Introduction

Terrain visualization can be used in many civilian and military applications such as wide-area environmental monitoring, search-and-rescue, surveillance, measuring, and robot guidance[1,4,5,6,8]. It can be applied in airborne, ground, indoor, outdoor, underwater, and subterranean situations making it extremely versatile in a variety of environments[5,6,8]. In 2001, the Defense Authorization Act set forth by the U.S. Congress, stated that by 2015 one third of all ground vehicles will be unmanned/autonomous[4]. Improvements in terrain visualization allows for parallel improvements in photogrammetry, computer vision, computer graphics, and robotics; allowing for continuous progress and dramatic expansion in the field of autonomous robots[5].

The government is pushing for autonomous robots and through competitions such as the DARPA Grand Challenge they are getting the community and researchers involved[3,4]. DARPA, which stands for Defense Advanced Research Projects Agency, is a government-funded sector of the United States Department of Defense[7]. The idea behind the DARPA Grand Challenge is to promote autonomous vehicles and related technology for future military use. In 2004 and 2005, the DARPA Grand Challenge was staged in an off-road environment and in 2007 it was an Urban Challenge staged in a mock urban environment. In the Grand Challenge, the vehicles navigated a 132 and 150 mile course equipped with sensors, positioning systems, and computers to keep them from navigating off course while avoiding obstacles[4]. The Urban Challenge was a 60 mile urban course which dealt with moving obstacles, stop signs, and traffic, bringing the competition to a more lifelike urban environment than previous years. While making autonomous vehicles will provide many benefits to our everyday lives and safety, there are many difficulties and issues that still need to be overcome before these systems can be put into place.

Devices such as LIDAR (light detection and ranging) give robots the ability to “see”, allowing us to take our environment and turn it into a virtual world in which autonomous robots can navigate.

The main objective for this research is to develop an indoor/outdoor autonomous robotic platform that gathers and wirelessly transfers data which can be converted to a 3D image at a secondary location allowing the robot’s activity to be controlled/monitored. In addition, other key objectives are to conduct this study in a safe manner and to produce a final product with a practical application.

Application

The robotic platform being used is an Invacare Ranger X power wheelchair (Fig. 1a). The 12V wheelchair system runs each of the two rear wheels independently while the front two casters can rotate and swivel freely. The wheelchair (autonomous systems not shown) has the ability to transfer disabled individuals autonomously and, with modification of the chair

Fig. 1a – Ranger X, [invacare.com]

Fig. 1b – Modified Ranger X, [3waylabs.com]

(Fig. 1b), there are possibilities to use the same system an autonomous platform to which many units can be attached, for transporting materials, etc. The broader and more distant application for these systems includes using them on autonomous personal and military use for transportation of individuals or supplies.

The setup of the wheelchair/robotic platform requires a calibration run in which three dimensional routes are saved to the robot or to a remote location so they can be recalled later. The individual calibrating the device first manually controls the platform from location to location while the onboard system collects data about the surrounding environment. The beginning and end locations of each route are given labels (such as building X, building Y...) which are provided to the occupant as an option on an input system which is either a hard button interface or touch screen monitor. Multiple routes can be stored and recalled allowing the wheelchair to be used in everyday life where the robot travels to different known locations. Also, entire areas can be scanned, such as full campuses so that the user can pick any location in the scanned area and be autonomously delivered rather than be limited to preset routes. Once the occupant makes a selection on the input device, the robotic platform can recall the previously saved map of that route and starts the autonomous navigation. This allows the robot to travel from building to building but more specifically from one room in building X to another room in building Y. In addition, controls such as stop and go back are provided for the user for safety.

The reason why it is important for the robot to be able to travel from one room in one building to another room in another is evident when considering the platform in a wheelchair application. With a disabled individual, he or she should be able to navigate to the building and enter the building as opposed to just making it to the general vicinity. This brings up an issue that is known in this area of research as SLAM (simultaneous location and mapping) which deals with the determination of the position and orientation of the robot in an unknown environment[2,5,6,9]. An environment where the robot cannot obtain a GPS signal and must rely on other input sensors is an example of when SLAM would come into play. This will be discussed in more detail in the 'Current Work' section.

Hardware

The main components needed for making the robot autonomous are the input devices which allow the robot to 'see'. For this study, it is assumed that the sensors include LIDAR, Camera, and IR rangefinders. The position/orientation sensors include: a position/orientation wheel and a GPS (Global Positioning System) and INS (inertial navigation system). In addition, added electronic controls and a wireless data transfer system are required to make the robot fully autonomous.

LIDAR

LIDAR (light detection and ranging) was originally used for mapping particles in the atmosphere using a stationary device, but when GPS was developed in the 1980s it allowed for LIDAR to be used on mobile platforms such as planes[8]. By coupling LIDAR with planes and other mobile platforms it allowed for terrain mapping/monitoring to occur at a faster rate. IMU (inertial motion units) allow for data to be more precisely gathered as it recorded the position (pitch and yaw) of planes and vehicle platforms[8]. This

Fig. 2 – A typical LIDAR sensor, [educatingsilicon.com]

allows for highly accurate three dimensional terrains to be generated even if the plane/vehicle cannot hold a steady path of travel.

The LIDAR component (Fig. 2) uses short pulses of light in specific directions which interacts with the molecules in the air or objects (scattering), sending a small fraction of light back to the LIDAR which is read by a photodetector making one 'pixel' in the three dimensional image[7]. The output of the LIDAR once run through the software is a three dimensional point cloud (Fig. 3) where each point shows the boundary between free and occupied space where the robot can and cannot travel[5].

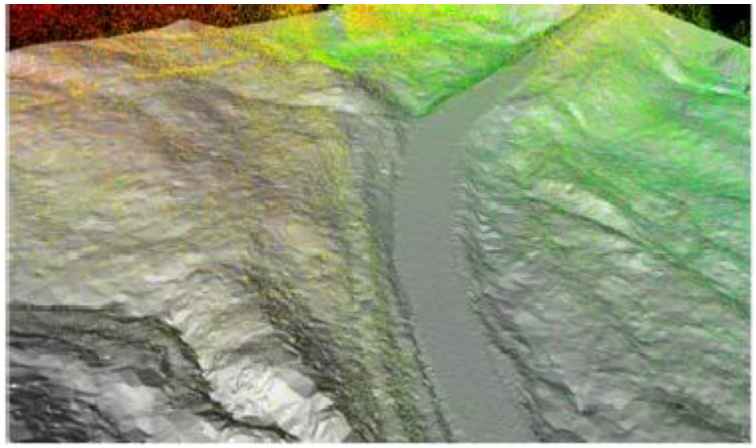

Fig. 3 - Output of a LIDAR Scan, NOAA Costal Service Center[7]

Early LIDAR systems were able to achieve 10,000 pulses per second[8]. LIDAR systems are now more mobile, more accurate, and can process multiple laser returns which allows the LIDAR to emit pulses before the previous pulse has returned allowing for up to 200,000 pulses per second[8]. There are multiple return systems which allow up to 5 return pulses from a single pulse to be read. This allows for distinction between the ground and the tree canopy in tree-dense areas (Fig. 4). By using the multiple return systems one can increase the amount of data by 30% as well as have the ability to toggle layers (trees) on

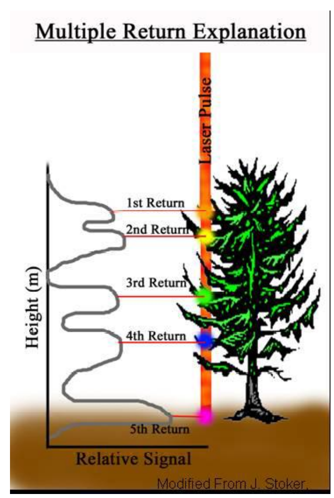

Fig. 4 - Multi return LIDAR system, NOAA Costal Service Center[7]

and off[8]. This gives the ability of mapping the ground below the tree canopy even though it may not be fully visible.

Once the LIDAR data is gathered with the robot it needs to be transferred to a simulator so that the robot can receive human input if needed. The pending problem is that the rate of data transfer from the robot to the simulator has to be fast in order to prevent simulator lag. This data transfer rate can cause issues when comparing high to low speed movement of the vehicle. When gathering data at a low speed there are a small number of informative data points per second that have to be gathered to get the resolution needed. But when the robot speed increases, the amount of unique data per unit time increases as well. If the data points are gathered at the same rate as the slow speed, there can be a loss of information. One way of eliminating this issue is to have an algorithm that tells the LIDAR system how many points to gather depending upon the vehicle's speed. Another way, the one used in this work, is to simply collect so many points that, even at the highest speeds the vehicle can move, there is no possible loss of critical information. This approach, however, puts a very large burden on processing a large amount of information very quickly.

Robot Position and Orientation

To monitor the position and orientation of the robot in space, a position/orientation caster (Fig. 5,6) is used. The position/orientation caster can be thought of as a glorified shopping cart caster in which the rotation of the wheel and swivel of the caster is monitored using optical encoders. The measured change in the wheel's location gives the robot's orientation and position based on the original starting point. The reason that GPS is not being utilized as the sole method of finding robot position is that a robot may often operate in locations with no signal (indoors, underground, deep tunnels, etc.) or places with limited signal (between buildings, under tree canopies, in urban canyons) which could cause the robot to loose connection and disrupt mapping. Odometry is also useful outside, as GPS does not give the orientation of the robot or vehicle.

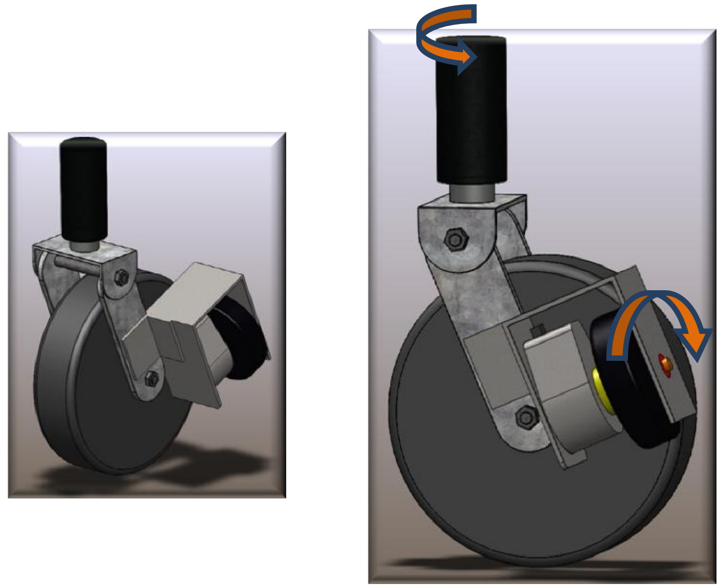

Fig. 5 – Initial Design: Position/orientation wheel concept - Filko, SolidWorks 2010

One of the issues with using a position/orientation caster wheel is that it limits the terrain the robot can drive on to relatively smooth surfaces. If the position/orientation wheel loses contact with the ground, the robot might “think” it has traveled a distance or made a turn that in fact not there, and such errors can cause a discontinuity and/or error in the map.

Fig. 6 – Assembled position/orientation wheels on wheelchair

The designed position/orientation wheel system uses an optical encoder with a stiff 3.5” foam wheel (commonly used on remote control airplanes) mounted on the optical encoder shaft. As shown in Fig. 6, the wheel is mounted perpendicular to the caster wheel and uses the sidewall of the caster wheel as the contact surface. This was done to avoid debris that may stick to the tire treads while the robot is maneuvering. The optical encoder is attached to the shaft of the contact wheel, and the shaft is supported by a dual bearing holder. The whole assembly is connected to the wheelchair caster by a machined aluminum bracket. The caster angle encoders are attached to the top of the caster shaft via a milled component which is directly connected to the optical encoder and mounted on a machined aluminum bracket. The encoder being used to monitor the wheel rotation and the caster swivel angle is the Avago® HEDS 5640 optical incremental encoder which has a resolution of up to 1024 counts per revolution. Each encoder has a lensed LED source and a metal code wheel that rotates freely between a focused LED emitter and multiple detectors. The encoders are connected to a pre-fabricated board which plugs directly into a purchased Arduino board and is used for converting the encoder data into a digital count that the Arduino can read. Using the Arduino, the output count is obtained which can be converted into distance traveled or caster angle using MATLAB software. In total there are four optical encoders; two per caster, one on the wheel monitoring the rotation of the wheel and one monitoring the caster angle. The wheel rotation encoders were specifically mounted on the inside of the caster wheels for protection while maneuvering through tight quarters such as hallways. When the robot travels in a backward or rotating motion though, the rotation encoders will be protruding from the side of the robot (Fig. 6) making them vulnerable to collisions with the environment. To avoid this, the floor space the robot takes up must include the area that the rotation encoder system takes up at any angle.

It was determined that a second position/orientation caster wheel was needed to correctly determine the position/orientation of the robot due to unknown variables in the calculations. The single position/orientation wheel was originally going to be an additional caster wheel trailing behind the robot but once it was determined that a second position/orientation wheel was needed, the system was switched to the two already existing front caster wheels on the robot. More specifically, with a single position/orientation caster wheel the system, the robot would not be able to differentiate the difference between certain motions that give equivalent encoder readings (Fig. 7). In the figure below, one can see that the readings produced by the position/orientation caster angle in both Case A and Case B are the same, even though the robot platform is traveling in different radii arcs. In Case A, both wheels are traveling forward while in Case B one is traveling forward and one backwards. These are different motions, but since the robot is reading the angle of a single caster, it interprets both motions as the same action. By adding a second position/orientation caster the problem is eliminated because it differentiates the two cases by providing the path of the other corner of the robot.

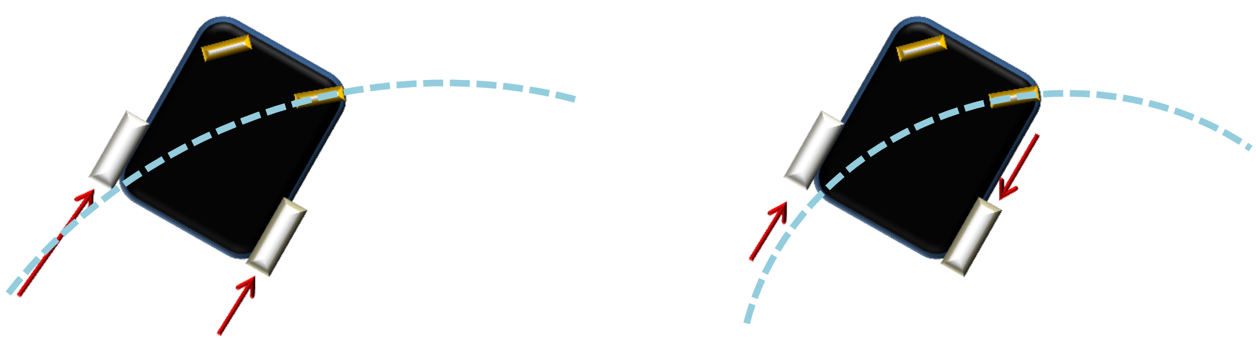

Fig. 7 – Single Position/Orientation Wheel System – Top View

The position/orientation caster system developed for this thesis is built for a slow traveling robot and would need to be built with stronger materials to operate at high speeds. But the concept could be applied to high-speed vehicles, and the calculations would remain the same. The caster wheel measurement system is modular in that it is bolted onto the caster and can be easily moved to be transferred to another system. This is beneficial because the system can be attached to any wheelchair or any other robot where two casters are available or can be attached.

Another option for determining the position and orientation of the robot is by monitoring the drive wheels and their rotation. A single encoder, one per each rear axle would provide accurate position and orientation of the robot. A problem with this is that errors would result if the drive tires slipped or rotated without the robot moving (ice, wet leaves, or other traction loss). In this case, the robot would think that it is in a position or orientation that it is not actually in, leading to errors and discontinuity in the generated maps. This odometry approach was considered but eventually bypassed because the likelihood of wheel slip was too large to guarantee this approach to be sufficient for determining position/orientation.

Robot Control

The wheelchair uses a low voltage inductance circuit to control its movements. The basic principle is to utilize voltage inputs into an inductance coil that is integral to the normal joystick on the wheelchair. In the circuit, voltage changes invoke direction functions (forward, backward, left, and right), and there are unpowered receiving inductance coils to read the joystick motion. When the joystick is in the neutral position, all of the direction function inductance coils are receiving the same voltage. But when the joystick coil is moved closer to one of the induction coils (by pushing the joystick in any direction) a voltage change is induced on the coil, and the wheelchair system processes the action.

To interface this guidance circuit, a custom circuit had to be produced to mimic the joysticks output to the chair so that it can be controlled by the computer (Fig. 8). The circuit uses an Arduino board that outputs a variable voltage from 0-5V and also constant -2.5V is supplied to the circuit from an external source. The four chips are 741 Operational Amplifiers and have the ability to invert and also multiply the positive or negative voltage that comes into them (Fig. 9). The 741 amplifiers are powered by +15V and -15V and have feedback resistors connecting the output pin and the non-inverting input pin. The output that is obtained from the

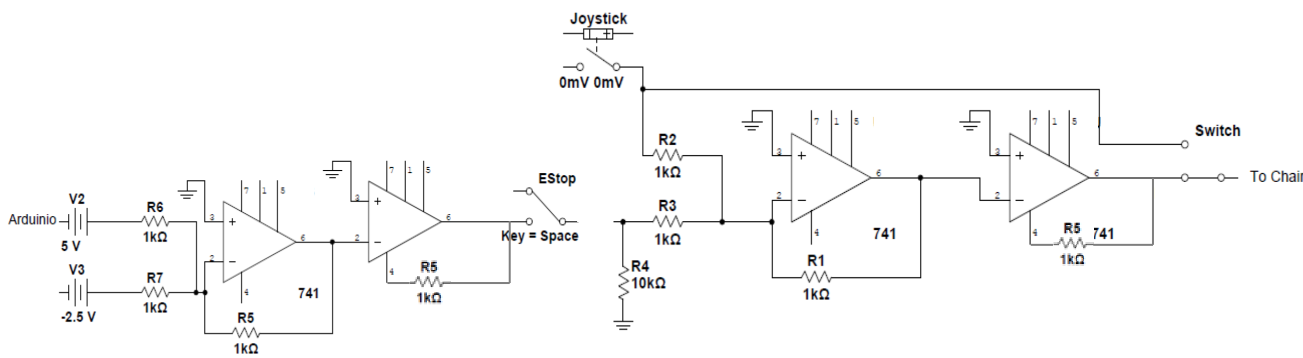

Fig. 8 – Inductance Joystick Circuit

first set of 741 amplifiers is a voltage ranging from -2.5V to 2.5V depending on the Arduino input voltage. After going through the second set of 741 amplifiers, the voltage is converted so that the output ranges from 1V to 4V which is the needed output to control the wheelchair. The circuit (Fig. 8) is for either left/right or forward/reverse functions so two of these circuits are needed to control the platform. For the forward/reverse circuit, a 4V output makes the wheelchair move forward and a 1V output makes the wheelchair move backwards (Fig. 10). By varying the voltage the speed and direction can be changed (2.5 volts being neutral and no movement from the robot). For the left/right circuit, 1V makes the

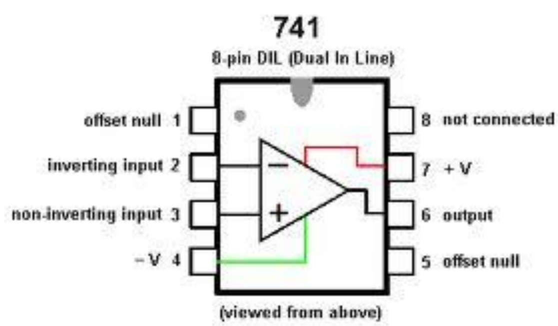

Fig. 9 – 741 Operational Amplifier, allaboutcircuits.com

chair turn left and 4V makes the chair turn right (2.5 volts being neutral and no movement from the chair). By using the two circuits in tandem any direction can be achieved.

In addition to the above voltage circuit, an emergency stop and an override switch are included for safety. One of the goals was to have the wheel chair so it can be made autonomous with all the attached systems then can be reverted back to a normal chair if needed. This requires that everything is modular and easily attachable and removable. The circuit in Fig. 8 plugs into the wheelchairs joystick via a socket that is soldered to the needed operation wires allowing for it to be easily removed.

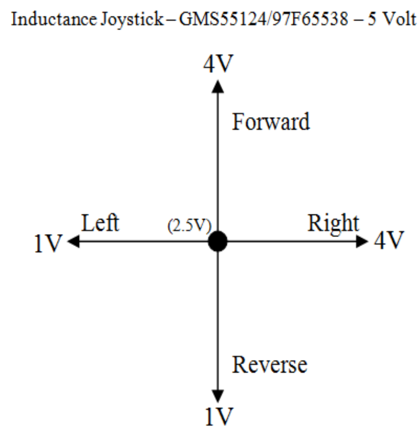

Fig. 10 – Inductance Joystick Functions Based on Voltage

Hardware Locations

The GPS, LIDAR, and camera are mounted above the occupants head (Fig. 11) to improve the camera and LIDAR viewing area, while the short-range IR Rangefinders are mounted low on the chair and help assist the system when navigating through tight/congested areas such as door entrances and through hallways. The IR Rangefinders are helpful in defining the local free space near the robot, to avoid situations where the robot tries to fit through a gap whose dimensions are too small.

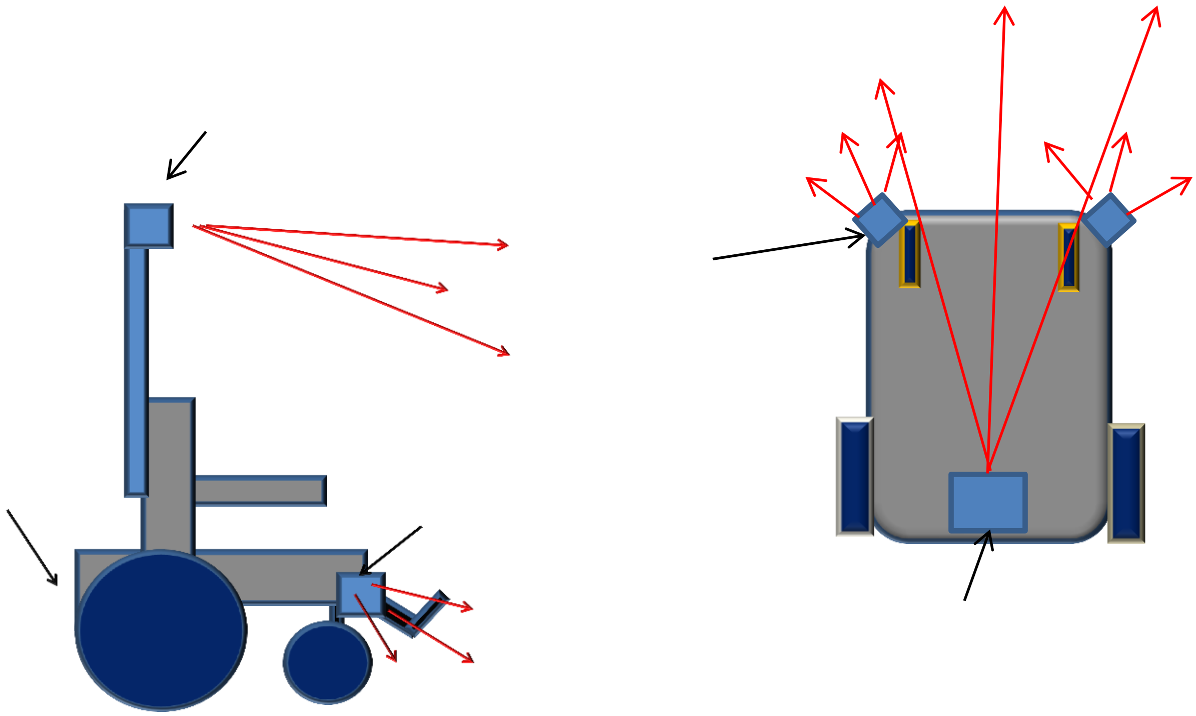

Fig. 11 – Robot/Wheelchair Hardware Layout, Side View and Top View

3D Map Generation

The construction of a three-dimensional map generation starts with the LIDAR system which scans the environment. The resulting "hits" occur as individual points, with each point represented by a radial distance from the robot and a theta coordinate relative to the centerline of the LIDAR. These point coordinates are then transferred using an 802.11 wireless system to a computer running Visual C++ which converts the points to XYZ and creates a 3D visualization for the observer. The generated map is then stored and can be recalled for use in situations such as the autonomous wheelchair scenario discussed earlier.

Path Planning/Collision Avoidance

Path planning and collision avoidance are essential to safely operate an autonomous vehicle nearby other people, for both the human user and the other people in its area of travel. The environments the autonomous vehicles are driving though can be highly structured environments such as a city, or a non-structured environment such as a desert. Regardless of the situation, the robot has to be able to notice changes and adapt in either scenario[3,4].

Path planning requires that the robot is able to gather information about potential obstacles, process the information, and based on the provided algorithms, determine a safe action[2,8]. All of this must occur in real time to ensure safety of the occupant or material being transported. When an object such as car moves into the robot's path, the autonomous robot must notice that there is an object in its path of travel with sufficient rapidity that the object can be avoided. It has to deviate from the planned path in a smooth manor as not to put the occupant, material, or surrounding environment in a dangerous situation. Techniques such as smoothing of the trajectory can be implied to make the motion of the robot more controlled[3,4,8]. Smoothing can also be used in applications such as lane-keeping so that the robot adjusts a small amount as opposed to turning sharply when it gets too close to one of the lane lines[4]. The main goal is to simulate human being control though use of algorithms that mimic human actions which allow the robot to process information in a smart manner[4]. Collision avoidance and path planning for the robot are based on processing speed of the information relative to the speed of the motion of the robot. If the robot is moving too fast and the software cannot process the data fast enough, the chair will act sporadically as the onboard systems cannot keep up with the motion.

Testing

Different hardware layouts were developed to test the robot on specific aspects and problematic issues that may occur with each layout (Fig. 12). In Case A, a robot with a standard camera is used to collect images and wirelessly send the video feed back to the simulator. The

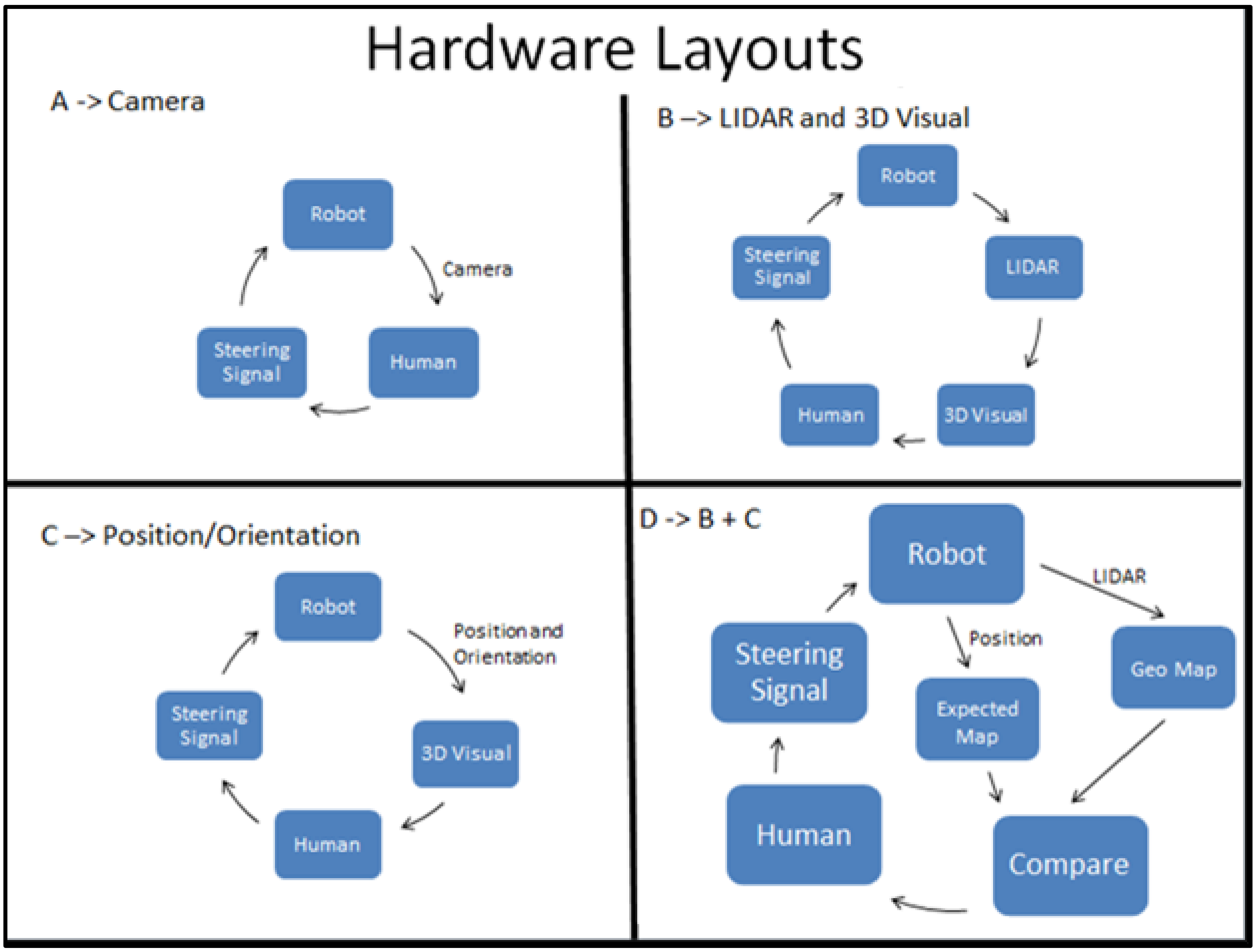

Fig. 12 - Hardware Layouts

operator is then able to view the video feed in the simulator and steer the robot via controls in the simulator. This mimics a remote control car and introduces a video feed with which the robot is remotely controlled. Case A tests the ability to send a camera feed wirelessly between the robot and simulator with minimal lag and serves as a basic service check.

In Case B, LIDAR data is gathered and converted into a three dimensional visual for the human to see in the simulator. The operator still has control over the robot like in Case A, but the visual is a three dimensional visualization (LIDAR point cloud output) versus the live camera feed.

Case B tests the ability for LIDAR data to be sent wirelessly between the robot and simulator. Case C testing the position/orientation aspect of the robot system and matching it with gathered LIDAR data. The position and orientation of the robot will be gathered by odometry, explained in the previous section, and be linked with the LIDAR data gathered at that specific orientation and position.

Case D is a combination of Case B and C in that it is uses the LIDAR system to generate a geographic map and position/orientation data to generate an expected map. The expected map and the geographic maps are then compared before being sent to the simulator for operator input. Case D tests the accuracy between the expected map and the geographic map. The reason for running the hardware layouts in this four step manor prior to automation is to eliminate the possibility of adding too many variables to the system at once, leading to less troubleshooting if there is an issue.

Current Work

Several individuals have done research relating to this thesis. The most prominent key researchers in my area of interest are Sebastian Thrun, Dmitri Delgov, and William ‘Red’ Whittaker.

Sebastian Thrun, the director of the Stanford AI Labs, has done research in the fields of path planning, robotic orientation and localization, and mapping with airborne and underground based robots[1,2,5,7]. Prof. Thrun has been involved in path planning through the DARPA Grand Challenge in which their car, Stanley, won in the 2005 competition[1,7]. His team also ran another car, Junior, who ran in the 2007 DARPA Urban Challenge and placed second[4]. Much of Dr. Thrun's research revolves around his work involved with DARPA.

Dr. William 'Red' Whittaker, a researcher at Carnegie Mellon, has dealt with using LIDAR and RADAR at high speeds, urban mapping, and camera and LIDAR fusion[2]. He has worked on a system for three dimensional mapping of underground mines in which different variations of SLAM were tested[2,6,9]. His involvement in high speed navigation of unrehearsed terrain was also research mainly conducted for the DARPA challenge.

The Robotics Institute at Carnegie Mellon University (Whittaker and the home of Thrun) has done ample research in autonomous mine mapping[2]. The four wheeler based robot named Groundhog was developed to navigate and map dry and partially flooded mine quarters that in some instances were deemed not safe for human entrance while retrieving highly accurate three dimensional maps[2]. The robot used laser range sensors, a night vision camera, gas detector, and a gyroscope to gather two dimensional, three dimensional, and photography images from the mine[2]. This is related to my current work in that while the robot is in the mine it must rely on other input sensors apart from GPS because signal is not achievable. SLAM comes into play when GPS is not available and it deals with the problem of obtaining a map with a mobile robot based of the robots position relative to other objects[2,5,6,9]. SLAM requires that a robot has to rely on other positioning sensors such as an IMU (inertial measurement unit), odometry, and LIDAR location[2,5,6,9]. The robots make their decisions based on algorithms which they use in coordination with their 'sensory' information devices such as cameras, scanning units, and or other sensors to navigate[6,8,9]. Relying on these sensors causes errors to accumulate over long time and distances making the ability of obtaining a highly accurate map difficult[8,9]. A fractional error in a distance or angle when determining the robots location can have a major difference in its 'determined location' which is why multiple locating systems are recommended when dealing with SLAM.

Dmitri Dolgov worked as a Senior Research Scientist in the AI & Robotics group at the Toyota Research Institute and was a Visiting Researcher at Stanford University working with Sebastian Thrun[1]. He also was involved in the DARPA challenge and involved in the path planning, finding structure in unknown environments, and stochastic planning and resource allocation[1]. Much of Dmitri's research is based on path planning and navigation which relates to my research. He reviews algorithms that can be used to in trajectories of the vehicle by looking at the needed results and the alternative paths that can be chosen to get there. Smoothness, proximity to obstacles, and lane keeping are some of the components that the robot can decide between based on the algorithm and its location in the space[1]. These algorithms can be used in the path planning of a robot and in Dmitri's case; a DARPA vehicle.

Implementation of Indoor and Outdoor Robots

In looking at autonomous navigation indoors where the robot is maneuvering on a single plane surface (floor) versus outdoor navigation where the robot is on a non-planer surface, both have their pros and cons. Indoors, the robot is driving on very predictable surfaces, but it cannot use certain systems such as GPS[5] . Further, the area of robot motion can in generally be classified as a smaller/tighter area such as a hallway, which increases the chances of collision with objects.

When driving outdoors, a robot is generally not driving on a planar surface and must have more information to process because of the extra levels of surface interaction. However, outdoor navigation allows access to GPS for location and even compasses to determine the orientation of the robot. Outdoor obstacles pose problems that are never seen indoors, including ditches, water (puddles), snow, mud, highly reflective surfaces, and highly variable weather conditions[4].

Stanford Universities AI Lab (Thrun and Montemerlo) developed a Segway-based robot whose purpose was to be used in indoor and outdoor environments while producing highly accurate three dimensional maps when transitioning from indoors to outdoors[5]. A problem the Stanford team ran into was the coordination of the GPS, IMU (inertial measurement unit), odometry, and LIDAR sensor in order to create a large scale urban map[5]. In their research they scanned for three types of obstacles: terrain that to steep or dangerous to drive on (steep gradient), protruding obstacles such as a piece of wood sticking out from the back of a truck, and holes in the roadway of the autonomous vehicle[5]. The first two examples are what they classify as “positive obstacles” and the third they classify as a “negative obstacle”. In their research they modified algorithms and path planning to deal with various positive and negative obstacles that are commonly encountered in an outdoor environment[5].

Future Thesis Work

Some suggestions for future extensions of this thesis research include: the expansion of the platform to off-road or unstable terrain, implementing the system into high speed applications such as a full size vehicle, and also developing control interphases such as voice recognition. Off-road or unstable terrain such as rocky or dirt terrain in deeply forested areas would bring up many SLAM issues since the castor wheels would have difficulty maintaining contact with the ground and the GPS might not be able to obtain a strong signal. Using the castor wheel odometry systems developed on high speed vehicles and other robots for determining position and location would be beneficial as they would replace the costly units that are currently providing the same data. A voice recognition system would be useful for voice commands such as “stop” and “go back” when dealing with an autonomous wheelchair or material carrying platform if one wants the robot to stop prior to it making it to its final destination. A smart charging system, similar to the Roomba® robotic vacuum systems where the robot finds its charging station automatically, recharges itself, then continues it job would be beneficial in an environment where the robot is working constantly with little or no human interaction. All of these systems would improve on the current system.

Conclusion

This type of indoor/outdoor robot has many applications currently such as delivering disabled individuals or materials from one location to another. The systems that were designed such as the inductance circuit and caster wheel odometry system can be used on other robotic platforms as well. The inductance circuit which controls the robot allows for this platform to be used easily for many other future projects/applications in Dr. Brennan’s research group. The Arduino interface allows allow students learning the Arduino system to program the platform with simple commands and is a basic platform for any autonomous and non-autonomous robots.

The caster wheel odometry system can be used on a multitude of robots and vehicles including tank robots and high speed cars eliminating expensive position and orientation monitoring equipment that is normally used. It is modular build makes it easily removable and attachable to any caster system. Developments in autonomous navigation such as the robot, equipment, and processes formed in thesis are the beginning steps that are needed to be taken to advance terrain visualization to the next and broader level of completely autonomous vehicles in everyday life for both personal and military use.

Thesis Experience

On the educational aspect, I have been working with many new programs and have learned so much in areas that I less knowledgeable in. Working with Dr. Brennan's graduate and other undergraduate students has been extremely helpful in obtaining needed information and skills for this project. This experience was challenging and frustrating at times but all of the new information I learned is irreplaceable and for that I am grateful. Using and learning programs such as Visual C++, CircuitSim, MATLAB, SolidWork, and the Arduino programming software aided in the thesis work. Hands on experience building the mechanical and electrical systems such as the inductance control circuit, position/orientation wheel systems, and other mechanical systems allowed for time at the Penn State Learning Factory and interactions with Electrical/Mechanical/Industrial Engineering undergraduate students. Working on this thesis has been a great learning experience and working with Dr. Brennan has been a pleasure.

Resources

1) Abbeel, Pieter, Demitri Dolgov, Andrew Y. Ng, and Sebastian Thrun. "Apprenticeship Learning for Motion Planning with Application to Parking Lot Navigation." *Intelligent Robots and Systems* (2008): 1083-090. Print.

2) Baker, C., A. Morris, D. Ferguson, S. Thayer, C. Whittaker, Z. Omohundro, C. Reverte, W. Whittaker, D. Hahnel, and S. Thrun. "Robotics and Automation." *A Campaign in Autonomous Mine Mapping*. Proc. of IEEE International Conference on Robotics and Automation (ICRA). 2004. Print.

3) Dolgov, Dmitri, and Sebastian Thrun. "Robotics and Automation." *Autonomous Driving in Semi-Structured Environments: Mapping and Planning*. Proc. of 2009 IEEE International Conference on Robotics and Automation, Japan, Kobe. 2009. 3407-414. Print.

4) Manduchi, R., A. Castano, A. Talukder, and L. Matthies. "Obstacle Detection and Terrain Classification for Autonomous Off-Road Navigation." *Autonomous Robots* 18.1 (2005): 81-102. Print

5) Montemerlo, Michael, and Sebastian Thrun. "Large-Scale Robotic 3-D Mapping of Urban Structures." *Experimental Robotics IX* 9th ser. 21 (2006): 141-50. Print.

6) El-Hakim, S. F., P. Boulanger, F. Blais, and J. Beraldin. "Sensor Based Creation of Indoor Virtual Environment Models." *Virtual Systems and MultiMedia, 1997. VSMM '97. Proceedings., International Conference on* (1997): 50-58. Print.

7) Petrovskaya, Anna, and Sebastian Thrun. "Model Based Vehicle Detection and Tracking for Autonomous Urban Driving." *Autonomous Robots* 26.2-3 (2009): 123-39. Print.

8) Schmid, Keil, Kirk Waters, Lindy Dingerson, Brian Hadley, Rebecca Mataosky, Jamie Carter, and Jennifer Dare. "Lidar 101: An Introduction to LIDAR Technology, Data, and Applications." *National Oceanic and Atmospheric Administration*. 2008. Web. 2 Mar. 2011. <http://www.csc.noaa.gov/digitalcoast/data/coastallidar/_pdf/What_is_Lidar.pdf>.

9) Thrun, Sebastian, Daphne Koller, Andrew Y. Ng, Zoubin Ghahramani, Hugh Durrant-Whyte, and Yufeng Liu. "Simultaneous Localization and Mapping with Sparse Extended Information Filters." *The International Journal of Robotics Research* 23.7-8 (2004): 693-716. Print.