NEURO-FUZZY GRASP CONTROL FOR A TELEOPERATED FIVE FINGER ANTHROPOMORPHIC ROBOTIC HAND

Maxwell Joseph Welyhorsky

Thesis submitted to the University of Ottawa
in partial fulfillment of the requirements for the degree of

Master of Applied Science in Electrical and Computer Engineering

Ottawa-Carleton Institute for Electrical and Computer Engineering
School of Electrical Engineering and Computer Science
Faculty of Engineering
University of Ottawa

Abstract

Robots should offer a human-like level of dexterity when handling objects if humans are to be replaced in dangerous and uncertain working environments. This level of dexterity for human-like manipulation must come from both the hardware, and the control.

Exact replication of human-like degrees of freedom in mobility for anthropomorphic robotic hands are seen in bulky, costly, fully actuated solutions, while machine learning to apply some level of human-like dexterity in underacted solutions is unable to be applied to a various array of objects.

This thesis presents experimental and theoretical contributions of a novel neuro-fuzzy control method for dextrous human grasping based on grasp synergies using a Human Computer Interface glove and upgraded haptic-enabled anthropomorphic *Ring Ada* dexterous robotic hand. Experimental results proved the efficiency of the proposed Adaptive Neuro-Fuzzy Inference Systems to grasp objects with high levels of accuracy.

Acknowledgements

I would like to say thank you to my supervisor, Dr. Emil M. Petriu, for his insight, advice, knowledge, and all the time he spent throughout the duration of my graduate studies for discussions and reviews.

I would like to thank Sindhu Radhakrishnan, for the brainstorming sessions, as well as her family for their kindness and giving me their time to answer my questions.

I would also like to thank all my friends that provided encouragement and entertainment throughout this process.

Finally, a very important thank you to my family, and my parents, for their support. At every point I knew you were encouraging me. I always knew you were there.

Table of Contents

List of Figures

List of Tables

Nomenclature

Abbreviations

A/D	Analog to Digital
ANFIS	Adaptive Neuro-Fuzzy Inference System
ANN	Artificial Neural Network
D/A	Digital to Analog
DC	Direct Current
DOF	Degrees of Freedom
FL	Fuzzy Logic
FSR	Force Sensing Resister
GA	Genetic Algorithm
GUI	Graphical User Interface
HCI	Human Computer Interface
IMF	Input Membership Function
IMU	Inertial Measurement Unit
MARG	Magnetic, Angular rate, and Gravity
MF	Membership Function
MISO	Multiple Input Single Output
NFS	Neuro-Fuzzy System
NRMSE	Normalized Root Mean Square Error
OMF	Output Membership Function
P2P	Peer to Peer
PC	Principle Component
PCB	Printed Circuit Board
PCA	Principal Component Analysis

PLA	Polylactic Acid
RMSE	Root Mean Square Error
ROS	Robot Operating System
RPC	Remote Procedure Call
SDK	Software Development Kit
SSR	Sum Squared Regression
SST	Total Sum of Squares
USB	Universal Serial Bus

Chapter 1 Introduction

The 1920 play Rossumovi Univerzalni Roboti [1], which can be translated to English from the Czech language as “Rossum’s Universal Robots”, was the first time the term ‘robot’ was used. The play refers to robots as artificial people made in a factory but having the capability of thinking and speaking. This is closer to the idea of androids, artificial beings composed of organic material, than that of modern-day humanoid robots, which resemble the shape of a human but are constructed from mechanical parts.

The rapid expansion of automation in industry during the Information Age reshaped the image of a robot. These robots performed tasks through computer programming but were not necessarily of an anthropomorphic nature. The earliest known industrial robot began operating in 1937 and was a device that resembled a crane with a gripper able to perform the simple task of stacking blocks [2]. The crane and gripper attachment were capable of basic hand operations.

There are several aspects that could be used to classify a hand as a hand whether it be the fingers, palm, the opposable thumb, as well as the ability to grasp, hold, and manipulate objects. The in-hand object manipulation is a unique defining quality for humans. The potential for robots to have similar dexterity as humans would allow them to have an immense positive impact in society, as nearly every tool is designed to be

manipulated with a human hand.

The first progress towards a realistic robotic hand was reported in 1977 by Asada and Hanafusa [3]. Mason and Salisbury continued this with a three finger robotic manipulator [4]. The work of these robotic pioneers is what has led the pursuit for true human-like robotic grasping capabilities. These hands were capable of stable grasp, and a more complex grasp than that of a three-clawed gripper.

The *manipulation paradigm* is defined as a research field that categorizes the different challenges and questions involved in robotic grasp [5]. Within this field, and with the aim of creating a human-like robotic grasping ability, *Task Planning, Hand Design, Grasping Issues, Control System, and Sensing* must all be considered.

While fully autonomous grasping is a topic yet to be solved, several areas of research continue to expand, including teleoperation and haptic exploration. Both of these topics significantly consider the five elements of the manipulation paradigm.

Haptic exploration is a key aspect in the perception of object surface properties using sensory devices, which allows a robotic system to learn about unknown objects in the surrounding environment. Humans have a very complex haptic sensory system, and to mimic sensor-based perception capabilities of humans would significantly enhance the ability of a robotic system to feel the shape, texture, and malleability of unknown objects while grasping them.

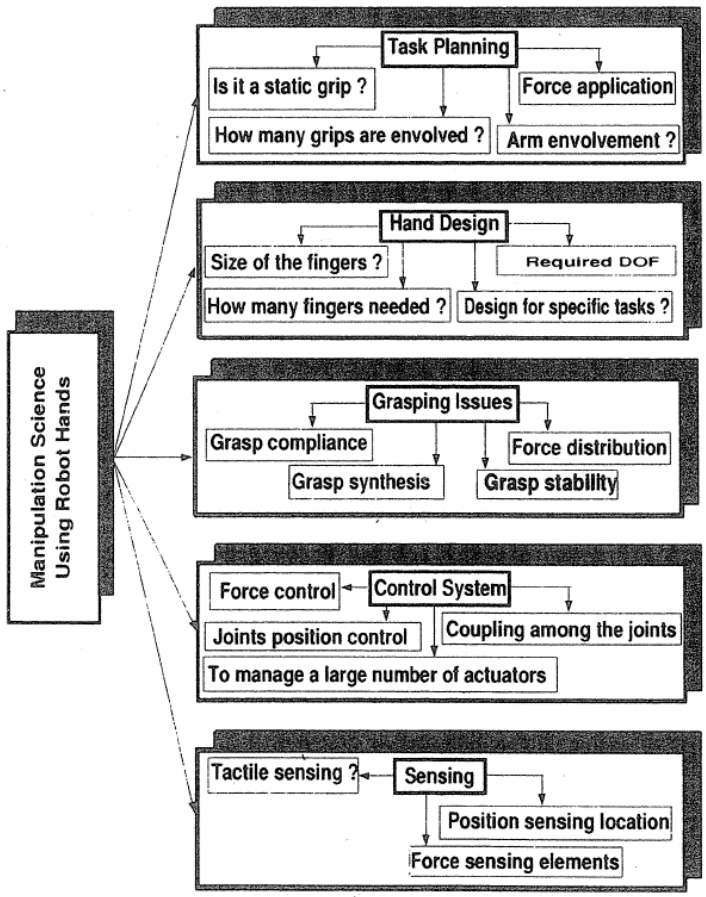


Figure 1.1 Manipulation paradigm [5]

Haptic exploration is exceptionally useful when operating in unstructured environments, as it does not necessarily utilize vision methods that would be useless in reduced lighting scenarios such as at night or underwater. This type of contact environment exploration, also referred to as haptic perception, has particular challenges for the Task Planning and Control Systems modules of the robotic manipulation paradigm shown in *Figure* 1.1 [6].

Robotic Telemanipulation aims to combine low-level robot computer control with higher-level perception and task planning abilities of a human operator equipped with adequate *Human Computer Interfaces* (HCIs) [7]. A multimodal, tactile, force, and kinesthetic HCI allows a human operator to interactively control the specialized robotic hand to perform dexterous manipulation operations.

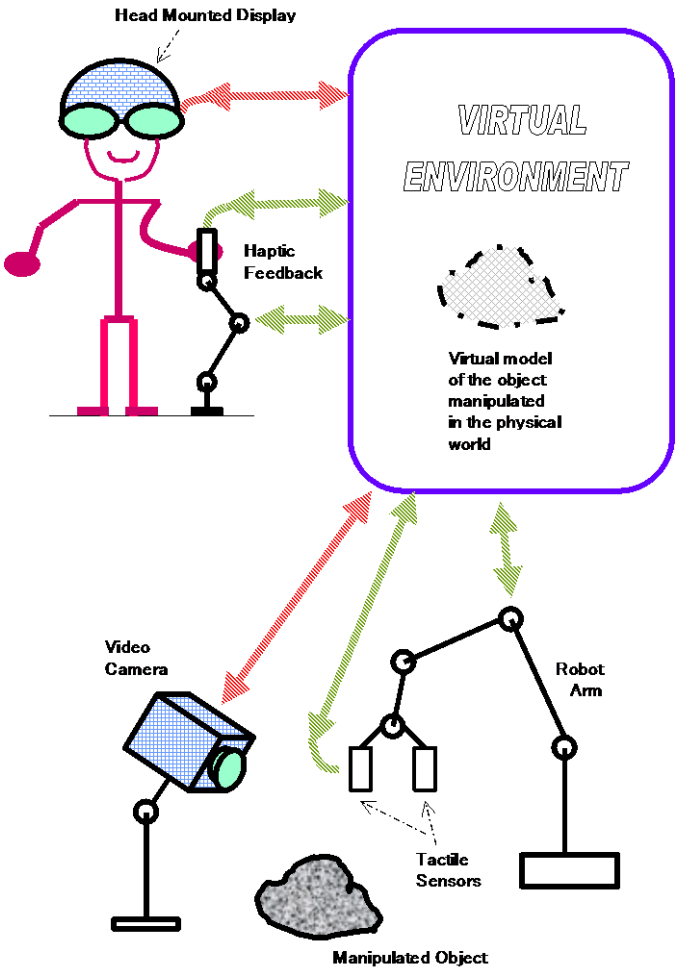


Figure 1.2 Robotic telemanipulation [7]

The HCI shown in the *Figure* 1.2 allows the human teleoperator, equipped with haptic feedback and head mounted display, to remotely control the robotic arm, which is equipped with tactile sensors and a video camera. Such robotic telemanipulation systems have various applications, for instance, handling hazardous material, difficult to reach environments, hands on virtual learning, and telemedicine, including surgeries.

1.1 Problem Definition

This thesis will address the specific problem of applying a machine learning technique to a dexterous anthropomorphic underactuated robotic hand system controlled through telemanipulation. It will use an Adaptive Neuro-Fuzzy Inference System to train grasp synergies that will map the human hand data glove to the robotic hand system. This thesis will also address the five elements of the manipulation paradigm: Task Planning, Hand Design, Grasp Issues, Control System, and Sensing shown in *Figure* 1.1.

The desire to have more human-like robots is ever prevalent in research today. To allow human operators to be removed from dangerous or uncertain environments while performing tasks would be of great practical value. The ability to exploit this value is limited by the complexity of dextrous manipulation. Traditional methods of human-like grasping require fully-actuated solutions involving many joints and actuators [8]. These solutions are bulky, expensive, and while they can perform the same full range of motions as a human, they are not actually particularly human-like.

Another challenge for efficient grasping by anthropomorphic dexterous robotic hands is that the machine learning models are often limited to fit a particular object or grasp. This prevents truly human-like grasping, as humans are capable of performing a variety of grasps on different objects with an incredible amount of precision and dexterity.

1.2 Thesis Objectives

The aim of this thesis is to combine the functional dexterity of a human-like haptic-enabled robotic hand with the higher-level perception and task planning abilities of a human teleoperator equipped with an adequate human-computer interface. The main objective was to study a neuro-fuzzy machine learning model to apply to the haptic-enabled teleoperation of the new experimental hand a *CyberTouch* glove [9].

1.3 Thesis Contributions

This thesis makes theoretical and experimental contributions to the development of a novel neuro-fuzzy control method for dextrous human grasping based on grasp synergies using a Human Computer Interface glove and upgraded haptic-enabled anthropomorphic *Ring Ada* dexterous robotic hand. Experimental results proved the efficiency of the proposed Adaptive Neuro-Fuzzy Inference Systems to grasp objects with high levels of accuracy.

1.4 Publications Arising from the Thesis

The following publication, [10], was produced based on the work presented in this thesis:

(1) Qi Zhu, Vinicius Prado da Fonseca, Bruno Monteiro Rocha Lima, **Maxwell Welyhorsky**, Miriam Goubran, Thiago Eustaquio Alves de Oliveira, and Emil M.

Petriu, “Teleoperated Grasping Using a Robotic Hand and a Haptic-Feedback Data Glove,” *Proc. 2020 IEEE Int. Systems Conf. (SysCon),* Montreal, QC, Aug. 2020, pp.1-7, doi: 10.1109/SysCon47679.2020.9275927

1.5 Thesis Organization

The thesis is organized as follows:

Chapter 1. Introduction. This chapter presents the problem definitions, thesis objectives, and summarizes the contributions.

Chapter 2. ***Literature Review***. This chapter presents a review of major research developments reported in the literature on human hands, robotic hands, vision and tactile feedback for robotic manipulation, the dexterous multi-fingered object grasping and manipulation from the synergy aspects, and the machine learning methods used for robotic grasp operations.

Chapter 3. System Overview. This chapter presents an overview of the hardware; the *CyberTouch* glove, and novel upgraded *Ring Ada* robotic hand, and of the Robot Operating System (ROS) based software used by the experimental system setup.

Chapter 4. Teleoperated Grasping using the Ring Ada Robotic Hand. This chapter further details the work of the research team reported in [10]. It includes the experimental results of the kinematic and postural synergy designs, as well as the feedback strategy used during the robotic hand grasping.

Chapter 5. Neuro-Fuzzy Grasp Control. This chapter discusses the effectiveness of using ANFIS for the sensor based synergistic control skills developed for the *Ring Ada* robotic hand.

Chapter 6. Conclusion and Future Work. The final chapter includes conclusions and provides ideas for future developments of the multimodal haptic HCI controlled *Ring Ada* robotic hand from a hardware point of view, as well as expanding the proposed machine learning method that would enhance the overall grasp and dexterity behaviours of the robotic hand.

Chapter 2 Literature Review

Nearly every object in our environment is designed to be touched and manipulated with our hands. Dexterous hands give humans a unique advantage over other creatures. We can perform great feats of strength while simultaneously retaining the ability to be extremely delicate. We can grasp an array of large objects, but also manipulate extremely small components. These grasp skills, the ability to seize and hold firmly or control skillfully, are preprogrammed into us. At birth, a baby can hold an object when placed in the palm; a reflex called “palmar grasp”. This evolutionary advantage is developed over time and taught to us.

Continuous developments are being made in the field of humanoid robotics to replicate that evolutionary advantage. The one key function for these robots is the use of hand held tools. In order to do that, these robotic hands and prosthesis used a variety of approaches to map hand movement, many of which involve machine learning. In order to advance the field of robotics, we must understand the human hand, the research involved in grasping, and the future of machine learning in robotic hands.

2.1 The Human Hand

In anatomy, the human hand is defined as “the portion of the upper limb distal to the radiocarpal joint, comprising the wrist, palm, and fingers” [11]. The human hand has 27 bones, 14 phalanges for the fingers and thumbs and 13 metacarpal and carpal bones defined in [12]. More commonly heard when discussing an infantry formation from Alexandrian tactics, the Greek word “phalange”, or “phalanx”, literally means “log” and describes the bones in our fingers. Latin “carpus” is derived from the Greek “karpo” which means wrist, where the root “carp“ translates to “pluck” - the action of the wrist.

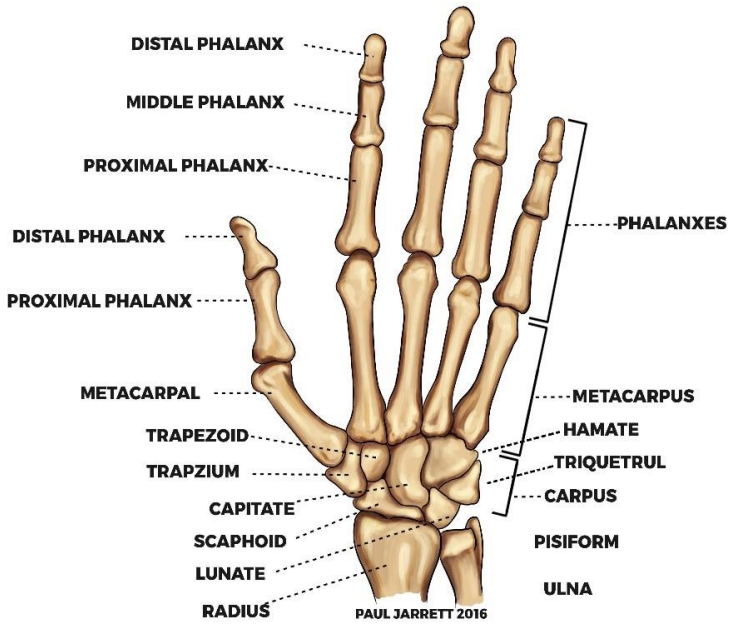


Figure 2.1 Human hand bone anatomy of the right hand [12]

Hands are often anthropomorphized in animals. Whether they are the paws of dogs, the talons of birds, or even to the claws of crabs and the praying mantis, they all have a grasping ability we characterize as a hand. However, in primates, the hand is unique. The hands of apes, monkeys, and humans have opposable thumbs [12]. This allows for advanced dexterity, grasping, and intricate hand movements that could never be possible to other creatures.

Every motion that is made with our hands has a certain level of complexity. *Figure* 2.2 shows the participation of the digits on a human in the form of four grasp types which clearly demonstrates the covariance between thumb and fingers [13].

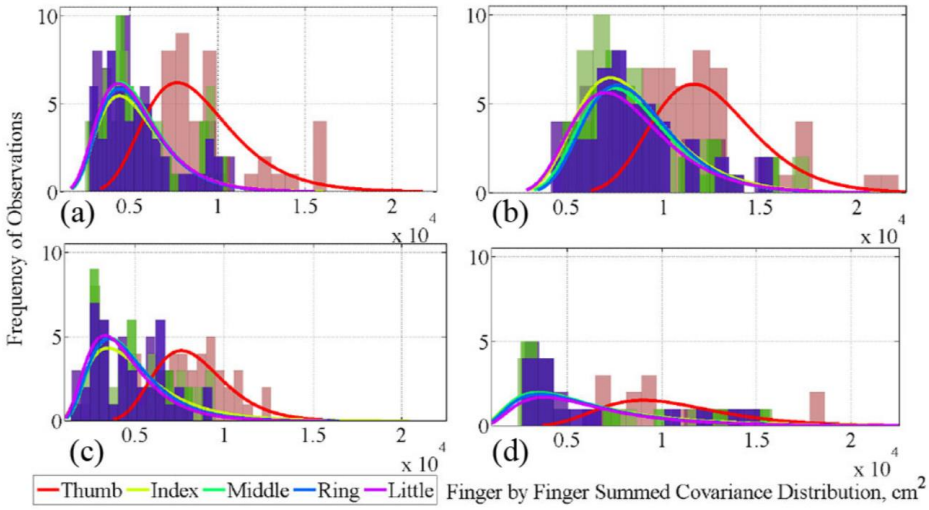


Figure 2.2 Participation of each finger during grasping (a)(b)(c)(d) power, precision, key and primate grasp respectively [13]

The motions able to be made by the hand are referred to as Degrees of Freedom (DOF) – independent movements of joints. The human hand has 27 DOF [14]: four in each finger, three for extension and flexion, and one for abduction and adduction. The

thumb has 5 DOF and is more complicated than the fingers. The wrist contains the final 6 for rotation and translation. *Figure* 2.3 shows the simplified kinematic structure.

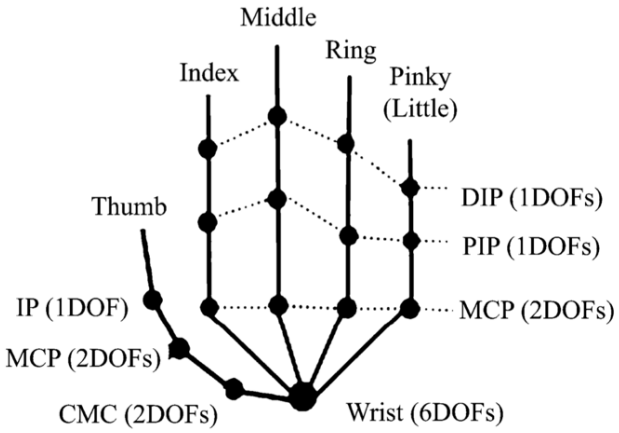


Figure 2.3 Human hand kinematic structure [14]

These different types of motion allow us to grasp. Everyday tasks hav been extensively studied and categorized since 1989 [15] where Cutkowsky created the Taxonomy Grasp Tree as shown in *Figure* 2.4.

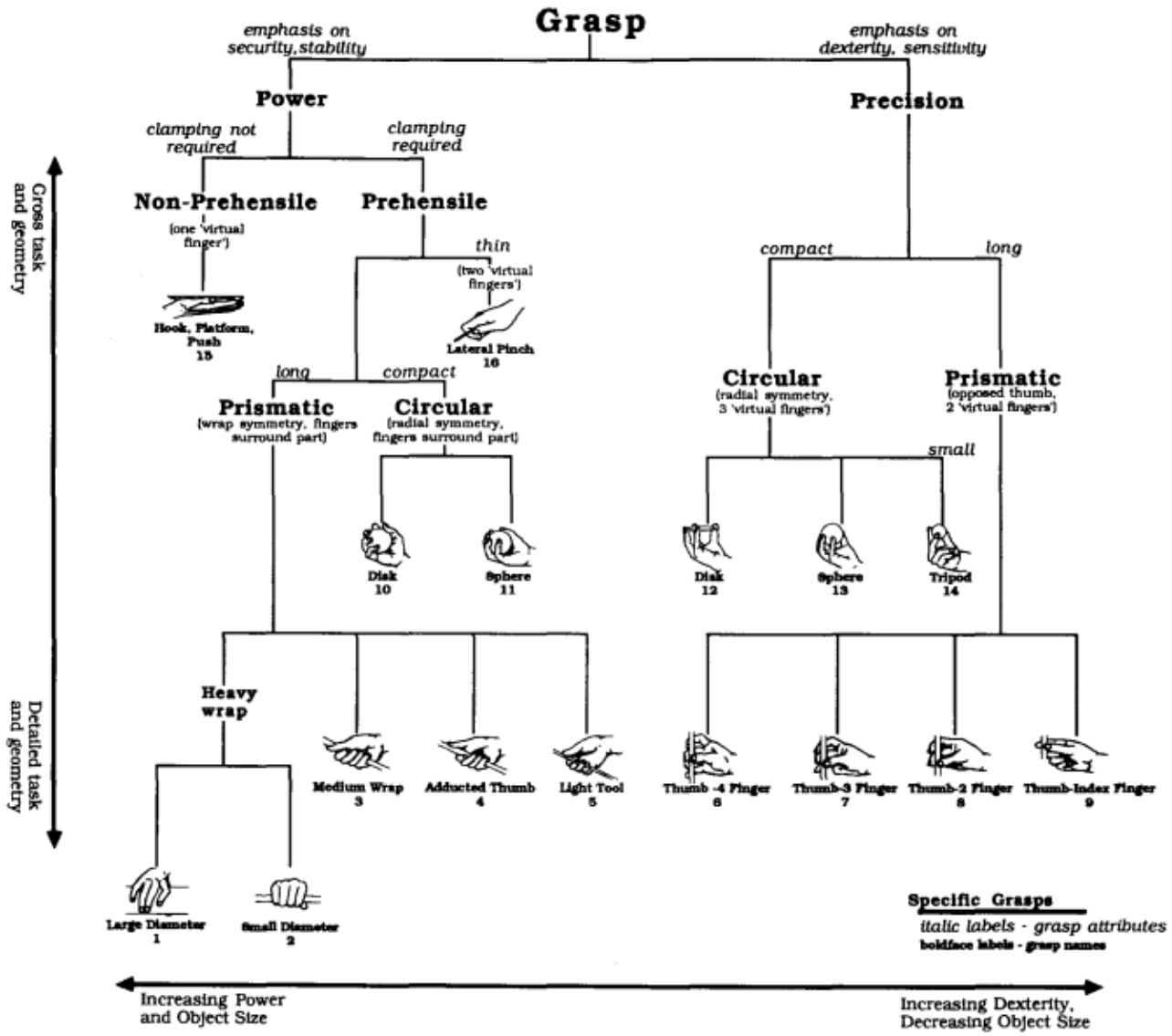


Figure 2.4 Taxonomy of grasp [15]

2.2 Robotic Hands

While robotic manipulators have been used for decades, the advancements in multi-fingered and anthropomorphic dextrous robotic hand designs have become more prevalent. A survey of bio-inspired robotic hands [16] lists several reasons why this may be, including; other manipulators like parallel jaw grippers have limited range of configuration, large industrial robots lack elements of sensitivity and space saving, the difficulty or inability to perform dextrous operations, and other manipulators not being suited to perform in unstructured environment.

Hand designs for the movement of finger and hand joints come in two forms, tendon-driven or actuator-based. Tendon-driven joints rely on a stiff or taut length of rope, wire, or plastic used in two ways: passive, using elastic behaviour, or active, being pushed and pulled by an actuator, to drive movement. There are several different types of tendon-driven joints described in a review of tendon driven robotic hands [17]. Actuator based joint movement is simply having an actuator at each joint performing control of one of multiple DOF.

Whether a hand design uses a tendon-driven or an actuator based joint movement, the robotic hand can be further classified as fully actuated or underactuated. Fully actuated designs have equal control inputs to degrees of freedom. Underactuated means the degrees of freedom exceed the number of control inputs.

The Shadow Hand [18] is a tendon-driven robotic hand which has 24 DOF. There are 20 actuated DOF and 4 underactuated movements. This hand contains a number of sensors for gyro and accelerometer, joint angle, force-torque, tactile, and temperature [19]. Being one of the most advanced robotic hand designs, the 4^{th} finger joint of the Shadow Hand makes the hand particularly different. This joint allows abduction and adduction that allow the ability to turn or rotate an object in the palm of the hand.

The KITECH-Hand [20] is an example of a fully actuated robotic hand with three fingers and thumb, and 16 independent torque-controlled joints. A small DC motor actuates each joint. The inexpensive solution of small DC motors, while maintaining good performance allowed this hand to be one of the first commercially available hands [21]. This hand design also has the ability for modular sensor selection of the fingertips.

2.3 Teleoperation

Teleoperation is the control of a device or machine remotely. Telemanipulation refers to the remote operation of a robotic hand and fingers. Performing tasks at a distance through teleoperation has always had benefits such as operator safety or increased efficiency. The opportunity for applications significantly increases when attempting to have the same dexterity and mobility as a human hand. People rely on visual and touch feedback every day for the most basic tasks, yet robotics have yet to achieve a perfect way to replicate the same feelings and abilities through telemanipulation let alone autonomously.

Teleoperation can be dated as far back as 1898, when Nikola Tesla developed a

remote-controlled boat [22]. Telemanipulation has much more recent roots in the late 1970s, when Guittet performed basic actions of grasping and moving objects with an industrial robot [23]. Soon after pick-and-place experiments showed position control using joysticks [24].

Telemanipulation research resulted in many current telemanipulation papers based on aspects of human manipulation including stable grasp, slip detection, and object recognition using anthropomorphic dexterous robotic hands [12], [25], [26], [27].

Qi Zhu et al. recently published [10] results of their research on teleoperated grasping using a robotic hand and haptic feedback data glove, which uses a modified 3D printed five fingered robotic hand to perform stable grasping in a similar way to Liarokapis et al. [28] who performed similar feats of teleoperation and grasping using a five fingered robotic hand. One main difference is that the effort of Liarokapis is mapping the kinematics of the human hand. Both papers use a *CyberTouch* glove for haptic feedback to the operator.

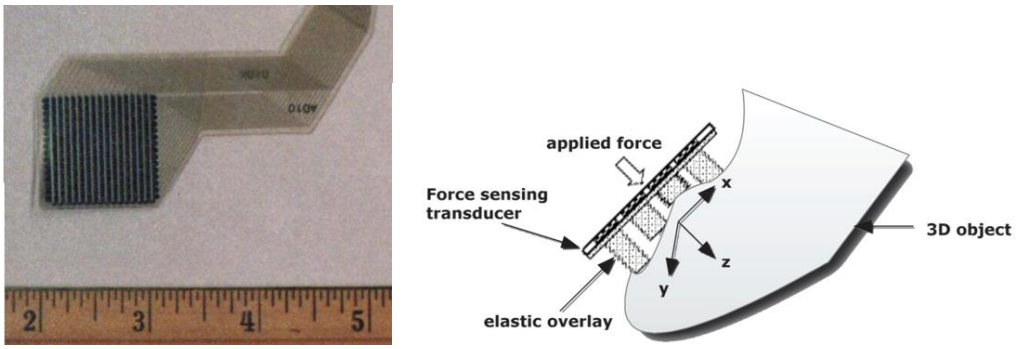
2.3.1 Vision

As real time video capabilities and data speed have increased, vision-based manipulation is now more effectively used in industrial robotics allowing for improved precision and productivity [29], [30], [31]. Vision-based manipulation with under actuated grippers use a camera to control the motion of the grippers to find and grasp the object [32]. A peripheral vision-based grasp force feedback method for telemanipulation, or interaction in virtual environments, was developed by Galambos et al. [33].

2.3.2 Tactile Sensors

Tactile sensing is a key feature for the robotic in-hand object manipulation. There are many types of sensors with the most recent work being done with multimodal sensing [34]. These advanced sensors are used to mimic human touch and feedback. They provide feedback for the force control of the robotic hand during the manipulation. Tactile feedback-based control consists of applications such as grasp stability estimation, and tactile surface exploration which is done by sliding a fingertip across an objects surface [35], [36].

Using tactile sensing is crucial for unstructured environments where visual feedback is not always available. However, it is possible to glance at an object with tactile sensing. This is commonly referred to as a “haptic glance” [37].



(left) *(right)*

Figure 2.5 FSR array (left), and the tactile sensor based on FSR array (right) [38]

A simple popular tactile sensing solution is a Force Sensing Resistor (FSR) array shown in *Figure* 2.5 (left). The resistance of the FSR changes when a force is applied to

the surface. A single FSR covers a relatively small surface area with low precision. A custom-designed elastic overlay having a tab on top of each node of the FSR matrix as shown in *Figure* 2.5 (right) [38] allows the material to expand without any stress in the x and y directions making possible compression in the Z-direction proportionally with the normal stress component.

Advancements are constantly being made in skin type tactile systems, particularly for pressure sensitive applications. Wu et al. [39] presented a novel design for a tactile sensor capable of measuring an ultrawide pressure range for areas from respiration to finger heart rate with excellent performance sensitivity. Other known pressure-sensitive tactile sensors for surface areas are the TeskanFlexiForce™ pressure ink technology [40], and the Quantum Tunnel Composites ® technology [41].

Figure 2.6 shows the biology-inspired multimodal tactile-sensing BioTac SP (Single Phalanx) sensor [42]. Multi modal sensing is the idea of using two or more sensing items to create sensory feedback, similar to how a human hand can tactilely perceive information. An advanced multimodal sensor can achieve thermal, pressure, tactile, and proximity sensing data. The BioTac [42] “provides simultaneous information about contact forces, microvibrations, and thermal fluxes, mimicking the full cutaneous sensory capabilities of the human finger.”

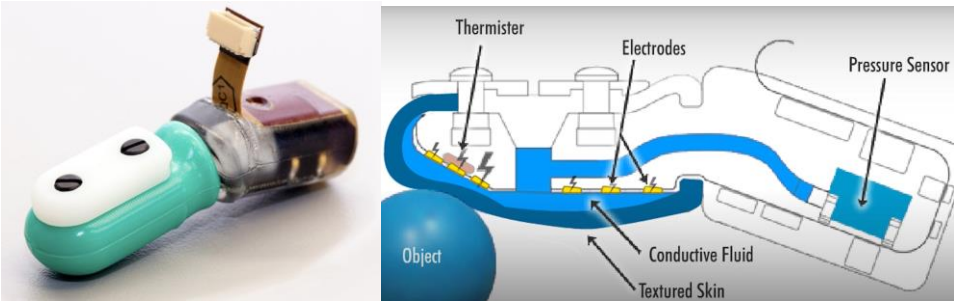
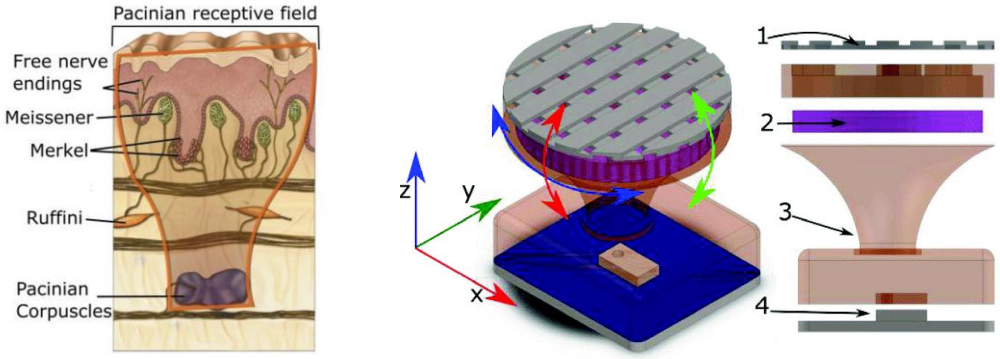


Figure 2.6 BioTac SP sensor from SynTouch [43]

The BioTac contains a conductive fluid that serves two purposes, contact force sensing and vibrations. The fluid changes thickness when a force is applied which causes a shift in impedance and measured by electrodes. Pressure transducers measure vibrations. Thermistors are used for temperature of touched objects [43].

Figure 2.7 presents a biology-inspired multimodal tactile sensor developed by T.E. Alves de Oliveira [44], inspired by the mechanoreceptors of human skin [45]. *Figure* 2.8 (right) shows the layers of the sensor: a 32-taxel tactile array, a MARG (magnetic, angular rate, and gravity), a passive compliant structure which helps solve the inverse problem in tactile sensing [46], and a barometer. The compliant structure consists of IMU and the barometer covered by polyurethane rubber.



(left) *(right)*

Figure 2.7 Human skin mechanoreceptors (left) [54], and Multi-modal tactile sensor module: 1 - tactile array; 2 - inertial measurement unit (IMU); 3 - cone compliant structure, and 4 - deep barometer (right) [44]

2.4 Synergy in Grasping

Synergy between components can be understood as multiple components working in tandem; the concept that the combined value and performance of the multiple parts will be greater than the sum of the separate individual parts.

Human-Robot Synergy “combines what humans are good at with what machines are good at. Prior knowledge, a great deal of "common sense", creativity and dexterity all are attributes that ordinary people have, but not machines. On the other hand, rapid computation, mechanical power, diverse sensory modalities and resistance to hazards are great advantages of machines that people do not possess” [47].

Three types of synergies have been proposed to map human hands to anthropomorphic robotic hands: **postural**, **kinematic**, and **dynamic synergies**.

In order to achieve robotic grasping that best mimics a human grasp, significant effort in characterizing posture of a hand has been made based on the definition by Napier of prehensile movements of the human hand [48], not to be confused with the grasp taxonomy. Santello et al. [49] show how **postural synergies** regulate the general shape of the hand, while also stating that “because the postural synergies did not coincide with grip taxonomies, the result suggest that hand posture may be regulated independently from the control of contact forces that used to grasp an object”.

Santello et al. recorded data for angular position of the fingers and thumb in order to measure static hand posture. They found joint angles did not vary independently of objects even though hand shapes were distinct. They performed a Principal Component Analysis (PCA) of 57 objects, reducing the dimensionality of datasets, increasing interpretability, and minimizing information loss while preserving the variability [50]. They found that the first two components; apple and banana, could account for over 80% of variance in grasp. Components are written as PC1 and PC2.

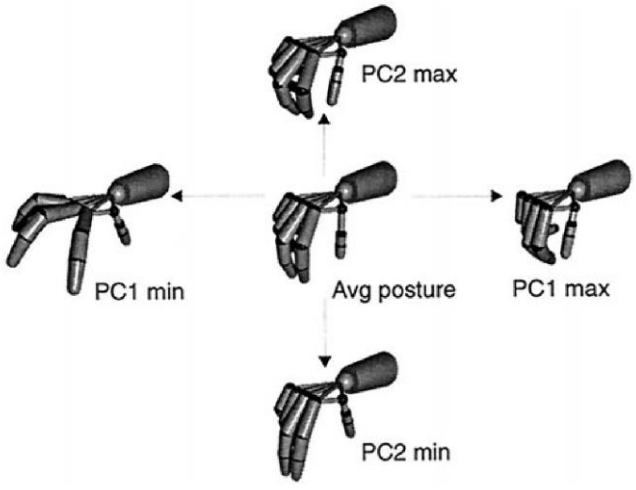


Figure 2.8 PC1 and PC2 [49]

Kinematic synergies describe the correlations between the joint angles during multi-joint finger movements, as there is a characteristic relation between maximum distance of the tips of the fingers on their way to an object and size of an object that is to be grasped [51]. *Figure* 2.9 depicts a defined PC1 which accounts for 97.8-99.7% of angular variance [52]. The trials consisted of PC1 closing the thumb and index finger.

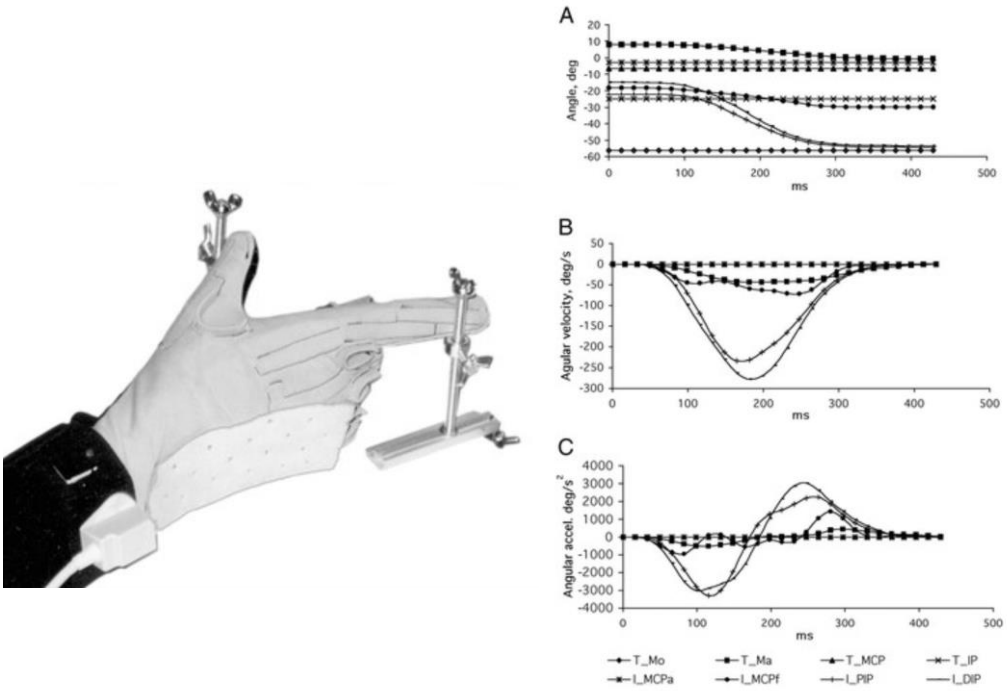


Figure 2.9 Human subject kinematics [52]

A: Thumb and index finger joint angles averaged over 20 trials of closing movements with normal velocity and normal aperture; B: Corresponding angular velocities; and C: Corresponding angular accelerations

Dynamic synergies were defined as stable correlations between the hand joint torques during precision grip movements [52]. Joint torques were found to be less correlated than joint angles.

These synergy types use a reduced number of parameters mapped from human and

robotic hand in order to provide efficient control. These parameters are mapped between hands with one of three techniques. J*oint-to-joint mapping, Cartesian mapping,* and *object-based mapping* [53] shown in *Figure* 2.10.

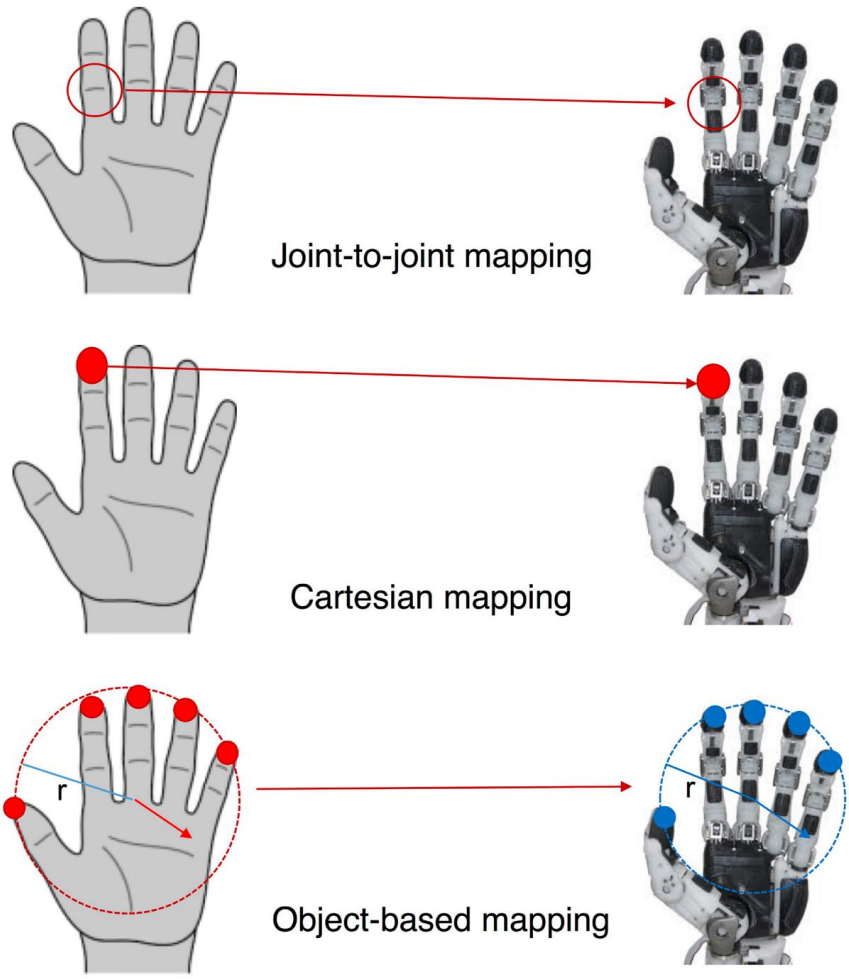


Figure 2.10 Schematic representation of the principal mapping techniques [53]

2.5 Neuro-Fuzzy Control

A Neuro-Fuzzy System (NFS) is a combined architecture of Fuzzy Logic (FL) and Artificial Neural Network (ANN) elements, which can both be used for cases where no mathematical models exist including pattern recognition, and regression. Many architectures of NFS have been developed [54]. The first instance of robotic grasping using Neuro-Fuzzy control was reported in 1993 [55].

Coincidentally, Adaptive Neuro Fuzzy Inference Systems (ANFIS) were also first presented in 1993 [56]. ANFIS is a well-studied architecture, *Figure* 2.11, which resembles a feed forward neural network with supervised learning capabilities.

A first order Sugeno model can be described with an example of two inputs, x and y, and one output

Rule 1: $If\ x\ is\ A_1 and\ y\ is\ B_1, then\ f_1 = p_1x + q_1y + r_1$

Rule 2: $If\ x\ is\ A_2\ and\ y\ is\ B_2, then\ f_2 = p_2x + p_2y + r_2$

The overall output can be expressed as

$$f = \frac{w_1}{w_1 + w_2} f_1 + \frac{w_2}{w_1 + w_2} f_2 = \bar{w}_1 f_1 + \bar{w}_2 f_2$$

$$= \bar{w}_1 x p_1 + \bar{w}_1 y q_1 + \bar{w}_1 r_1 + \bar{w}_2 x p_2 + \bar{w}_2 y q_2 + \bar{w}_2 r_2$$

Where w_i is the normalized firing strength from layer 3, and p_i, q_i, r_i are the parameters of the node [57], [58].

The ANFIS structure has 5 layers, layer 1: **input nodes**, layer 2: **input terms (partition) nodes**, layer 3: **rule nodes**, layer 4: **normalization nodes**, and layer 5: **output node** [59]

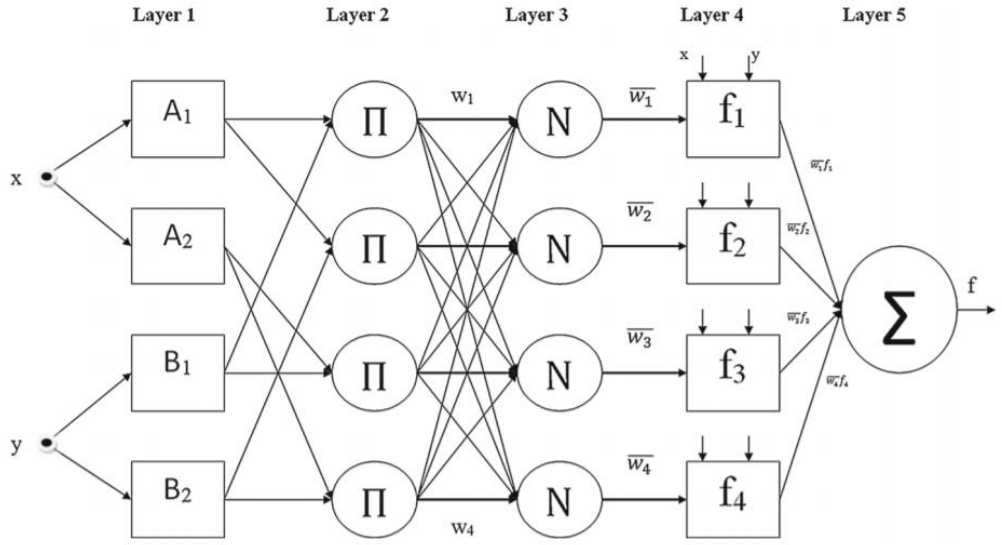


Figure 2.11 Structure of an Adaptive Neuro-Fuzzy Inference System [58]

The first layer maps the input to each membership function which can be referred to as the fuzzification layer. The rules are created in the second layer. The third layer normalizes the rule strengths. The fourth layer performs the defuzzification. The final layer is the sum of those parameters in the form of the global output [58].

Using different algorithms such as grid partitioning or subtractive clustering provide ANFIS the ability to generate membership functions. A Neuro-Fuzzy Controller uses the ANN to tune the membership functions while keeping the basic elements of the fuzzy logic controller intact [60].

Neuro-Fuzzy grasp control with pressure sensing was tested and validated in 1993 [55], where a best first grasp for a robotic hand with pressure sensing was determined by assigning fuzzy membership values to aspects of candidate grasps attempted with a modified genetic backpropagation neural network controller. ANFIS has been used in prediction of the grasping object weights by an underactuated system that is a new sensing capability [61].

ANFIS has also been employed as a means of control in prosthesis grasping, using the electrical signals in the arm [62] – [64]. A hybrid control technique using ANFIS and genetic algorithms (GA) were used in the grasp control for a smart prosthesis [65].

Chapter 3 System Overview

This chapter presents the hardware and software aspects of the experimental teleoperation setup. The aim of this experimental system architecture was to combine human-like dexterity and tactile feedback. The two main system components are discussed in section *3.1 Hardware* and section *3.2 Software*.

The *hardware subsystem* consists of three main components: (i) the *CyberTouch* glove, (ii) the upgraded *Ada* robotic hand, herein called the *Ring Ada*, and (iii) the computer-based controller. The software subsystem consists of a *Robot Operating System* (ROS) framework for the custom-developed drivers for the *Ring Ada hand* and *CyberTouch* data glove, and a *Python* software module developed to provide the grasp teleoperation. More details are presented in Section 3.3.

3.1 Hardware

Many Human Computer Interfaces (HCI) have been developed over the years allowing humans to more naturally interact with different computer applications [66]. These interfaces range from implanted devices in the brain to smartphone apps providing for better, more natural, interactions between humans and machines.

Well before the emergence of HCIs, it was known that the objective evaluation of performance on a task was the time taken and the error rate of the actions involved in the task [67]. The speed and accuracy being inseparable as the objective measures for a

meaningful performance evaluation. As computer technology advanced, computers assisted in the tasks so fewer errors were made. It is at this point when HCI design came into consideration. A good HCI would provide an easily interpretable and reliable task-related interface. As a result, the learning curve will be reduced, and time for tasks should decrease.

Human Computer Interaction is now the combination of computer science, cognitive science, and human factors engineering (HFE) [68].

3.1.1 CyberTouch Glove

The *CyberTouch* data glove was chosen as it has small vibrating tactile sensors on each of the fingers and on the palm, which can be individually programmed. The glove is able to record the real-time finger-joint angles of the human operator hand and provide linear vibration feedback to both the fingers and the palm, as shown in *Figure* 3.1.

The *CyberTouch* glove has 18 embedded flexible sensors positioned between the joints of each finger. One sensor is positioned between the intermediate phalange and the proximal phalange, and the other between the proximal phalange and the metacarpal joint.

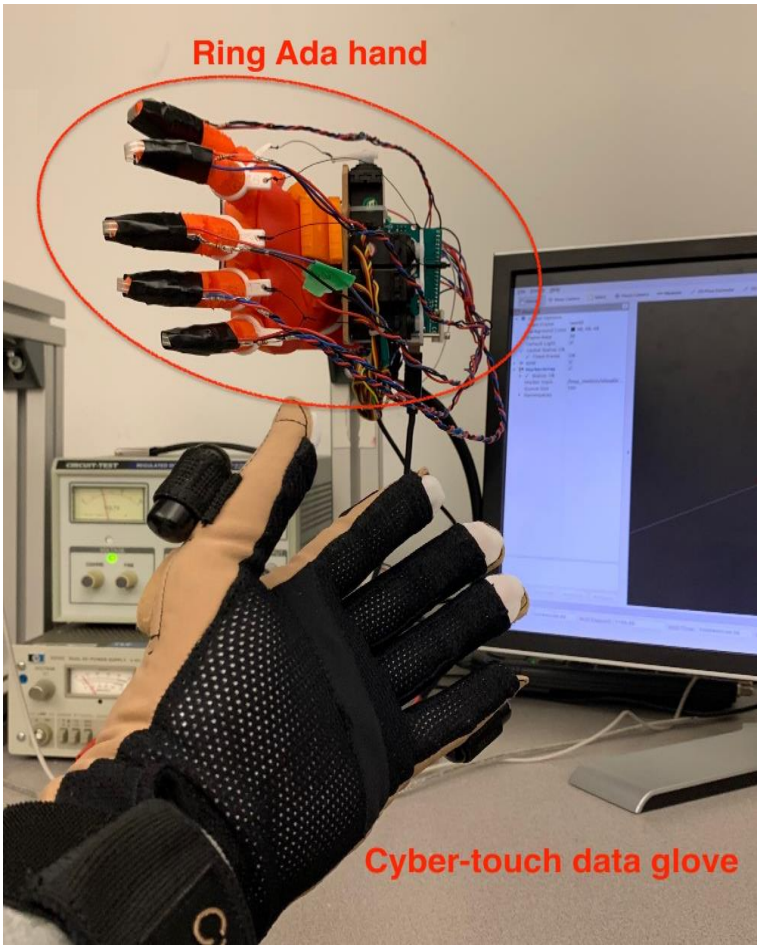


Figure 3.1 The experimental setup

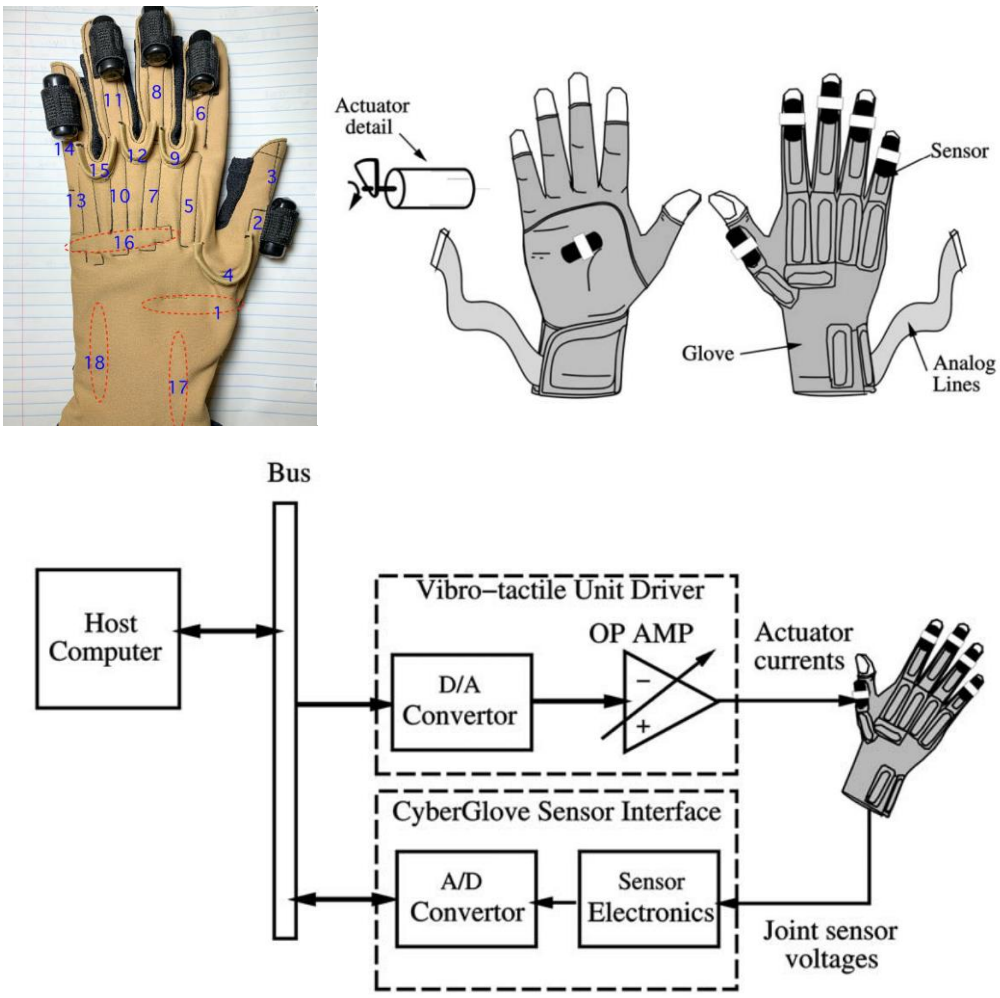


Figure 3.2 The CyberTouch Glove (top) and the Control Interface (bottom) [69]

Figure 3.2 (top) shows the 18 joint sensors and the 6 vibrating actuators built into the *CyberTouch* glove, which provides the haptic feedback to the hand of the human operator [69]. The red dotted circles indicate the sensors inside of the glove structure. One vibrating actuator is placed on each fingertip of the glove and the sixth is placed in the palm of the glove.

Figure 3.2 (bottom) shows how the data is transferred between sensors and actuators from the host computer using a Serial-to-USB protocol. The joint angle sensor

data which goes through the Analog to Digital (A/D) converter has an 8-bit format. The built-in vibrating actuators positioned at each tip of the glove and the center palm provide linear intensity vibrations controlled by 8-bit digital signals coming from the host computer via the Digital to Analog (D/A) converters.

3.1.2 Ada Hand

The open-source 3D printed anthropomorphic robotic hand is provided by Open Bionics as part of the Open Hand Project [70]. The underactuated tendon-driven hand is based closely on the human hand which makes it a viable candidate for easily implementable teleoperation given a few requisite modifications.

This hand has the ability to close relatively quickly and to make basic grasping operations due to the performance of the linear motors housed in the 3D printed palm – where all the components are located as shown in *Figure* 3.3 [71].

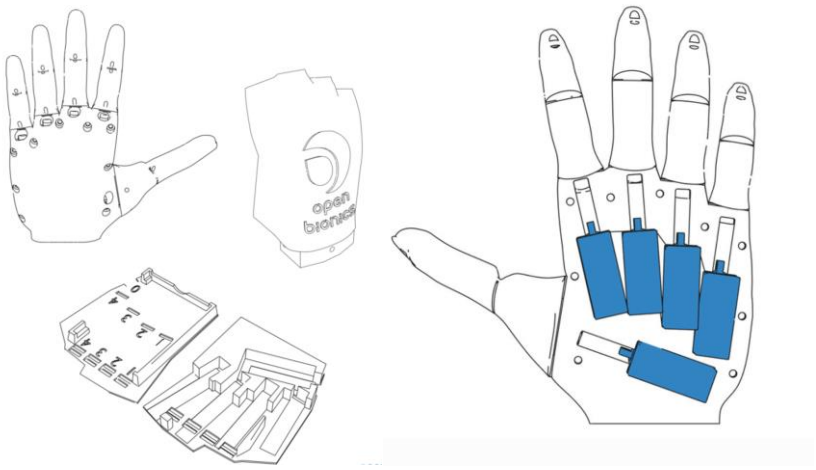


Figure 3.3 Ada hand structure (left), and the linear motor layout inside the palm (right) [71]

The hand was 3D printed using two different materials: (i) a harder Polylactic Acid

(PLA) material was for the back cover of the hand, the Printed Circuit Board (PCB) tray, and for the upper and the lower part of the hand, and (ii) a more compliant elastic Ninja-Flex material, which provides the ability to deform and spring back to original shape was used to print the palm.

Figure 3.3 (right) depicts the placement of the linear actuators used to power the fingers and thumb. The linear actuator motors operate with a 12V input and have a usable stroke length of approximately 18 mm. Each motor also provides internal position feedback. Each finger is positioned by a tight tendon linked to a hole positioned at the backside of the distal phalange as shown in *Figure* 3.3. This allows the finger to be closed completely. A microcontroller based controller was developed in the BioIn Robotics laboratory at the University of Ottawa.

3.1.3 Upgraded hand design

The rings shown in *Figure* 3.4 were designed and 3D printed as special pieces to be added to the easily modifiable Ada Hand [72]. The resulting upgraded version of the hand was called the *Ring Ada* Hand. The rings were 3D printed with rigid PLA material. Each ring is attached at the middle of the proximal phalange for each finger as shown in *Figure* 3.4 (right).

Figure 3.4 'Rings' (left), the ring attached to the hand finger (right) [10]

The larger ring opening is for the attachment to the finger, while the smaller opening is for the connection of the ring to the wire driven by the servo motor which actuates the proximal phalange.

A tendon attached to the ring is used to pull back the fingers when the linear actuators stop pulling the fingertips. This provides additional dexterity and a more natural claw-like behaviour as shown in *Figure* 3.5 (middle). This allows the hand to only close the intermediate joint and distal phalange while the proximal phalange stays in the same position. The ring allows controlling the angle between the proximal phalange and the palm, while the main linear motor is being pulled.

Figure 3.5 illustrates the experimental 9 DOF *Ring Ada* hand. Copper wires are used as tendons linked to the servo motors. The rigid characteristic of the copper wire allows joint control to be more stable control of the finger joint during the pull and push movements.

FSR sensors are attached to the distal phalanges and on the palm, as shown in

Figure 3.5. These sensors are placed corresponding to the sensor positions of the actuators on the *CyberTouch* glove, so they provide a natural haptic feedback for the human operator equipped with the *CyberTouch* glove.

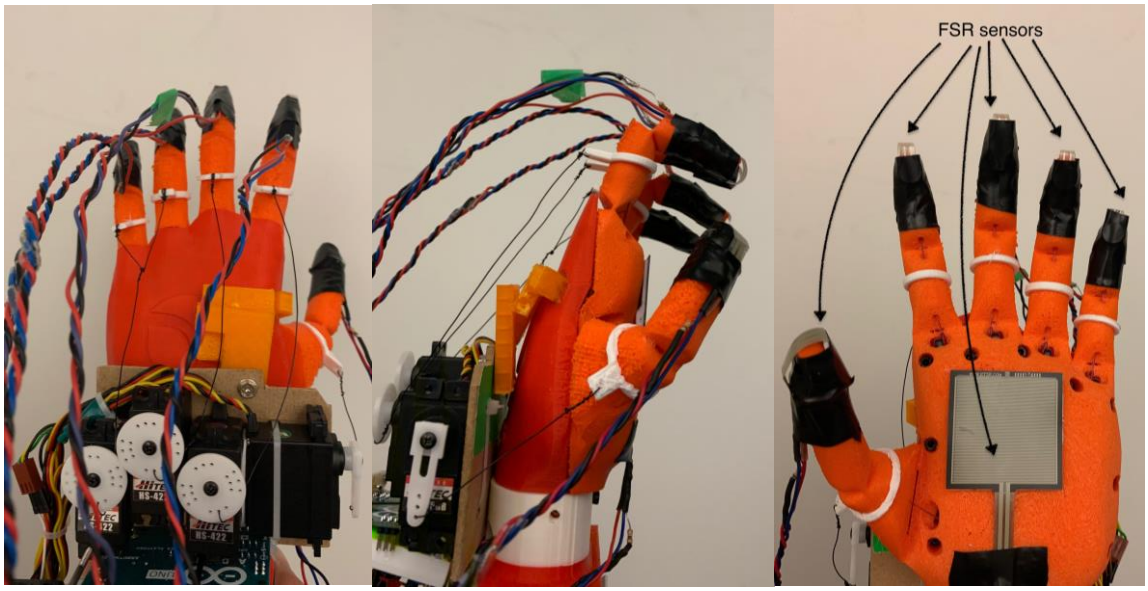


Figure 3.5 Back view of the Ring Ada hand (left), the side view of the Ring Ada with 'Claw' behavior (middle), and the palm view of the Ring Ada hand with FSR pressure sensors on each fingertip and on the palm (right) [10]

There are two tendons cooperating to control each finger position. The linear motor shrink distance of l, and the g^i as the gap between the intermediate and proximal phalanges identical with the gap g^p, and α the gap between proximal phalange and the palm. The servo motor only restricts g^p, the servo position δ can then be calculated as:

$$\delta = {}^{\pi * g^p}/_{\alpha}, \qquad g^p \leq (2\alpha - l)$$

As long as g^p is controlled by the servo motor, the gap between the intermediate and proximal phalanges g^i follows the three positions in *Figure* 3.6, with the kinematics defined as

$$g^i = 2\alpha - l - {}^{\alpha\delta}/_{\pi} \begin{cases} l \leq 9\text{mm}: condition < 1{,}2 > \\ l \geq 9\text{mm}: condition < 3 > \end{cases}$$

The distal phalange of the hand is designed to mount FSR sensors or multi-modal tactile sensors, which would add stiffness to the intermediate joint. It can be confidently assumed that the resistance of the proximal joint is lower than the intermediate joint, and therefore always has higher priority to be pulled by the linear motor, which simplifies the control.

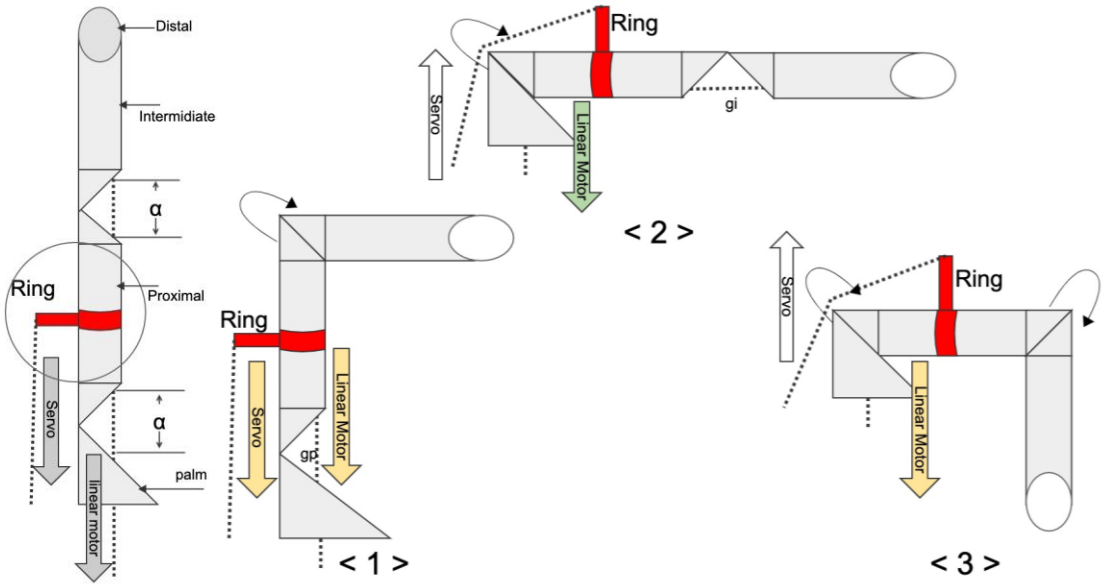


Figure 3.6 The ring, highlighted in red, which adds another DOF for the Ring Ada hand finger (1), typical finger movement (2 and 3) [10]

According to the manipulation taxonomy graph [73] and the grasp taxonomy given in [74], there is no defined unique instance when the pinky and the ring finger are moving separately. It is for this reason one servo motor is used to control the proximal part of the ring and pinky fingers, herein called the *Ringpinky* servo.

3.1.4 Role of the thumb

It is a well-known fact that the role of the thumb is important for human grasping, as clearly shown in the taxonomy of grasping [74] and manipulation classification [75], the loss of the thumb corresponds to the loss of 40% of hand performance [76].

Figure 3.7 shows the experimental setup used in [10] which covers the analysis of thumb movement in different robotic hands.

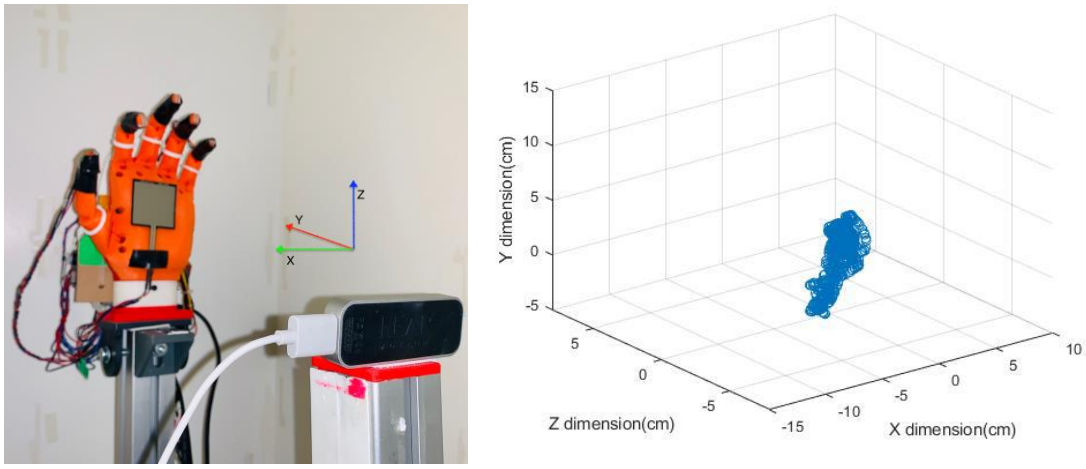


Figure 3.7 Depth camera setup for measuring thumb fingertip positions of the Ring Ada hand (left), and the 3D diagram of the reachability space for the Ring Ada thumb (right) [10]

Figure 3.7 (left) shows the experimental system setup using a Leap Motion 3D camera to find the reachability space for the thumb of the *Ring Ada* hand. *Figure* 3.7 (right side) illustrates the 3D reachability space recovered from 900 thumb fingertip position data set collected during the experiments.

3.2 Software

The *CyberTouch* glove, the *Ring Ada* robotic hand system, and ROS nodes communicate through a USB connection. The *CyberTouch* uses a USB to RS232 interface, and the two microcontrollers used in the *Ring Ada* hand follow the ROS standard [77].

3.2.1 Robot Operating System (ROS)

ROS is an open-source software designed for robotic system software development and inter-robot communication [78]. It includes the ability to access hardware through programming interfaces, low-level driver management, common function execution, inter-program messaging, and program distribution package management. The most important aspect is that ROS provides compatibility with other large-scale software modules, such as Software Development Kits (SDKs), allowing to develop new algorithms for learning, controlling, and analysis. The user-friendly communication structure it provides allows the implementation of a Peer to Peer (P2P) network between modules, which enables multiple computers to share resources. An example is the Remote Procedure Call (RPC) where one computer can execute software on another computer over a shared network.

Some ROS packages are designed to provide easy management for reusable software and keep system use minimal at the same time. The packages contain nodes that communicate through the two actions of *Subscribing* and *Publishing* [79]. Nodes use RPC communication, such as topic streaming, when managing continuous information

flow like sensor data. A node will use a *Publisher* to send data to a topic. Another node will use a *Subscriber* to receive information from that same topic. These packages may contain ROS-independent libraries, datasets, configuration files, third-party software, or anything else that constitutes a useful functional module [80].

Figure 3.8 presents the data flow diagram for the active ROS nodes, represented by ellipses, during teleoperation.

The *Ring Ada* communication is done through USB cables driven by a ROS serial package installed on an *Arduino* microcontroller which acts as a ROS node. The BioIn Robotics laboratory custom designed PCB controller also acts as a ROS node.

An example of topic *Subscribing* and *Publishing* in the experimental system is where the operation of the linear motor occurs only after the ROS node receives a command from the ROS '/position' topic and publishes the '/feedback' topic to reflect the real-time position at the same time.

The *CyberTouch* data glove uses an RS-232C 9-pin to USB connector for the serial connection to the computer. The data converter for RS-232 to USB is driven by a PL-2303 USB to serial bridge controller chip inside the connector shell, with functional block diagram shown in *Figure* 3.10 [81]. The *CyberTouch* glove serial output runs at a 115.2 baud rate.

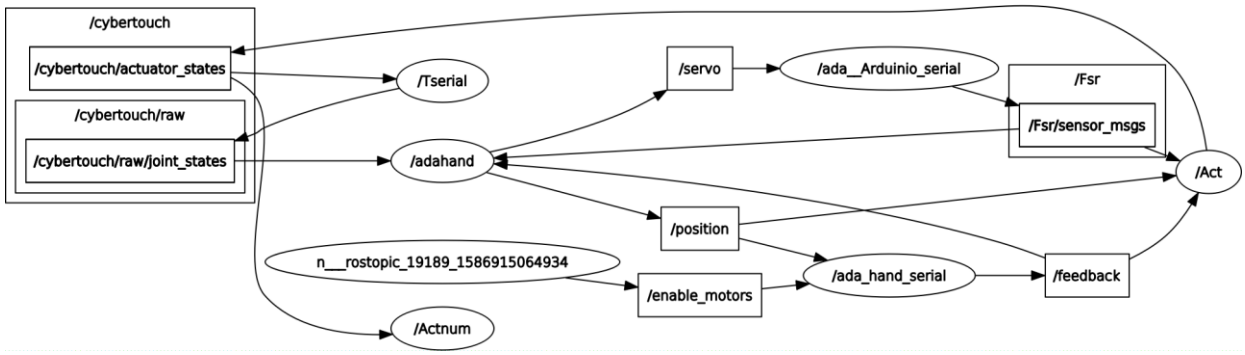


Figure 3.8 System data flow diagram for the active ROS nodes during teleoperation using the CyberTouch glove and the Ring Ada hand

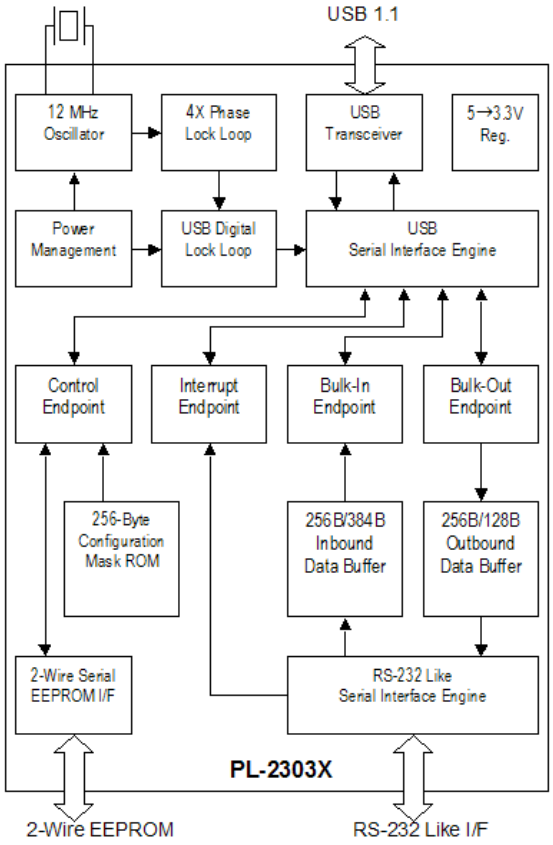


Figure 3.9 Functional block diagram [81]

3.3 Limitations

The *CyberTouch* glove flex-sensor readings have an under 1-degree resolution. However, due to the different hand sizes, shapes, and the habits of the operators, the overall repeatability of the system has only a 3-degree accuracy. In order to compensate for this issue, the glove is calibrated and configured in real-time for the flex sensor data. The *CyberTouch* data extractor node shown in *Figure* 3.8 detects the maximum and minimum angle in real-time and remaps them as 8-bit values for the joint angle range of 0 - 255. This value is then normalized between 0.0 - 1.0.

During the experiments, it was found that the actuator mounted on the center of the palm can lose touch with the human operators hand as shown in *Figure* 3.10. However, as long as the contact is constant while grasping an object, the actuator will continue to provide feedback will work without issue.

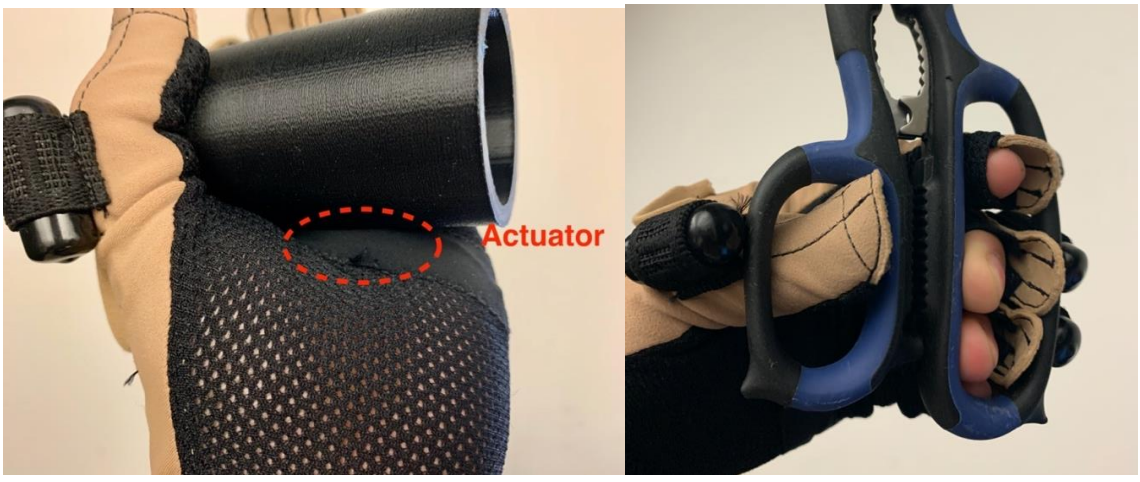


Figure 3.10 Grasp limitations of the CyberTouch glove

The thickness of the actuator shell is similar to that of a finger, which may affect the dexterity of the operator during some grasping operations where fingers may overlap or external objects are nearby.

A limitation of the *Ring Ada* hand is due to the link that exists between the ring and pinky fingers. The dexterity of the hand could be further increased by separating these two fingers which would allow for a much finer in hand object manipulation. This separation would also allow the *Ring Ada* hand to be used for American Sign Language (ASL) applications [82]**,** [83].

The *Ring Ada* hand also has inaccurate elements, most notably being the soft joints, the stretchable tendons, the tolerance of the linear actuators and servo motors, and the point where the ring is attached on the finger. During grasping the soft joints have the potential to bend, which will occasionally cause motor input commands to end up with significant differences compared to the intended final finger positions. The tactile feedback from the FSR sensors could be used to help address some of these teleoperator-*Ring Ada* hand position tolerance issues.

Chapter 4 Teleoperated Grasping Using the *Ring Ada* Robotic Hand

The aim of the *Ring Ada* upgraded robotic hand shown in *Figure* 4.1, was to achieve better teleoperated grasping and understanding of the elements which allow grasping. Synergy based grasp was a method derived through the Principal Component Analysis (PCA) of the range of motions for the *Ring Ada*. *Figure 4.2* shows the full input/output system block diagram grouped by system element using synergies as a closed loop control process.

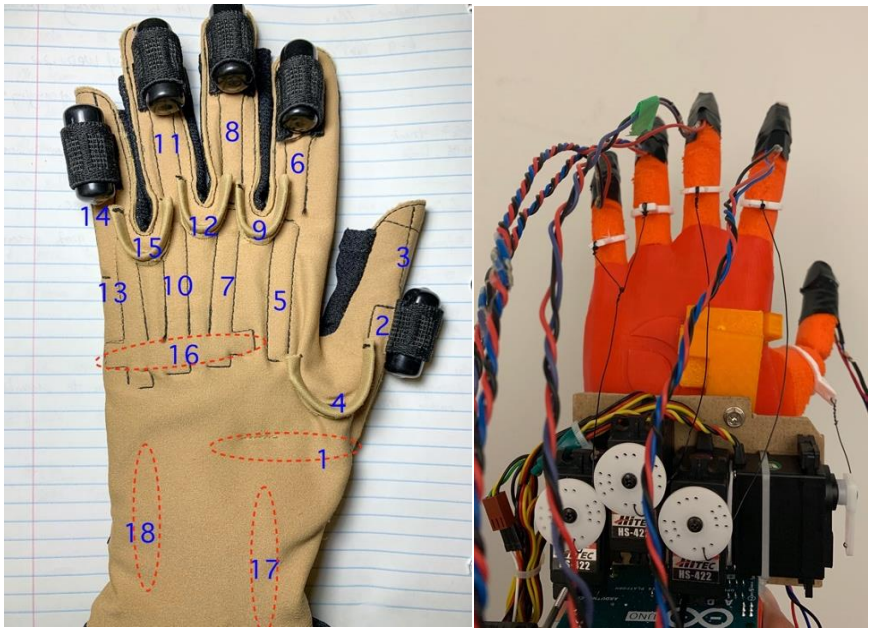


Figure 4.1 The CyberTouch glove (left), and the Ring Ada robotic hand (right)

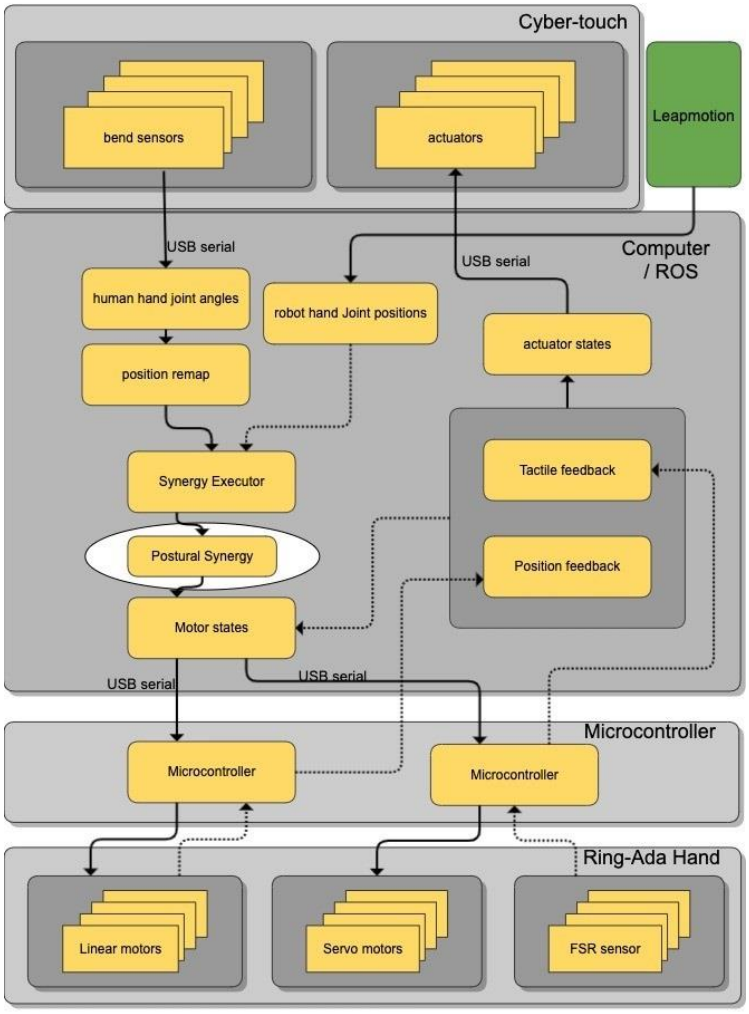


Figure 4.2 Detailed input-output modules of the Ring Ada hand teleoperation [10]

4.1 Kinematic Synergy

Kinematic synergy was used to perform remapping of the joint-to-joint between the human operator wearing the *CyberTouch* glove and the servo motors of the *Ring Ada* proximal phalanges. This is seen as the “position remap” module of *Figure* 4.2.

The operator was required to open and close the glove three times in order to

calibrate and set the reference data for the sensors. This remapping is what would allow to dynamically calculate the proportion of flex sensor deformation according to *Table* 4.1, which ROS then sends a position command to the linear and servo motors.

As discussed in Chapter 3, if a servo motor actuated a proximal phalange which holds gap g^P open, the linear motor would reach the maximum. The maximum linear positions for the *Ring Ada* hand fingers are shown in *Table* 4.1.

	Thumb	Index	Middle	Ring	Pinky
$l_{max}\ when\ g^i = 0$	0.56	0.4	0.4	0.4	0.5

Table 4.1 The maximum value for linear motors when $g^i = 0$ and $g^p = \alpha$.

4.2 Postural Synergy

As previously mentioned, the concept of postural hand synergies for tool use based on a typical set of human hand gestures was introduced by M. Santello et al in the late ‘80s [49]. The idea is to define transformations from one gesture to another, each transformation is being considered as a Principal Component (PC). Such combinations of Principal Components would allow the *Ring Ada* to achieve dexterous grasping. This function is represented by the “Postural Synergy” block in *Figure* 4.2.

The two basic PC1 and PC2 shown in *Figure* 4.3 were selected to implement the additional 4 DOF of the *Ring Ada* hand while using the ‘ring’ element of the hand.

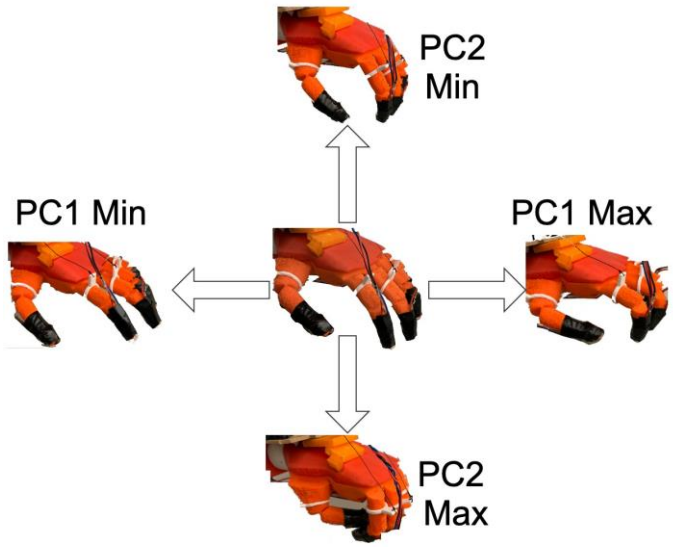


Figure 4.3 Minimum and the maximum values of PC1 and PC2 for the Ring Ada hand [10]

4.2.1 Kinematic Pattern of the Two Predominant Synergies

The MATLAB synergy toolbox *SynGrasp*, developed by the SIRSLab provides a convenient “analysis of fully or underactuated robotic hands with compliance” [84], [85]. *SynGrasp* was used to perform virtual simulations of different grasps and corresponding finger position configurations in order to better assess the synergies, and was also used to visualize the limitations of the *Ring Ada* robotic hand.

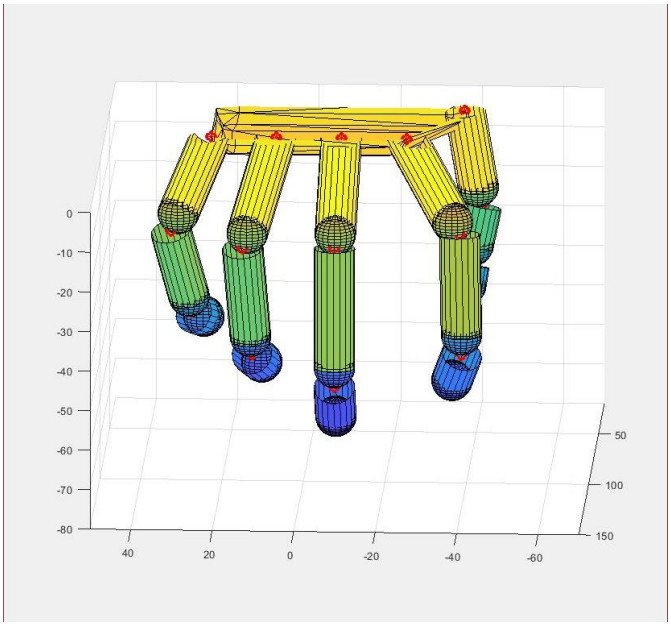


Figure 4.4 The 3D view of a robotic hand in SynGrasp [84]

The polar graphs in *Figure* 4.5 and *Figure* 4.6 represent the postural synergy vectors which are used to identify the joints whose rotations are involved in each synergy [98]. The ‘zero’ points in the polar graphs correspond to the position of the *Ring Ada* at rest which is defined in *4.2.2 Synergy Control*. The numbers 1 to 9 represent the nine actuated joints.

The red areas on the circular graphs represent the area travelled when different joint actuators are activated: actuators 1 and 2 for the thumb; 3 and 4 for the index finger; and 5 and 6 for the middle finger. The ring and pinky each have a linear actuator, 7 and 9, but are controlled by one servo, 8. The symmetry between 7 to 8 and 8 to 9 show that these actuators follow the same path.

The hand shown in *Figure* 4.5 performs the gesture while moving from an ‘open’ to ‘claw’ state due to the proximal phalange action. As shown in the polar graph, the servo actuated joints 2, 4, 6, and 8 are far away from the zero point. The ‘Claw’ is due to the distal phalange action and is the most important movement in the grasp phase. This motion dominates the angle between distal and proximal phalanges.

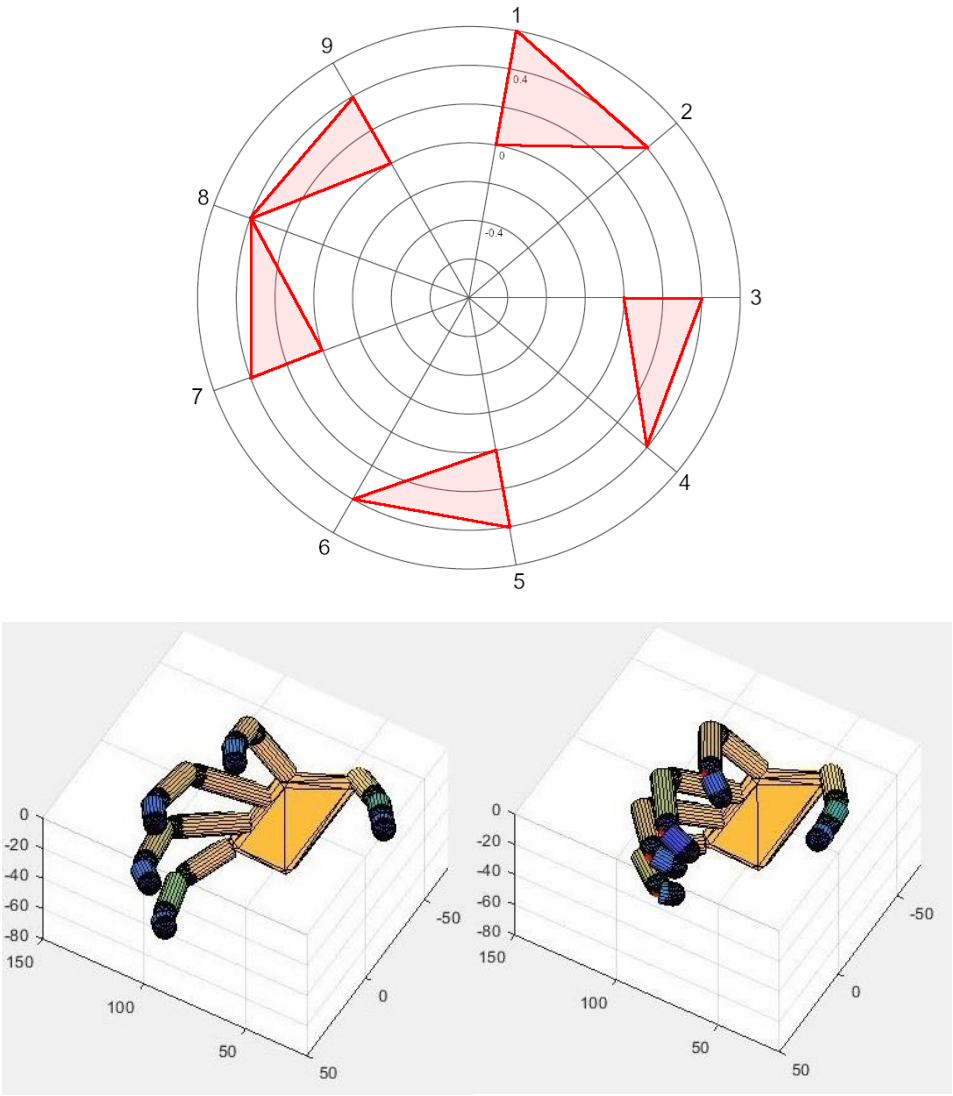


Figure 4.5 First synergy PC1, 1,2 for the thumb; 3,4 for the index; 5,6 for the middle finger, 7,8 for the ring finger; and 8,9 for the pinky (top), The minimum (bottom left) and the maximum (bottom right) PC1 present in simulation through SynGrasp

Figure 4.6 shows the second synergy of the 'fist' gesture to a gesture that relaxes all fingertips, which are pointing forward. *Figure* 4.6 (bottom left) and (bottom right) shows the fingers 'closing' towards the palm from the zero.

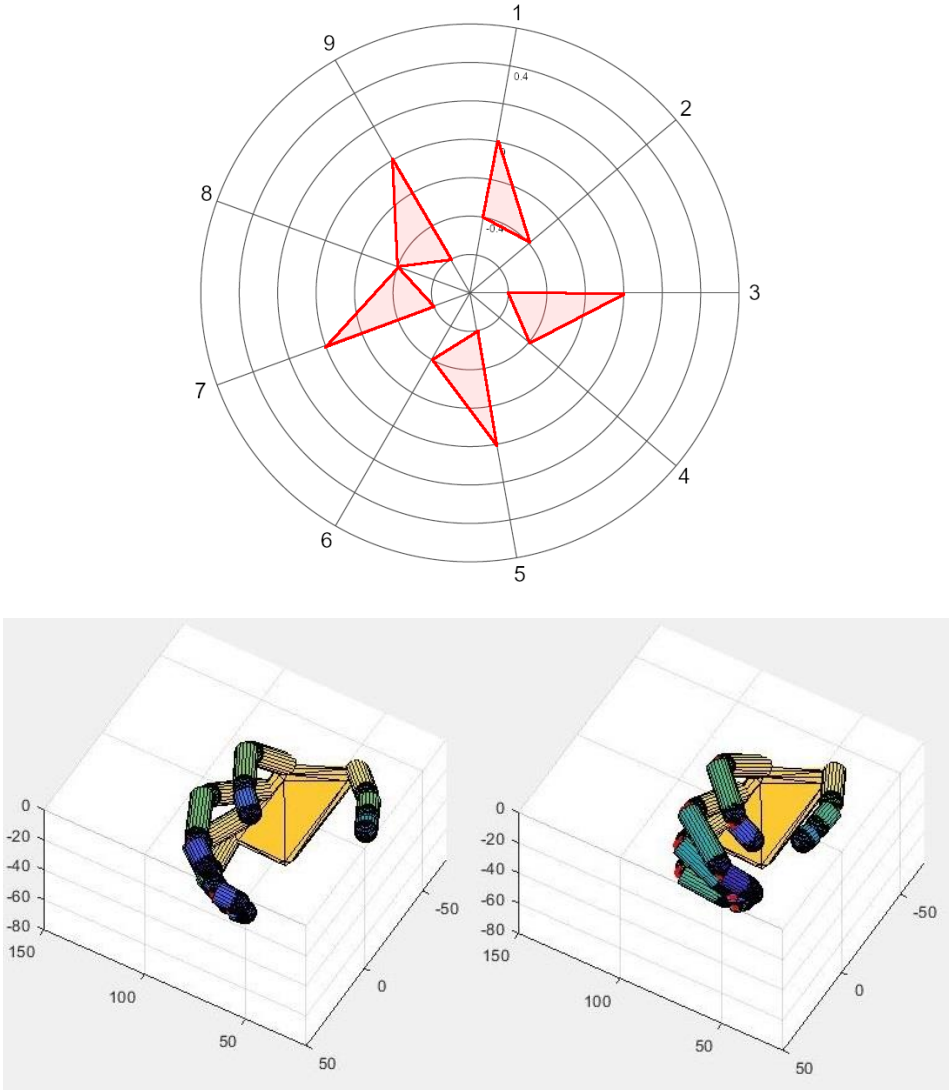


Figure 4.6 Second synergy PC2, 1,2 for the thumb; 3,4 for the index; 5,6 for the middle finger, 7,8 for the ring finger; and 8,9 for the pinky (top), minimum (bottom left) and the maximum (bottom right) PC2 present in simulation through SynGrasp

4.2.2 Synergy Control

Synergy-based control with tactile feedback is used to find the relationship between the hand joint motions and the sensory data from the haptic feedback glove.

As previously mentioned in *4.1 Kinematic Synergy,* while wearing the *CyberTouch* glove, the operator was required to perform a data reference calibration by opening and closing the hand three times then relaxing the hand for an initial reference point.

Through this calibration it was found that when an operator was at rest, the servo motors, R_s, and linear actuators, L_l, had a consistent point where they would stop moving.

	Thumb	Index	Middle	Ring	Pinky
R_s	140°	60°	60°	60°	
L_l	0.4	0.6	0.6	0.6	0.6

Table 4.2 The position of the servo motors and the linear actuators when the CyberTouch glove is worn and the hand of the operator is in a relaxed state

The grasping hand movement is controlled by ϵ, which is the average deformation between sensors 5 and 2, to control the middle phalange, and 6 and 3, to control the fingertips. The sensors are shown in *Figure 4.1*. ϵ is normalized between 0; the minimum, and 1; the maximum. The implementation of ϵ allows the actuators to work together while restricting some motion to prevent interruption or damage between them. P_s and P_l pattern values seen in the circular plots.

Through grasping experiments it was determined that the servo motor parameters needed to be scaled in order to achieve the appropriate operation range for the servos. These values were found to be 10, for the thumb, and 15, for the index finger. The servo motor parameters, V_s and V_l, are:

for the *thumb*:

$$V_s = 10 * P_s + R_s, \qquad (-0.4 < P_s < 0.4)$$

$$V_l = P_l + L_l, \qquad (-0.4 < P_l < 0.6)$$

for the *index*, *middle, ring*, and *pinky* fingers:

$$V_s = 15 * P_s + R_s, \qquad (-0.4 < P_s < 0.6)$$

$$V_l = P_l + L_l, \qquad (-0.6 < P_l < 0.4)$$

The motor control voltages equation can be defined as:

$$V_{s/l} = [\epsilon_1 P_{s1/l1}, \epsilon_2\, P_{s2/l2}] \cdot [PC_1, PC_2]^T$$

Due to the symmetry in the circular plots, it was determined that a single ϵ could be sufficient for each pattern value for full hand grasps. Individual finger movement with synergies is possible by performing averaging of corresponding sensors of the middle, ring, and pinky fingers to the thumb sensors. The ring and pinky sensors are first averaged together as they are controlled by a single servo motor. How we define ϵ is valuable in real-world situations where individuals who may not have full function of their hand, or do not have all fingers, can still perform teleoperations.

4.2 Sensory Feedback

The sensory feedback provides the ability for the operator to be aware of likelihood of a stable grasp.

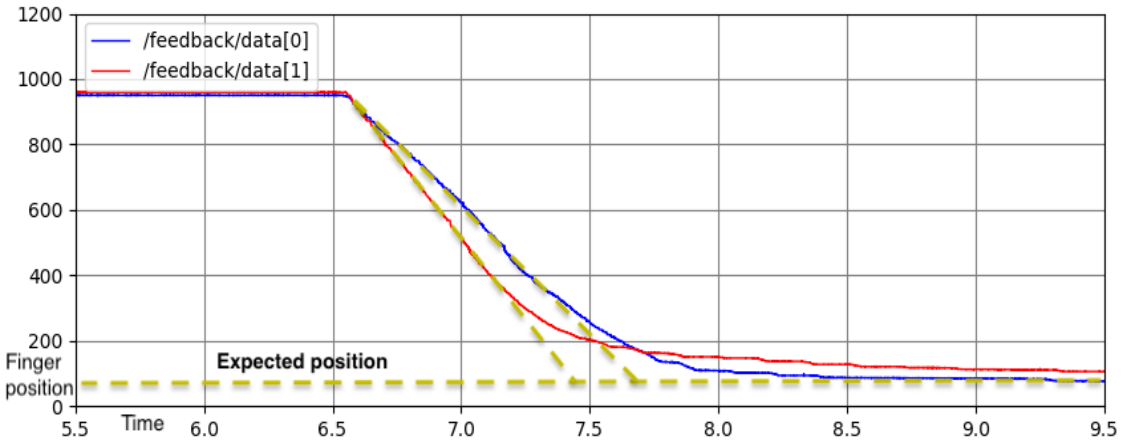


Figure 4.7 Actual vs Expected position of feedback data for the data[0] thumb, and the data[1] index finger collected during grasp operation [10]

Figure 4.7 shows the actual position f^i of the linear motor vs the expected positon l^i for the motor command. The position feedback is calculated as the difference between f^i and l^i. It could easily be seen that the two begin to diverge when the finger was facing the object. The actuator provides a stronger haptic feedback v^l when the difference $\bar{d}$ increases. The system is capable of reaching a stable grasp without any further adjustments. The v^l with $\bar{d}$ follow linear positive correlation.

When using the FSR sensors, the feedback was defined as

$$v = \max(v^l, v^{FSR})$$

4.2.1 Experimental Results

In order to maintain the consistency of the experimental grasping tests, the same set of 3D printed rectangular and cylindrical objects shown in *Figure* 4.8, was used as employed in [86]. The three objects used in these tests have the dimensions: $50mm, 70mm$, and $90mm$, for the cylindrical object diameters and also for the rectangular object side lengths.

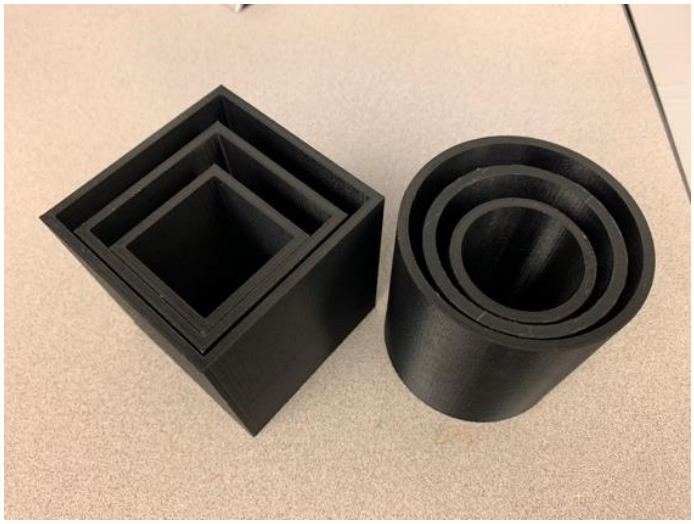


Figure 4.8 The 3D printed rectangular and cylindrical objects used for grasping experiments [10]

During the human operator calibration phase, an object was positioned within reach, near the palm, of the *Ring Ada* hand. This allowed the teleoperated *CyberTouch* glove operator to attempt a grasp and interpret the feedback and determine when a stable grasp was achieved for validation purposes.

Due to the uniform nature of a cylinder shape, no special consideration needs to be given when attempting a grasp.

During the grasp of the smallest cylinder of $\phi 50mm$ *Figure* 4.9 shows that all fingers had similar levels of activation for a stable grasp. It is clear the object is handled through a power grasp – when all fingertips touch the object. The results also show the thumb not being able to fully close, as represented by the absence of data and confirmed in *Figure* 4.9 (left).

The results obtained during the grasp of the mid sized, $\phi 70mm$ cylinder, *Figure* 4.9 (middle) shows the thumb as the first digit that reached a stable grasp, followed by the middle, ring, and pinky fingers.

Grasping the largest diameter cylinder of $\phi 90mm$ *Figure* 4.9 (right) shows only the thumb and ring fingers achieved a stable grasp; however, this was sufficient to hold the cylinder.

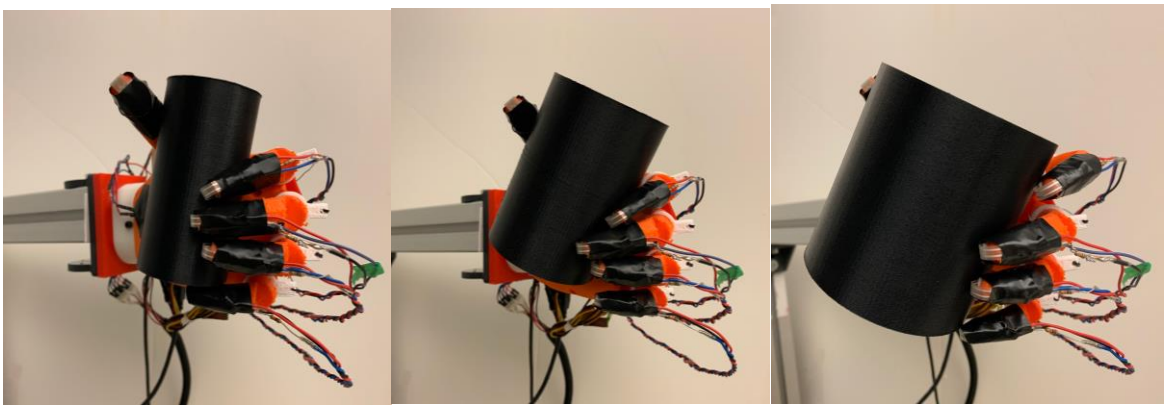


Figure 4.9 Stable grasping of cylindrical objects with different diameters ϕ50mm (left), ϕ70mm (middle), and ϕ90mm (right) [10]

Rectangular shapes need special consideration in placement during the experimental testing due to the limited approach capabilities of the *Ring Ada* robotic hand.

Figure 4.10 (left) shows the result of grasping the smallest rectangle with side length $50mm$. The results from this grasp show significant similarity with that obtained when grasping the smallest cylinder. Due to the differences in the rectangular shape of the object, the thumb is able to successfully make contact on the object as well as the other fingers when performing a stable power grasp.

Figure 4.10 (middle) and (right) show results which are more easily interpreted by the teleoperator. The feedback from the FSR and linear motor causes significant vibrations generated at the same time, causing the output to show significant jitter.

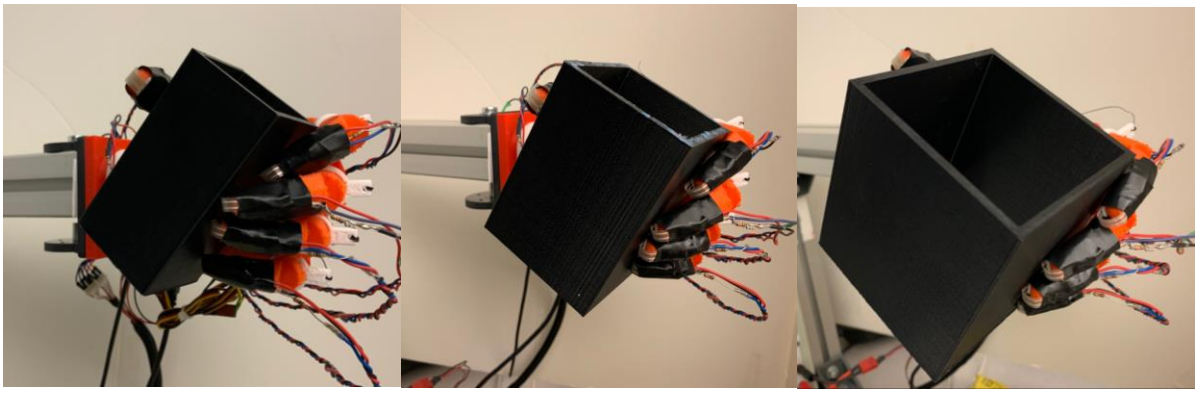


Figure 4.10 Stable grasping of rectangular objects with different dimensions 50mm (left), 70mm (middle),and 90mm (right) [10]

4.3 Conclusion

The experiments presented in this chapter validate the efficiency of grasping operations by tactile-enabled multi-finger robotic hands remotely controlled by a human teleoperator using a haptic-feedback data glove.

The non-instantaneous response is an important aspect to take into consideration for the teleoperated object grasping using haptic feedback. During the experiments, an approximately 0.5 seconds delay for every feedback update from the *Ring Ada* robotic hand to the *CyberTouch* glove of the human teleoperator was noted. It is reasonable to expect that higher speed connections and greater computing power will decrease the delays to more acceptable values that will less affect the ability to grasp.

As the fingers of the robotic hand uses a soft material, the slip of the grasped object was found to happen approximately 10% of the time. Using convenient finer materials that provide better contact points could further help in mitigating the slip issues that may occur while grasping.

Chapter 5 Neuro-Fuzzy Grasp Control

As discussed in section 2.5, a Neuro-Fuzzy System is a combined architecture of Fuzzy Logic and Artificial Neural Network elements. The Neuro-Fuzzy Controller of a *multiple-input, single-output* (MISO) dynamic system has a finite number of states that can be defined by n number of variables $X_1, X_2, \dots X_n$. The process of fuzzification uses "if-then" rules which describe the control action, the controller output, for a particular state using fuzzy linguistic variables. An inferencing process uses these rules to determine the fuzzy linguistic value of the output. The rules are evaluated to generate a specific output fuzzy set, which is then defuzzified in order to obtain the specific crisp output values.

The Neuro-Fuzzy Controller uses the ANN to tune the membership functions for cases where no mathematical models exist, while keeping the basic elements of the fuzzy logic controller intact [60].

Figure 2.11 presents the neural-like structure which provides the ability for the learning procedure, for ANFIS. It uses either backpropagation for *input membership function* (IMF) parameters, or a hybrid approach of backpropagation for IMF and least squares estimation for *output membership function* (OMF) parameters. The input, the rules, and the output resemble neurons, while the I/OMF act as the adjustable weights.

This chapter will present an ANFIS solution for the control of a teleoperated robot hand using the hand synergies, in order to allow for the teleoperated hand to be better positioned in anticipating a stable grasp without fingertip sensors to control pressure.

The work of Santello et al. [49] described the use of PCA when grasping two objects, an apple and banana, which resulted in over 80% of variance in grasping. The work presented in this chapter will use similar representations of objects to teach the ANFIS model. The three objects chosen were a tennis ball, a banana, and a rectangular prism. The work reported in this chapter has inherent limitations imposed by the experimental setup which has only one hand.

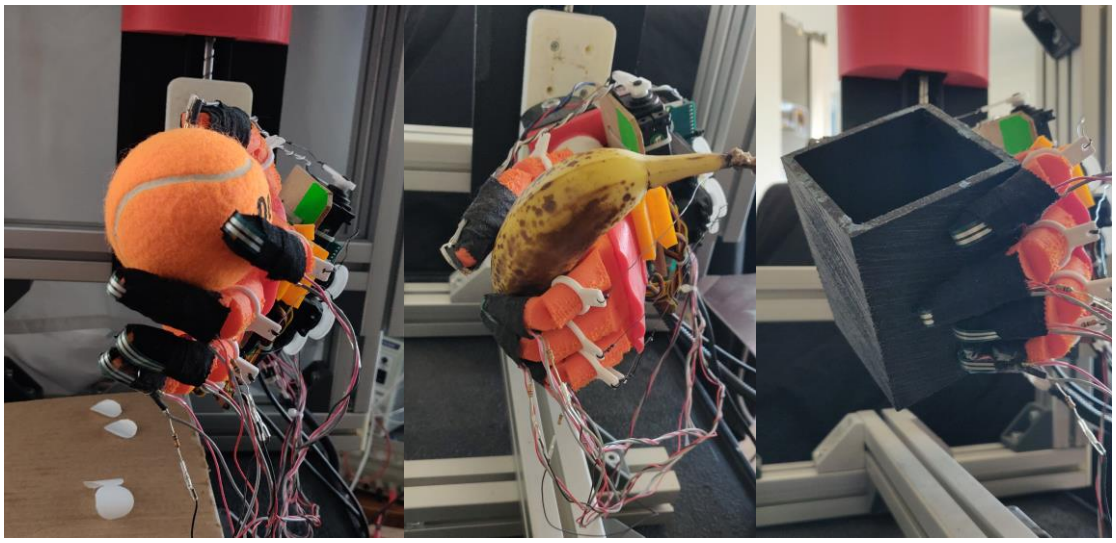


Figure 5.1 The three objects used in the grasp training experiments, ball (left), banana (middle), and rectangular prism (right)

The hand actuation system is divided into four separate systems: 1. *Thumb,* 2. *Index,* 3. *Middle,* and 4. *Ringpinky* corresponding to the four servos which the control system has to be trained to control. The numbers in *Figure* 5.2 identify the specific servos and fingers that are controlled.

Figure 5.2 The four servos

5.1 Methodology and Data Acquisition

A qualitative and quantitative approach will be employed to assess the use of ANFIS in teleoperation. The manipulation paradigm concepts in *Figure* 1.1 were used as a framework to address the Task Planning, Grasp Issues, and the Control System functions.

Data collected from the *CyberTouch Data Glove* were used to train and test the Neuro-Fuzzy system. The data collection consisted of three basic steps: (i) the operator must wear the data glove, (ii) the object must be placed within grasp of the *Ring Ada* as there is no arm or wrist movement available, and (iii) the object must be grasped only by the *Ring Ada* without falling out of the grasp.

As discussed in chapter 4, the feedback data are sampled at a different rate than the position data. After the sampling was adjusted there was no further need for post processing of the data. The data needed to reflect real grasp scenarios and any distortion that came with the grasp from the movement of the operator during grasping.

As the glove is only capable of controlling the 5 fingers of the hand, the system was set up such that during the training, the object was conveniently suspended in the palm of the anthropomorphic robotic hand while the glove wearing teleoperator grasped the object. The orientation of the grasp was chosen in such a way that in the case of a poor grasp the object would fall, requiring another attempt at the grasp.

Analysis was conducted by evaluating i) Root Mean Square Error (RMSE), ii) the correlation coefficient, R^2, and iii) by a visual analysis of the plots.

5.2 Experimental Results

The MATLAB Neuro-Fuzzy toolbox, and Graphical User Interface (GUI) shown in *Figure* 5.3, allows for several different parameter selections. After loading the training data, the plot area will show the output of this data. It will also show the training error against training epochs and an overlay of testing data against the FIS outputs.

FIS selection is split between (i) a user defined FIS, or (ii) the use of one of two methods: grid partitioning, and subtractive clustering. Training the model with the data and generated FIS provides the ability to set the error tolerance and number of epochs, where 0 and 3 are the default values.

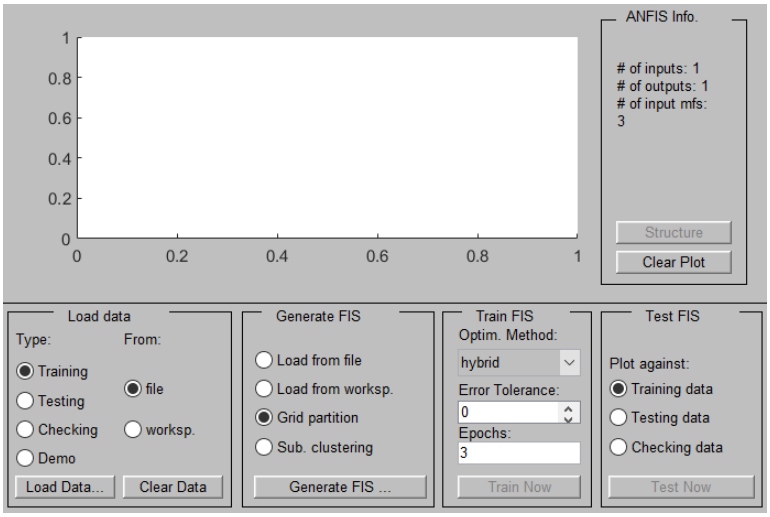


Figure 5.3 Neuro-Fuzzy Designer Graphical User Interface (GUI)

5.2.1 FIS Generation and Evaluation

The two basic FIS generation methods, Grid Partitioning and Subtractive Clustering are discussed in detail in literature [87]. The Grid Partitioning divides the *n*-dimension input space into specific regions and compares the result of inference in each specific region. As an example, if there are 5 input variables and each input variable is divided into 3 membership functions, it is divided into $3^5 = 243$ distinct areas, with one rule for each specific area. In the Subtractive Clustering, a user sets the radius of the cluster, which is the range of influence from the center of the cluster when the input space is considered as a unit hypercube with a normalized size between 0 and 1. The smaller the radius of a cluster set, the greater number of clusters, and thus an increase in the number of fuzzy rules. The contrary also holds, if the radius of the cluster is set to be large, the size of the cluster becomes so large that the number of clusters decreases and the number of fuzzy rules becomes small.

It was found from the two FIS generation methods that using the Grid Partitioning with the Gaussian membership function (MF) type, selected as *Gaussmf* in the Neuro-Fuzzy Designer, with the GUI shown in *Figure* 5.3*,* and choosing 3 MF per input variable provides the best output solution for the *Thumb*.

Modifying the Squash Factor (SF) in the Subtractive Clustering mode changes the number of membership functions and the following specific values were chosen: *Index*: SF 1.25, *Middle*: SF 0.7, and *Ringpinky*: SF 0.75. This resulted in 4, 4, and 8 MFs respectively. The Hybrid Optimization method was used which consists of back propagation for the parameters associated with the input MFs, and least squares estimation for the parameters associated with the output MFs.

The defuzzification for the FIS uses the Center of Gravity method. This is represented with a red line in the bottom right most graph of the rule plots *Figures* 5.27-5.30.

	Thumb	Index	Middle	Ringpinky
FIS Generation Method	Grid Partitioning	Subtractive Clustering	Subtractive Clustering	Subtractive Clustering
Parameters	Gaussmf	Range of Influence: 0.5 Squash Factor: 1.25 Acceptance Ratio: 0.5 Rejection Ratio: 0.15	Range of Influence: 0.5 Squash Factor: 0.75 Acceptance Ratio: 0.5 Rejection Ratio: 0.15	Range of Influence: 0.5 Squash Factor: 0.75 Acceptance Ratio: 0.5 Rejection Ratio: 0.15
Input/Resulting MF	3	4	4	8
Optimization Method	Hybrid	Hybrid	Hybrid	Hybrid

Table 5.1 FIS Generation Parameters

The resulting ANFIS models of each finger are shown in *Figures* 5.4 – 5.7. These structures correspond to the ANFIS framework shown in *Figure* 2.11.

The membership functions for the *Thumb*, *Index*, and the *Middle* fingers have 3 inputs, and the *Ringpinky* has 6, which are shown in *Figures* 5.8 – 5.11. Each input for

the membership function properties has a corresponding membership function cluster. These clusters are the output described in *Table* 5.1 as generated clusters. These clusters are shown in *Figures* 5.12 – 5.26

Each FIS rule plot depicts an example input point. The membership functions act as the “if” part of each fuzzy rule. The yellow the area under the membership function curve in *Figures* 5.27-5.30 visualizes the fuzzy membership value. The corresponding set of plots to the right side of the membership functions which show a blue line are the “then” part of each fuzzy rule. The defuzzified output value is shown by the thick red line of the aggregate plot [88].

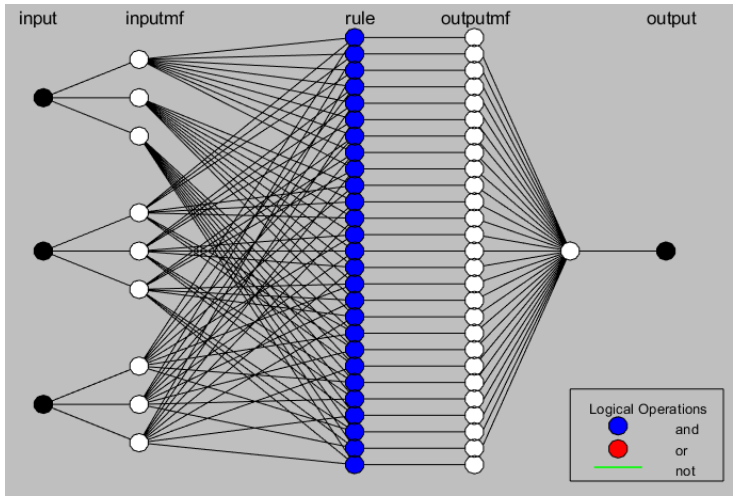


Figure 5.4 Thumb ANFIS structure

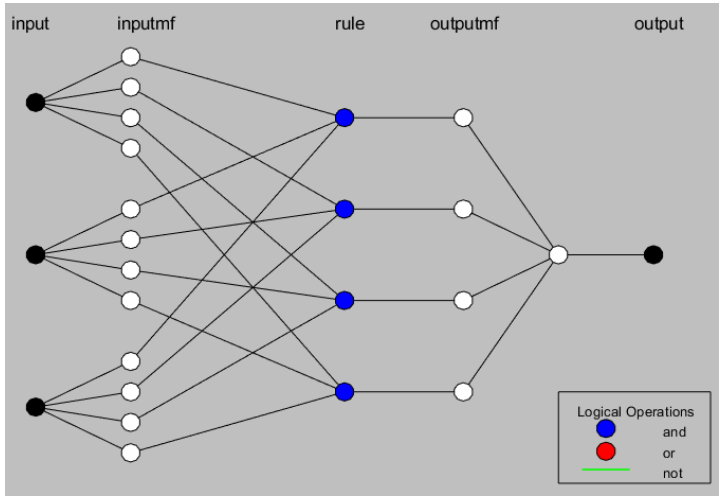


Figure 5.5 Index ANFIS structure

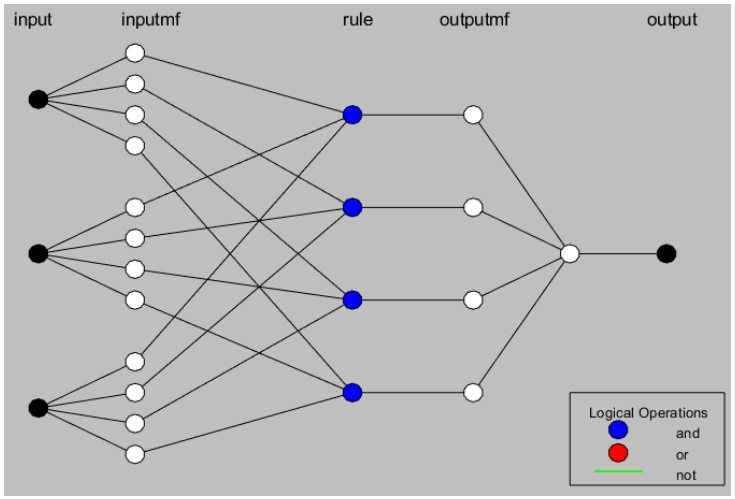


Figure 5.6 Middle ANFIS structure

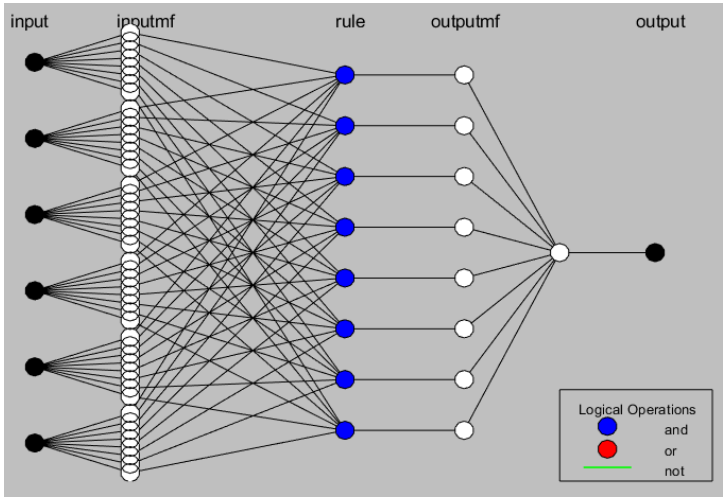


Figure 5.7 Ringpinky ANFIS structure

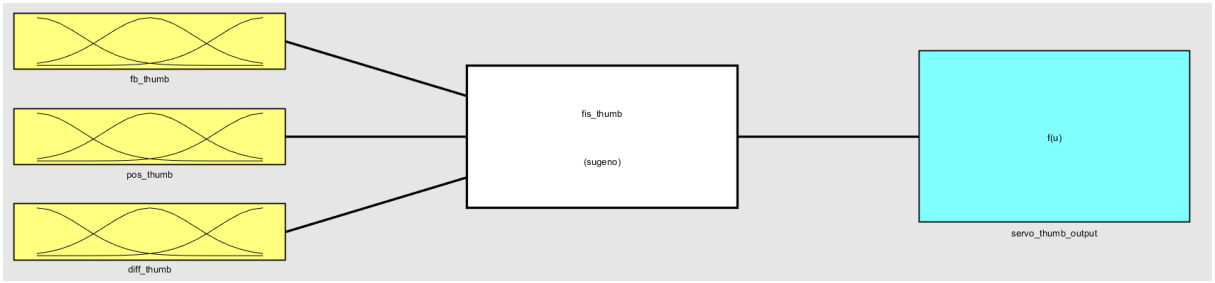


Figure 5.8 Thumb membership properties

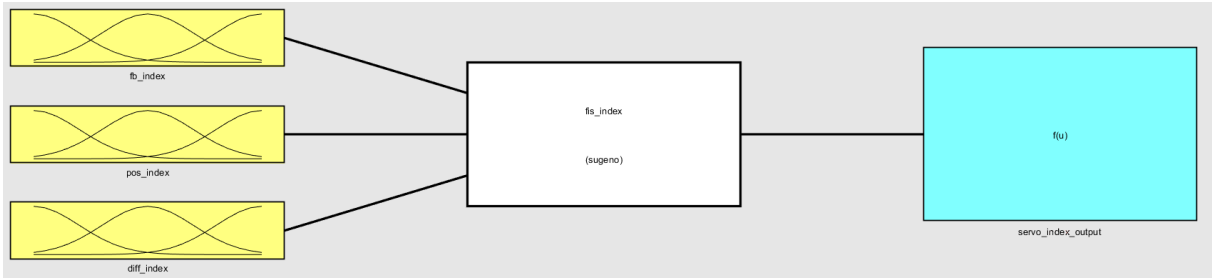


Figure 5.9 Index membership properties

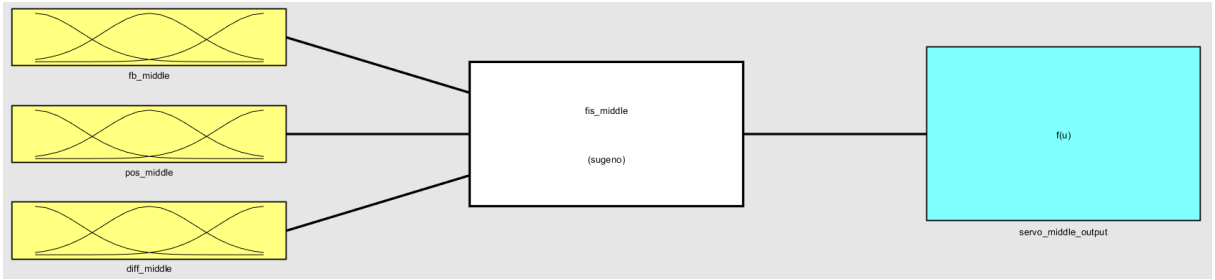


Figure 5.10 Middle membership properties

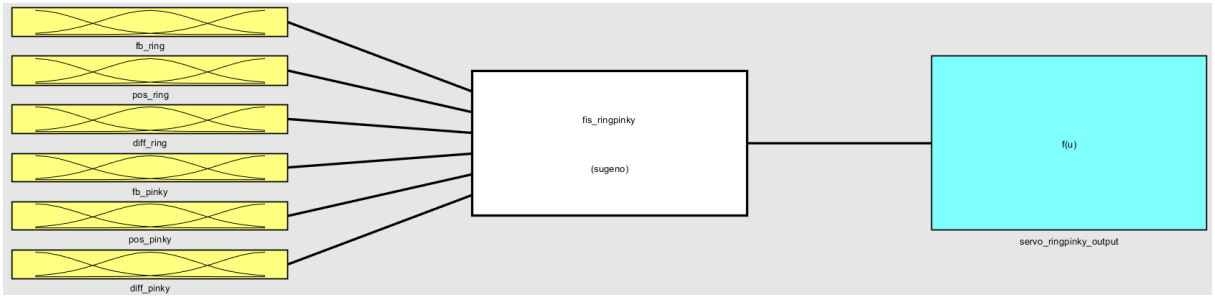


Figure 5.11 Ringpinky membership properties

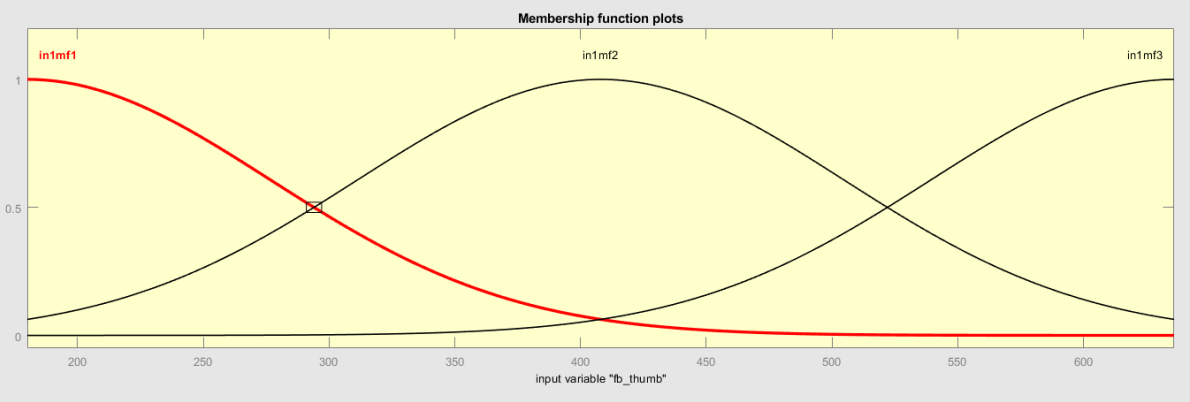


Figure 5.12 Thumb feedback membership functions

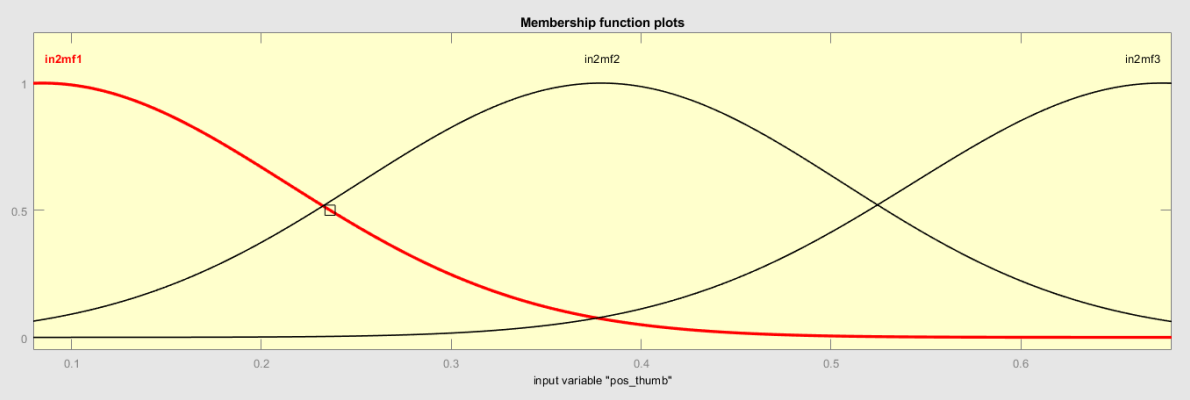


Figure 5.13 Thumb positon membership functions

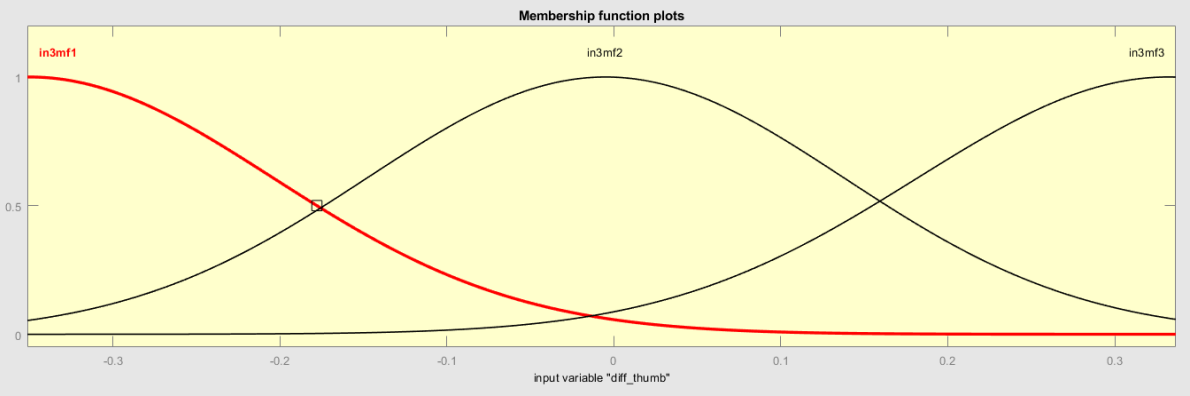


Figure 5.14 Thumb difference membership functions

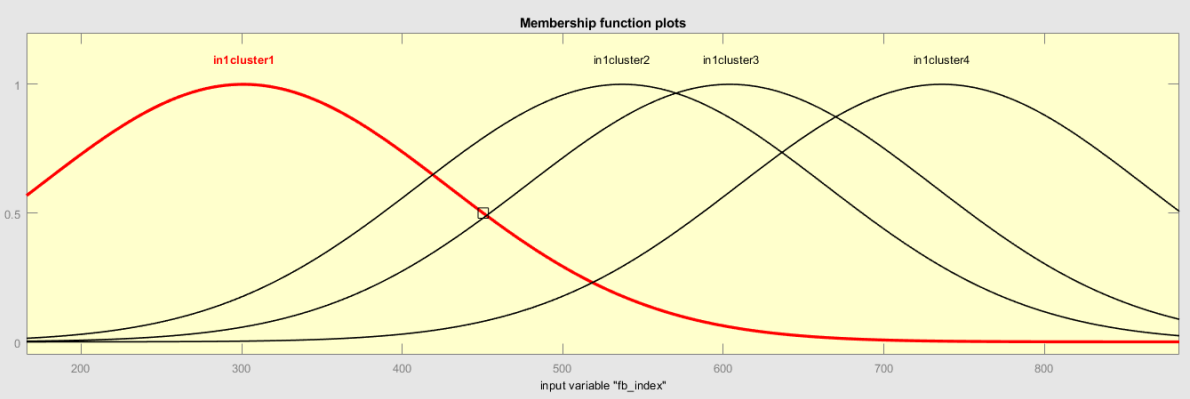


Figure 5.15 Index feedback membership functions

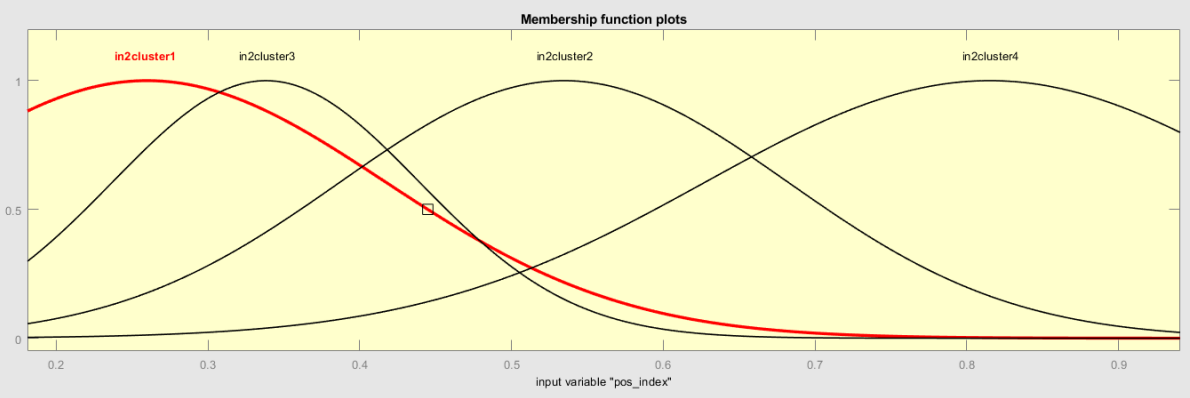


Figure 5.16 Index position membership functions

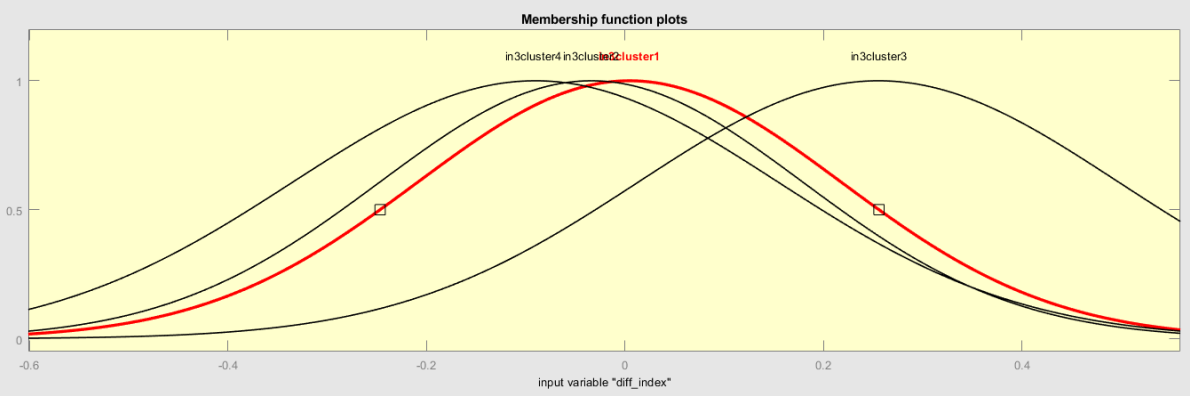


Figure 5.17 Index difference membership functions

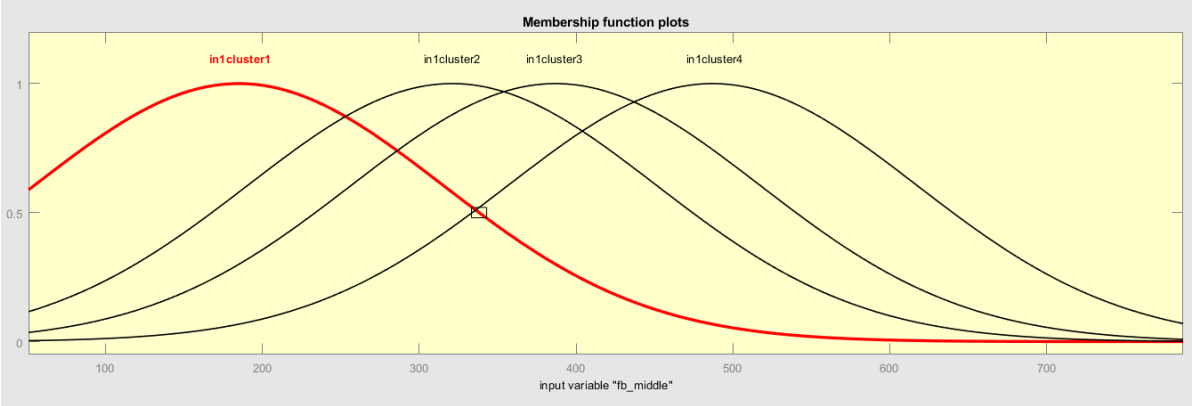


Figure 5.18 Middle feedback membership functions

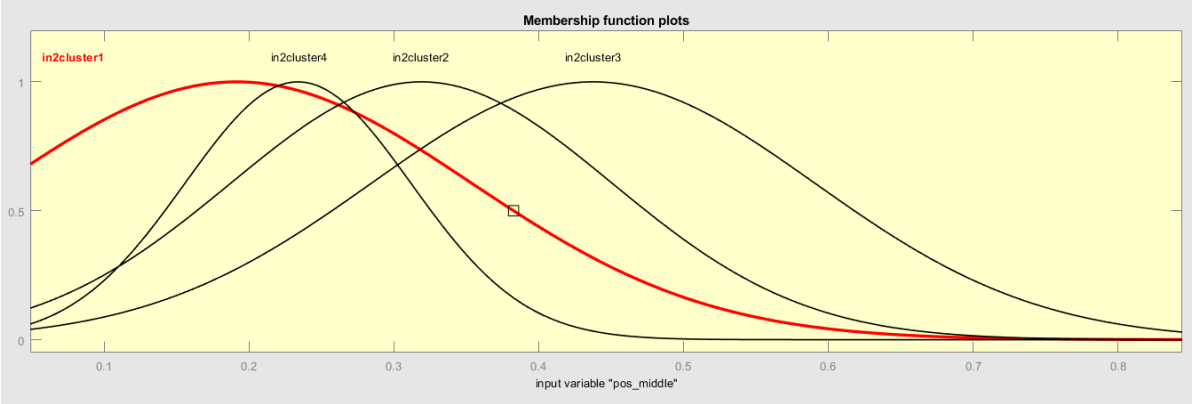


Figure 5.19 Middle position membership functions

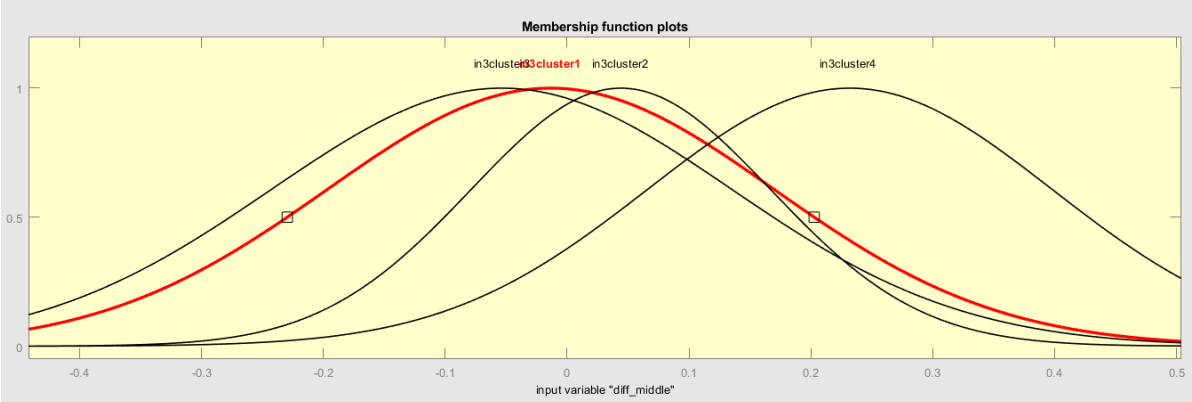


Figure 5.20 Middle difference membership functions

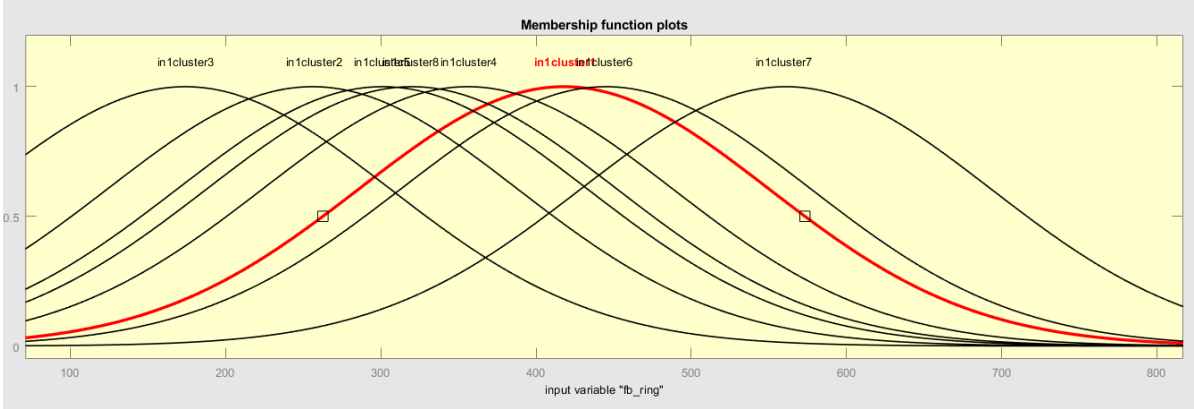


Figure 5.21 Ring feedback membership functions

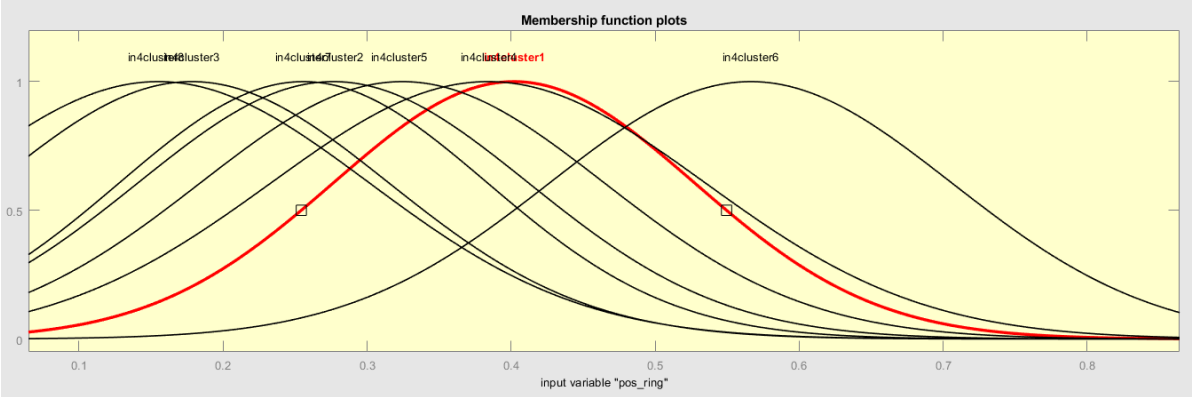


Figure 5.22 Ring position membership functions

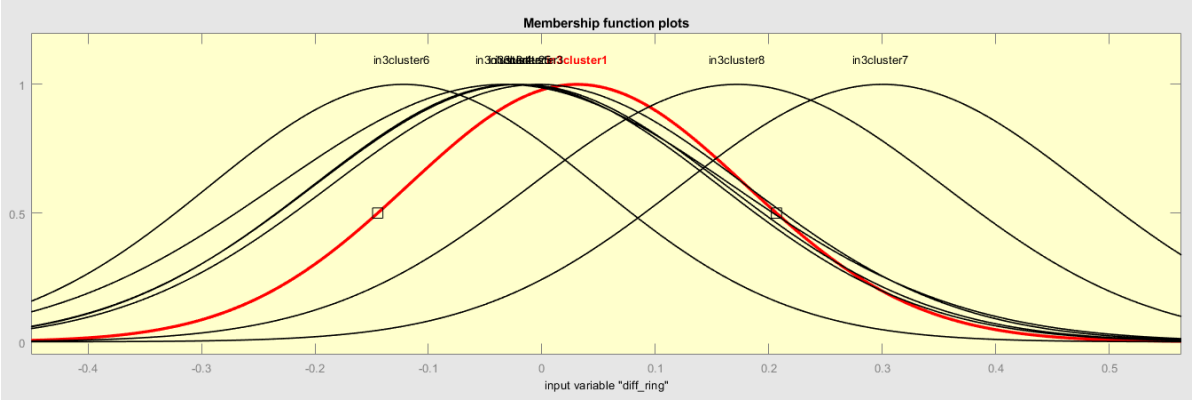


Figure 5.23 Ring difference membership functions

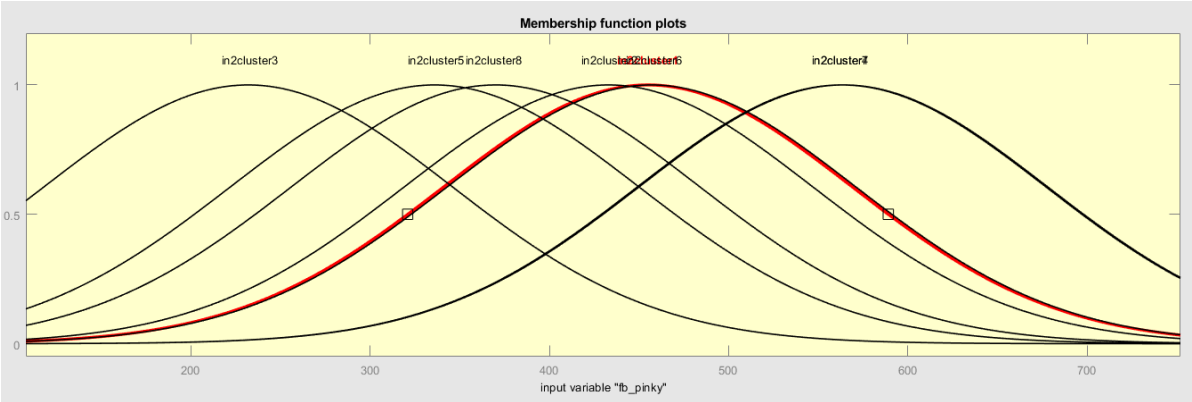


Figure 5.24 Pinky feedback membership functions

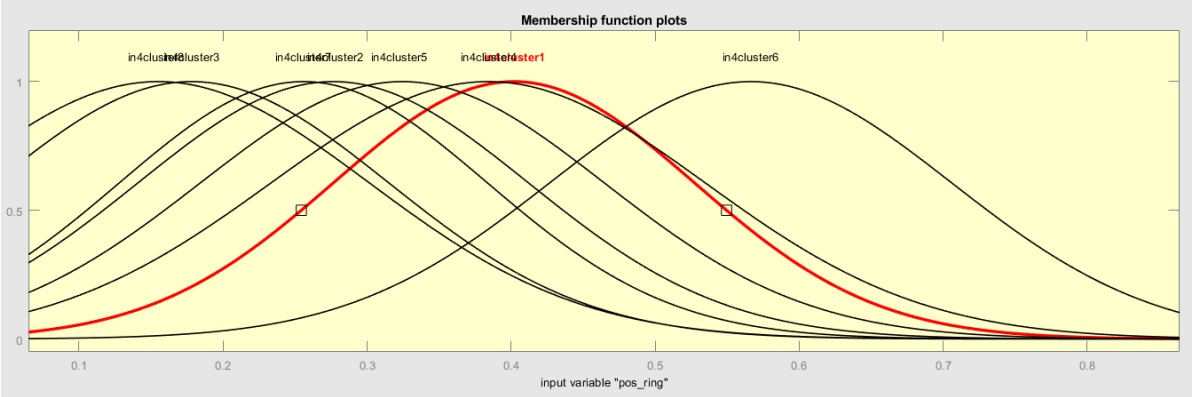


Figure 5.25 Pinky position membership functions

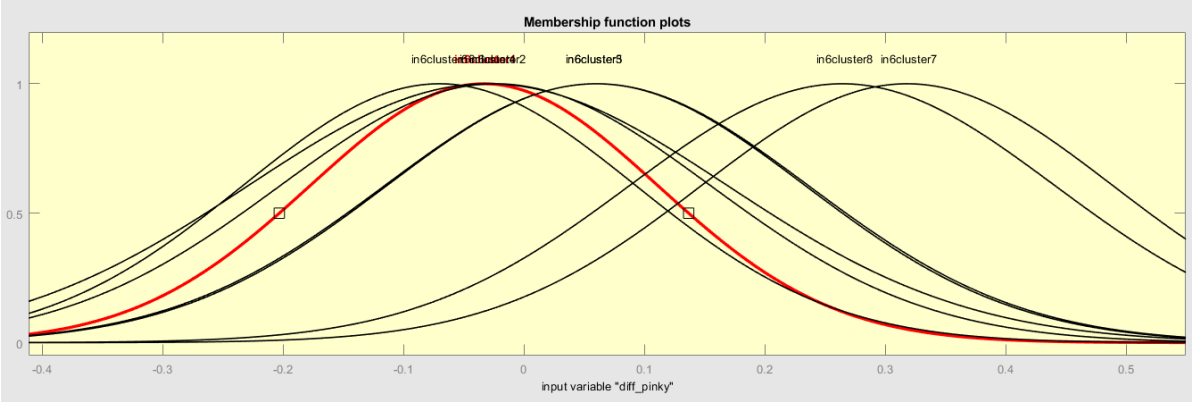


Figure 5.26 Pinky difference membership functions

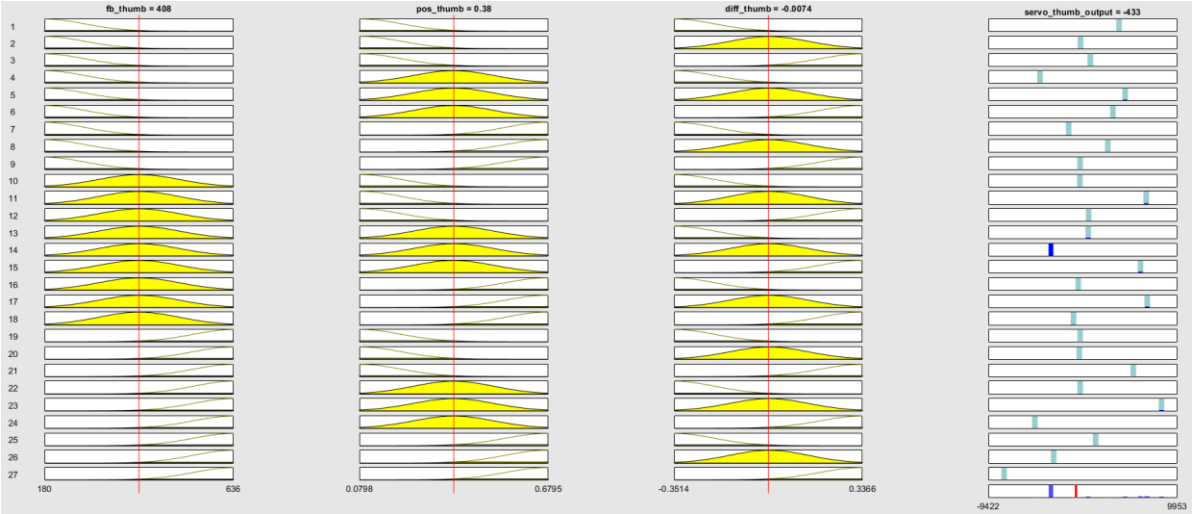


Figure 5.27 Thumb rules

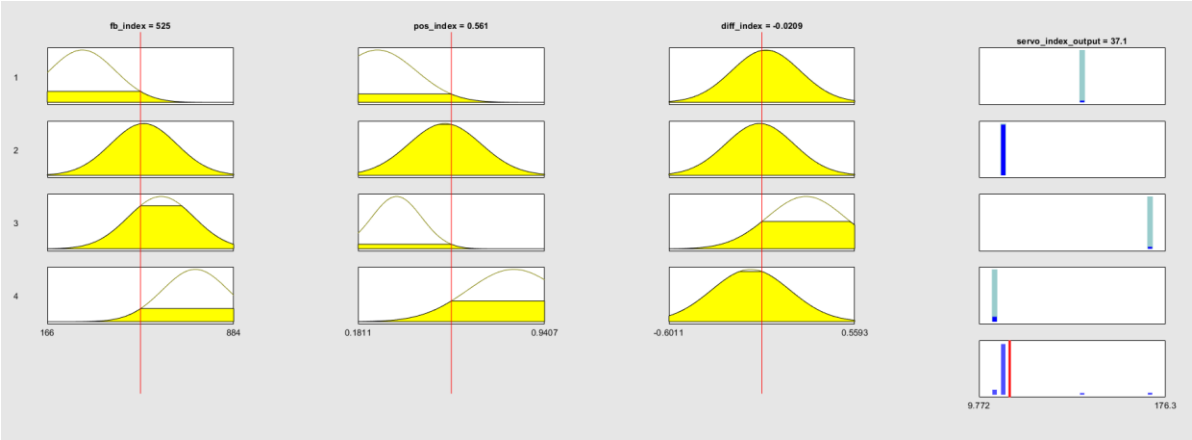


Figure 5.28 Index finger rules

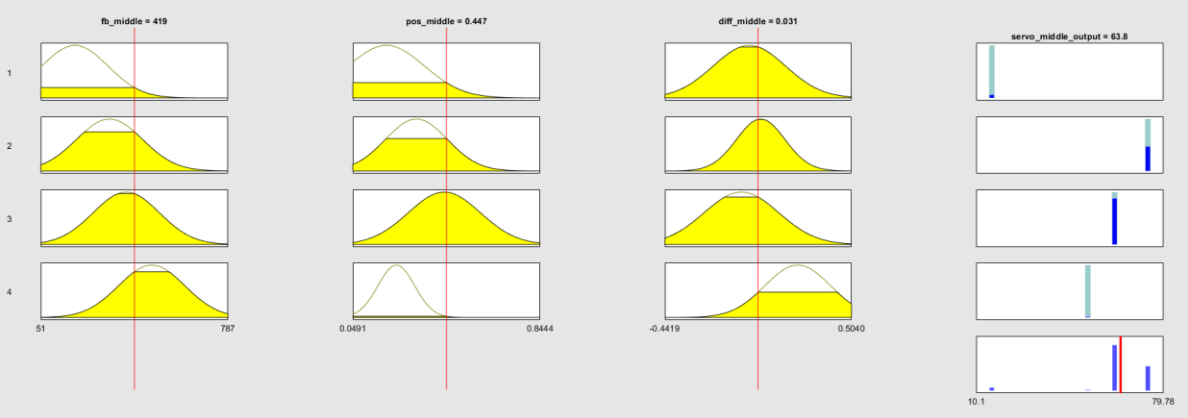


Figure 5.29 Middle finger rules

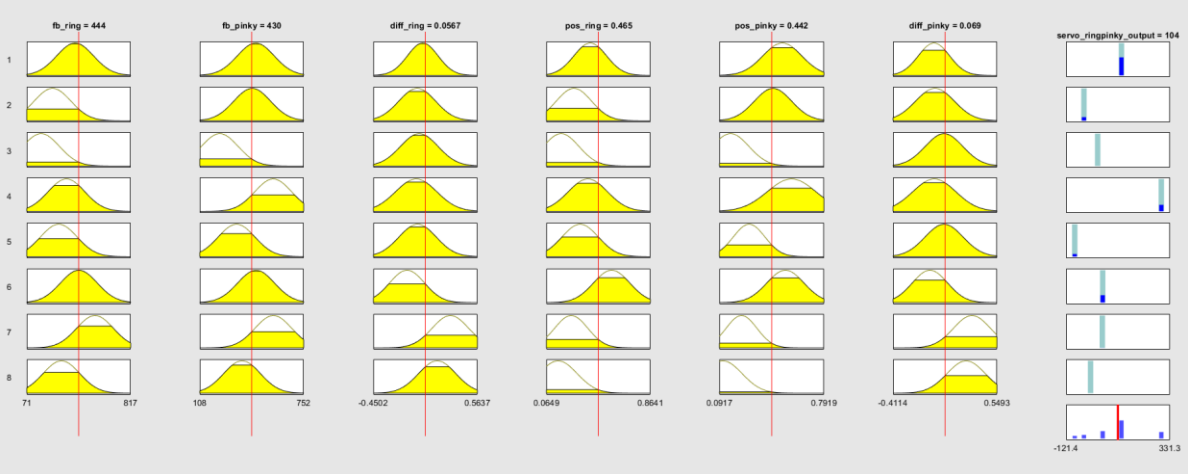


Figure 5.30 Ringpinky rules

5.2.2 Training

For training purposes, one may use *N* grasps for each of the objects in the training object set. *N* being defined as the number of times an individual may grasp an object differently. In these experiments, five grasps were performed for each object in the training set: *ball*, *banana*, and *brick*.

The feedback, position, and their difference were used as inputs for the *Thumb*, *Index*, and the *Middle* finger servos. The *Ringpinky* servo used feedback, position, and their difference of each the ring finger and pinky finger for a total of six inputs. As the feedback is sampled at a higher rate than position, the data first needed to be resampled at the same rate. After the data was resampled, all data from the five grasps for all three chosen objects were appended to a single matrix to be fed into the ANFIS GUI.

During training of the FIS, the Neuro-Fuzzy GUI plots the training error as stars and checking error as dots for each training epoch. The checking error decreases up to a certain point in the training, and then it increases. This increase occurs at the point where the training starts overfitting the training data. The Neuro-Fuzzy designer selects the FIS associated with this overfitting point as the trained ANFIS model [89]. An epoch number of 15 was selected for training.

Figures 5.31 – 5.34 show the training error against epochs for each servo.

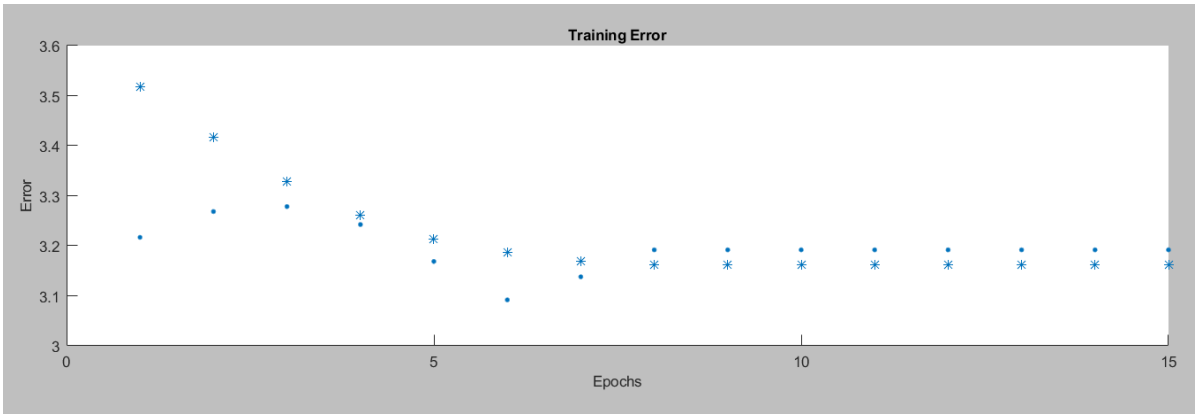


Figure 5.31 Thumb training error using 15 Epochs

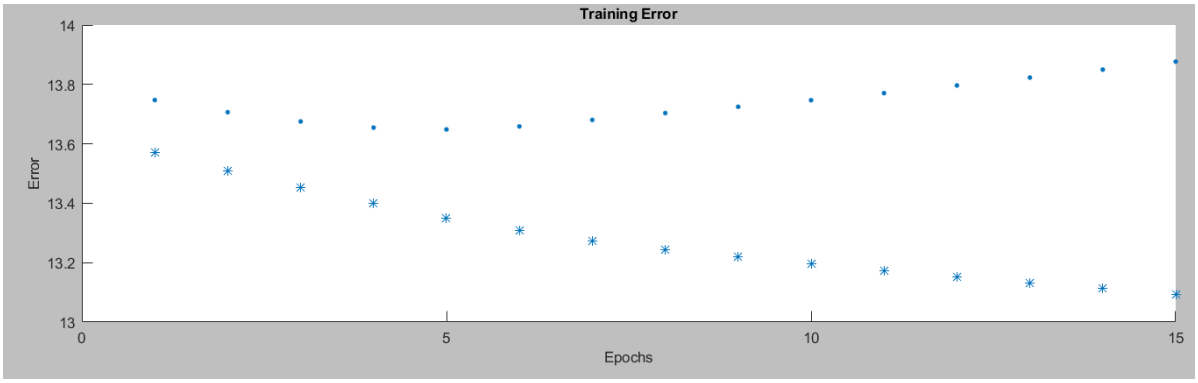


Figure 5.32 Index training error using 15 Epochs

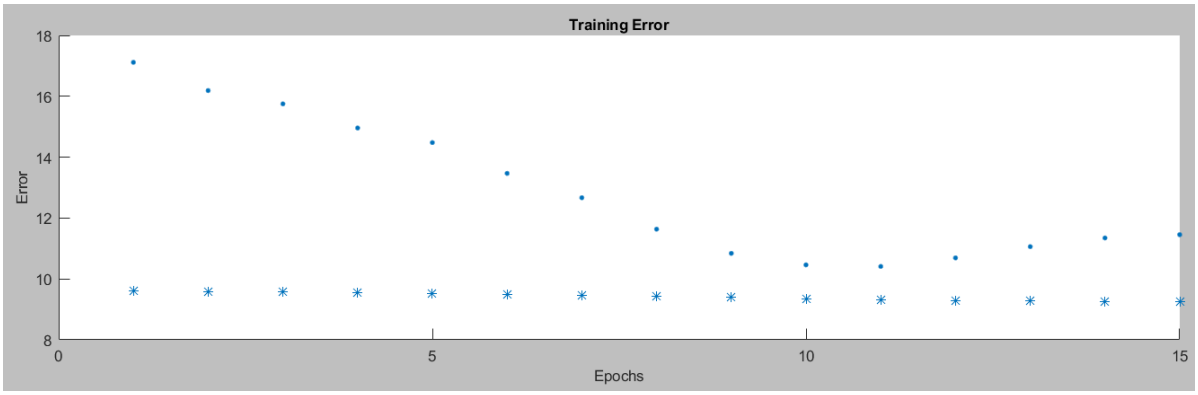


Figure 5.33 Middle training error using 15 Epochs

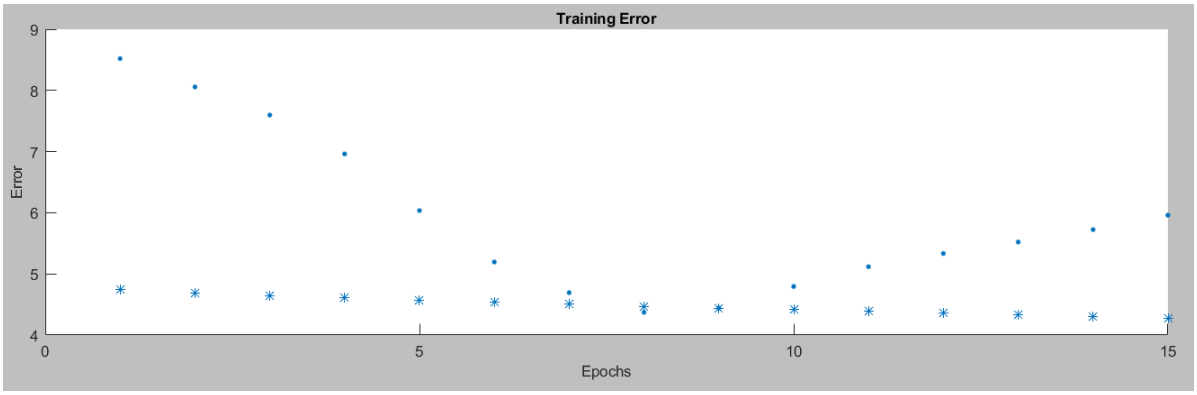


Figure 5.34 Ringpinky training error using 15 Epochs

5.2.3 Testing

Two objects shown in *Figure* 5.35 (left) and *Figure* 5.35 (right) were used to test the grasp, (i) a bottle, and (ii) a medium sized, ϕ70mm, cylinder with the approximate diameter of an everyday mug or water bottle. The medium sized cylinder could also represent a banana and brick type grasp. The inverted bottle which presents a prominent taper, has a section at the bottom more akin to a banana type grasp, where the upper half could resemble more of a ball and brick type grasp. These objects represent some variation of single hand grasping found in everyday life.

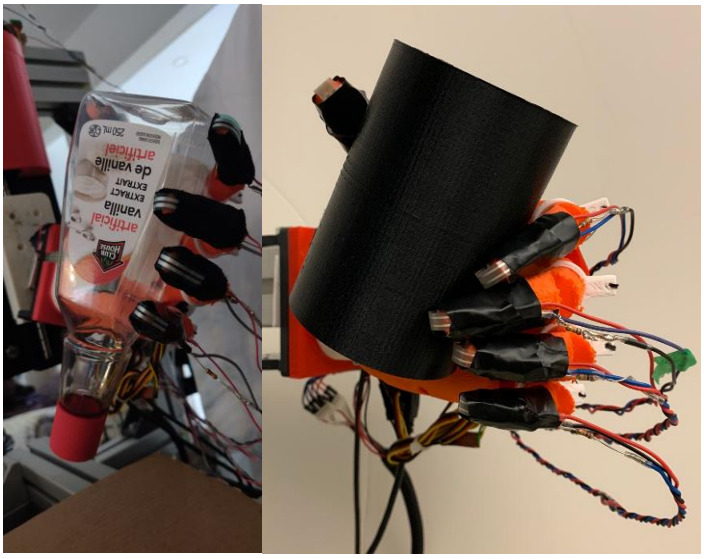


Figure 5.35 Two grasp test experiments, inverted bottle (left), and medium cylinder (right)

Two grasp tests are shown in *Figures* 5.36-5.43 and the *Tables* 5.2 – 5.4 with the associated metrics.

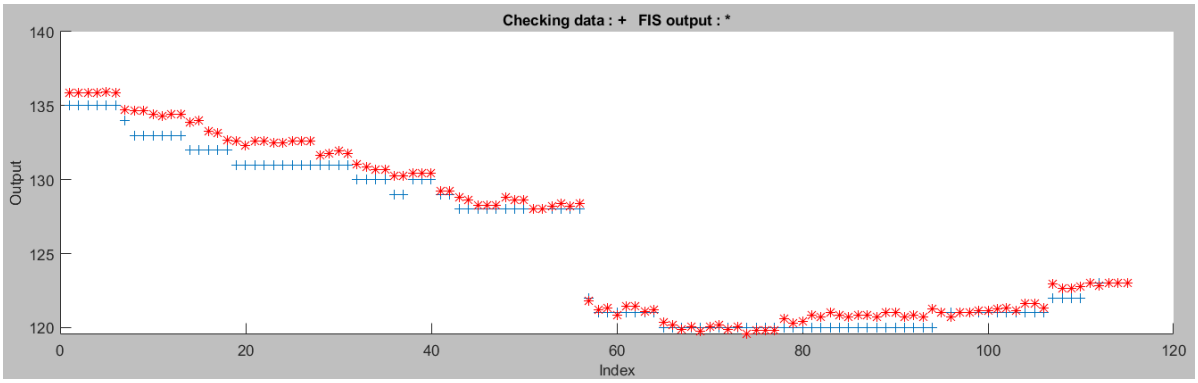


Figure 5.36 Actual vs ANFIS output for the thumb

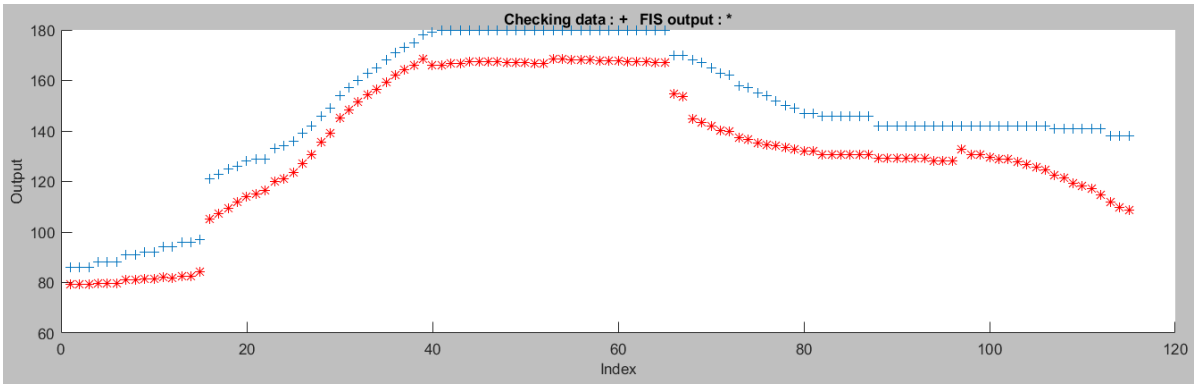


Figure 5.37 Actual vs ANFIS output for index finger

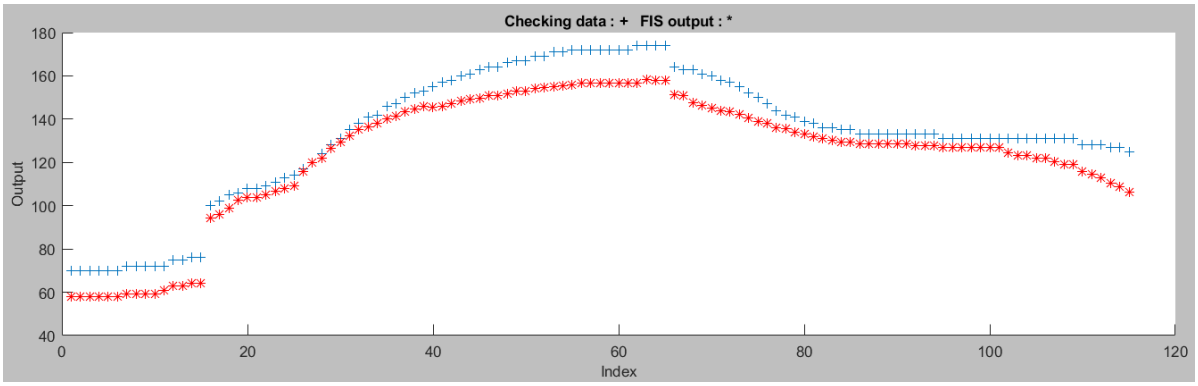


Figure 5.38 Actual vs ANFIS output for the middle finger

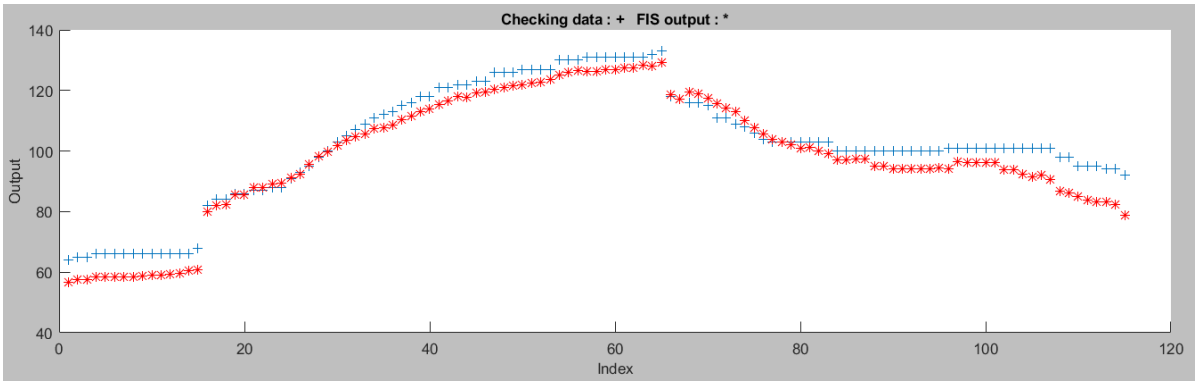


Figure 5.39 Actual vs ANFIS output for the ring and pinky fingers

	Thumb	Index	Middle	Ringpinky
R^2	0.96	0.88	0.88	0.92
RMSE	1.09	14.21	10.63	5.52
Normalized RMSE (NRMSE)	3.03%	9.94%	6.64%	5.21%

Table 5.2 Metrics evaluation for the inverted bottle

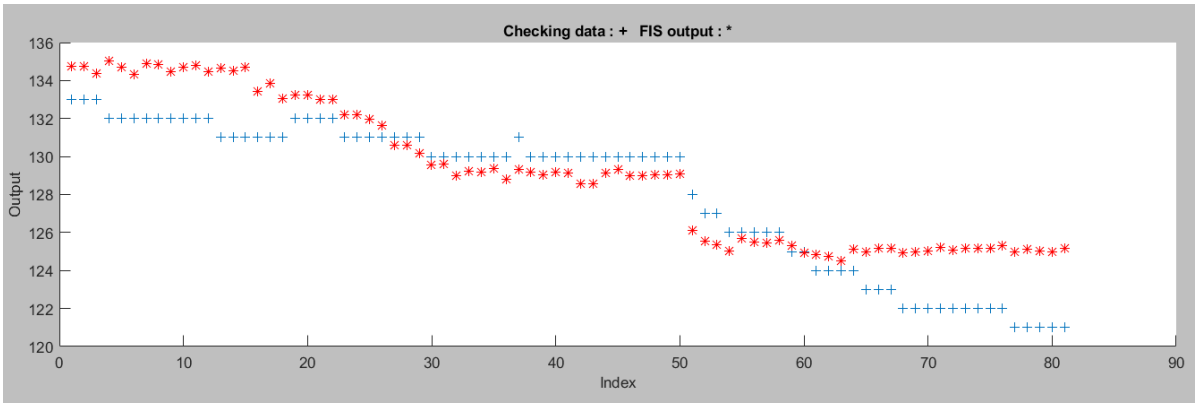


Figure 5.40 Actual vs ANFIS output for the thumb

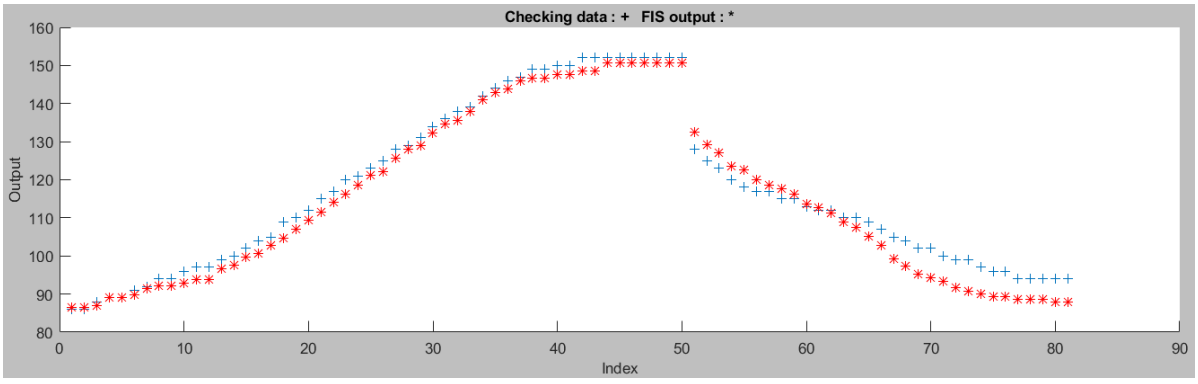


Figure 5.41 Actual vs ANFIS output for index finger

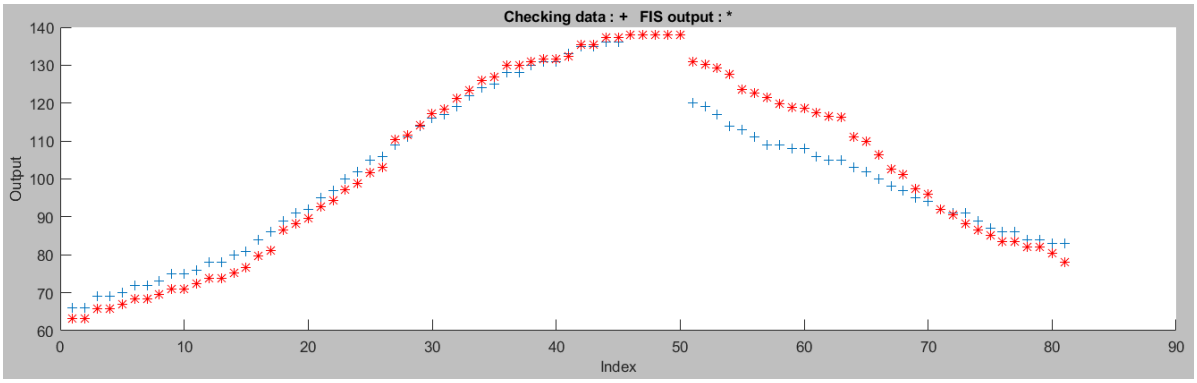


Figure 5.42 Actual vs ANFIS output for middle finger

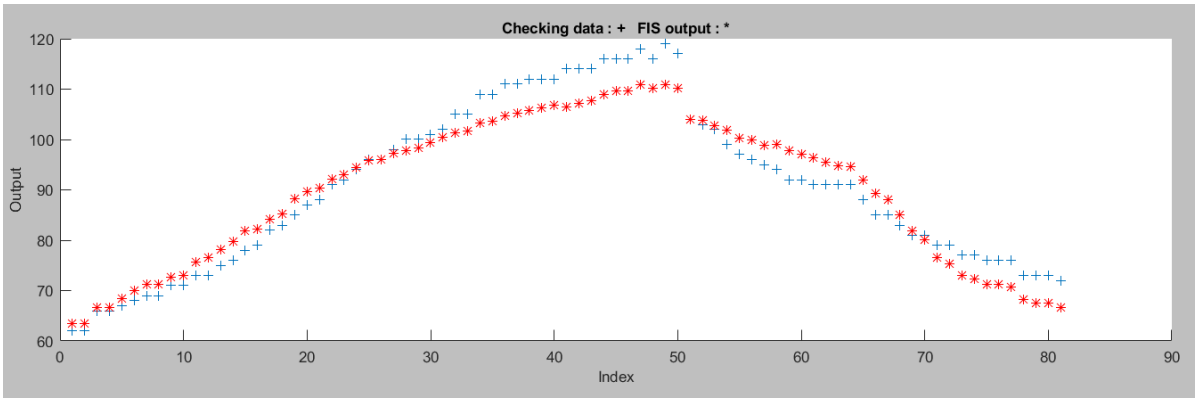


Figure 5.43 Actual vs ANFIS output for the ring and pinky fingers

	Thumb	Index	Middle	Ringpinky
R^2	0.71	0.97	0.94	0.93
RMSE	1.77	3.04	4.36	3.48
Normalized RMSE (NRMSE)	4.93%	2.13%	2.73%	3.28%

Table 5.3 Metrics evaluation for medium size cylinder

Three tests of each shape were evaluated with the metrics in the *Table* 5.4.

	Thumb	Index	Middle	Ringpinky
R^2	0.71-0.96	0.88-0.99	0.86-0.94	0.90-0.95
Normalized RMSE (NRMSE)	4.93%-3.03%	9.94%-2.02%	6.69%-2.73%	5.43%-3.19%

Table 5.4 Metrics ranges for the 6 tests, 3 of the Medium Cylinder and 3 of the inverted bottle

The metrics in *Tables* 5.2-5.4 provide the observed movements of the hand in a numerical format to provide valuable understanding for the effectiveness of the Neuro-Fuzzy model.

R^2 is a statistical measure representing the goodness of fit, or the relative measure of fit. R^2 is calculated using the sum squared regression (SSR), and total sum of squares (SST). SSR represents the sum of residuals squared, SST is the sum of distance the data is away from the mean all squared [90].

$$R^2 = 1 - \frac{SSR}{SST} = 1 - \frac{\sum(actual - observed)^2}{\sum(actual - \ overall\ average\ of\ observed)^2}$$

$$R^2 = \ 1 - \frac{\sum(y_i - \ \hat{y}_i)^2}{\sum(y_i - \ \bar{y}_i)^2}$$

Root Mean Square Error answers the question of how spread out the prediction errors are for a predicted model compared to the observed values. RMSE can also be understood as how concentrated the data is around the line of best fit.

$$RMSE = \frac{\sum(actual - observed)^2}{Number\ of\ points}$$

$$RMSE = \frac{\sum(y_i - \hat{y}_i)^2}{N}$$

NRMSE was found to be less than 10% for all tests and is further discussed in the next section.

$$NRMSE = \frac{RMSE}{(Max\ value\ training\ data - Min\ value\ training\ data)}$$

5.4 Discussion

The same inputs were used as in [10] while using the individual finger synergies described in 4.2.2. In order to select the model which best simulates accurate servo motor action, training and testing of the FIS was required for all of the possible membership functions provided by the ANFIS toolbox. There is no simple way of determining the best method, so each needed to be evaluated using an iterative process for both Grid Partitioning and Subtractive Clustering methods.

The Grid Partitioning steps are:

1. Select Membership Function type
2. Choose number of Membership Function
3. Train
4. Test
    a. Evaluate RMSE
    b. Evaluate R^2
    c. Visually assess predicted outputs against actual outputs

For each Membership Function type: *trimf, trapmf, gbellmf, gaussmf, gauss2mf, pimf, dsigmf,* and *psigmf*, the chosen number of Membership Functions was iterated from 2 – 8 inclusive as 9 or more became burdensome on the computational time to train. The *Ringpinky* did not undergo this testing process as Grid Partitioning with 6 variables was unable to be achieved due to the lack of available computer memory.

The iterative process continued with Subtractive Clustering which is generally used when there is the unknown of how a cluster data set should be defined [91].

1. Adjust squash factor
2. Train
3. Test
    a. Evaluate RMSE
    b. Evaluate R^2
    c. Visually assess predicted outputs against actual outputs

The increment of adjusted squash factor was 0.05 for all values between 0.5 and 1.25. The range of influence, accept and reject ratios were left on default settings of 0.5, 0.5, and 0.15 respectively in order to minimize some complexity. Due to the significantly reduced number of connections seen when comparing *Figure* 5.4 to 5.5 - 5.7 as defined from the definitions in 5.2.1 FIS Generation and Evaluation, more membership functions were able to be considered during this process as it was less computationally taxing.

RMSE was the first metric calculated and is an indicator of absolute measure of fit. The normalized RMSE, (NRMSE) was calculated against the training data to avoid bias. Computing the NRMSE as a percentage gives an idea of the error the model has in relation to the dependent variable. There is a relationship with R^2, the second calculated value. If the correlation coefficient is 1, the RMSE will be 0, as all the points lie on the regression line and therefore no errors [92]. The NRMSE was found to be less than 10% for all iterations, this makes sense as high R^2 values are achieved.

The servo motor has a very small circumference with increments of rotation from 0-

255. Due to the angular displacement of each step being so minute, an error of less than 10%, or 25 steps, would be insignificant. When assessing a less than 10% NRMSE, this represents less than 10% of the difference between maximum and minimum training points for that servo. As the servo never actually achieves ranges around 255 or 0, the 10% will certainly be less than 25 steps further supporting the acceptance criteria of 10% NRMSE.

However, RMSE does not present as uniform error. Single large errors cause larger RMSE. This means it could be such that many points fall along the line of best fit, with one or two being erratic and notably larger than 10%. It is for this reason other approaches need to be taken.

The second metric was the R^2 values, which is the relative measure of fit. It indicates how much variation of the dependent variable is explained by the independent variables in the model. The higher the R^2 for this model, the more precise the model for predicted outcomes which means a high correlation between observed and modelled values of servo motion. Lower R^2 scores were seen with one or multiple single large errors referenced with RMSE.

In order to complete the evaluation process, the modelled data was visualized. These plots made it very easy to distinguish when there was an outlier causing larger errors. A plot with large errors would eventually be discarded in favour of one without such large outliers. Using this metric alone would be difficult to determine a better model between similarly valued outputs, as seen between Grid Partitioning with *trapmf* and Subtractive Clustering of the Thumb and Middle servos.

Figures 5.44 – 5.46 illustrate the use of *trapmf,* the trapezoidal membership function, which is very common in fuzzy systems.

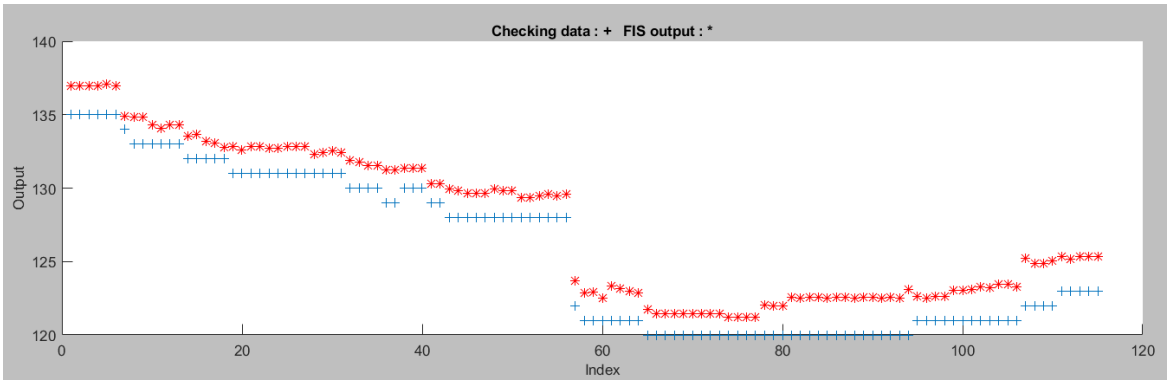


Figure 5.44 Actual vs ANFIS output for the thumb when using trapmf

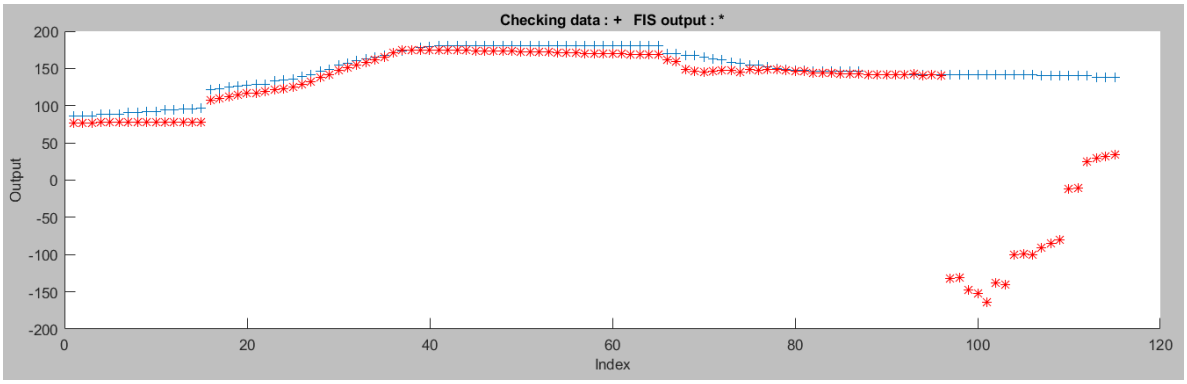


Figure 5.45 Actual vs ANFIS output for the index finger when using trapmf

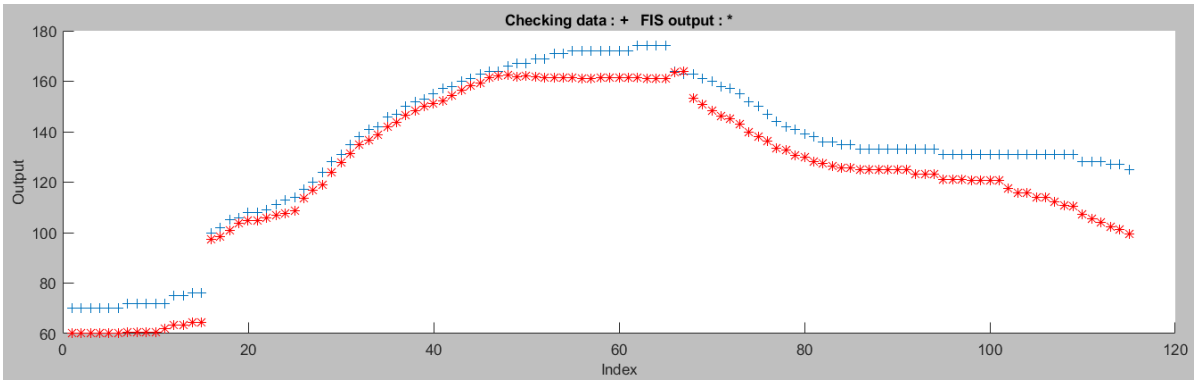


Figure 5.46 Actual vs ANFIS output for the middle finger when using trapmf

	Thumb	Index	Middle	Ringpinky
R^2	0.88	-10.47	0.87	N/A
RMSE	1.93	93.47	10.77	N/A
NRMSE	5.39%	65.36%	6.73%	N/A

Table 5.5 Metrics evaluation of the inverted bottle case using trapmf

The above *Figures* 5.44 – 5.46 and *Table* 5.5 show very similar data for the thumbs and the middle fingers for the inverted bottle though the RMSE is twice that of the models using Subtractive Clustering. *Trapmf* begins to do a very good job for the *Index* but it ultimately behaves erratically. RMSE penalizes larger errors more due to the squared factor, though it is uncertain what caused these errors from the data. Grid partitioning could not be used for *Ringpinky* due to the lack of available memory space of the computer.

The use of all three allowed for much better evaluation criteria compared to any single metric. These R^2 values with low RMSE and a positive visual comparison of the predicted value indicate the selection of objects to represent grasp variance was a good representation of generic grasping. However, it should be noted that some discrepancy in the R^2 results may stem from sampling, as there may have been some inconsistencies from the system when collecting data. This would mean that the training data and or testing data could be flawed should there have been data loss due to issues with physical connections, sampling rates, or speed of grasping.

Another observation is that the R^2 and RMSE have some dependency on how many data points are collected. *Figures* 5.40 - 5.43 presented approximately one third fewer data points due to the time it took to grasp the object. *Figure* 5.40 shows the ANFIS output has a similar shape when compared to the actual output, and maintains a low RMSE. The R^2 is not very high; however, due to the limited range of motion of the servo this is still very good. *Figures* 5.40 - 5.43 show the uniform grasping behaviour of fingers for a uniform object, the medium cylinder, and the difference in grasping behaviour can be seen when comparing to a non-uniform shape, the bottle, in *Figures* 5.36 - 5.39.

The best R^2 values and visual comparisons were observed when using the Subtractive Clustering method for every finger over the Grid Partitioning method, except in the case of the thumb.

This is interesting as it suggests all motion with the thumb, regardless of what is grasped, moves the in the same way. This is likely due to the DOF in the thumb and is noted as part of the construction of hand design. In the membership function clusters in

the other digits, similar cluster locations can be observed which is good because during grasping, a fully functional 5 finger human hand grasp will have the thumb moving similarly to other fingers.

If an individual were to grasp an object multiple times, there is no guarantee the grasp behaviour would be the same. Humans are imperfect, though always continuously adjust to the situation and adapt – especially when it comes to grasping. It is because of this lack of consistency that the robustness of the system must also be discussed. Several aspects were considered to increase robustness: no post processing of collected data, variations of objects in the tests, and the use of multiple grasp attempts for trained data. The results show evidence that these aspects provided a robust model for remapping a human hand to the robotic system behaviour.

As grasping is so complex, even very small adjustments or changes in positions would make a difference. In a system which were to have increased dexterity it can be seen how training could be provided to encompass a larger variety of shapes to provide a more comprehensively trained system.

Chapter 6 Conclusion and Future Work

6.1 Conclusion

This thesis presents theoretical and experimental contributions to the development of an upgraded haptic-enabled anthropomorphic *Ring Ada* dexterous robotic hand using an ANFIS model of human-like grasp behaviours for the teleoperated grasping operations using a *CyberTouch* Human Computer Interface (HCI) glove.

While applying grasp synergy methodology for machine learning models, developments have been made to further define grasp synergies and how they can be used for human-like grasping operations.

The proposed ANFIS model was analyzed to assess the validity for the teleoperated control of an underactuated *Ring Ada* robotic hand during grasping. Experimental results have convincingly proved the efficiency of this ANFIS system to grasp with a high level of accuracy.

6.2 Future Work

A future improvement would be to continue developing the ANFIS model by adding additional sensors for autonomous grasp operations.

Adding advanced sensors, such as the multi-modal bio-inspired cutaneous tactile

sensors developed at the University of Ottawa [44], could provide significant potential for even more human-like grasping and object handling behaviours. Using these sensors would allow for enhanced grasp stability, could be applied for slip detection, and more delicate tactile feedback. These sensors could be combined with ANFIS to further address these new applications.

Figure 6.1 shows a set of the multi-modal sensors placed in the palm of a basic Ada robot hand. It would be possible to add them to the fingertips, and along the fingers, which could potentially significantly improve sensitivity. Future work should assess the efficiency of these novel sensors for the teleoperated control of the *Ring Ada* hand.

Figure 6.1 Prototype of an Ada robot hand with palm embedded multi-modal bio-inspired tactile sensors

Developing object classification algorithms using ANFIS would be a relatively straight forward addition, which could be used on more objects beyond the ball, cylinder, or rectangular prism shapes used in this thesis. In order for the human teleoperators to

use this classification ability, closer mapping of tactile sensors on the *CyberTouch* must match the *Ring Ada* structure.

The current *Ring Ada* system has no wrist or arm motion, which makes it challenging to use for more advanced object handling operations. In order to overcome this limitation, a human-like 3D printed arm and wrist should be developed. As the *CyberTouch* has accelerometers, this could be used to raise and lower the arm, and rotate the wrist. This improvement could be further enhanced with the addition of the multi-modal sensors which would allow to better assess the difference between human movements and the robotic hand motions.

References

[1] E. Guizzo and E. Ackerman, “When robots decide to kill,” *IEEE Spectrum*, vol. 53, no. 6, Jun. 2016, doi: 10.1109/MSPEC.2016.7473151.

[2] “An Automatic Block-Setting Crane,” *Meccano Magazine*, pp. 172-undefined, Mar. 1938.

[3] H. Hanafusa and H. Asada, “A Robot Hand with Elastic Fingers and Its Application to Assembly Process,” *IFAC Proceedings Volumes*, 1977, doi: 10.1016/s1474-6670(17)66592-3.

[4] M. T. Mason and J. Kenneth. Salisbury, *Robot hands and the mechanics of manipulation*. MIT Press, 1985.

[5] E. Al-Gallaf, A. Allen, and K. Warwick, “A survey of multi-fingered robot hands: Issues and grasping achievements,” *Mechatronics*, 1993, doi: 10.1016/0957-4158(93)90018-W.

[6] A. M. Okamura and M. R. Cutkosky, “Feature detection for haptic exploration with robotic fingers,” *International Journal of Robotics Research*, vol. 20, no. 12, pp. 925–938, 2001, doi: 10.1177/02783640122068191.

[7] E. M. Petriu, “Haptic Sensors & Interfaces for Robotic Telemanipulation,” 2006. http://www.site.uottawa.ca/~petriu/HapticDexterTelemanip-2018a.pdf.

[8] E. N. Gama Melo, O. F. Avilés Sánchez, and D. Amaya Hurtado, “Anthropomorphic robotic hands: a review,” *ingeniería y desarrollo*, vol. 32, no. 2, pp. 292–293, Jun. 2014, doi: 10.14482/inde.32.2.4715.

[9] “CyberGlove II — CyberGlove Systems LLC.” http://www.cyberglovesystems.com/cyberglove-ii.

[10] Q. Zhu, V. P. da Fonseca, B. Monteiro Rocha Lima, M. Welyhorsky, M. Goubran, T. E. Alves De Oliveira, and E. M. Petriu, “Teleoperated Grasping Using a Robotic Hand and a Haptic-Feedback Data Glove,” in *2020 IEEE International Systems Conference (SysCon)*, 2020, pp. 1–7, doi: 10.1109/SysCon47679.2020.9275927.

[11] T. L. Stedman, *Stedman’s Medical Dictionary for the Health Professions and Nursing*, 5th ed. Lippincott Williams & Wilkins, 2005.

[12] V. P. da Fonseca, B. Monteiro Rocha Lima, T. E. Alves De Oliveira, Q. Zhu, V. Z. Groza, and E. M. Petriu, “In-Hand Telemanipulation Using a Robotic Hand and Biology-Inspired Haptic Sensing,” 2019, doi: 10.1109/MeMeA.2019.8802139.

[13] G. Cotugno, K. Althoefer, and T. Nanayakkara, “The Role of the Thumb: Study of Finger Motion in Grasping and Reachability Space in Human and Robotic Hands,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 7, Jul. 2017, doi: 10.1109/TSMC.2016.2531679.

[14] N. A. Davidoff and A. Freivalds, “A graphic model of the human hand using CATIA,” *International Journal of Industrial Ergonomics*, vol. 12, no. 4, pp. 255–264, 1993, doi: 10.1016/0169-8141(93)90095-U.

[15] M. R. Cutkosky, “On grasp choice, grasp models, and the design of hands for manufacturing tasks,” *IEEE Transactions on Robotics and Automation*, vol. 5, no. 3, pp. 269–279, 1989, doi: 10.1109/70.34763.

[16] E. Mattar, “A survey of bio-inspired robotics hands implementation: New directions in dexterous manipulation,” *International Robotics and Autonomous Systems*, vol. 61, pp. 517–544, Jan. 2013.

[17] E. Nazma and S. Mohd, “Tendon Driven Robotic Hands: A Review,” *International Journal of Mechanical Engineering and Robotics Research*, vol. 1, no. 3, pp. 350–357, Oct. 2012.

[18] S. R. Company, “Shadow dexterous hand C5 technical specification,” 2008. http://www.shadowrobot.com/downloads/shadow_dextrous_hand_technical_specification_C6P6.pdf.

[19] “Shadow Hand - ROBOTS: Your Guide to the World of Robotics.” https://robots.ieee.org/robots/shadow/.

[20] D. H. Lee, J. H. Park, S. W. Park, M. H. Baeg, and J. H. Bae, “KITECH-Hand: A Highly Dexterous and Modularized Robotic Hand,” *IEEE/ASME Transactions on Mechatronics*, 2017, doi: 10.1109/TMECH.2016.2634602.

[21] J.-H. Bae, S.-W. Park, J.-H. Park, M.-H. Baeg, D. Kim, and S.-R. Oh, “Development of a low cost anthropomorphic robot hand with high capability,” Oct. 2012, doi: 10.1109/IROS.2012.6386063.

[22] A. Marincic and D. Budimir, “Tesla’s multi-frequency wireless radio controlled vessel,” Sep. 2008, doi: 10.1109/HISTELCON.2008.4668708.

[23] J. Guittet and M. Parent, “Programming automatic reflex actions in telemanipulation,” Dec. 1979, doi: 10.1109/CDC.1979.270104.

[24] Won Kim, F. Tendick, and L. Stark, “Visual enhancements in pick-and-place tasks: Human operators controlling a simulated cylindrical manipulator,” *IEEE Journal on Robotics and Automation*, vol. 3, no. 5, Oct. 1987, doi: 10.1109/JRA.1987.1087127.

[25] J. De *et al.*, “Learning of operator hand movements via least angle regression to be teached in a manipulator,” *Evolving Systems*, vol. 0, p. 3, doi: 10.1007/s12530-018-

9224-1.

[26] R. A. Romeo and L. Zollo, “Methods and Sensors for Slip Detection in Robotics: A Survey,” *IEEE Access*, vol. 8, 2020, doi: 10.1109/ACCESS.2020.2987849.

[27] S. Fan *et al.*, “Object shape recognition and grasping by five-fingered robotic hand based on E-ANFIS model,” Dec. 2009, doi: 10.1109/ROBIO.2009.5420858.

[28] M. v. Liarokapis, P. K. Artemiadis, and K. J. Kyriakopoulos, “Telemanipulation with the DLR/HIT II robot hand using a dataglove and a low cost force feedback device,” Jun. 2013, doi: 10.1109/MED.2013.6608758.

[29] J. Miura and K. Ikeuchi, “Task-oriented generation of visual sensing strategies in assembly tasks,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1998, doi: 10.1109/34.659931.

[30] T. H. Park, H. J. Kim, and N. Kim, “Path planning of automated optical inspection machines for PCB assembly systems,” *International Journal of Control, Automation and Systems*, 2006.

[31] A. Dutta, G. R. Muzumdar, V. T. Shirwalkar, K. Jayarajan, D. Venkatesh, and M. S. Ramakumar, “Development of a dextrous gripper for nuclear applications,” doi: 10.1109/ROBOT.1997.614358.

[32] B. Calli and A. M. Dollar, “Vision-based model predictive control for within-hand precision manipulation with underactuated grippers,” 2017, doi: 10.1109/ICRA.2017.7989331.

[33] P. Galambos, A. Roka, P. Baranyi, and P. Korondi, “Contrast vision-based grasp force feedback in telemanipulation,” Jul. 2010, doi: 10.1109/AIM.2010.5695857.

[34] T. Xue, W. Wang, J. Ma, W. Liu, Z. Pan, and M. Han, “Progress and Prospects of Multimodal Fusion Methods in Physical Human–Robot Interaction: A Review,” *IEEE Sensors Journal*, vol. 20, no. 18, Sep. 2020, doi: 10.1109/JSEN.2020.2995271.

[35] Q. Li, R. Haschke, and H. Ritter, “A visuo-tactile control framework for manipulation and exploration of unknown objects,” Nov. 2015, doi: 10.1109/HUMANOIDS.2015.7363434.

[36] N. F. Lepora, K. Aquilina, and L. Cramphorn, “Exploratory Tactile Servoing With Active Touch,” *IEEE Robotics and Automation Letters*, vol. 2, no. 2, Apr. 2017, doi: 10.1109/LRA.2017.2662071.

[37] S. J. Lederman and R. L. Klatzky, “Extracting object properties through haptic exploration,” *Acta Psychologica*, 1993, doi: 10.1016/0001-6918(93)90070-8.

[38] A. M. Cretu, T. E. A. de Oliveira, V. Prado Da Fonseca, B. Tawbe, E. M. Petriu, and V. Z. Groza, “Computational intelligence and mechatronics solutions for robotic

tactile object recognition," 2015, doi: 10.1109/WISP.2015.7139165.

[39] C. Wu *et al.*, "A new approach for an ultrasensitive tactile sensor covering an ultrawide pressure range based on the hierarchical pressure-peak effect," *Nanoscale Horizons*, vol. 5, no. 3, 2020, doi: 10.1039/C9NH00671K.

[40] "Embedded Force Sensors | Tekscan." https://www.tekscan.com/products-solutions/embedded-force-sensors.

[41] "Peratech - Standard Products." https://www.peratech.com/standard-products/.

[42] "BioTac® Product Manual." BioTac, 2018, [Online]. Available: https://www.syntouchinc.com/wp-content/uploads/2018/08/BioTac-Manual-V.21.pdf.

[43] M. Borzage, "Startup Spotlight: SynTouch Seeks to Enable Robots With 'Machine Touch' - IEEE Spectrum." https://spectrum.ieee.org/automaton/robotics/robotics-hardware/startup-spotlight-syntouch.

[44] T. E. Alves De Oliveira, A. M. Cretu, and E. M. Petriu, "Multimodal Bio-Inspired Tactile Sensing Module," *IEEE Sensors Journal*, 2017, doi: 10.1109/JSEN.2017.2690898.

[45] G. A. Dale Purves David Fitzpatrick William Hall Anthony-Samuel Lamantia Leonard White, *Neuroscience. 5th Edition*. 2012.

[46] W. C. Nowlin, "Experimental results on Bayesian algorithms for interpreting compliant tactile sensing data," 1991, doi: 10.1109/robot.1991.131606.

[47] S. Zhai and P. Milgram, "Human-robot synergism and virtual telerobotic control," *Proceedings-HUMAN FACTORS ASSOCIATION OF CANADA*, 1992.

[48] J. R. Napier, "THE PREHENSILE MOVEMENTS OF THE HUMAN HAND," *The Journal of Bone and Joint Surgery. British volume*, vol. 38-B, no. 4, Nov. 1956, doi: 10.1302/0301-620X.38B4.902.

[49] M. Santello, M. Flanders, and J. F. Soechting, "Postural hand synergies for tool use," *Journal of Neuroscience*, vol. 18, no. 23, pp. 10105–10115, 1998, [Online]. Available: http://www.jneurosci.org/content/jneuro/18/23/10105.full.pdf.

[50] I. T. Jolliffe and J. Cadima, "Principal component analysis: a review and recent developments," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 374, no. 2065, Apr. 2016, doi: 10.1098/rsta.2015.0202.

[51] R. Vinjamuri, Z. H. Mao, R. Sclabassi, and M. Sun, "Time-varying synergies in velocity profiles of finger joints of the hand during reach and grasp," in *Annual International Conference of the IEEE Engineering in Medicine and Biology -*

Proceedings, 2007, pp. 4846–4849, doi: 10.1109/IEMBS.2007.4353425.

[52] I. v Grinyagin, E. v Biryukova, and M. A. Maier, “Kinematic and dynamic synergies of human precision-grip movements,” *Journal of Neurophysiology*, vol. 94, no. 4, pp. 2284–2294, 2005, doi: 10.1152/jn.01310.2004.

[53] G. Salvietti, “Replicating human hand synergies onto robotic hands: A review on software and hardware strategies,” *Frontiers in Neurorobotics*, 2018, doi: 10.3389/fnbot.2018.00027.

[54] P. V. de Campos Souza, “Fuzzy neural networks and neuro-fuzzy networks: A review the main techniques and applications used in the literature,” *Applied Soft Computing*, vol. 92, Jul. 2020, doi: 10.1016/j.asoc.2020.106275.

[55] A. L. Tascillo, V. A. Skormin, and N. Bourbakis, “Neurofuzzy grasp control of a robotic hand,” in *Neural Networks for Signal Processing III - Proceedings of the 1993 IEEE-SP Workshop*, 1993, pp. 507–516, doi: 10.1109/NNSP.1993.471837.

[56] J.-S. R. Jang, “ANFIS: adaptive-network-based fuzzy inference system,” *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 23, no. 3, 1993, doi: 10.1109/21.256541.

[57] S. Prakash and S. K. Sinha, “Simulation based neuro-fuzzy hybrid intelligent PI control approach in four-area load frequency control of interconnected power system,” *Applied Soft Computing*, vol. 23, pp. 152–164, 2014, doi: 10.1016/j.asoc.2014.05.020.

[58] D. Karaboga and E. Kaya, “Adaptive network based fuzzy inference system (ANFIS) training approaches: a comprehensive survey,” *Artificial Intelligence Review*, vol. 52, no. 4, Dec. 2019, doi: 10.1007/s10462-017-9610-2.

[59] O. M. Ahtiwash and M. Z. Abdulmuin, “An Adaptive Neuro-Fuzzy Approach for Modeling and Control of Nonlinear Systems,” 2001.

[60] D. van Cleave and K. S. Rattan, “Tuning of fuzzy logic controller using neural network,” doi: 10.1109/NAECON.2000.894925.

[61] D. Petković, M. Issa, N. D. Pavlović, L. Zentner, and Ž. Ćojbašić, “Adaptive neuro fuzzy controller for adaptive compliant robotic gripper,” *Expert Systems with Applications*, 2012, doi: 10.1016/j.eswa.2012.05.072.

[62] B. Peerdeman *et al.*, “Myoelectric forearm prostheses: State of the art from a user-centered perspective,” *The Journal of Rehabilitation Research and Development*, vol. 48, no. 6, 2011, doi: 10.1682/JRRD.2010.08.0161.

[63] C. Potluri *et al.*, “sEMG based fuzzy control strategy with ANFIS path planning for prosthetic hand,” Sep. 2010, doi: 10.1109/BIOROB.2010.5627782.

[64] G. Ouyang, X. Zhu, Z. Ju, and H. Liu, “Dynamical Characteristics of Surface EMG Signals of Hand Grasps via Recurrence Plot,” *IEEE Journal of Biomedical and Health Informatics*, vol. 18, no. 1, Jan. 2014, doi: 10.1109/JBHI.2013.2261311.

[65] C.-H. Chen, D. S. Naidu, A. Perez-Gracia, and M. P. Schoen, “Hybrid control strategy for five-fingered smart prosthetic hand,” Dec. 2009, doi: 10.1109/CDC.2009.5400771.

[66] “Human Computer Interface Examples,” *GetSmarter Blog*. 2018, [Online]. Available: https://www.getsmarter.com/blog/market-trends/14-human-computer-interaction-examples/.

[67] “Human–machine interfaces for medical imaging and clinical interventions.” https://reader.elsevier.com/reader/sd/pii/B9780128161760000387?token=C994A5ED181CCAAA06058E83525D49D45ED88A2932D3777B2054EA0C421520B52377156F214F25865B7DBED4F8EB7E4F.

[68] “What is Human-Computer Interaction (HCI)?,” *The Interaction Design Foundation*. https://www.interaction-design.org/literature/topics/human-computer-interaction.

[69] G. C. Burdea and P. Coiffet, *Virtual Reality Technology, Second Edition with CD-ROM*. Wiley-IEEE Press, 2003.

[70] “The Open Hand Project - Home.” http://www.openhandproject.org/.

[71] “Ada V1.1 Assembly Instructions — Open Bionics.” https://openbionicslabs.com/obtutorials/ada-v1-assembly.

[72] D. Prattichizzo, M. Malvezzi, M. Gabiccini, and A. Bicchi, “On motion and force controllability of precision grasps with hands actuated by soft synergies,” *IEEE Transactions on Robotics*, vol. 29, no. 6, pp. 1440–1456, 2013, doi: 10.1109/TRO.2013.2273849.

[73] I. M. Bullock, R. R. Ma, and A. M. Dollar, “A Hand-Centric Classification of Human and Robot Dexterous Manipulation,” *IEEE Transactions on Haptics*, vol. 6, no. 2, pp. 129–144, 2013, doi: 10.1109/TOH.2012.53.

[74] T. Feix, R. Pawlik, H.-B. Schmiedmayer, J. Romero, and D. Kragi, “A comprehensive grasp taxonomy,” in *ROBOTICS, Science and systems conference: workshop on understanding the human hand for advancing robotic manipulation*, 2009, pp. 2–3.

[75] I. M. Bullock and A. M. Dollar, “Classifying human manipulation behavior,” 2011, doi: 10.1109/ICORR.2011.5975408.

[76] “Guides to the Evaluation of Permanent Impairment,” *JAMA: The Journal of the American Medical Association*, 1971, doi: 10.1001/jama.1971.03190110063016.

[77] “rosserial_arduino/Tutorials - ROS Wiki.” http://wiki.ros.org/rosserial_arduino/Tutorials.

[78] M. Quigley *et al.*, “ROS: an open-source Robot Operating System,” p. 6.

[79] “sensor_msgs - ROS Wiki.” http://wiki.ros.org/sensor_msgs.

[80] “Packages - ROS Wiki.” http://wiki.ros.org/Packages.

[81] N. Kang, “PL-2303X Edition USB to Serial Bridge Controller Product Datasheet,” *Interface*, vol. 7, no. 48, pp. 1–18, 2004.

[82] D. Zhi, T. E. A. de Oliveira, V. P. da Fonseca, and E. M. Petriu, “Teaching a robot sign language using vision-based hand gesture recognition,” 2018, doi: 10.1109/CIVEMSA.2018.8439952.

[83] Clayton. Valli and Ceil. Lucas, *Linguistics of American Sign Language : an introduction*. Gallaudet University Press, 2000.

[84] M. Malvezzi, G. Gioioso, G. Salvietti, and D. Prattichizzo, “SynGrasp: A MATLAB toolbox for underactuated and compliant hands,” *IEEE Robotics and Automation Magazine*, vol. 22, no. 4, pp. 52–68, 2015, doi: 10.1109/MRA.2015.2408772.

[85] M. Malvezzi, G. Gioioso, G. Salvietti, D. Prattichizzo, and A. Bicchi, “SynGrasp: A MATLAB toolbox for grasp analysis of human and robotic hands,” in *2013 IEEE International Conference on Robotics and Automation*, 2013, pp. 1088–1093, doi: 10.1109/ICRA.2013.6630708.

[86] A. J. Spiers, M. v Liarokapis, B. Calli, and A. M. Dollar, “Single-Grasp Object Classification and Feature Extraction with Simple Robot Hands and Tactile Sensors,” *IEEE Transactions on Haptics*, vol. 9, no. 2, pp. 207–220, 2016, doi: 10.1109/TOH.2016.2521378.

[87] C.-U. Yeom and K.-C. Kwak, “Performance Comparison of ANFIS Models by Input Space Partitioning Methods,” *Symmetry*, vol. 10, no. 12, p. 700, 2018, doi: 10.3390/sym10120700.

[88] J.-S. R. Jang and N. Gulley, “MATLAB ® User’s Guide Fuzzy Logic Toolbox Computation Programming Visualization,” 1984. [Online]. Available: http://www.mathworks.com.

[89] “Neuro-Adaptive Learning and ANFIS - MATLAB & Simulink.” https://www.mathworks.com/help/fuzzy/neuro-adaptive-learning-and-anfis.html.

[90] “Coefficient of Determination, R-squared.” https://internal.ncl.ac.uk/ask/numeracy-maths-statistics/statistics/regression-and-correlation/coefficient-of-determination-r-squared.html.

[91] D. Rodić, M. Sekulić, M. Gostimirović, V. Pucovsky, and D. Kramar, “Fuzzy logic and sub-clustering approaches to predict main cutting force in high-pressure jet assisted turning,” *Journal of Intelligent Manufacturing*, vol. 32, no. 1, pp. 21–36, 2021, doi: 10.1007/s10845-020-01555-4.

[92] “RMSE: Root Mean Square Error,” *Statistics How To*. https://www.statisticshowto.com/probability-and-statistics/regression-analysis/rmse-root-mean-square-error/.