EEnable: An Affordable Myoelectric Powered Prosthetic Hand

Final Report

University of Notre Dame
Electrical Engineering Senior Design
Spring Semester 2019

Jose Montalvo Ferreiro
Jacob Kapala
Conner Futa
Alexis Magana
Simon Kiang-Hinojosa

Abstract **3**

Introduction **3**
- Background 3
- Motivation 3
- Problem/Goal Statement 4
  - Ease of Use 4
  - Affordability 4
  - Customization 4

System Requirements **4**
- Control Requirements: 4
- Power Requirements: 5
- Mechanical Requirements: 5
- User Interface Requirements: 5

Project Description **6**
- Software 6
  - Data interpretation 6
  - Machine Learning 6
  - Motor Movement 7
- Hardware 7
- Electronics 7

System Testing **7**
- Inter-module communication 7
- Motor Control Fine Tuning 8
- Machine Learning Accuracy 8

User Manual **8**
- How to Train 8
- How to place pads 8

Future Improvements **8**

Conclusions **8**

Abstract

This project proposes to accurately categorize a hand gesture based on the data acquired from an EMG (Electromyography) and use that information to be able to move motors which move motors in order to perform the specified hand gesture. We have 4 different channels that acquire electrical signals from a person's remaining limb. Whenever the person attempts to do a hand gesture, which they used to be able to do, a pattern is gathered, and that data used to determine what kind of gesture the user intended to recreate.

Introduction

Background

"There are 185,000 amputations in the US every single year, currently there are about 2.4 million amputees in the US alone, this number is expected to double by 2050."[1] Most of the prosthetics solutions currently available in the market are priced between $35,000.00 and $120,000.00. The most expensive part of this prosthetics is the EMG and signal analysis system.

With this project we have been able to develop a machine learning algorithm that grabs data from an individual's arm (through the use of an EMG), interprets it and then is able to predict what hand gesture is the amputee attempting to recreate.

In order to do this we researched and used different machine learning algorithms such as Naive Bayes, Convolutional Neural Networks, Resource Description Frameworks, Logistic Regression, Linear Discriminant Analysis, etc. In the document you are going to be able to find a detailed comparison between each one of those and a final decision on which one we chose to finally implement. We are also going to explain the mechanical aspects involved in this project.

Motivation

Current prosthetics in the market can only reach specific people that are able to adequate their muscles in order to fit a predetermined range of EMG signals, this prosthetics are also extremely expensive, which means only a limited amount of people are actually able to get them. Our main motivation is reaching that "forgotten" segment of the market that is not able to purchase a prosthetic arm, and allow them to regain previously lost hand movement, allowing them to regain the quality of life they once had.

---

[1] Amputee Statistics You Ought to Know. (n.d.). Retrieved from https://www.advancedamputees.com/amputee-statistics-you-ought-know

Problem/Goal Statement

Current myoelectric prosthetics are expensive, hard to use, and in most cases don't work for every amputee. For this project we set out to make an easy to use, affordable, and customizable prosthetic that will improve the quality of life of the user.

Ease of Use

This was one of our main goals. Most amputees complain about how difficult prosthetics are to control. Mechanical ones make the user do complex movements in order to pick something up, electrical ones make you press a button or use an external device to control, and in order to control myoelectric ones, amputees have to go through months of physical therapy in order to be able to use.
This is why we design a system that adapts to the user and learns from the signals that the user is more proficient at.

Affordability

As we explained in the abstract, current costs for prosthetics are in the hundreds of thousands of dollars range. This means that most people are not able to afford this kind of prosthetic, instead they have to go for a mechanically activated or a place-holder hand.

Customization

No two amputations are the same, each person should have what is best for them and there is where customization comes in. if a young boy needs a hand, the design will be completely different from a forty year old, and that is why we need to be able to customize the hand to each individual patient.

System Requirements

Goal: A safe, lightweight, and affordable prosthetic that can be easily trained for recent transradial amputees.

Control Requirements:

The embedded intelligence must be capable of rotating the motors in order to acquire the correct hand gesture. It must also know what hand gesture is being requested from the signals acquired by the EMG. It should know the battery remaining and send a signal if it is running low. Finally, it should be able to know when the system is being trained and when training has stopped.

Power Requirements:

The Power Control System is the subsystem that powers the hand. It needs to provide enough voltage and current to drive 6 motors (one for each finger, plus two for the thumb), the microcontroller, and ADCs. The motors we are going to be using are rated at 6V, the max power is 1.3 Watts. The microcontroller requires a voltage between 3.3 and 5V. Ideally, the battery should be able to power the hand for several hours so that multiple recharges per day are not necessary.

In addition to supplying the necessary power, this subsystem also has to fit other design requirements such as size, weight, durability, and safety. The system should also be transportable, so batteries are an obvious choice. The battery system should be compact, lightweight, and robust enough to withstand a patient's day to day activities.

Mechanical Requirements:

The prosthetic hand weight and size requirements are supposed to closely match the weight distribution of a biological arm. Weight should be roughly 2-3% of their total bodyweight to be accurate to average arm weighting, size should be equivalent to human hand measurements. We are working with a 7 year old girl. She weighs about 55 pounds, this means that the weight of the prosthetic should be around 1.1 - 1.5 pounds. The size should match the size of a 7 year old with some room to grow without it looking odd. The average width of a 7 year old's hand is 4.5 cm which means we should be around that same ballpark.

User Interface Requirements:

Not much interface is required, user only needs to attach the EMG electrodes in the proper positions and activate or deactivate training mode via a button press. There is no wireless interfaces as everything is self contained within the hand and microcontrollers, no data is needed to come out or go into the hand once it is trained and in use. EMG electrodes are attached to the bicep and forearm areas for signal reading. The 3D printed limb is attached to the arm through some harness.

Project Description

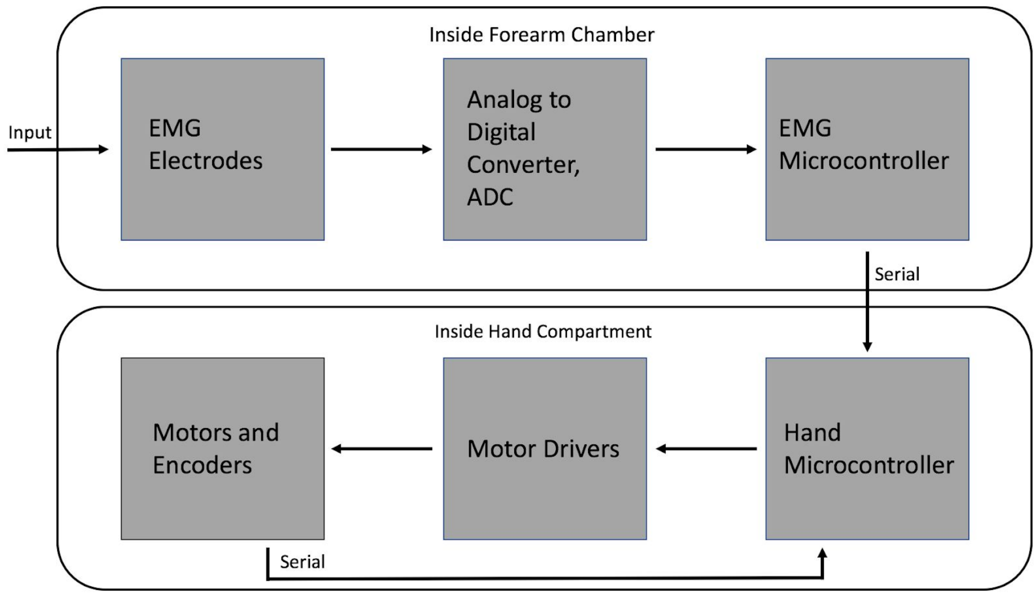

Figure 1. Block Diagram

The project consists of 3 major aspects, software, hardware and electronics. They are all important do to the fact that if one of them is missing, the rest wouldn't be able to work.

Software

The main parts of the software are:

Data interpretation

The data comes into the chip as an unsigned character, three of those characters create the actual number we need. Hence, we need to "clean" the data in order to be able to "see" what is actually happening.
First we receive the data through SPI, all 24 (3 per channel for 8 channels) unsigned characters are gathered in an array which is then sorted out in groups of 3. Transformed into binary, shifted, added up and finally sent through SPI using the terminal (if in training mode) or used in the machine learning algorithm (if in test mode).

Gesture Recognition

The machine learning algorithm can be divided into 2 parts. Training, and Testing. In order to train, the data is sent out of the chip through UART and read in the computer using a serial port.

This data is then entered into the machine learning algorithm which generates a certain group of weights and standard deviations to be used when testing, this is then copied into the program. For testing, the data stays within the chip, the new number is used to be multiplied to the current weights and that gives out the likelihood of each one of the gesture predictions, the highest likelihood is the one that is chosen as the predicted class.
We attempted to improve Naive Bayes by chunking (grouping) the values that ended in the machine learning algorithm. The data flows out of the EMG at a rate of 15 points per second. Then a prediction is generated for each one of them. Those predictions are then stored in an array. After 1 and a half seconds of holding a hand gesture, the mode is computed in that array. Since the efficiency of the Naive Bayes is about 72%, then the mode is usually the correct prediction for the hand gesture, increasing the accuracy to somewhere around 80%

Motor Movement

Motor movement speed is controlled by PWM signals, PWM basically changes the amount of power that goes in to the motor by regulating how often voltage is put out of the microcontroller. Depending on how fast the motor is moving, then the time for the finger to reach the next position can be varied. This is fine tuned in order to make the movement look more natural.

Hardware

This is a fully 3d printed prosthetic. All of the components were printed individually and then put together using pins, or other mechanical objects that can be found in any hardware store. The base is printed in PLA material, while the parts that have to endure the most activity (for example the joints) are printed in a PLA/Carbon Fiber mix.

Electronics

The leads of the system feed into and ADS 12983 that converts the analog signals into digital ones and then sends that to the Pic33. The Pic33 contains the machine learning algorithm as well as the motor control algorithm. The Pic33 sends PWM signals to the DRV8833 motor drivers that control the motors.

System Testing

Testing can be done in several different parts, those are:

Inter-module communication

In order to test the inter module communication, a logic analyzer is required. First you need to test that the ADS1298 board is actually sending out the information we need. In order to do this, the first step is to connect all leads to a common ground, then start the ADS, and connect the logic analyzer accordingly. All of the values that appear there should be around the 16M range.

Motor Control Fine Tuning

This is a trial and error process, run the motors for a 700ms at the PWM you desire, then adjust up or down accordingly. The motors should move the hand smoothly and once the full range of motion is achieved the motors should stop.

Machine Learning Accuracy

Software is provided in order to test, if the % accuracy is not above the 60% range, consider placing pads in different locations, or choosing a more distinct gesture.

User Manual

How to Train

In order to train, you have to introduce a “0” to the variable “training” in the software. Once that is done, connect to the computer and then just run the training software and follow the detailed instructions in there.

How to place pads

Place the pads at the end of your remaining limb. They should go around the arm with equal distance one from another.Place one of the leads on your elbow to act as the ground. Once they is placed, you are good to go.

Using the Hand

Once you have placed the pads and trained the gestures for the algorithm, plug in the battery to the arm and strap it to yourself.

Future Improvements

Future improvements include creating custom actuators to provide better movement of the fingers and reduce the stress experienced on the individual fingers. This includes encoders to tell the position of the finger to better control the fingers rather than the time delay that is currently used. Pressure sensors on the fingertips would aid with grip as the sensors could aid the motor control of when to stop when holding an object. We would also want to improve the leads by using higher quality pads that don’t degrade as quickly, as well as creating a sleeve to ease putting the leads on, as well as have greater consistency in placement. It would be better to use one motor driver per motor in order to be able to provide greater power to each motor without fearing reaching the current limit. Adding an encoder to the motors would also allow greater precision in moving the motors and eliminate guesswork when moving the motors.

Conclusions

This is the first iteration of our myoelectric powered hand that is affordable, and simple to assemble. We started off with the mechanically powered hand of the Enable group and modified it so that it is more natural and more functional. The total cost of the hand came to be under $500. With the ability to recognize 4 gestures based on muscle signals, it is comparable to similar designs in the $1500 range that can recognize 6 gestures. Due to its functionality as well as its affordability we view this design as a success and we were able to meet the goals we set out to accomplish.