine the shortest path that also avoided open areas not marked with features [2]. These two maps combined allowed Minerva to create her maps and determine the best route for the tour on demand. Through the use of Markov localization the robot is able to keep track of its location. Minerva uses lasers, sonars, and cameras to read the input regarding its environment. A third robot found through research was RoboX who was created by BlueBiotics SA and uses two cameras, one to interact with humans and the other for location. A color camera is used to track the face of visitors and through the use of the Intel Image Processing Library follows patches of skin color to maintain contact with a visitor [3]. The second camera is a gray scale camera that is pointed to the ceiling to track the robot's location. Each of these projects was fully autonomous and used location algorithms to dynamically give tours. One of the projects found that better mirrored the work done on Leroy by the Klondike Robotics team was Brown University's autonomous robot. Brown University used an infrared depth camera to follow a person. They were able to follow a person without losing that person amongst others. Through this camera they used gestures as commands to tell the robot what it was expected to do [4].

IV. System Description:

LEROY, pictured in Figure 1, is designed to be an autonomous robot that will follow a tour guide throughout the tours given at Ohio Northern University. The robot can be divided up into four systems, the structure and motion system, the obstacle avoidance system, the vision system, and the computing systems. The structure and motion system is made up of a motorized base and a shelving unit. The obstacle avoidance system is to be programmed and designed to avoid any obstacles that will get in the way of the robot by using an array of sonar sensors mounted on its frame. LEROY will follow the tour guide during the tour using images captured by the vision system consisting of a camera mounted on the top of the robot. The computing systems, containing an onboard laptop and Arduino microprocessor, will process the information received by both the obstacle avoidance and vision systems and output commands to the motorized base.

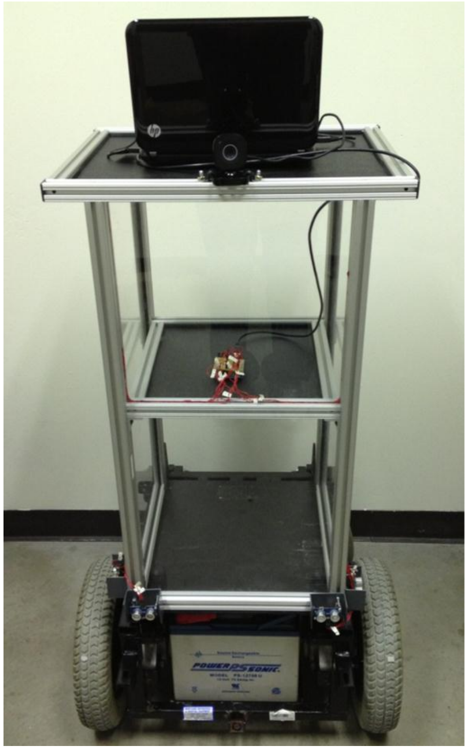

Figure 1: Photo of LEROY

IV-A. Structure and Motion System:

A Jazzy 1122 motorized wheelchair, donated to the college, is used with the seat removed as the motorized base for this project. The wheelchair has four wheels in total, two driven pneumatic wheels and two non-driven non-pneumatic wheels. Each driven wheel is connected to a 24 volt braking DC motor to provide movement of the robot while the non-driven wheels allow turns to be achieved. Two twelve volt batteries are used in series to provide the necessary voltage to operate each open loop motor and disengage their brakes. The wheelchair includes a charging circuit for the batteries run by 120 volts AC.

The shelving unit that further defines the structure of LEROY is built to stand on top of the motorized base. Aluminum framework was used with Plexiglas walls to create a rectangular prism. The walls of the unit encase 3 out of 4 sides of the cart leaving an open space for material storage. The cart has three plastic shelves for storage of additional items needed by the tour guide, the top of the cart, a middle shelf, and the top of the motorized base. These features can be found in the image in Figure 1.

IV-B. Obstacle Avoidance System:

The Ultrasonic Range Finder sonar sensors are used for obstacle detection and are mounted at the bottom of the shelving unit as shown in Figure 2. The five sonar sensors are powered from a

5 volt DC source provided by the Arduino microcontroller and have a range of 0.03-3.0 meters with a precision of ±0.02 meters. Sensors 1 and 3 are used to detect objects in front of LEROY while sensors 4 and 5 determine if there are any objects on the side. Sensor 2 is used to detect when LEROY is approaching a descending staircase.

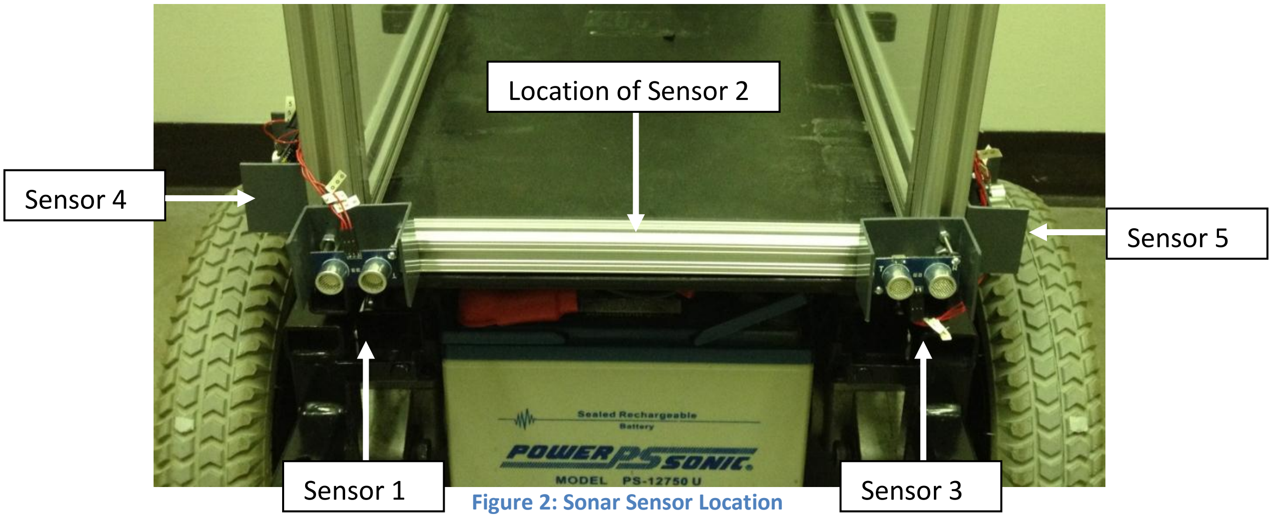

Figure 2: Sonar Sensor Location

IV-C. Vision System:

LEROY vision system uses a Microsoft webcam, Model 1407, with 30 frames per second to track an object that the user will wear. The object being tracked is a circular pendant four inches in diameter. The pendant is yellow and is a modification upon the design used by the previous team. The images taken by the webcam are transmitted to a laptop for processing via USB connection. This USB connection also provides the necessary power for the webcam to operate. The camera is mounted at the top of LEROY's shelving unit.

IV-D. Computing Systems:

Two different programming devices are used on LEROY to perform the computations needed to track the LED pendant, a laptop and a microcontroller. The laptop used on LEROY is a HP Mini 1104 netbook with 2GB of ram and Intel Atom CPU N2600 processor that runs at 1.6 GHz. The netbook is in charge of processing the images taken from the camera, locating the pendant, and sending data to the Arduino microcontroller for motor control over a USB connection. Using a C# program on the netbook the pendant is located using a Euclidean filter combined with blob detection on the image. Based on the width in pixels of the pendant the distance of the pendant from the camera is calculated. A command is then sent to the Arduino Duemilanove microcontroller. The Arduino interprets this command sent by the netbook along with signals sent from the sonar sensors and outputs the appropriate command to the motor interface with the sonar inputs taking priority.

V. Advancements:

The tasks assigned to this year's autonomous tour guide robot team were to make improvements on the current components and develop and implement the tracking and following algorithms. The components being used at that time included a motorized Jazzy 1122 wheelchair base with its original motor controller, a shelving unit, an Arduino controller, four sonar sensors, a netbook, a pendant with five LED lights attached, and a Microsoft webcam. Testing had to be done on the four different systems of LEROY to ensure required capabilities. Upon completion of the testing it was determined whether or not modifications were needed in order to accomplish the requested task. If modifications were made additional testing was done to ensure that the changes produced the desired outcomes. Throughout this process the implementation and intercommunication of the structure and motion system, obstacle avoidance system, vision system and computing systems were accomplished.

V-A. Structure and Motion System:

The wheelchair base was chosen due to the accessibility of the controller and its ability to carry the payload of the shelving unit. After running the robot, during an initial sonar code test, it was found that when given a forward command signal the robot would continually veer off of its commanded path to the right. Tests were run on the motors to determine the rotational speed of each motor. When the wheels were blocked up, a tachometer was used to determine the speeds of each motor, by checking the speed of each wheel during a ten second test period. The results indicated that the right wheel spun at an average of 97.787% of the left wheel. The existing controller being used did not allow individual motor control which could be used to match the speed of the right wheel to the left. Through research and attempts to reverse engineer the wheelchair base controller, it was determined that the best option was to invest in a new controller having the ability of individual motor operations.

The replacement controller chosen was a Sabertooth 2x60, a dual 60 ampere DC motor driver shown in Figure 3. It requires a 6-30 nominal voltage supply and will output 60 amperes continuous current per channel. The dual channel allows for independent motor control of a two DC motor system and provides the ability to compensate for motor wear over time. The Sabertooth controller is capable of operating in four different modes but only the simplified serial mode is useful for this project. The simplified serial mode can be used with either the Arduino or the Laptop as the signal source. This will only use one of the signal inputs on the Sabertooth. The signal, ranging in value from 0-256, will be divided into two portions, the lower half, 1-127, controlling the left motor and the upper half, 128-255, controlling the right motor. A value of zero will command a zero speed to both motors simultaneously.

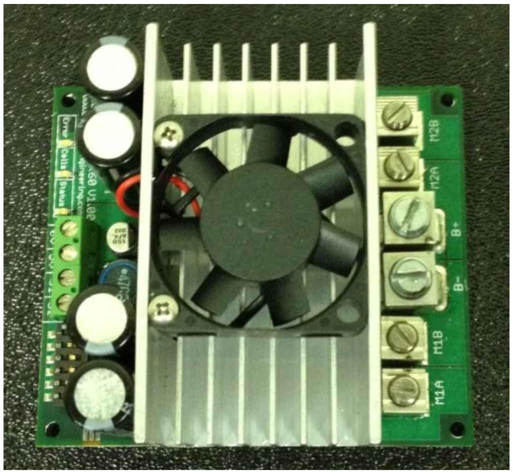

Figure 3: Sabertooth Controller Module

Tests on the Sabertooth were run to determine the percentage decrease needed to match the left motor speed to that of the right motor. The brakes needed a supply of 10 volts DC to disengage them and were connected to one of the 12 volt batteries to achieve this. The Sabertooth was originally connected to the 24 volt source provided by the batteries and each motor, with the speed commands being received over a serial transmission cable from the Arduino controller. Testing was done on each wheel individually to find each motors input voltage levels from the controller and output speeds obtained from specific commanded values. For example the left motor would be commanded a value of 10 while the right motor 0, then 11 and 0, 12 and 0 and so on. This was followed by the right motor being commanded a set of values and the left motor a stop command. These tests showed that the motor speed and voltage levels, while linear for each motor, were offset from each other by a constant value with the right motor being constantly slower than the left.

Due to the fact that the Sabertooth command signals are integer values, the motor speeds cannot be matched up exactly. It was demised that a duty cycle would have to be run in order to adequately match the speeds of the right and left motors. Before the duty cycle was calculated, the supply voltage for the Sabertooth controller was lowered to 12 volts instead of the originally tested 24 volts. This change allowed for smaller step sizes to be achieved for the same commanded signals, which effectively lowered the offset between the two linear equations for the motors. This change limits the maximum speed possible for this system to exactly half the original speed but this was deemed acceptable due to only slower speeds being needed during tours. The test was repeated with the supply voltage changed from 24 volts to 12 volts and the results can be found in Figure 4.

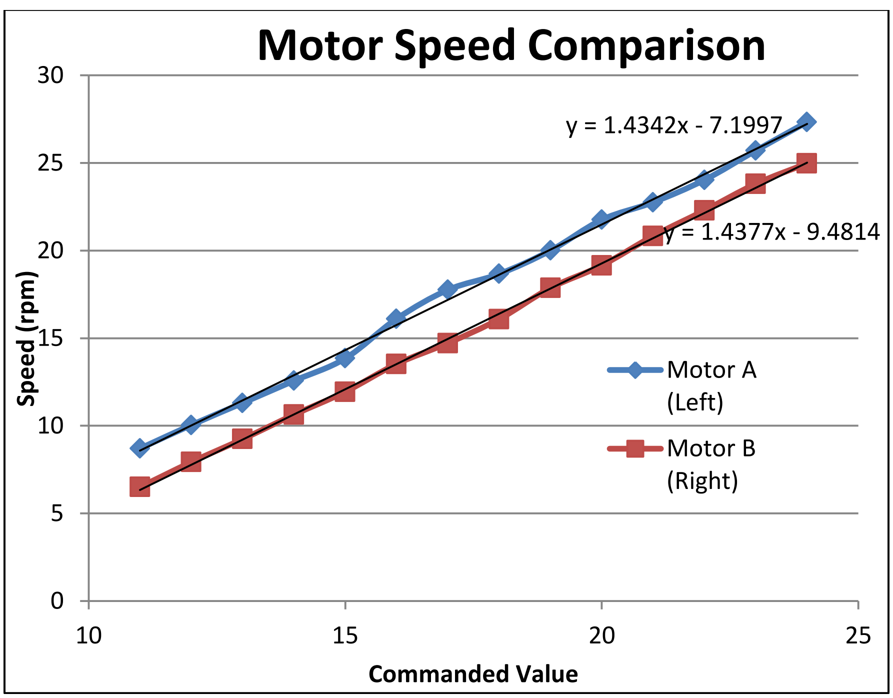

Figure 4: Motor Speed Comparison with 12 volt supply

The duty cycle was found using this data. With the left motor running at commanded value ‘n’, the duty cycle to match the right motor speed is 0.57(n+2) and 0.43(n+1). This duty cycle was tested on LEROY with the wheels being loaded and it was found that this did not hold true. The appropriate duty cycle for LEROY while loaded was then found empirically to be 0.55(n+2) and 0.45(n+1). The corresponding period of this duty cycle was also empirically based on the responsiveness of the motors to the commanded values. This period was found to be 5 seconds.

V-B. Obstacle Avoidance System

Testing was done on the sonar sensors to confirm their accuracy and precision. Each sensor was tested three times at five distances, 3 centimeters, 50 centimeters, 100 centimeters, 200 centimeters, and 300 centimeters. Each of the sensors had an accuracy of 2 centimeters at a distance of 3 centimeters. At the distances of 50, 100, 200 and 300 centimeters the sensors were within 1-2 centimeters of accuracy. However this accuracy is dependent on the object being directly in front of the sensor. Any variance to either side at these measured values would cause the readings to be larger than the actual distance. Despite this variance, these results provided evidence that the accuracy and precision achieved by each of the Ultrasonic Range Finders were sufficient for the application of this project.

Further tests were run to identify the areas that were undetectable by the sonar sensor array. For this test a tall slender wooden pole was used. The pole was placed directly in front of LEROY in between sensors 1 and 3. It was then moved straight out in increments of 5 centimeters until the pole was detected by both sensors 1 and 3. Once the pole was detected it was then moved back towards the robot in 0.5 centimeter increments until it was again undetectable by sensors 1 and 3. This distance of the pole was found to be 29.5 centimeters measured from the front of the frame at the midpoint of sensors 1 and 3. At this distance any movement towards sensor 1 would cause sensor 3 to lose its detection of the pole and vice versa. Even though the actual distance the pole was found to be 29.5 centimeters from the cart, the sensors were giving feedback of readings in the range of 60 centimeters because it was not detected directly in front of each sensor. From this it was determined that the stop distance would have to be 60 centimeters when approaching an object. In order to establish the array of detection the pole was then moved to 35 centimeters. At this distance the pole was shifted to the side of sensor 1 by 1 centimeter increments until it was no longer detectable by sensor 3. This process was repeated shifting the pole towards sensor 3 until sensor 1 could no longer detect it. In both cases the pole was shifted by 4 centimeters before it became undetectable by the appropriate sensor.

As LEROY is to be used on the second floor of the engineering building at Ohio Northern University, a method for detecting stairs as an obstacle was needed. From this the decision was made to install a fifth sonar sensor, Sensor 2 in Figure 3. This sensor is mounted on the front of the frame and directed at a negative angle pointing towards the ground. Unless there is an obstacle that is too close to LEROY or LEROY approaches a set of declining stairs, the value being read by this fifth sensor would be constant. Upon any change a stop command will be immediately issued to the motor controller. Because of the nature of the sound signals produced by the sonar sensors, sensor 2 had to be mounted at an extreme angle in order to attain a constant reading. This angle allowed for a distance of 10 inches or 25.4 centimeters for the stop command to take effect before reaching the detected object or staircase.

V-C. Vision System:
First, the design included a pendant with flashing LED lights; the LED lights were included to make tracking the pendant simpler. The pendant was made up of 5 green LED's against a black background as shown in Figure 5. These LED's are part of a 555 timer blinking circuit, powered by a nine volt battery, which increases the accuracy of the pendant detection by the webcam.

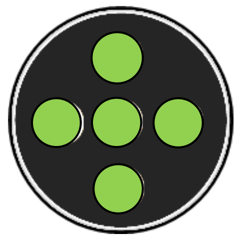

Figure 5: LED pendant

In determining the best method of processing an image from the camera, multiple computer vision methods provided through the .NET framework were considered. The three initial methods compared and contrasted included edge detection, canny edge detection, and stereo vision. The edge detection algorithm analyzes each pixel in the image and determines the difference between each of the eight surrounding pixels, if the difference is greater than the threshold the pixel is set to white otherwise to black. Canny Edge detection is a more detailed form of edge detection. First, a Gaussian filter is applied on the image to reduce noise. Next, the intensity gradient of the image is taken to identify horizontal, vertical and diagonal edges. Then, the rounded gradient angle previously obtained is used to determine a local maximum. Finally, a binary image, consisting of a set of edge points, is attained. Another method that was considered was stereo vision. Stereo vision allows the computer to process the location of objects using stereo triangulation.

Stereo vision was never implemented due to the need for two cameras to perform the triangulation on the image. Another con was the need to still determine how to distinguish the LED pendent from the surrounding environment. Edge detection was implemented in conjunction with changing the background color of the pendent to increase the contrast between the pendent and the LED lights. This was chosen because it was the solution that provided the best image allowing a human to find the LED lights and was believed to avoid wasting computational resources.

The initial plan was to use a Sobel filter and Canny edge detection to determine the position of the tour guide wearing the LED lit pendant. The edge detection provided an image that contained only the edges; however, determining the location of the pendant and tracking the pendant became an issue. Using the edge detection there were still issues in discerning the LED lights. After inverting the image the results improved but the background of the pendent was still too similar to the LED light color to apply a threshold that didn't turn the entire pendent black or the entire picture white. When a black background, made of electrical tape, was used to cover the pendent, leaving the LED lights visible, the results improved and the edge between the LED lights and the pendent were easily discernible. Canny edge detection provided even better results than the edge detection method provided in the .Net framework, allowing the edges of each item and creases in these items to be seen. However, many of the edges in the result were unnecessary and distracting. Many of the edges provided in the resulting image would have added computation time to determine that the edge being addressed was not an edge of one of the LED lights. Through the use of Canny edge detection a lookup table was to be used to determine the distance of the Pendant from the camera based on the circumference of each LED light in pixels as well as the distance of each LED to the surrounding lights also in pixels.

After implementing the methods of edge detection and finding cons with these methods provided in the .Net Framework, research was done on image dilation and erosion. Dilation takes the light pixels in the image and expands them upon the dark pixels, erasing the noise from the picture and enlarging the light colors in the image. With the pendant being covered in black and the LED lights being yellow, the lights become larger despite farther distances. Erosion, inversely, expands the dark pixels in the image upon the lighter pixels. Through this method the LED lights on the pendant are larger making it easier to find in the image. Dilation became a less than ideal solution due to the light in all objects being enhanced, areas in the image with light colored objects became difficult to discern from the pendant LED lights. The other issue that was noticed during these various processes was the rate that the camera grabbed and processed the image made the blinking of the LED lights difficult to detect. The LED lights were found to be blinking at a frequency of 7.47Hz. With the methods being used, the frame rate of the captured images was not able to be specified to match the frequency of the LEDs.

Through further research the framework AForge.Net was found as a viable option. In order to allow for the input video stream to be processed quicker, AForge.Net, which was designed for developers working in computer vision and robotics, was used for the project. AForge.Net is a C# framework designed to enhance computer vision development. The libraries provided through AForge.Net allow for quick processing of video streams. Due to the ability of the framework to quickly process video input streams and the classes and methods provided we will be able to set an ideal frame speed for the camera allowing for the blinking of the LED lights to be detected. The libraries that are needed for the expectations of LEROY are AForge, AForge.Video.DirectShow, and AForge.Controls, AForge.Imaging.Filters, and AForge.Imagaing. Also, through the use of these libraries a combination of filters will be used to detect the LED lights. The ideal solution is to filter the image by color, leaving only the desired colors of the pendant, originally the LED lights, then to track the object through blob detection, which categorizes sections of the image based on characteristics such as color or lighting compared to the surrounding pixels.

First, changing the rate of the LED lights blinking was considered. However, using the libraries provided by AForge.net, the frame rate of the camera was changed to better detect the LEDs. The capabilities of the software require that the frame rate be a whole number, instead of matching the rate of the blinking of the LED lights perfectly, a multiple of the lights rate was used. The frame rate of the camera was set to 15fps, as close to double the speed of the LEDs allowed considering the software restrictions. The size of the LED lights began changing with the corresponding of the frames per second and the rate of the lights; however, the change in size of the light was not easily discernible. In an effort to better detect the LED lights filters were applied to the image. Using the AForge.net framework the image was filtered using a Euclidean filter and a radius on the color wheel. The Euclidean filter takes a specific color value and filters any color that does not match that specific color. The radius on the color wheel is what allows a range of colors to be filtered for. The specific color is a point on the color wheel and the radius is applied around that point. Using this method to filter for yellow/green, circles appeared around the outside of the LED lights but left the center black. After testing color variations and varying the radius size to allow more or less of a variance to the color and seeing little improvement on the visibility of the center of the lights, white was tested due to the knowledge that on the rgb scale white is the presence of light or all the colors. Through this adjustment the LED lights were visible as small white dots. An issue arose due to reflections of light off of objects and the color of the walls being off-white. The radius was made smaller and while this removed some of the noise from the image there was still excessive noise not filtered out. Also, the size of the LED lights became smaller as the radius decreased. Because the lights already become small at distances over 4 feet this would make detecting the pendant at distances over 4 feet difficult.

Despite this setback, progress continued and blob detection was implemented. Blob detection works by looking for variances in the characteristics of the pixels. These variances include color and light. Each blob is stored as a rectangle object through the AForge.net library where the rectangles position is known relative to the control displaying the video steam. Using these rectangle objects, code was added to draw rectangles around each blob. While filtering for white, there were many blobs detected and the processing time increased drastically. In an attempt to find just the pendant, the color being filtered was changed to yellow, the background color of the pendant. The pendant was easily detectable to both the human eye as well as through the software as a blob object, as shown in figure 6. While much of noise from the image was taken care of by changing the color filtered for, there were still a few other small objects detected in the image. Using the blob class provided in the framework, it is possible to set a minimum and maximum size in pixels that the objects detected can be. The minimum size was increased and the only object detected was the pendant.

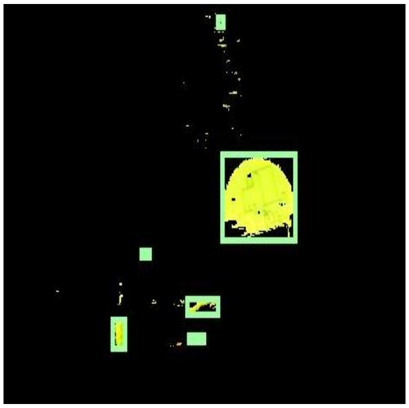

Figure 6: Filtered image with detected blobs

The camera feed was tested with the LED lights on and off and results were better with the lights off. Due to this, the LED lights no longer serve a purpose to this project and consideration is being given to various shapes or lettering on a shirt of specific colors as the object to be detected.

Originally the plan was to use a matrix on the image to track the pendant. This would allow for less computation because once the pendant has been found, the algorithm would start its search in the grid space that the pendant was last located in. Due to the blob object being stored with location relative to the video component on the screen, this searching of the grid is no longer necessary. However, the matrix idea will still be used to determine commands to be sent to the Arduino controller. As the pendant moves to either side, this is analyzed to indicate turning. Commands will then be sent from the netbook to the Arduino based on the pendant or object's location relative to edges of the matrix.

V-D. Computing Systems

The Arduino controller takes inputs from both the sonar sensors and the laptop. The inputs from the sonar sensors are used to determine if an object is too close to the robot. These inputs have the highest priority in the program. This section of the program will start to ping each sonar sensor individually to determine the distance reading from that sensor and returns the data in centimeters. If the front sensors detect an object that is too close, the code will proceed to check if it is safe to turn on either side and maneuver around it. In the instance that the side sensors are also blocked, then the robot will wait and continue to check for a clear path. Lastly, if the sensor that is pointed at a negative angle detects a change greater than 5 centimeters then the robot will come to a complete stop in order to not cause harm to itself.

Upon a clear path for LEROY, the laptop input commands the motor signals that are necessary for the continuous tracking of the pendant. When these signals are retrieved, the Arduino will then go through a switch statement of commands, for example stop, forward, drift left, turn left, etc., and send the appropriate signal to the Sabertooth to make the motors move in that direction. It is important to note that the commands sent to the motors are programmed to have a smooth acceleration and deceleration. These switch statements also directly correspond to a value sent from the laptop. If the signal from the laptop is not being received, the program will tell the robot to stop and wait until the tour guide restarts the program.

VI. Educational Value:

Throughout the work on this project many lessons have been learned contributing to our range of engineering education and future workforce experience. Among these lessons, there are five main concepts that we learned. Two of these, program management and hardware implementation reinforce concepts that have been readily used in the classroom and lab setting of school. The other three, time management, documentation and communication and budgeting skills, have a direct impact on what will be required in future engineering endeavors. Through all the work on this project we not only received these valuable skills but also were able to give back to Ohio Northern University through our achievements.

Program management:

Throughout this project program management has been necessary due to programs being written that need to communicate with each other; this was accomplished by dividing up the program divisions among the team members. One large hurdle that needed to be addressed was figuring out how to communicate between the Arduino and the netbook that was controlling the camera. The complexity of each program segment increased out of necessity of making the connection between the different platforms sound. However, by applying different tactics this communication was allowed to take place. A common communication language between the multiple languages that were being used was necessary. Each language then translates commands to the appropriate action that a language requires of it. An example of this obstacle was the communication from the laptop to the Arduino. The language being used for communication with the camera was C# while the language used with the Arduino is similar to C. Communication between these two languages was overcome by sending the data from the camera code to the Arduino code as a stream of bytes. Specific members had a code portion that he/she worked closest with but that member was still required to have complete understanding of the other divisions of code. A programming style that helped the team members understand each division of code was using modularity.

Having to develop a large-scale software program requires an organization that smaller applications can be managed without. Smaller applications can be managed without separating each file into an appropriate folder. When a real-world application is being developed, code is re-used and made into a library for this re-use. Throughout this project we've had to reconfigure the way we design our code to be more manageable and reusable. To go along with this we still maintain the idea of simplicity in the code when allowed. This simplicity is needed in a larger application such as this in order to increase the efficiency and responsiveness of the code while contributing to the goal of allowing the end user to be able to understand and maintain the code for future use. Programming is a skill that most every engineer acquires throughout his or her schooling. However certain aspects that are specific to larger projects are not always learned. For example, how to manage multiple different programming platforms that need to be interconnected, while focusing on maintaining the simplicity of this connection and the code for each individual platform. We have learned that these two tasks are opposing forces and have worked to make these two goals work in tandem.

Hardware implementation:

The second skill mentioned was that of hardware implementation. This is a necessary step that takes the project from an idea on paper to a working prototype and eventually a final product. Even though most of the hardware implementation was done previously, experience was gained due to unexpected issues. One major issue that presented itself was steering control. Initially LEROY, who moved using the base of a motorized wheel chair, would constantly veer to the right when given the command to go straight. The right motor of the wheel chair base was rotating a slightly slower speed than the left motor. The first thought was that new wheels would need to be purchased since the existing motor controller was incapable of independent motor control. If this ability had been available, a simple command to the left motor to rotate at a constant percentage of the right to compensate for the veering could have been sent. Through research a replacement motor controller with independent motor control was found. This new controller, however, did not support control of the brakes for the dual motor system. Because of this, a new circuit for the motor brakes needed to be developed to allow for proper engaging and disengaging of the motor brakes. Through a series of tests both the new controller and the braking circuitry in the hardware were successfully implemented. Experience was gained in reverse engineering a product by bypassing the current controller that the base already had installed and figuring out the wiring and how much current that each component needed. In the real world a project may be given that was started and one is requested to improve upon the design with only the documentation that came with it.

Time Management:

Time management, a skill that is relevant in every aspect of life, is another important lesson that was further developed while working on this project. Even though we are in our senior year of college, our schedules are still very busy. On top of a full course load and senior design each one of the group members had ongoing extracurricular activities and/or a part time job. All of these different responsibilities take up a significant amount of time. Unfortunately we found that most of this significant amount of time was spread out in such a way that very few free hours were common among each other's schedules. We quickly realized the importance of time management as a team and also individually in order to keep the project on pace. This skill along with the flexibility required for all of these tasks completion is synonymous to the multiple project load that we will have in the work force.

Through the development of this project, our team as a whole has had to configure their schedules to find common work times and each individual member had to learn to manage their time to work on the automated robot and appropriate time for outside work. This is similar to a real-world experience where each member of a team has multiple projects and must find common time and allot their time for each project the members are working on. Our team had a difficulty of finding a common meeting time and meeting to work on the robot. We decided to work on different parts of the projects during our own personal time. We then meet to bounce ideas off each other and to work together on the project that we could. In the real world, we would be working on different projects and have to decide which of the projects get focus than others.

Budget:

As the project commenced, a budget that the robot will have to follow as it's built was necessary. Since all projects in the real world require some kind of budget, this robot has helped us learn how to budget for projects and get the project done in the budget. The initial course of action was to take the cheapest route possible; this was decided before thorough testing was done. Extra money was allotted for events beyond control, for example if the processor all of a sudden broke and a replacement was necessary. After finding malfunctions in various hardware components the lesson learned was to thoroughly test as early as possible. Without the proper testing naive decisions were made that later cost time, energy, and money putting us farther from the budget set. Another thing we learned from this process was that the cheapest option isn't always the best option. Cost-benefit analysis was done but due to improper testing the knowledge used to analyze the options was faulty. Halfway through development a new controller was purchased that would give more control over the system than the previous controller that had been provided with the wheel chair system. The delayed purchase of the new controller required a change in design and the topology of the various parts of the system. Throughout the design of the project budget has played a large part on the equipment available and hardware that could be used within the system.

The' tour-ies 'experiences:

This project is designed to give information to the tour group and be helpful to the tour guide. By displaying PowerPoint's and giving a space to store information packets that the tour guide can use it gives a positive learning experience for the tour group. The automated robot tour guide will allow the tour guide to carry materials, such as pamphlets and flyers, during the tour. This will allow the tour guide to remain better organized, giving the prospective students a sense of how well Ohio Northern University presents itself and the students of the university. Also, once the tracking and following systems are correctly working, there are plans to add room and display images from classes that often take place in the specified room. This would provide the prospective students a brief visual of how classes are conducted and what projects students participate in.

Showcase ONU:

By using this senior design project as an interactive measure with prospective students, they can see what opportunities are available through Ohio Northern. The prospective students will be able to see some of the skills acquired through the different majors and various ways of applying these skills. This automated robot that will assist in tours will demonstrate to the prospective students that Ohio Northern is an innovative place where they will be given the opportunity to create large, real-world applications. Ohio Northern University will be presented as community where the students are able to work on current applications in their field and a safe environment to try new and challenging projects. Through the demonstration of the automated robot, students will notice that through the education at Ohio Northern University they will be provided a knowledge and education that will prepare them for challenging, real-world, team projects. It will also help show what the different kinds of learning experiences that Ohio Northern University is capable of giving to students of the campus. They will learn that Ohio Northern University has the students work on teams to test their knowledge and have them work on projects that will help them with their learning experiences.

VII. Conclusions:

The robotic tour guide assistant, LEROY, uses a motorized wheel chair base with independent motor control to follow a tour guide. Sonar sensors are used to detect when obstacles are present in LEROYs path. LEROY uses a camera to uniquely solve the location problem. This is done by using the camera to track the lit LED pendant worn by the tour guide. This simple tracking method provides a unique approach to solving the location problem by eliminating the need for globalized positioning. It also provides a method to correct the local tracking errors incurred by system imperfections. Despite the solution that tracking an LED pendant provides to this project, LEROYs dependency of this pendant is also a source of limitation. The tracking is only applicable when the pendant is visible to the camera. Therefore if an object obstructs the view of the camera, the tracking method is temporarily ineffective. However within the scope of this project this tracking procedure provides a simple and efficient solution.

References:

[1]"TPR-Robina." . N.p., 25 Oct 2009. Web. 10 Oct 2012. <http://www.plasticpals.com/?p=16011>.

[2] Thrun, Sebastian, Maren Bennewitz, et al. MINERVA: A Second-Generation Museum Tour-Guide Robot. School of Computer Science Carnegie Mellon University, Computer Science Department III University of Bonn, Web. <http://robotics.usc.edu/~maja/teaching/cs584/papers/thrun.icra_minerva>.

[3] Tomatis, N., R. Philippsen, et al. Building a Fully Autonomous Tour Guide Robot: Where Academic Research Meets Industry. BlueBotics SA, Autonomous Systems Lab, EPFL Swiss Federal Institute of Technology Lausanne, Web. <http://infoscience.epfl.ch/record/97477/files/isr02.pdf>.

[4] ÒGrifantini, Kristina. "Robot Plays Follows the Leader - Technology Review." technology review. MIT, 17 Mar 2009. Web. 8 Oct 2012.<http://www.technologyreview.com/view/412585/robot-plays-follows-the-leader/>.