Gesture Based Navigation and Localization of a Smart Wheelchair using Fiducial Markers

by

Jayam Patel

A Thesis

Submitted to the Faculty

of the

WORCESTER POLYTECHNIC INSTITUTE

In partial fulfillment of the requirements for the

Degree of Master of Science

in

Robotics Engineering

by

April 2015

APPROVED:

Professor Taskin Padir, Thesis Advisor

Professor William Michalson

Professor Michael Gennert

Abstract

With the rise in aging population, about 6.8 million American residents are dependent on mobility devices for their day to day activity. More than 40% of these users have difficulty in moving the mobility device on their own. These numbers serve as a motivation on developing a system than can help in manipulation with simple muscle activity and localize the mobility device in the user's home in case of medical emergencies. This research is aimed at creating a user interface of Electromyographic Sensor, attached to the forearm, incorporated with present smart wheelchairs and a simple localization technique using fiducial markers. The main outcome of the research is a simulator of the smart wheelchair to rapidly analyze the results of our research.

Acknowledgements

I would like to express my sincere gratitude towards Professor Taskin Padir for giving me the opportunity to work on the projects mentioned in this thesis and his invaluable support and direction throughout the duration of this work. I would also like to the thank Professor William Michalson and Professor Michael Gennert for being a part of my thesis committee. Robotics and Intelligent Vehicles Research(RIVeR) laboratory has been a great place to collaborate and learn. Thanks to Dmitry Sinyukov and Vinayak Jagtap for their patience during my learning phase and guidance throughout the projects. Thanks to Nirav Patel and Manik Jahagirdar for their help during the projects.

Lastly, I would like to thank my family for the motivation and encouragement that was required for taking the tough decisions and choosing the right path forward.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Nearly 6.8 million American residents are dependent on devices to help them with mobility. This can be further classified to 1.7 million wheelchair users or scooter riders and 6.1 million users of other mobility devices, such as canes, crutches, and walkers [1]. More than 40% of these users have difficulty in moving the wheelchair on their own and hence one nurse is always required to monitor the whereabouts of the patient. To target this problem, there is a need of a system that can help the patients to navigate the mobility device with the least effort and also monitor the position of the patient without human agent involved.

1.1 Background

The number of users of mobility devices increase drastically with the age group. Yet there is a huge difference in the users using manual wheelchair vs electronic wheelchair as shown in Figure 1.1. One major reason and the only advantage in electronic wheelchair is that it can be controlled using a joy stick. While patients who have difficulty in making limb actions go with manual wheelchair and hire a nurse to help them with their mobility. This suggests that there should be specific

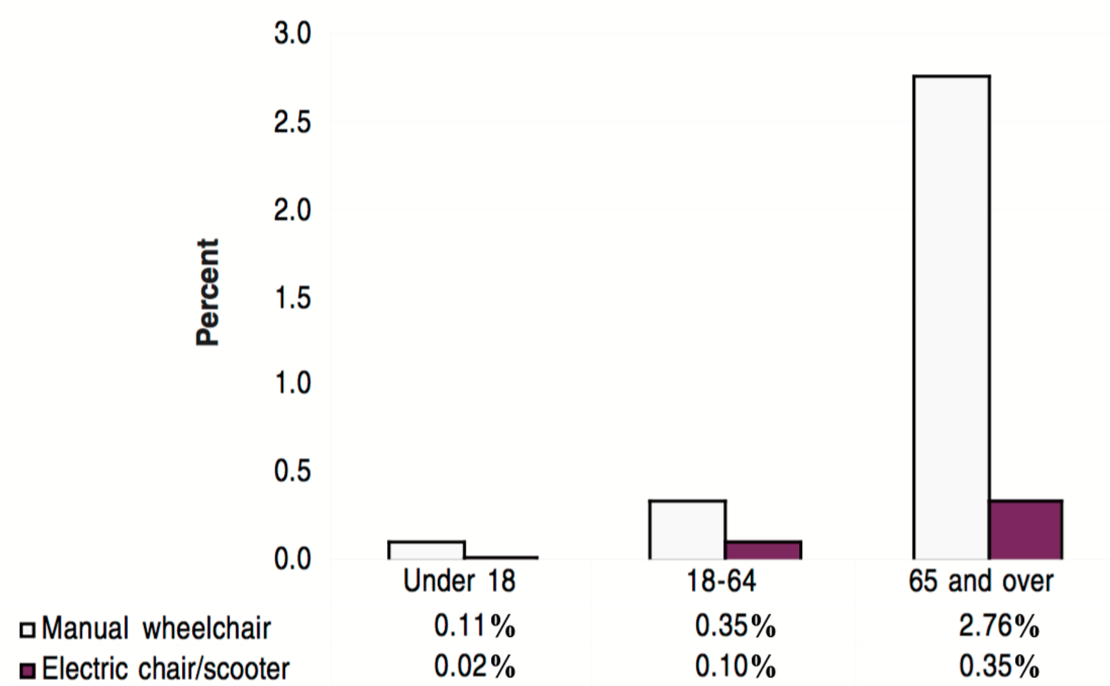


Figure 1.1: Proportion of population using manual wheelchair vs. motorized device, by age [1]

solutions for the entire group of patients who cannot navigate the wheelchair in their daily life and depend on others to do so, as suggested by numbers in Table 1.1.

Table 1.1: Leading conditions associated with wheelchair or scooter use, all ages.

	Conditions	**Persons (1000s)**	**Proportion of device users (%)**
	All Conditions	1,629	100.00
1	Cerebrovascular disease	180	11.05
2	Osteoarthrosis and allied disorders	170	10.43
3	Multiple sclerosis	82	5.02
4	Absence or loss of lower extremity	60	3.68
5	Paraplegia (paralysis of both legs)	59	3.63
6	Orthopedic impairment of lower extremity	59	3.62
7	Other forms of heart disease	54	3.30
8	Cerebral palsy	51	3.11
9	Rheumatoid arthritis and other inflammatory polyarthropathies	49	3.00
10	Diabetes	39	2.40

Figure 1.2: Anna, the semi-autonomous wheelchair

1.2 Semi Autonomous Wheelchair

Semi autonomous wheelchairs are automated systems but controlled by the human agent behaving as the operator of the system. This system is designed in such a way that the authority to navigate the mobility device lies with the operator, human. This authority can be either shared with framework of interfaces, like Brain Control Interface (BCI), Google Glass or be completely operated by the human with a joy-stick.

Figure 1.2 shows the semi autonomous wheelchair, Anna, developed at WPI. This semi autonomous wheelchair consists of two Light Detection And Ranging(LiDAR) sensors for environment mapping and obstacle avoidance, camera for perception and cliff sensors to avoid cliff fall risks. The wheelchair operator can select different interfaces depending on the suitability of the patient.

1.3 EMG Based Navigation

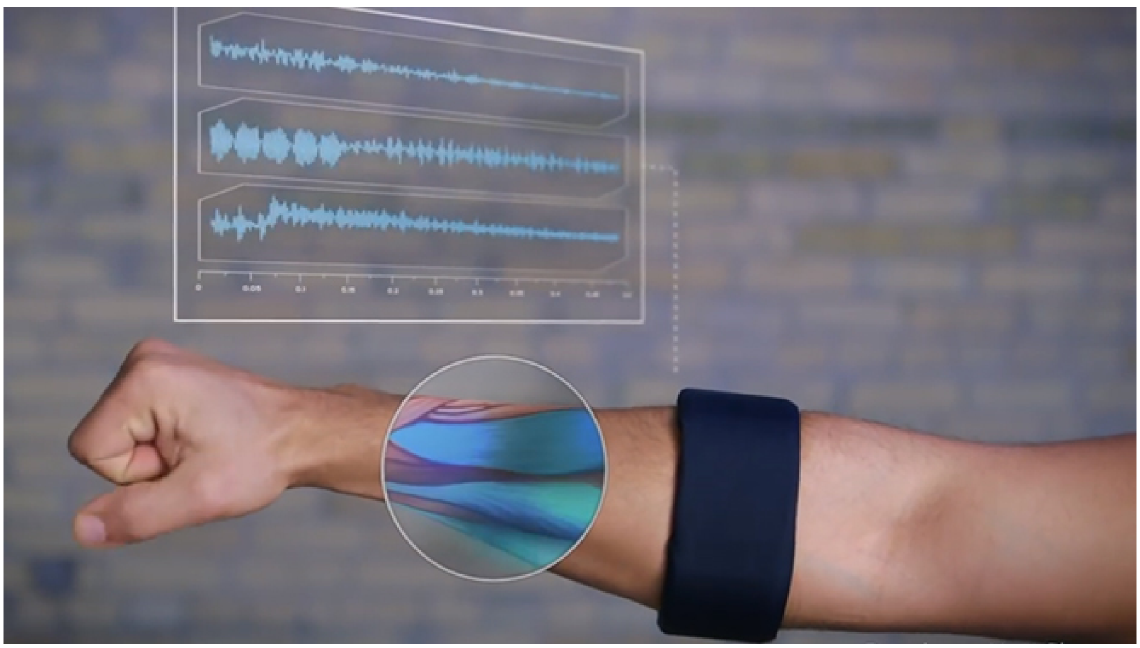


Figure 1.3: EMG Gesture Sensor

Electromyographic sensor, or EMG sensor, records the electrical activity of muscles to measure the functioning of nerves' electrical activity. EMG readings can provide us with muscle turning on and off, i.e. contraction and expansion of the muscle at a particular time [2]. These readings can be interpreted using signal processing and can be applied to a human interfaced wheelchair [3] [4].

The location of placing EMG sensor have to be carefully selected as the sensor should be placed on the longitudinal mid-line of the muscle and between the joints performing motor functions. EMG sensor if placed near the tendon, tissue that holds muscles with joints, the muscle activity cannot be correctly interpreted as the muscle fibers become thinner and fewer in number. Also multiple tendons are attached with the joint which can increase the chances of recording cross talks of different muscles [5]. The best part of a human body, serving all the restrictions of EMG sensor placement, would be forearm. Placement and removal of the sensor strips would also be easy when attached to forearm.

Although, these readings are noisy with minor variation in the analog signals in comparison with the readings obtained from any other sensor, they are effective and helpful to patients who are incapable of making other motions. Capturing the EMG signals may require a set of EMG sensors attached to forearm giving us a touch free control of any technology with hand gestures and motion as shown in Figure 1.3. These motions and gestures can be interpreted, after removing the noise, as set of instructions for the robot to move in all the directions.

1.4 Indoor Localization of Wheelchair

Over the evolution of indoor localization, researchers have started using various methods that are helpful both indoor and outdoor. Compared to outdoor localization, indoor localization has proved to be more challenging in terms of accuracy and precision. In outdoor localization GPS [8] is available which solves most of the problems but indoor localization contains obstructions and various points of interests. Researchers have implemented navigation system based on the RFID Tags to locate household items [6]. Wireless sensor network based approach has also been implemented and tested in dynamic environment [10]. Also, researches have developed smart wheelchair which can not only navigates using the ceiling lights as landmarks but also avoids the obstacles [7]. Along with these methods Sonar and camera data have also been used to localize the wheelchair without specular reflection due to Sonar in corners [9]. Localization tryouts have also been done using magnetic sensors [11] and Kinect Sensor & Monte Carlo Localization Method [12].

All the localization methods mentioned above either requires active landmarks or costly sensors mounted on the wheelchair which generate large amount of data just to localize the wheelchair. It is challenging to provide easily scalable and feasible solution. This calls of a system that is efficient localization system for an assistive device being used in a predefined, static environment.

1.5 Flow of Thesis

The flow of thesis is as follows. Section 2 describes about the wheelchair simulator which details the wheelchair and environment modeling in a physics engine running simulator. This includes the design of the smart environment set up at WPI. Section 3 describes the EMG based navigation of a wheelchair using a sensor called MYO. Section 4 describes localization technique using low cost April Tags to locate the wheelchair in an actual and a simulated environment.

Chapter 2

Wheelchair Simulator

With the constant evolution of the smart wheelchair, Anna, the risks of testing a technology directly on the wheelchair also rises, considering the fact that a disable human will be operating the wheelchair in reality the simulator have to be reliable and robust [16]. These risks may damage the wheelchair hardware or the user operating the wheelchair. Hence this called for a need of a test bed for the wheelchair which is not only realistic but also follows the law of Physics. The simulator has two major parts, a robot model and a world in which the robot can navigate and perform the required test cases.

2.1 Wheelchair as a Mobile Robot

A mobile robot is defined as a system that is capable of autonomous navigation in a dynamic or static environment using the a particular type of driving mechanism. The most common driving mechanism is a differential drive consisting of two motors controlling two wheels and few casters for free rotation. A simple differential drive mechanism is shown in Figure 2.1. The navigation of the differential drive robot can be implemented by just controlling the translation on x-axis and y-axis, and rotation about z-axis, shown by ω, with an angle θ with positive x-axis. A wheelchair can be compared to mechanism with addition of 4 caster wheels to support and balance the whole setup.

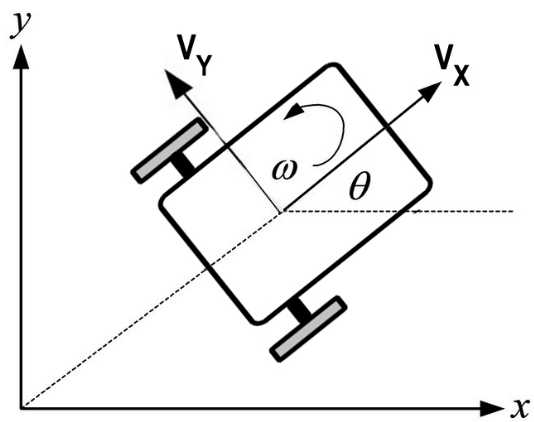


Figure 2.1: Simple Differential Drive Mechanism

2.2 ROS Integration

Robot Operating System(ROS) provides standard operating services like hardware abstraction, communication link between processes, driver support, and several other tools and packages [1]. ROS also provides support with simulators which have the hardware configuration of a robot or a system in a pseudo realistic computer generated environment. Simulators like GAZEBO or V-REP also have a physics engine running with them which simulates robot in a ideal environment. We can not only test the functioning of the robot but also create a test world and then parse data between the simulator and ROS. ROS gives up the capability of using the navigation message type, TWIST message for our application to control the translation velocity on x-axis and y-axis, and angular rotation on z-axis.

[1]http://wiki.ros.org/ROS/Introduction

2.3 Gazebo Simulation

Gazebo is a standalone 3D simulator that can be integrated with ROS to simulate robot. Gazebo takes two files as their input, a world configuration file and a robot configuration file. The world description file contains all the elements of a simulation, including furniture, walls and lights. This file is written in XML format, and typically has a .world extension. A configuration file is the description of the robot in nearest scale and joint movements. Robot configuration file contains all the elements of the robot, including sensors, links, static joints and transmission joints. This file is written in XML Macro format(XACRO) which is very similar to Unified Robot Description Format(URDF) but a much simpler version than URDF. In XACRO we don't have to redefine robot parts which are similar in characteristics, for example if the robot contains more than one wheel then we can create a generalized structure and then just feed information which changes in rest of the identical objects, which is not the case in URDF.

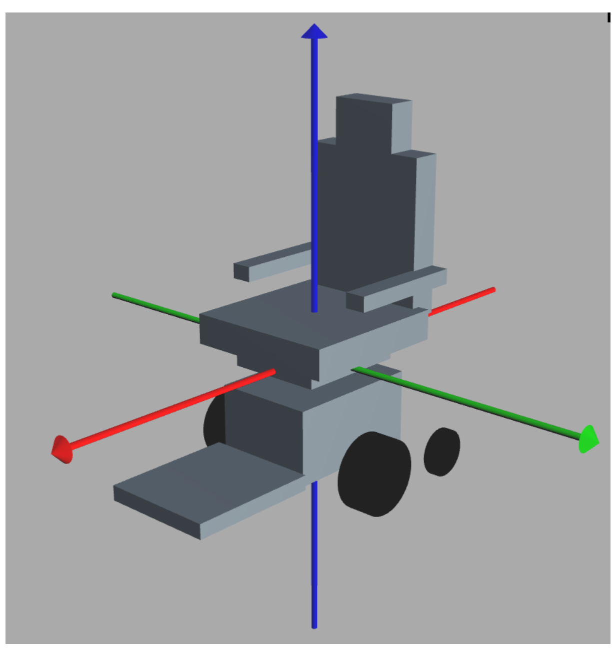


Figure 2.2: Wheelchair Model in Gazebo

2.3.1 Gazebo Description Files

Wheelchairs description is in XACRO format with the closed dimensions of the real wheelchair, shown in Figure 2.2. Details of sensors installed on real wheelchair, including two LiDARs, two wheel encoders and one camera, are also provided in the description file. The world file is created in closest dimension of the WPI's Smarthome environment where the wheelchair is actually being tested as shown in Figure 2.3 and Figure 2.4.

Figure 2.3: Smarthome World Model in Gazebo

Gazebo uses Open Dynamics Engine(ODE), a physics engine for simulating rigid body dynamics. ODE also provides Gazebo with advanced joint types and integrated collision detection with friction. Gazebo can simulate LiDAR scans, depth data,

Figure 2.4: Smarthome at WPI

odometry and camera data from the sensors installed on the wheelchair. Every sensor requires a Gazebo Plugin, a chunk of code to acquire data from sensors to publish in ROS and control the properties of a sensor, and our gazebo configuration files contains such plugins for both LiDARs, encoders and camera. LiDARs cannot be graphically constructed in the gazebo for which we import its mesh file with is in COLLADA (COLLAborative Design Activity) format [2]. There are two LiDARs installed on the wheelchair, one on the battery top and one on the foot plate as shown in Figure 2.5.

Simulators lack the capacity of displaying the outputs of the sensors and for this functionality a visualizer is needed. A visualizer displays the outputs as seen by robot and for this utility ROS provides a package called Robot Vizualiver (RViz).

[2]https://collada.org

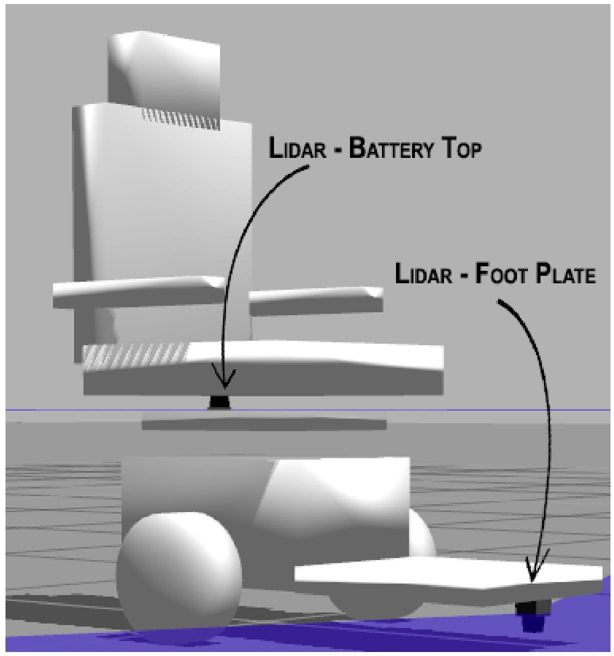


Figure 2.5: LiDAR installation on Wheelchair

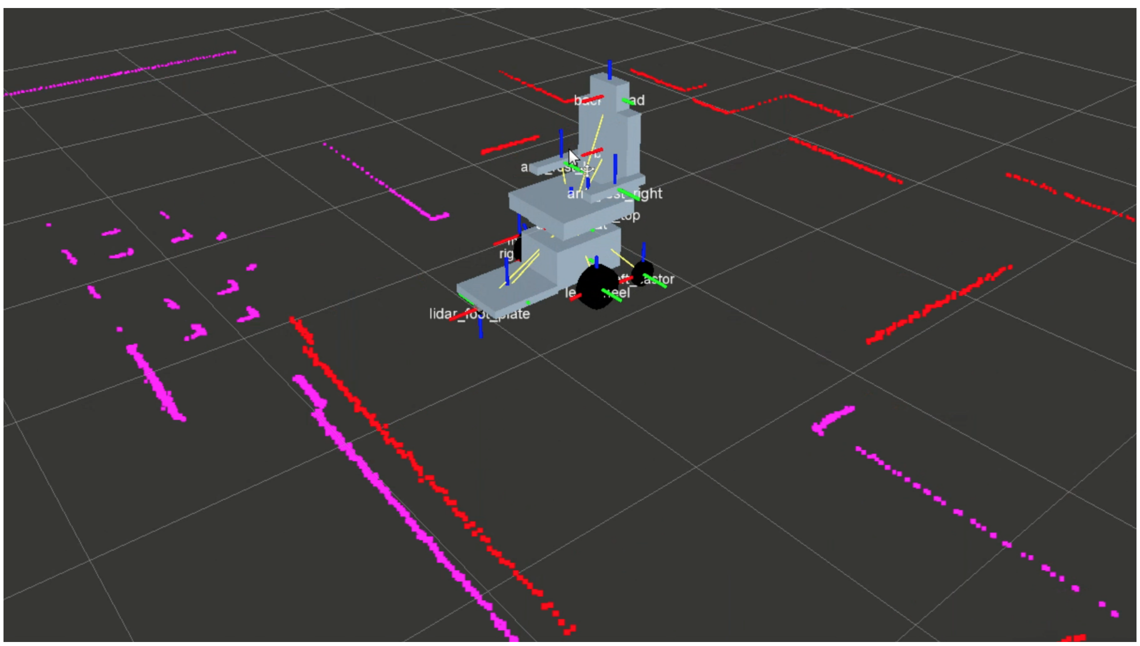


Figure 2.6: LiDAR plot in RViz

Using RViz, we can also display live representations of sensor values coming over ROS Topics including camera image data, LiDAR depth data and odometry data encoders. Figure 2.6 shows the depth points as perceived by LiDAR, where the red points signify the data generated by LiDAR over the battery top and pink points signify the data generated from LiDAR on the foot place.

Hence, a Gazebo ROS package for wheelchair in smart home is an outcome of this section enabling the dynamics of the system to test all the future interface and algorithms before its deployment on the wheelchair. This package also generates sensor data of the fabricated environment in the simulator.

Chapter 3

MYO Based Navigation

3.1 Wheelchair Navigation

Safe navigation of a wheelchair is always an essential feature because the user depends on the device for its safety. Safe navigation is observed if no collisions or no unsafe conditions are met with moving the robot.

Anna, the smart wheelchair is equipped with user interfaces of navigation with voice control, brain computer interface and a joystick. Voice Control Interface is for patients who can only use speech function, Brain Computer Interface for patients incapable of physical movement and joystick for patients who can use their hands, but an interface for patients who can only make few gestures with their hand to control the navigation of the wheelchair can be added as a replacement to all the interfaces defined above.

3.2 Electromyography

Electromayograhy(EMG) is a diagnostic technique which records and evaluates the electrical activity produced by muscle movement, generating an electromyogram.

The movement of muscles leads the muscle cells to activate and deactivate. This on and off of the cells can be analyzed and electromyogram can be generated. Primarily EMG is undertaken to diagnose a neuro-muscular disease or disorder of motion control.

Recently researchers have started using EMG to control smart systems. The main site to extract EMG is from the forearm and the muscle activity can be read using few sensors. Although this is a readily available technique, it is not that efficient. The signal retrieved from the electromyogram are very noisy and unreliable. But this can be overcome using signal filters, like high pass or band pass filters, to obtain the useful signals.

3.3 MYO - Gesture Detection

Figure 3.1: MYO by Thalmic Labs

MYO is gesture recognizing band manufactured by Thalmic Labs, shown in

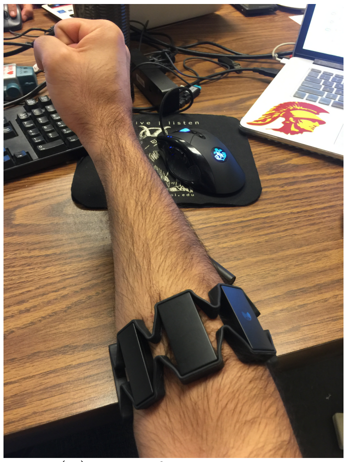
Figure 3.1. MYO uses EMG signals to determine the hand gesture, an accelerometer to measure acceleration and a gyroscope to determine the orientation of the hand. It uses low power Bluetooth connected to an ARM processor with a rechargeable lithium ion battery. Haptic feedbacks can also be provided on MYO band. The device is compatible with Windows, IOS, OSX and android. Thalmic Labs do not provide support for Linux based systems. Hence a MYO-ROS package is created by converting the drivers and making them publish data on ROS topics, which can be easily accessed by any robotics technology that runs using ROS.

The MYO-ROS package consists of a MYO node, which lets us connect MYO with ROS through Linux and an interpreter package that lets us convert the incoming data in terms of ROS messages. The most frequently used ROS message type is the TWIST message provided by geometry stack and the MYO-ROS package is capable of publishing it.

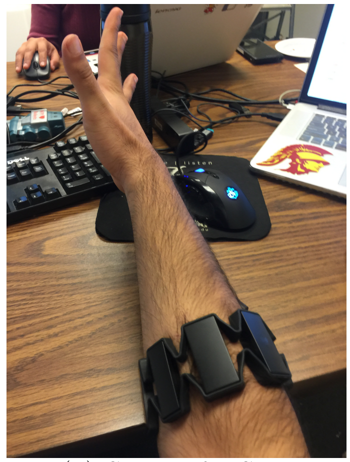
The data that is generated by the MYO-ROS package are forwarded to following ROS Topics:

- IMU Raw Data
- EMG Raw Data
- Arm Muscle Movement Data
- Gesture
- Action Description

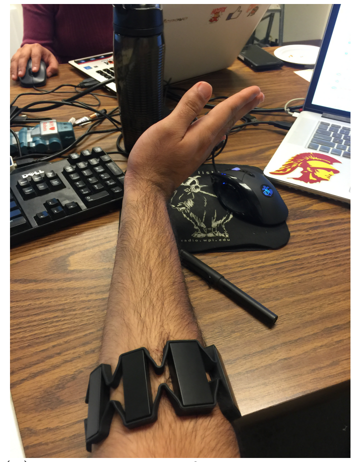
The MYO-ROS package is capable of recognizing four hand gestures as shown in Figure 3.2 and each of these gestures are defined with a particular command for navigating the wheelchair.



(a) Fist for Forward



(b) Stretch for Stop



(c) Right Wave for Right Turn

(d) Left Wave for Left Turn

Figure 3.2: Gestures to control navigation

3.4 Matlab-Robotics System Toolbox Integration

Robotics System Toolbox is a Matlab toolbox that provides ROS data sharing compatibility with any system that can run Matlab. The toolbox lets you share the address of the ROS and topics being published in that instance. Toolbox has some inbuilt algorithms that can be merged with ROS applications to generate interfaces that are faster and easier to code. Matlab has better mathematical functions to analyze and understand the sensor signals which helped in bring a better understanding of the EMG Signals. Figure 3.3 shows the output for the hand gesture in

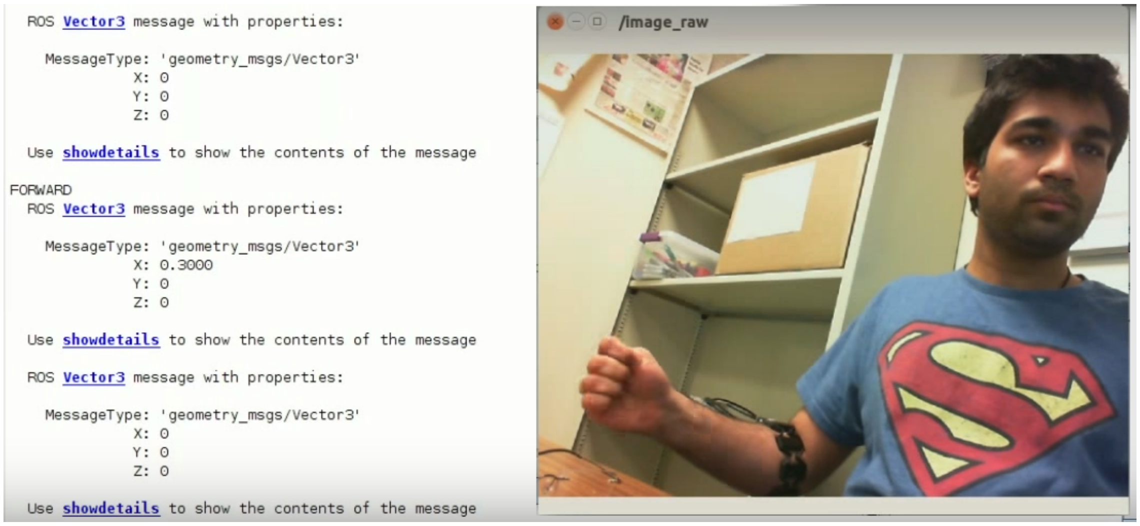
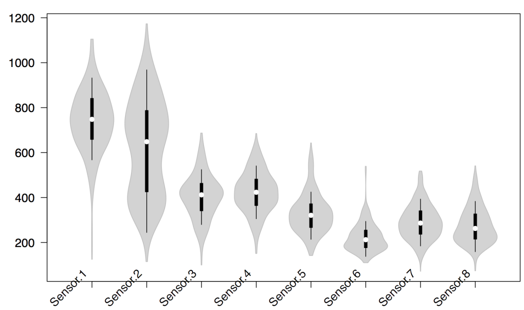
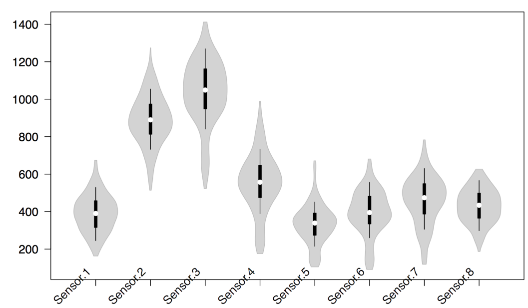


Figure 3.3: Interfacing with MYO and Matlab using ROS

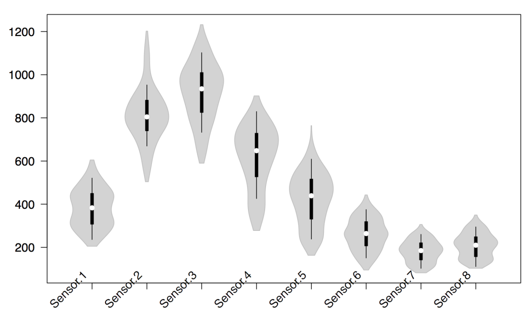
Matlab using ROS Core on another system for logging the gesture data for further analysis.



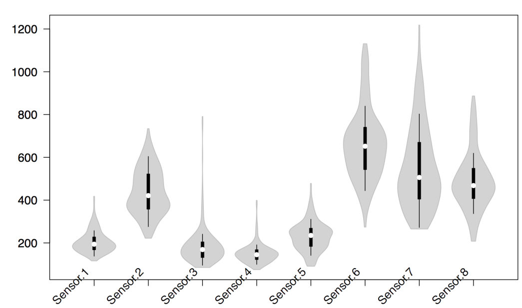
(a) Plot for Fist Gesture



(b) Plot for Stretch Gesture



(c) Plot for Right Wave Gesture



(d) Plot for Left Wave Gesture

Figure 3.4: Violin Plots of all the EMG Gestures [15]

After recording 500 Samples of each gesture a violin plot, shown in Figure 3.4, was generated to understand the distribution of the signals. A violin graph is a method to plot numerical data in form of distribution along with reflecting the standard deviation and the mean of the sample [15]. For one gesture we get inputs from 8 EMG Pads, hence each violin graph shows 8 sensors generating amplified voltage signal for that particular gesture.

Chapter 4

Localization using Fiducial Markers

4.1 Localization Techniques

Localization of a wheelchair in a known but cluttered environment poses many challenges. There has been lot of effort put by the research community to address the challenges and develop feasible solutions. Researches have tried to adopt outdoor navigation approaches in the indoor environment. Developing an accurate indoor localization system can benefit a large segment of aging population. A camera based localization method is presented which uses April Tags [1], as Fiducial Markers, to precisely find the location of wheelchair in a predefined environment. Using off-the-shelf web camera and Intel Galileo embedded board, an embedded localization device is developed and an accuracy of ±6 inches is achieved. The development of this project started as a course project in a team of three. The flow of course project was divided in three parts, the hardware configuration and testing, April

[1] https://april.eecs.umich.edu/wiki/index.php/AprilTags

Tag information decryption and data exchange with server. My contribution was to read the April Tags using OpenCV Library to generate the pose and orientation of the camera with respect to the tags.

This divice is tested in Smart Home Environment at WPI which is equipped with a PHANT [2] server. The PHANT server is a dedicated server for Internet of Things (IOT) application, that records and transmits data to all the connected devices in Smart Home.

4.2 Model of Physical Process

Landmark are the physical quantities which are required to be identified and decoded. A simple landmark with the least information detail in them are April Tags which on decoding gives only the tag ID and the family ID of tags it belongs to. A fixed focus camera mounted on the wheelchair is used to identify and locate the tags. Modeling of the physical process needs consideration of certain parameters. Those parameters change or affect the physical model and are listed as bellow:

s:- Physical size of the April-tag

l,w:- Dimensions of the picture frame (l=w)

L,W:- Dimensions of the Room

h:- Height from camera to ceiling

2α*:- Viewing angle of the camera*

V_x, V_y*:- Velocity components*

f:- frames per second (fps)

*x*y:- Resolution of the image*

[2] http://phant.io

d:- distance between the outer edges of April-Tag

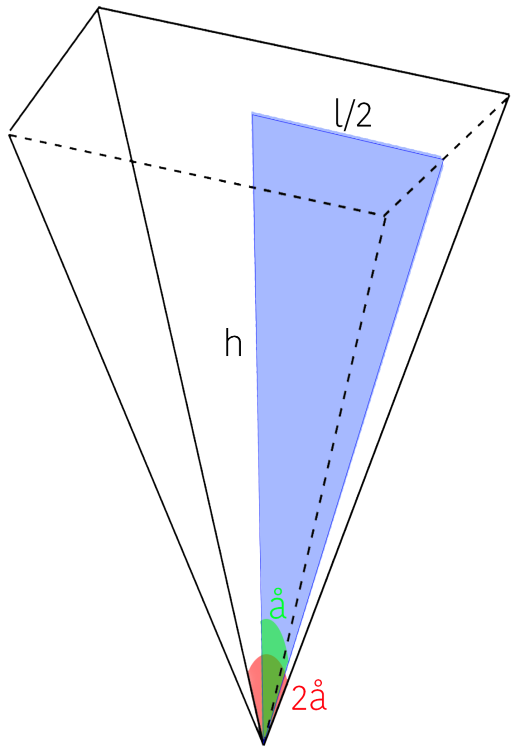


Figure 4.1: Square Pyramid formed by Camera

Wing angle α and height of the ceiling form the camera h, the area which can be covered by a single image in the camera frame can be obtained by calculating the size of the base of the pyramid formed by the camera viewing angle or in other words a view-port. As shown in Figure 4.1 size of the base of the camera view-port pyramid can be obtained as follows

$$tan(\alpha) = \frac{\frac{l}{2}}{h} \tag{4.1}$$

$$l = 2h * tan(\alpha) \tag{4.2}$$

The maximum possible distance between two adjacent April Tags, to be placed on the ceiling, can be found by using camera viewing angle and hence area covered by the image pyramid should be used. Assume that at time t_{-1}(blue) and at time t_0(red) frames are being formed by camera. Considering the two consecutive frames and the size of the image captured, ideal distance between two April Tags should be given by:

$$d < \sqrt{2}d < l \tag{4.3}$$

$\sqrt{2}$d is the efficient way when considering an April Tag in diagonal translation.

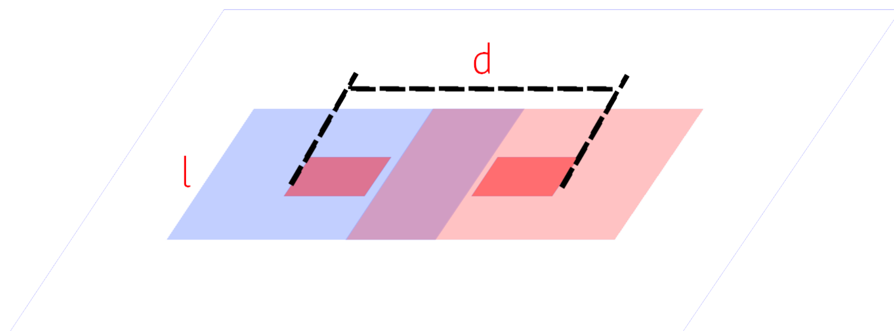


Figure 4.2: Distance between adjacent tags

Now while the image is being processed the wheelchair will still be in motion, hence by the time the decoded location is obtained, the wheelchair would have traveled to a different location. Thus wheelchair velocity should be constantly monitored and the position should be updated as well. Let x_{t-1}, y_{t-1} be coordinates at time t-1 and x_t, y_t at time t, then:

$$x_t = x_{t-1} + V_x \tag{4.4}$$

$$y_t = y_{t-1} + V_y \tag{4.5}$$

We also need to calculate the size of the tags. s_{min} which can be found by camera parameters, obtained by calibrating camera in OpenCV, while s_{max} can be estimated by supposing that picture frame contain 4 large April Tags with no space left and dimension s_{max} * s_{max},

$$l^2 = 4s_{max}^2 \Longrightarrow s_{max} = \frac{l}{2} \tag{4.6}$$

But, from experiments performed by using different size for April Tags, it has been found that April Tags printed on an A4 paper are sufficiently large enough to be found in a single camera image. Also, distance between the adjacent April Tags is found to be 5 feet for a room having height of 8 feet, considering the device is place at 3 feet height from the ground.

4.3 Localization Model

Localization model is based on environment description and visual processing of the landmarks where the April Tags placed on the ceiling. Figure 4.3 depicts the localization model used to locate the wheelchair in a predefined environment.

As shown in the figure, locations of the April Tags in the environment is known and described in a file given as an input to the localization device. The transform T_A^H is read from the file along with the tag ID and its location in the environment with respect to a predefined origin. The transform T_C^A is calculated by decoding the April Tags using the camera mounted on the wheelchair. The transformation with respect to April Tags is generated by using April Tags C++ Library [14] and then it is converted to transform from Tag to Camera T_A^H. Once these two transforms

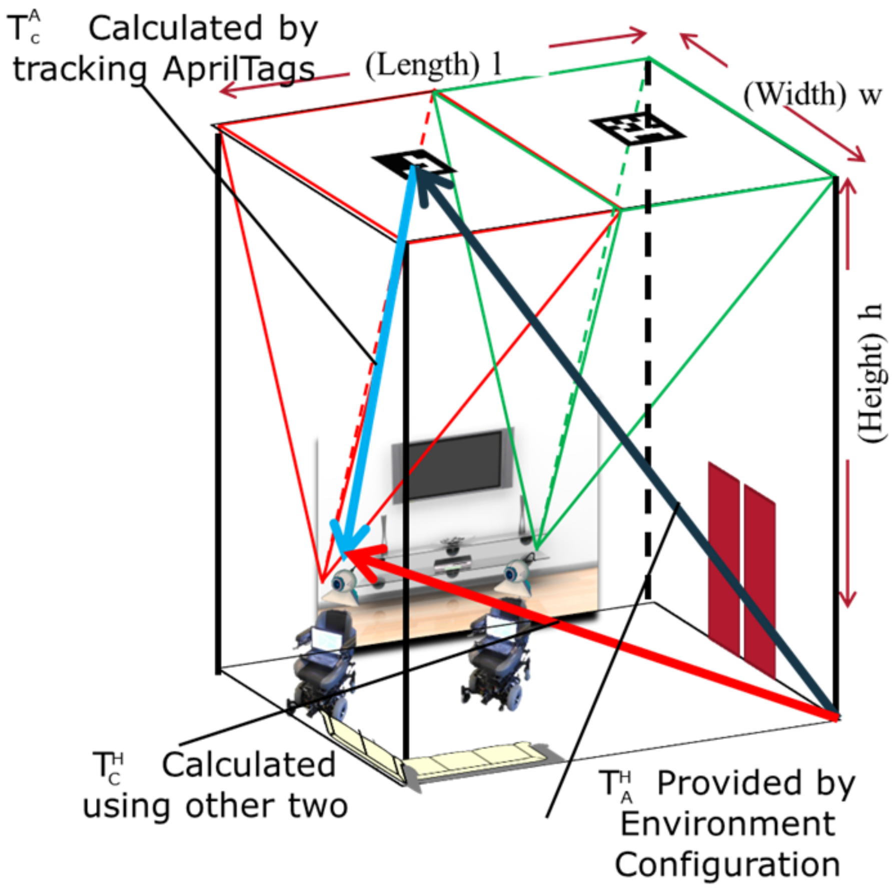


Figure 4.3: Localization Model

are known, location of the wheelchair in the environment can be given as

$$T_C^H = T_A^H * T_C^A \tag{4.7}$$

4.4 Finite State Machine

Finite State Machine(FSM) is a model of a system which consist of states, inputs, intermediate outputs and final system outputs. Figure 4.4 depicts the FSM for the localization device which is implemented using Intel Galileo board. As shown in the FSM, during system startup, first a check for the connection of camera with the main board is done. On successful camera connection, image is acquired and passed over to the processing unit which decodes the image to detect April Tags. At the end of processing, algorithm returns 6-D pose of April-Tag in camera frame which is transformed to camera pose in April-Tag frame which is further processed to calculate the camera location in environment frame. Calculated 6-D pose is sent to the central PHANT server for storage and tracking.

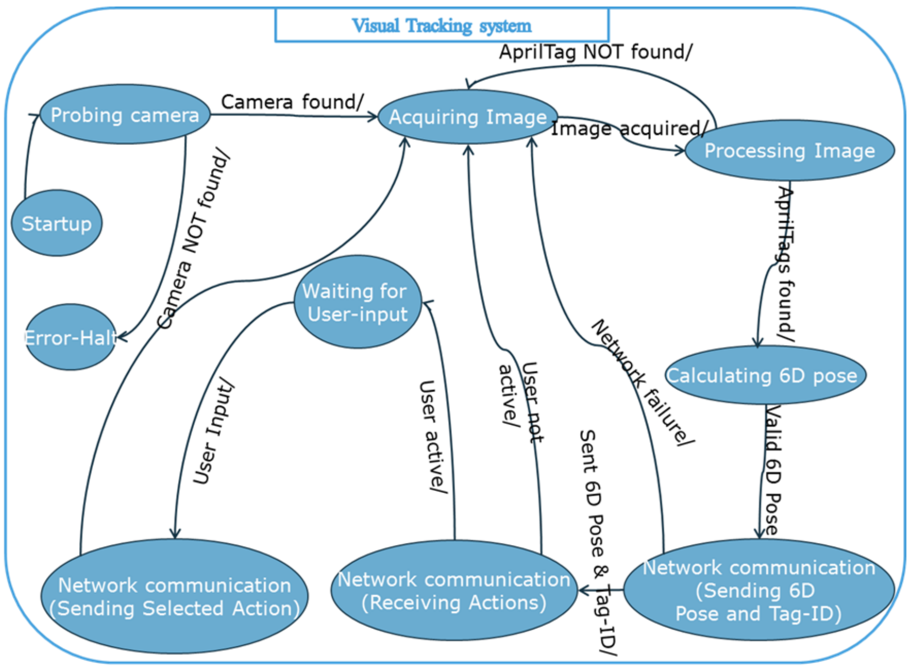


Figure 4.4: Finite State Machine for localization device

The central control system is shown in Figure 4.5

4.5 Experimental Setup and Result

April Tags comparatively has less information as compared to a QR code or bar code but results in faster inspection. To localize the disabled person using assistive tool, April Tags should be placed at regular intervals on the roof of the room such that at least one April code is visible at a time. The wheelchair has embedded system based

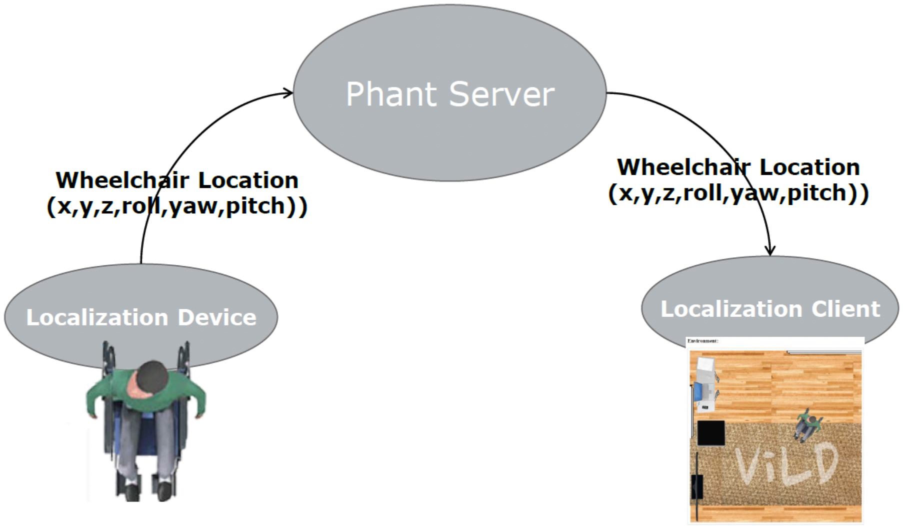


Figure 4.5: Finite State Machine for Current Implementation

on camera to scan the April-code and decode it on-board. The decoded April-code information is sent to central logging server using HTTP GET/POST requests. The current implementation of the system uses following hardware components.

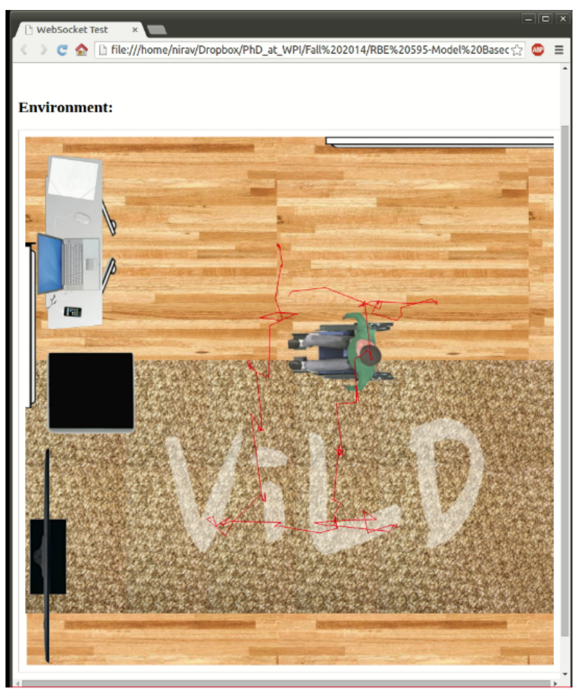


Figure 4.6: Web Application for Environment Description

1. Main processor: Intel Galileo V. 1

2. Vision system: Off-the-shelf web camera

3. Communication: Wi-Fi mPCIe shield for Intel Galileo

A web client to pop data from the PHANT server was also made to fetch the localized data of the wheelchair through any computer on network. Figure 4.6 shows trace of wheelchair locations traced using the from camera data on Intel Galileo board.

Chapter 5

Experiments and Results

5.1 MYO and Wheelchair

EMG signals can generate uncertainty in data because of their low amplitude. Taking 100 samples for each gesture using the action ID generated in our package and checking the uncertainty in the data, it was found that most errors were in fist gesture and stretching of fingers with a fail rate of 9% and 8% respectively. While the right wave gesture and left wave gesture have the fail rate of 6% and 4% respectively. This data is reflected in form of graph in Figure 5.1 for n=100 samples.

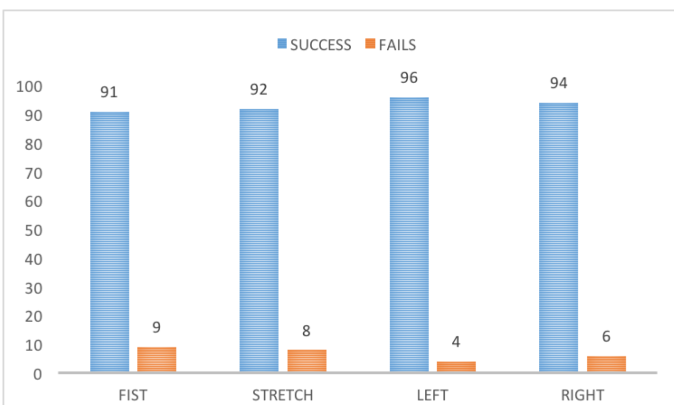


Figure 5.1: Success vs Fail Rate for EMG Gestures using 100 samples each

The trials mentioned above were taken while providing wheelchair simulator input and cross verifying if the wheelchair is responding as per the signal provided by the MYO-ROS package and it was observed that the wheelchair would behave as per the output given by the MYO-ROS package. During these trials the world file was replaced with an empty file to avoid collisions in cluttered environment.

The final test of wheelchair simulator package and MYO-ROS package was tested in the simulated Smart Home while publishing the data from MYO and from the wheelchair. The wheelchair data from LiDARs and odometry data are visualized in RViz, along with MYO-ROS published action on ROS topics is shown in Figure 5.2.

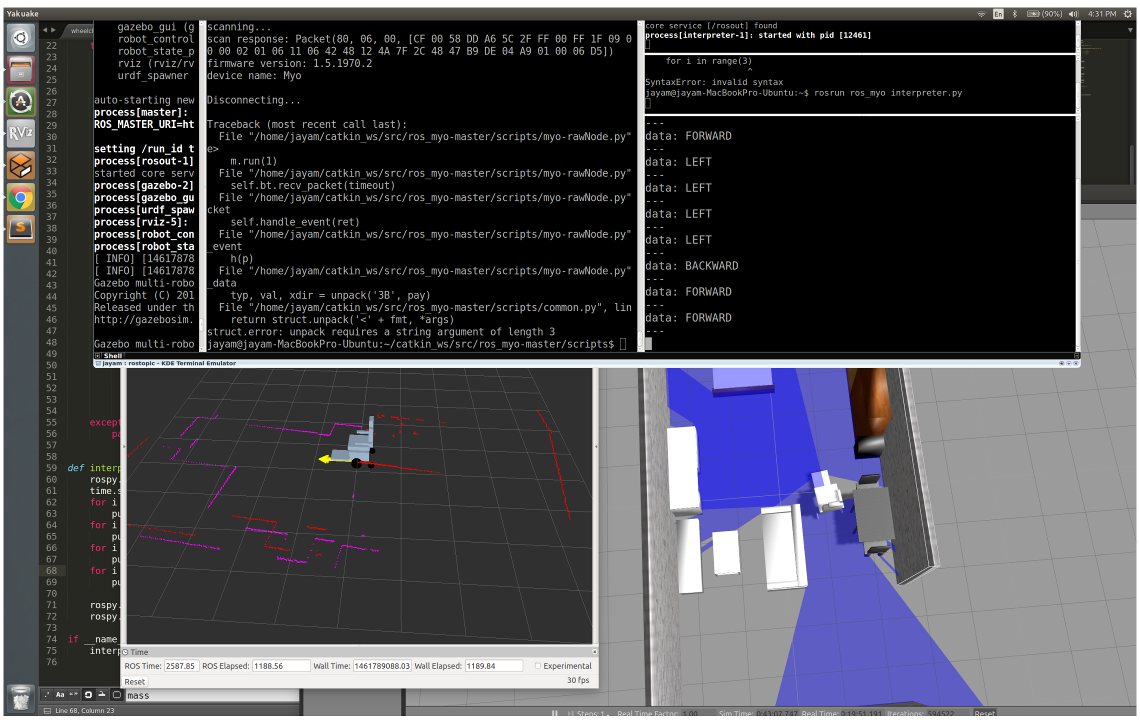
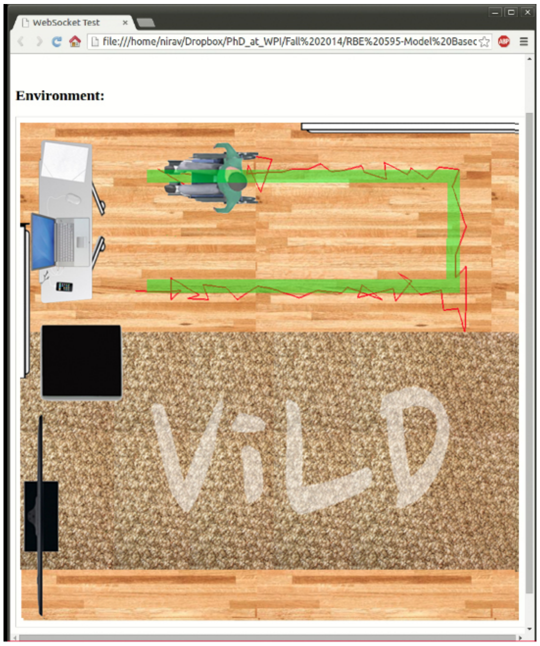
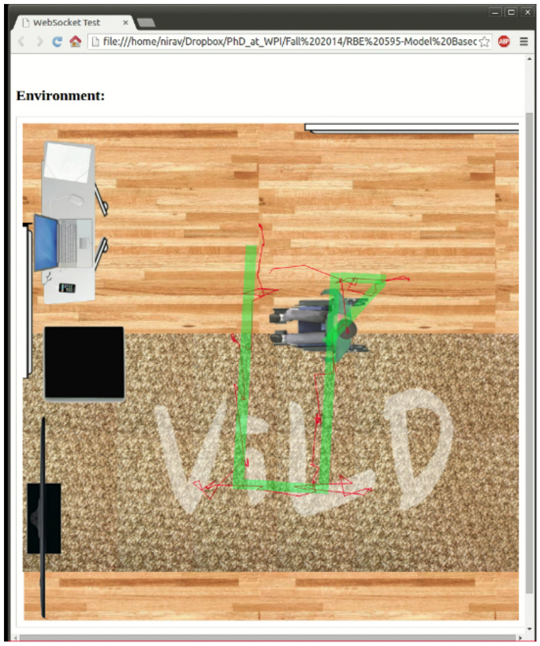


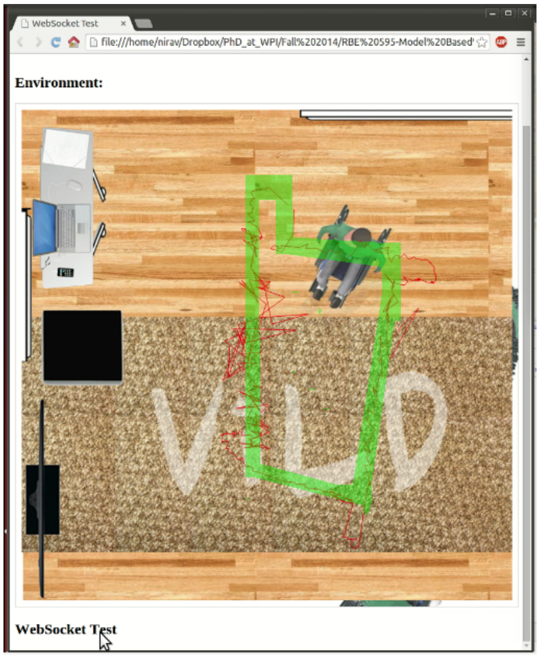
Figure 5.2: Testing of MYO-ROS Package on Wheelchair Simulator



(a) Test 1



(b) Test 2



(c) Test 3

Figure 5.3: Test Case of April Tag Localization Device

5.2 Analysis of April Tag Localization

The setup of April Tag localization device was using Intel Galileo board and an off the shelf camera. April Tags were printed with size of 6.53543" x 6.53543" and pasted on the roof in the test environment. After performing tests and taking images of the received trajectory through the web client it was evident that there was error of ±6" from the expected path. Figure 5.3(a), Figure 5.3(b) and Figure 5.3(c) show the test cases performed on April Tag Localization Device, where the red line justify the recorded trajectory and the green path shows the intended motion. As seen from the red line and green path that the localized information is within 6" and some jerks in red line were also formed because of vibration of the device on the wheelchair, while rest of the jerks are errors by the device.

Chapter 6

Discussion

The thesis aims on creating a gesture based navigation interface for wheelchair users who depend on others for mobility. Using MYO, EMG band with 9-Axis IMU by Thalmic labs, a ROS package was created to convert the EMG signals to action messages that can be understood by the smart wheelchair at WPI. The package later was made modular in a way that it can be used by any robotic technology using ROS libraries. As discussed in the previous section the hand gestures have a fail rate less than 9%. Currently, the package only interprets the EMG signal and in future it can be appended with 9-Axis IMU interpreter, which can help in arm movement recognition along with gesture recognition. This arm movement recognition could be applied to control and manipulate the JACO [1] arm attached on the smart wheelchair.

Also, it is risky to test any new interface on the wheelchair directly as it can endanger the hardware and the person using the wheelchair. Hence, a test bed i.e. a simulator was made to test all the future algorithms or technologies of wheelchair before deploying it on the wheelchair framework to avoid any possible damage. This

[1] www.kinovarobotics.com

includes the replicated model of wheelchair and Smart Home in Gazebo.

Lastly, to keep a check on the whereabouts of the wheelchair user in a known environment an April Tag Localization Device was devised. This device can localize the wheelchair and push data to server which can be accessed by our web client to provide the followed trajectory and the current position. Intel Galileo was used as the development hardware for the device along with an off the shelf camera. As discussed in the results, if there is a sudden change in the pose of the wheelchair then an error occurs in the localized output, but in all the test case the localized output is within the range of ±6". These test case also showed that the update speed of web client, i.e. the speed of fetching data from the server, depends on the network bandwidth.

In conclusion, the thesis presents a wheelchair simulator, a gesture based navigation interface using MYO and an April Tag localization device.

Appendix A

Appendix

A.1 Differential Drive Kinematics

A simple mobile robot have six degrees of freedom (DOF) expresses as (x, y, z, Roll, Pitch, Yaw). x, y, z denotes the position and Roll, Pitch, Yaw denote the pose of the robot. For a differential drive robot, we reduce the control of 6 DOF to only 3 DOF i.e. x, y, θ, where θ denotes the Yaw.

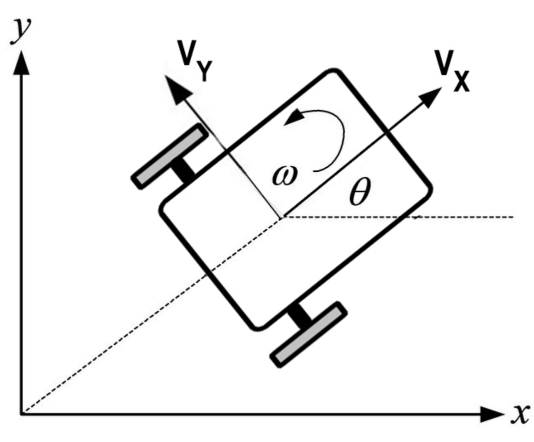


Figure A.1: Simple Differential Drive Mechanism

The navigation can be controlled by controlling the velocities in the reduced 3 DOF. Figure A.1 shows a simple mobile robot on 2D Plane.

While translating the robot in X or Y direction we assume that no-slip conditions are met. For all the wheels on the robot they should have they should have a common center by which they rotate which is known as Instantaneous Center of Curvature (ICC). The rotation happens around the ICC circle with a radius r. R denotes the distance between ICC and the center of the wheel axis and l is the length of the wheel axis.

So if a wheel speed is v in time t to complete one turn around ICC and angular velocity is ω, then we can get a combined equation

$$v = \frac{2 * \pi * r}{t} \tag{A.1}$$

$$v = \frac{2 * \pi}{t} \tag{A.2}$$

$$v = r * \omega \tag{A.3}$$

Here, r and v for both wheels will result in same ω and converting above equation in independent values of v and r for both the wheels yields the following equation

$$\omega(R + l/2) = v_r \tag{A.4}$$

$$\omega(R - l/2) = v_l \tag{A.5}$$

Solving these equations we can find R and ω.

Now, the as the robot rotate in δt then the θ will change to θ'

$$\theta' = \omega * \delta t + \theta \tag{A.6}$$

Using all these equations we can compute the final equations as follows

$$\begin{bmatrix} x' \\ y' \\ \theta' \end{bmatrix} = \begin{bmatrix} cos(\omega * \delta t) & -sin(\omega * \delta t) & 0 \\ sin(\omega * \delta t) & cos(\omega * \delta t) & 0 \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} x - ICC_x \\ y - ICC_y \\ \theta \end{bmatrix} + \begin{bmatrix} ICC_x \\ ICC_y \\ \omega * \delta t \end{bmatrix} \tag{A.7}$$

A.2 MYO Specifications

MYO EMG Band's specifications are as follows:

1. Size and Weight Specifications
    (a) Arm size: Expandable between 7.5 - 13 inches (forearm circumference)
    (b) Weight: 93 grams
    (c) Thickness: 0.45 inches
2. Compatible Devices
    (a) Windows: Windows 7 and above
    (b) Mac: OSX 10.8 and above
    (c) Android: v4.3 and above
    (d) IOS: IPhone 4s and above
3. Sensors
    (a) Nine-axis IMU containing three-axis gyroscope
    (b) Stainless Steel EMG sensors
    (c) Three-axis Accelerometer

(d) Three-axis Magnetometer

4. Processor: ARM Cortex M4 Processor
5. Haptic Feedback
6. Communication Link: Bluetooth 4.0
7. Battery: Lithium Ion Battery

A.3 Intel Galileo Specifications

Intel Galileo Board's specifications are as follows:

1. Processor: Intel Quark SoC X1000 (16K Cache, 400 MHz)
2. RAM: DDR3 800 - 256MB
3. I/O Specifications: USB 2.0 3 Ports, 1 Serial, 1 LAN
4. PCI Support: PCI Express
5. Wi-Fi: mPCIe shield

Bibliography

[1] Kaye, H. Stephen, Taewoon Kang, and Michell P. LaPlante. "Mobility Device Use in the United States. Disability Statistics Report 14." (2000).

[2] Kim, Jonghwa, Stephan Mastnik, and Elisabeth Andr. "EMG-based hand gesture recognition for realtime biosignal interfacing." Proceedings of the 13th international conference on Intelligent user interfaces. ACM, 2008.

[3] Moon, Inhyuk, et al. "Wearable EMG-based HCI for electric-powered wheelchair users with motor disabilities." Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on. IEEE, 2005.

[4] Moon, Inhyuk, et al. "Intelligent robotic wheelchair with EMG-, gesture-, and voice-based interfaces." Intelligent Robots and Systems, 2003.(IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on. Vol. 4. IEEE, 2003.

[5] Takala, Esa-Pekka, and Risto Toivonen. "Placement of forearm surface EMG electrodes in the assessment of hand loading in manual tasks." Ergonomics 56.7 (2013): 1159-1166.

[6] Deyle, Travis, Matthew S. Reynolds, and Charles C. Kemp. "Finding and Navigating to Household Objects with UHF RFID Tags by Optimizing RF Signal Strength."

[7] Wang, Honbo, et al. "Auto-navigation of a wheelchair." Artificial Life and Robotics 1.3 (1997): 141-146.

[8] Gao, Chao, Michael Sands, and John R. Spletzer. "Towards autonomous wheelchair systems in urban environments." Field and Service Robotics. Springer Berlin Heidelberg, 2010.

[9] Luo, Yuan, Shuai Shao, and Yi Zhang. "Localization and Path Planning of Intelligent Wheelchair Based on Data Fusion." Advanced Materials Research 217 (2011): 1086-1092.

[10] Mistry, Hetal P., and Nital H. Mistry. ”RSSI Based Localization Scheme in Wireless Sensor Networks: A Survey.” Advanced Computing & Communication Technologies (ACCT), 2015 Fifth International Conference on. IEEE, 2015.

[11] Fan, Ruolin, et al. ”Localizing a wheelchair indoors with magnetic sensors.” Ad Hoc Networking Workshop (MED-HOC-NET), 2013 12th Annual Mediterranean. IEEE, 2013.

[12] Theodoridis, Theodoros, et al. ”Kinect enabled monte carlo localisation for a robotic wheelchair.” Intelligent Autonomous Systems 12. Springer Berlin Heidelberg, 2013. 153-163.

[13] Olson, Edwin. ”AprilTag: A robust and flexible visual fiducial system.” In Robotics and Automation (ICRA), 2011 IEEE International Conference on, pp. 3400-3407. IEEE, 2011.

[14] Michael Kaess. ”AprilTags C++ Library.” http://people.csail.mit.edu/kaess/apriltags/ , MIT.

[15] Hintze, Jerry L., and Ray D. Nelson. ”Violin plots: a box plot-density trace synergism.” The American Statistician 52.2 (1998): 181-184.

[16] Niniss, Hafid, and Takenobu Inoue. ”Electric wheelchair simulator for rehabilitation of persons with motor disability.” Symposium on Virtual Reality VIII (Proceedings), Belm (PA). 2006.