Teleoperated Grasping Using an Upgraded Haptic-Enabled Human-Like Robotic Hand and a CyberTouch Glove

by

Qi Zhu

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

M.A.Sc. in Electrical and Computer Engineering

School of Electrical Engineering and Computer Science
Faculty of Engineering
University of Ottawa

Abstract

Grasping, the skill to hold objects and tools while doing in-hand manipulation, still is in many cases an unsolvable problem for robotics, but a natural act for humans. An efficient grasping requires not only human-like robotic hands with articulated fingers but also tactile, force, and kinesthetic sensors for the precise control of the forces and motions exerted during the manipulation.

As a fully autonomous robotic dexterous manipulation is too difficult to develop for changing and unstructured environments, an alternative approach is to combine the low-level robot computer control with the higher-level perception and task planning abilities of a human operator equipped with an adequate human-computer interface (HCI).

This thesis presents theoretical and experimental contributions to the development of an upgraded haptic-enabled anthropomorphic *Ring Ada* dexterous robotic hand and a biology-inspired synergistic real-time control system for teleoperated grasping of different objects using a *CyberTouch* HCI data glove. A fuzzy logic controller module was developed to efficiently control the underactuated *Ring Ada'* robotic hand during grasping. A machine learning classification system was developed to recognize grasped objects.

Experiments have convincingly demonstrated that our novel *Ring Ada* robotic hand equipped with kinematic position sensors and touch sensors is able to efficiently grasp different lightweight objects through teleoperation.

Acknowledgements

I would like to thank my supervisor, Dr. Emil M. Petriu, for his continuous encouragement and insightful advice throughout my thesis research. Without his incredible and detailed support, my thesis could not be completed.

I also would like to express my gratitude to all the colleagues in the BioIn Robotics Lab, for their assistance and suggestions throughout my research. My sincere thanks to Vinicius for his professional perspectives and cooperation provided for my project. Also, thanks to Bruno for inspiring helpful ideas, and to Maxwell for help with proofreading. I am so happy to have them as colleagues.

Finally, a special thanks to my beloved family, my parents, and my girlfriend for their endless love, understanding, and financial support. This accomplishment would also belong to them by their great dedication.

Table of Contents

List of Figures

List of Tables

Nomenclature

Abbreviations

CMC	Carpometacarpal
DIP	Distal Interphalangeal
DOF	Degree of Freedom
FLC	Fuzzy Logic Control
FSR	Force Sensing Resister
HCI	Human Computer Interface
IDE	Integrated Development Environment
IMU	Inertial Measurement Unit
K-NN	K-Nearest Neighbors
MARG	Magnetic, Angular rate, and Gravity
MCP	Metacarpophalangeal
MLP	Multilayer Perceptron
PC	Principle Component
PCB	Printed Circuit Board
PIP	Proximal Interphalangeal
PLA	Polylactic Acid
ROS	Robot Operating System
RPC	Remote Procedure Call
SDK	Software Development Kit
SVM	Support Vector Machine
SVR	Support Vector Regression

Chapter 1. Introduction

Since the mid 20th century, robots are slowly integrating into our daily life as helpful partners offering tireless and high precision assistance. For over 50 years industrial robotic arms are very efficiently making and assembling a variety of products. Special robots were designed to face extreme environmental condition and go to unreachable places such for instance the International Space Station, or the Mars planet. A small humanoid robot was developed and is commercially available to interact with people as house pets. However, for robots' skills to match those of humans, there still are great challenges to be solved [1].

Robots can improve their overall skills by developing a more effective in-hand object manipulation ability which is crucial for many human-like activities[2]. Robots' new ability influences human productivity, just as all tools are innovated by human beings in history.

The idea of developing human-like hands for robotics could call back to 1977 with the work of Asada and Hanafusa [3] in the field of robot grasping, and become more practical with Salisbury's first attempts to develop a three-fingered robotic hand [4]. For the robotic hand, we not only expect it just to hold the object sturdy but also to perform complex key grasping or even in-hand manipulation operations.

Haptic exploration research is thriving nowadays since it's an important mechanism by which the users learn about the surface properties of unknown objects. Shapes,

textures, stiffness, and temperature information could be collected instantly through a single touch. This ability to have such a so-called 'haptic glance' would allow robotic hands to explore unknown environments, instead of staying in well-known, structured environments. However, we should not forget that the haptic exploration is an active perception act that involves direct interaction with the object being explored, which presents significant challenges in both sensing and control [5].

Teleoperated dexterous object exploration, grasping and manipulation can also allow developing a new generation of intelligent robots with advanced, human-like, tactile perception capabilities enabling them to perform complex in-hand telemanipulation operations under poor or nonexistent visibility conditions, such as underwater, in space, in hazardous or high-risk security operational environments like highly infectious hospital rooms, where touch feeling is of paramount importance.

As presented in [6] "Robotic telemanipulation is an object-oriented act which requires not only specialized robotic hands with articulated fingers but also tactile, force and kinesthetic sensors for the precise control of the forces and motions exerted on the manipulated object. When a fully autonomous robotic dexterous manipulation is impractical in changing and unstructured environments, an alternative approach is to combine the low-level robot computer control with the higher-level perception and task planning abilities of a human operator equipped with adequate Human Computer Interfaces (HCI)", as illustrated in *Figure* 1.1.

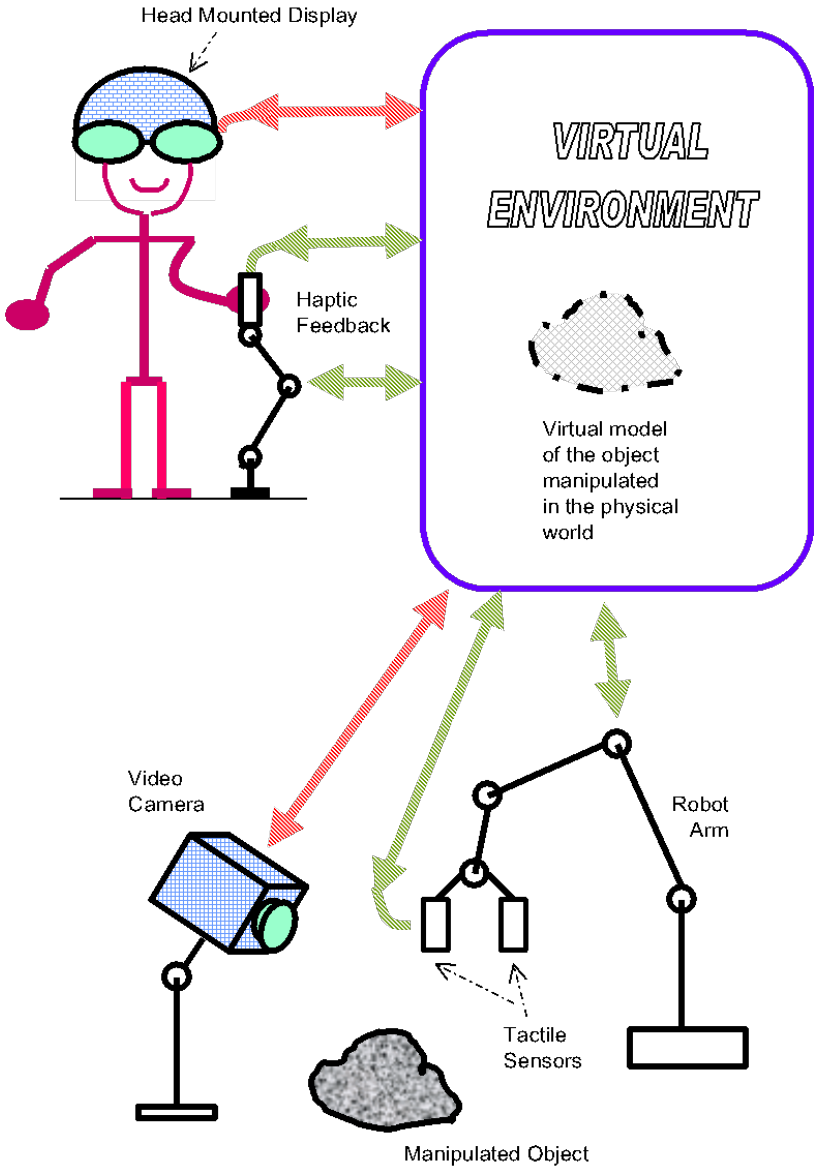


Figure 1. 1 Robotic telemanipulation [6].

Telemanipulation robotic systems should have a bilateral architecture that allows a human operator to connect as much as possible in a transparent manner to a remote robotic manipulator. Convenient Human Computer Interfaces (HCI) should be used to provide easily perceivable and task-related sensory feedback, which fit naturally the perception capabilities of the human operator.

1.1. Problem definition

This thesis will attempt to address specific aspects related to the very complex and challenging sensor-enabled dexterous robotic telemanipulation problem, which requires not only human-like robotic hands with articulated fingers but also tactile, force, and kinesthetic sensors for the precise control of the forces and motions exerted during the manipulation. As fully autonomous robotic dexterous manipulation is too difficult to develop in changing and unstructured environments, an alternative approach is to combine the low-level robot computer control with the higher-level perception and task planning abilities of a human operator equipped with adequate human computer interface.

Dexterous object grasping is a complex operation in which multiple fingers cooperate to carry out this task. Controlling a dexterous robotic hand poses major design challenges for the tactile sensors which are the first line of contact to detect the contact pressure, and then for the tactile and kinesthetic feedback-based coordinated control of finger movements and applied forces.

The human hand is one of the most versatile multifigured gripper available in nature, which can handle objects of different sizes, shapes, and weights with a high level of dexterity. A reach body of literature on the human hand structure, touch, and dexterous grasp are available to be used by engineers as a source of inspiration for a more advanced robotic haptic sensing and perception, and object grasping and manipulation.

Object grasping combines four major elements called ‘manipulation science’ [7]

include: the task, object, robotic gripper, and controller modules, as illustrated in *Figure* 1.2.

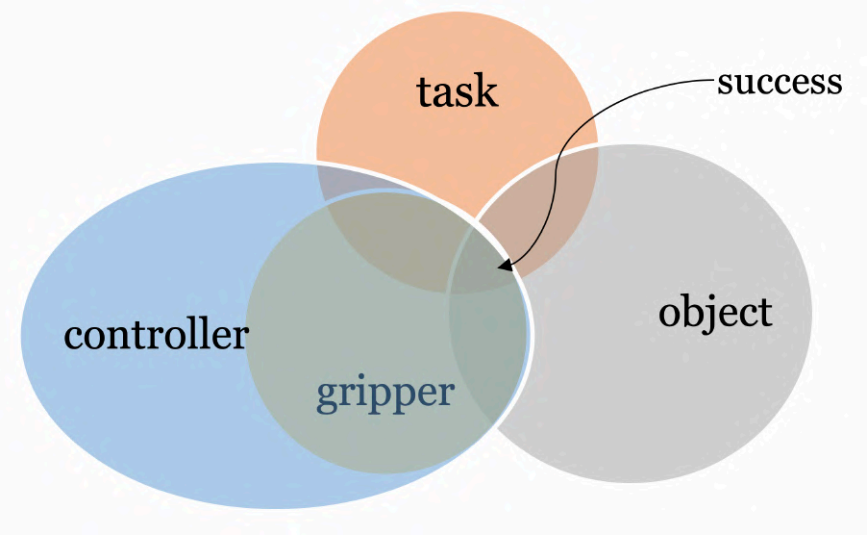


Figure 1.2 The concept of successful grasping

The grasping task has to adjust to fit the gripper design constraints. The object size and shape should fit in the reachable space of the gripper and its surface texture needs have enough friction to avoid getting loose. A proper controller design has to take into consideration the robotic gripper's structural parameters, and its actuators and sensors' performance.

Although all functions and equipment are well prepared, the objects still cannot always be picked up in our desired configurations. However, dexterous manipulation enables a robot to reposition an object while holding within its hand[8]. More predictable and robustness in the grasp execution can be achieved through incorporating data provided by the haptic sensors disposed on the robotic hand such as the contact force,

the relative velocity at the contact points, or even the material rigidly.

Grasping, the skill to hold objects and tools while doing in-hand manipulation maintaining a stable grasp, still is in many cases an unsolvable problem for robotics, but a natural act for humans, even for kids.

A new approach must focus on the robotic hand's compliance, tactile sensing, smart actuation [1], and multi-sensor control [7], all these combined with teleoperation haptic-feedback HCIs.

1.2. Objectives

We aim 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 thesis objectives are as follows:

- Develop an experimental haptic-enabled human-like multi Degrees of Freedom (DOF) with independent actuated joints dexterous robotic hand.
- Study an efficient control system for haptic-enabled teleoperation of the new experimental hand using a *CyberTouch* glove [9].
- Study an object classification system of the grasped objects in order to assist the human teleoperator to better carry on the telemanipulation.

1.3. Thesis Contributions

In this thesis, an upgraded five-finger dexterous tactile enabled underactuated robotic hand was designed and build based on opensource resources. The thesis makes the following contributions:

- Developed an experimental haptic-enhanced human-like five-finger robotic hand, called the *Ring Ada* hand, by adding a set of 'Rings' in order to add DOFs to an open-source available Ada hand [10], and also adding FSR tactile sensors to the robotic hand's fingertips and palm.

- Experimentally studied the reachability of the new *Ring Ada* robotic hand.
- Developed kinematic synergy and postural synergy control system for the *Ring Ada* robotic hand to perform teleoperated stable grasping.
- Developed an FLC (fuzzy logic controller) for the new *Ring Ada* hand, using data from the FSR touch sensors on the fingertips and the hand's linear actuators and servo motors, to perform stable object self-grasping.
- Experimentally implement the object classification system using position output from the FLC of the robotic hand's linear actuators and servo motors. This could assist the human teleoperator to better carry on the telemanipulation.

1.4. Publications Arising from the Thesis

The author has published four international conference papers on topics directly related to the work presented in this thesis:

(1) *Qi Zhu,* Vinicius Prado da Fonseca, Bruno Monteiro Rocha Lima, Maxwell Welyhorsky, Mirian Goubran, Thiago Eustaquio Alves de Oliveira, and Emil M. Petriu, "Teleoperated Grasping Using a Robotic Hand and a Haptic-Feedback Data Glove," IEEE International Systems Conference - SysCon 2020, (accepted).

(2) Bruno Monteiro Rocha Lima, Vinicius Prado da Fonseca, Thiago Eustaquio Alves de Oliveira, *Qi Zhu*, and Emil M. Petriu, "Dynamic Tactile Exploration for Texture Classification using a Miniaturized Multi-modal Tactile Sensor and Machine Learning." *IEEE International Systems Conference - SysCon 2020*, (accepted).

(3) Vinicius Prado da Fonseca, Bruno Monteiro Rocha Lima, Thiago Eustaquio Alves de Oliveira, *Qi Zhu*, Voicu Z. Groza, and Emil M. Petriu, "In-Hand Telemanipulation Using a Robotic Hand and Biology-Inspired Haptic Sensing," *Proceedings of the 2019 IEEE International Symposium on Medical Measurements and Applications (MeMeA),* [11].

(4) Bruno Monteiro Rocha Lima, Thiago Eustaquio Alves de Oliveira, Vinicius Pradoda Fonseca, *Qi Zhu*, Miriam Goubran, Voicu Z. Groza, and Emil M.

Petriu, “Heart Rate Detection Using a Miniaturized Multimodal Tactile Sensor,” *Proceedings of the 2019 IEEE International Symposium on Medical Measurements and Applications (MeMeA),* [12].

1.5. Thesis Organization

The thesis is organized as follows:

Chapter 1. Introduction. The problem definitions, objectives, a list of contributions, and a list of author’s published papers are presented in this chapter.

Chapter 2. ***Literature Review***. This chapter presents a review of major research developments from the past to the most recent, reported in the literature on human hands, robotic hands, vision and tactile for robotic manipulation, the dexterous multi-fingered object grasping and manipulation from the synergy aspects, and a brief introduction of machine learning.

Chapter 3. System Overview. This chapter presents an overview of the hardware (such as the *CyberTouch* HCI teleoperator glove, and the novel upgraded *Ring Ada* robotic hand) and the software (such as the Robot Operating System (ROS), and the software for Leap Motion controller), components used by our experimental system setup.

Chapter 4. Teleoperated Grasping. This chapter illustrated the design of the kinematic and postural synergies, the tactile sensors, and the feedback strategy used during the robotic hand grasping. Experimental results illustrating the effective haptic

feedback actuation and the teleoperated stable grasping for different objects are provided.

Chapter 5. Grasp Control Skills. This chapter discusses different aspects of synergistic control skills developed for the Ring Ada robotic hand. A fuzzy logic controller was developed to drive self-explored grasping operations. Using a human-like object approaching behavior, the *Ring Ada* achieve basic object classification skills using machine learning algorithms.

Chapter 6. Conclusion and Future Work. This final chapter includes conclusions about the work reported in the thesis. It also provides ideas on how the results of this work could be used in the future to develop a new generation of anthropomorphic robotic hands that will further reduce the gap between the robotic and the human hands.

Chapter 2. Literature Review

From Human Hand to Robotic Hand

Dexterous hands have helped humans build up the history of civilization. The delicate hierarchic controlled mechanical structure and its haptic sensing system let the human hand be the main tool for humans to interact with their surroundings. Tools have been built over millennia to optimize their efficiency when used by human hands.

It was quite logical that the robotic hand designs have been inspired from the very beginning by the human hand.

A robot is not anymore just a bulky, rigid, and clumsy machine. One of the biggest challenges that are currently undertaken is to study and develop robots that are more agile and dexterous. Significant advances over the past decade have been made in robotics, sensors. artificial intelligence and other related fields, allowing the development of highly sophisticated bio-mimetic robotics systems [13]. So, one can wonder, if the human hand's astonishingly dexterous structure and perception capabilities could be now used as a base for an anthropomorphic design, how much this would magnificently improve the capability of the next generation of humanoid robots?

2.1. The Human Hand

The human hand is a true masterpiece from the mother nature, which evolved from a 'foot' functionality to a more versatile dexterous manipulation tool function.

Dexterous object exploration, grasping and in-hand manipulation are complex operations in which multiple fingers, equipped with cutaneous touch and kinesthetic sensors, cooperate to carry out these operations. Cutaneous touch sensors provide information about the geometric shape, contact force, elasticity, texture, and temperature of the touched object area. Kinesthetic information about the positions and velocities of the hand's kinematic structure is provided by sensors on muscles, tendons, and joints.

Even though there are already many robot gripper and hand designs, the human hand's unbeatable flexibility and adaptability have not yet been matched by any robotic artifact. And another bonus for anthropomorphic robotic hand design is higher acceptability for the human environment. So, before designing and building any human-like robotic gripper, we need to better understand how the human hand works.

Figure 2.1 Human hand bone anatomy, right hand (from [11]].

Figure 2.1. shows the bone anatomy of a human hand according to [11]. A human hand consists of a palm and five fingers. The grasping is performed by the fingers, which more specifically, the phalanges. Phalanges are the small bones that constitute the skeleton of the fingers and thumb in hand and feet for most vertebrates.

We could see that the thumb of the human hand is positioned separate from the other four fingers, [11] and [15]. And considering the carpal part of the hand, the index, middle, ring, and pinky would not move individually without any other finger participate. The thumb has the most complex structure amongst the hand fingers and differs from the other fingers in that it contains only two phalanges and has 5 DOFs. The thumb also has the highest flexibility and could turn towards the palm, which makes the grasping possible.

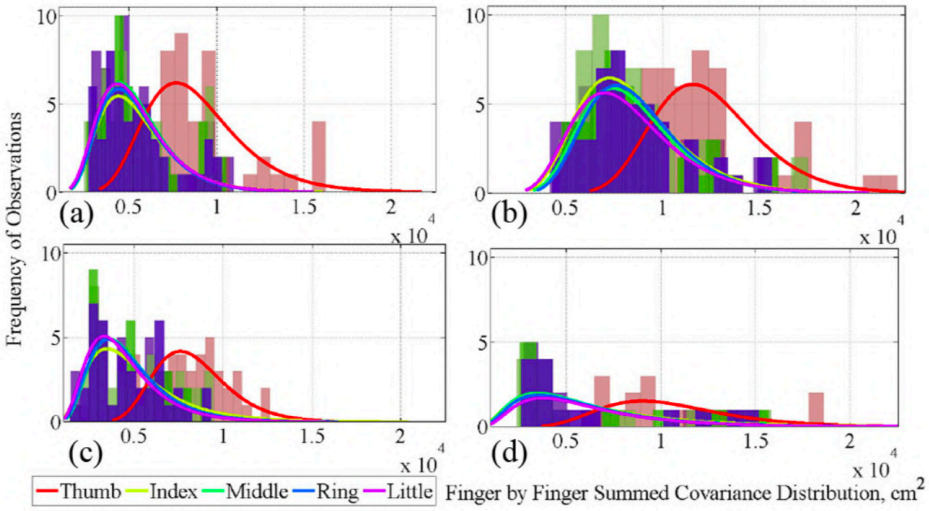


Figure 2.2 Participation of each finger during grasping. (a)(b)(c)(d) respectively are the power, precision, key and primate grasp.

The thumb dexterity helps to handle objects of various sizes and shapes. In *Figure* 2.2, each *Figure* highlights the strong magnitude of covariance between the thumb and the other fingers. In other words, the motion of the thumb is dominating the displacement of the other fingers. In addition, this trend can be observed in all four posture classes.

Figure 2.3 shows the simplified kinematic structure of an anthropomorphic hand having a grand total of 27 DOFs. The thumb CMC parts has 2 DOFs and more than 55° adduction angle helping the thumb fold towards the palm. The index finger has the greatest range of motion amongst the fingers with the extension/flexion movement of 80°, 110°, and 90° at the DIP, PIP, and MCP joints respectively. The abduction and adduction angle has been measured as 20° at the MCP joint in the index finger based on the literature [16].

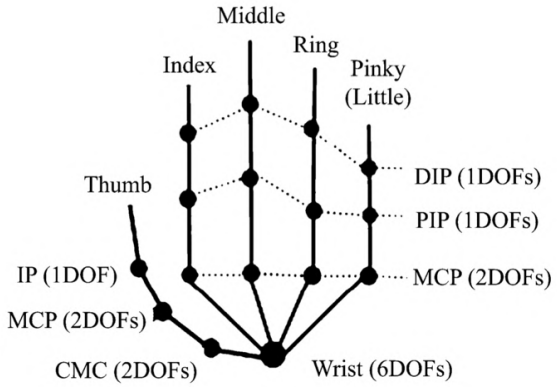


Figure 2.3 Simplified human hand kinematic structure

Many robotic hand designs aimed to mimic the human hand structure having between 15 to 25 joints. *Figure* 2.4 shows as an example the 25 DOF kinematic model of a human hand [17].

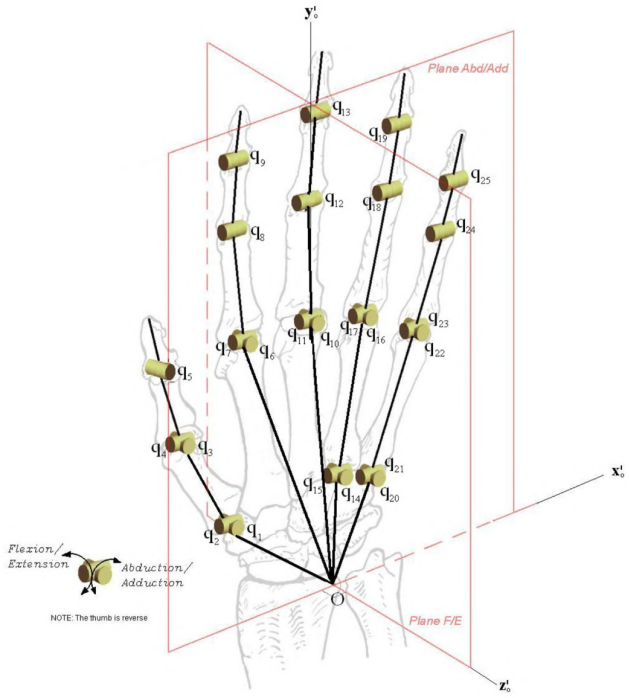


Figure 2.4 The DOFs of the human hand [17].

The wrist is a key feature of the human hand. As shown in *Figure* 2.1 the wrist consisted of eight carpals (or carpus) bones, these bones could be divided into two sub-categories, the proximal and the distal rows. All these bones are connected together as an interweaving network of ligaments. Ligaments work their own way through the wrist jointing the carpals to each other, and also connecting the carpal bones to the radius and ulna. The wrist has three DOFs: flexion to front/extension to back (pitch), radial deviation to center/ulnar deviation to outward (yaw); pronation/supination (roll) which occurs between the distal radius and ulna, although 2–12° occurs between the radius and the carpals at extremes of forearm rotation [18].

According to medical studies [19][20], the main design requirements for artificial wrist implants are : (1) relieve pain; (2) be stable; (3) provide a functional range of motion; and (4) correct deformity.

The picture in *Figure* 2.5 explains the basic movement of the wrist from the literature [21], which discusses seven different wrist implant designs and the materials used for the artificial recreated joints.

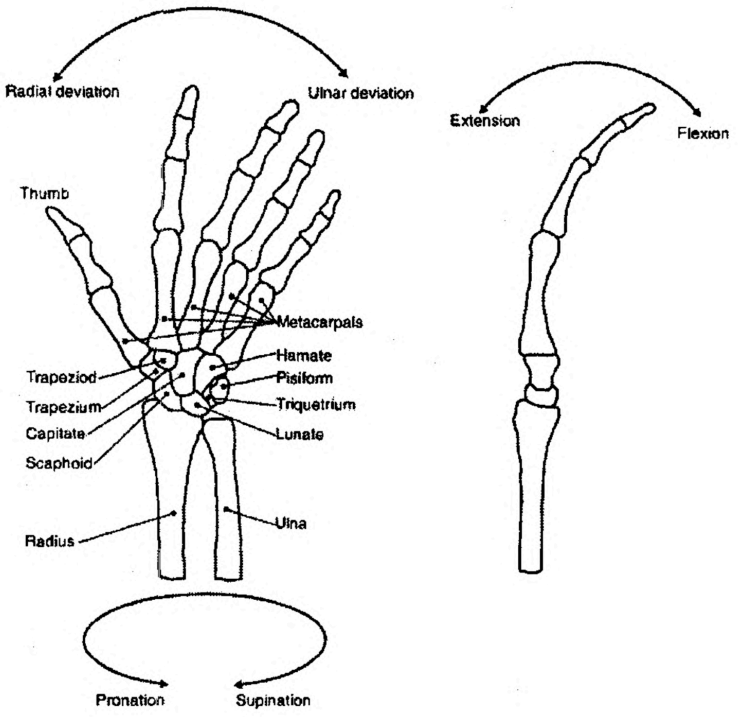


Figure 2.5 The basic movements of the wrist

The complicated structure made the human wrist hard to mimic. So, most of the robotic hand designs focus only on the palm and fingers. Only a few pieces of research using a group of complementary tendons succeeded to achieve relatively free movement in all three DOFs, for example, the *Shadow Dexterous Hand* which will be introduced in the following section.

2.2. Robotic Hands

We could find over 70 robotic hand designs inspired by the human hand structure in the literature. Every research team attempted to develop a design that offers the best possible performance on a limited budget.

All things considered, for the anthropomorphic robotic hand designs, packing more than 20 DOFs architecture into a hand size space especially in for a finger remains a significant challenge. Efforts have been made in eight areas: 1) structures; 2) internal sensors; 3) external sensors; 4) actuation systems; 5) remote systems; 6) covers; 7) communication networks; and, 8) computation systems [22].

Two distinct actuation solutions are currently available: (i) tendon-driven robotic hands, and (ii) actuator-joint base robotic hands.

Tendon driven robotic hands

The tendon is a basic actuation organ used by almost all creatures on the planet. Benefiting from the improved electric motor designs, the tendon driven actuators are offering an efficient solution to replace the cumbersome fluid drive actuators [23] in small scale systems such as the robotic hands. The robotic manipulators could use different types of tendons, for instance, belts, tapes, cables, ropes, chains, or specially designed fibers (bio-inspired tendons having similar performance as the human tendons). These tendons allow actuating manipulators remotely without a bulky terminal energy transformation system. They have the advantages of low inertia, backlash, friction, and minimal end-effector volume.

There are three main tendon-driven systems as shown in *Figure* 2.6 [24].

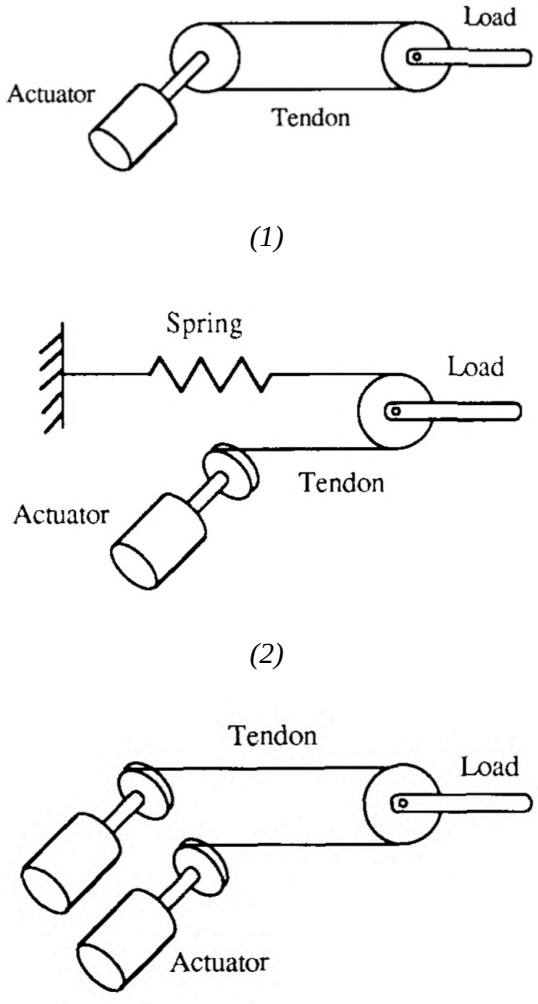


Figure 2.6 (1) Single joint driven by a rotary actuator. (2) Spring & actuator for the single joint. (3) Dual actuator drives one single joint. (from [24]).

The first system, shown in *Figure* 2.6 (1), uses a continuous cable driven by a single actuator. The motor and the cable are relatively fixed which means the actuator is driving the joint continuously.

A three-finger robotic hand has been developed using this first rotary actuator configuration [25]. This approach requires pre-tensioning of the system, which should be

strong enough not only to keep the link off the joint but also to prevent the cable slipping or allow the tendon slack when the joint moves at higher velocities. The advantage of this solution is to keep the simplicity and easy to design. However, because of the pre-tension, friction and backlash are significant, which obviously decreases the overall performance.

The second system is also driven by a single actuator, but instead of the closed-loop cable, a permanent positioning spring is used, as shown in *Figure* 2.6 (2) [24]. This approach was used by the *Hitachi Hand* [26] developed by Nakano and Hosada. This configuration prohibits small co-contractions when hard springs are used, which are necessary for high extension force and rapid response time. High energy dissipation is also expected since the actuator is required to pull constantly to maintain zero torque at the joint. The other disadvantage is that if any tendon slips, the joint will experience a large amount of error. It may be worth noting that the *Open Bionics Ada* robotic hand [10] is using this system.

The third tendon-driving system shown in *Figure* 2.6 (3) [24] is a true push-pull actuator. Although this approach increases the volume of the actuation package, the configuration provides for low co-contraction forces, independently controlled joints, and equal-strength actuators and tendons. The advantages of this configuration are:

- relatively minimal amount of the drag strength on the opposing actuator.
- the smallest tension required of the tendon (the lowest co-contraction)
- very good position control.
- simple implementation.

The most representative application of this push-pull tendon driving solution is the popular sophisticated multi-fingered robotic hand *Utah/MIT Dexterous Hand* [22], [27]. Version 3 of the *Utah/MIT Dexterous Hand* is shown in *Figure* 2.7 [28]. It has 32 specially designed actuator powered tendon driven system design and an elegant 4 finger mechanical structure. This hand features: 1) 16 degree of freedom; 2) very high active and passive performance; 3) good reliability; 4) allows studying a broad variety of tactile sensing systems; and, 5) through its modularity offers the possibility of geometric reconfigurations to address different experimental objectives [22].

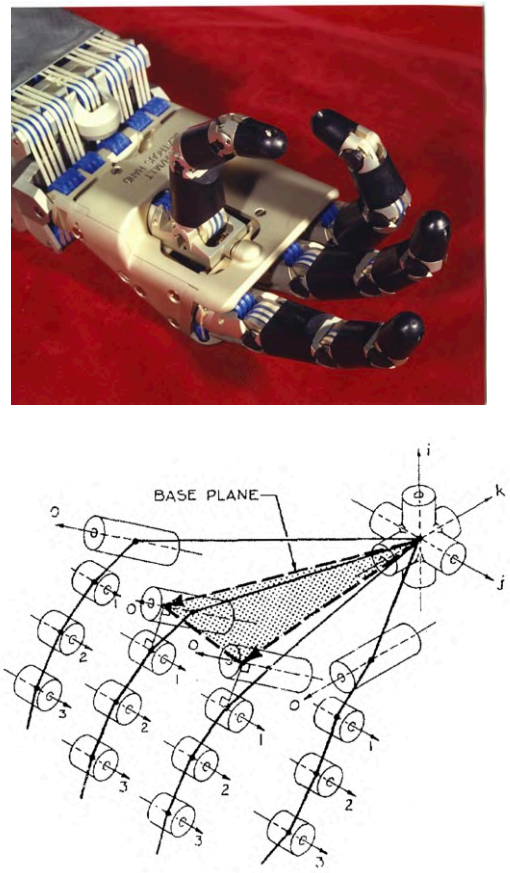


Figure 2.7 Utah/MIT Dexterous Hand version 3 and its kinematic structure (from [28])

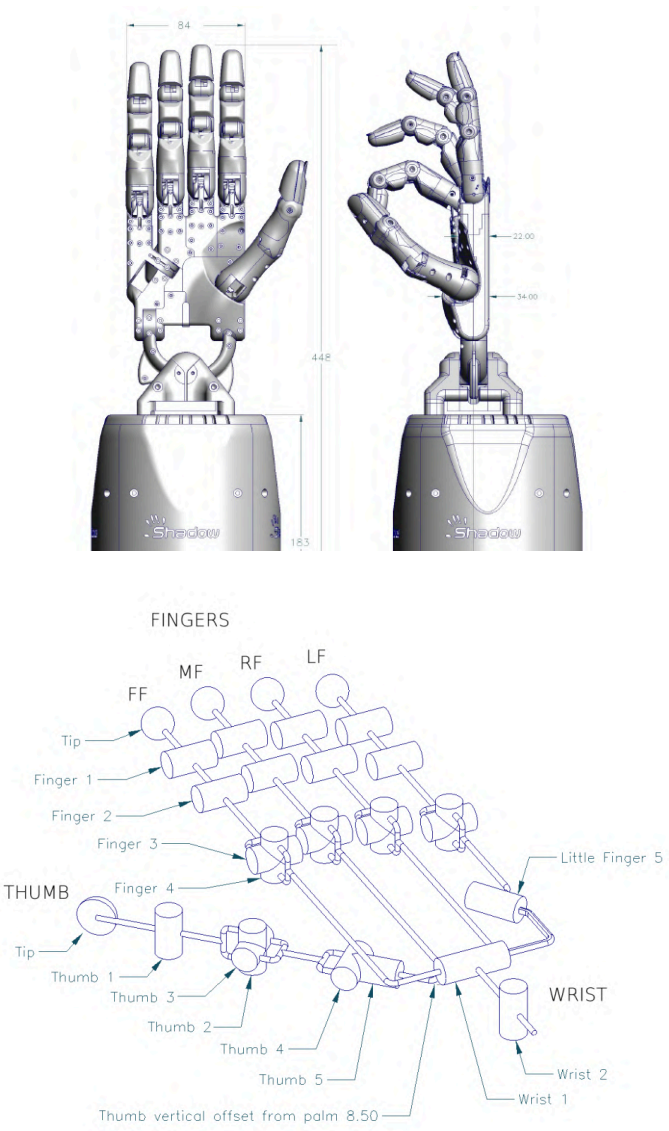


Figure 2.8 Shadow Dexterous Hand and its kinematic structure (from [29]).

Figure 2.8 shows the tendon driven Shadow Dexterous Hand™ [29]. It provides a total of 24 total DOFs (20 actuated degrees of freedom and a further 4 under-actuated movements for a total of 24 joints, using continuous cable tendons) and packed in six-axis gyro and accelerometer, joint angle sensors, force-torque sensors, tactile sensors, current and temperature sensors [30], up to 129 sensors in total [31].

As shown in *Figure* 2.8 the thumb has 5 DOFs, especially the 'thumb 2' and 'thumb 5' joints imitate the human hand muscle functions to provide similar kinematics that imitates better the human hand functions and allows the thumb to turn towards the palm and touch other fingers. And the 'little finger 5' joint is designed in such a way to allow the little finger draw into the palm in order to better fit different object shapes into the hand.

The *Shadow Dexterous Hand* has one special part which does not exist in other types of robotic hands, the joint 'finger 4' that is designed to allow the stretch and draw of the four fingers. This expands the dexterity of the hand and makes it possible to actually turn an object inside the hand, which provides for better in-hand manipulations and allows to execute special hand gestures.

The fully actuated human-like joints let Shadow Hand become the benchmark of finger dexterity and grasping. it keeps updating through the years. The Shadow Hand was developed to easily adapt to different robotic platforms. It should be noted that it was one of the first supporting interaction with the *Cyber Glove* [32] systems.

Actuator-joint robotic hands

In the case of the tendon-driven robotic hands, each finger joint is controlled by a tendon driven by a remote actuator usually positioned in the forearm of the robotic manipulator.

Benefiting from the recent development of new motor technologies, the finger actuators can finally have the size, torque, low power consumption, and low weight that

will allow them to be placed directly in the finger joints. There is no need to have actuator motors or pneumatic muscles positioned in the forearm, which will leave precious room for other parts of the robotic arm.

A representative example is the 16 DOFs *KITECH-Hand* [33] shown in *Figure* 2.9, which has the motors directly located in the respective finger joints. The PIP and DIP are single-DOF joints allowing flexion and extension. The MCP is a 2-DOF joint enabling flexion and extension and the abduction and adduction of the finger.

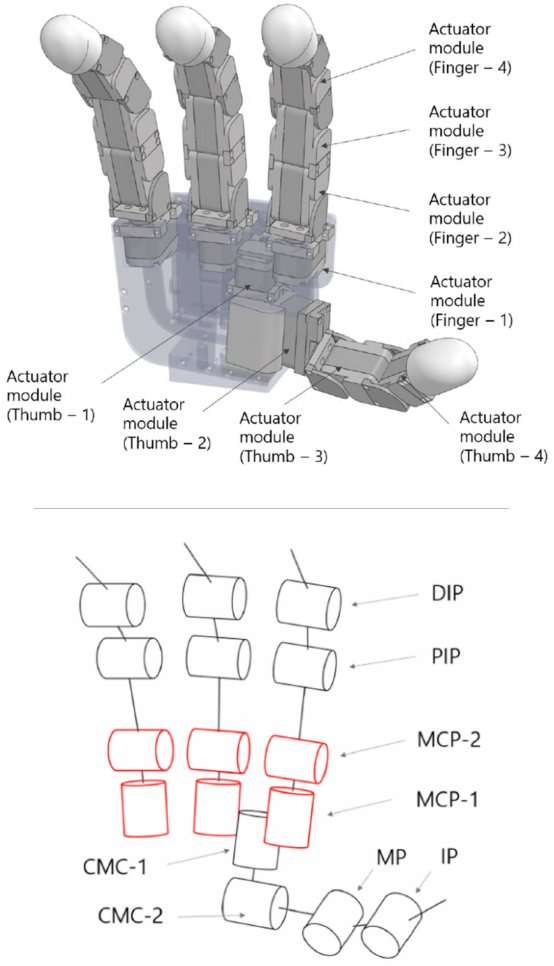


Figure 2.9 The KITECH-Hand and *its kinematic structure (from [33]).*

The lightweight is another advantage of this direct actuator-joint humanoid robotic hand. The *KITECH-Hand* is designed to weigh less than 1 kg, including the embedded electronics. The motors in this hand are small RC servo modules, which are very cheap and easy to obtain. While the use of this solution leads to a slight inaccuracy, it nevertheless allows us to minimize the size and complexity of the robotic hand [34].

While all these robotic hands provide a lot of DOFs, one major problem which remains is their cost, e.g. even the cheapest *KITECH-Hand* costs some 10,000 USD [34]. On the other hand, it is becoming clear that while the simpler three-finger robotic hands have allowed conducting some valuable research on grasping and in-hand manipulation [35] [36], these relatively simple hands are not actually dexterous enough and may easily lead to unstable grasping and in-hand manipulation [37], [35].

2.3. Sensors for Robotic Manipulation.

Locating, grasping, and manipulating objects is a natural behavior for humans, but it remains still a major challenge for most of the robotic hands. Even the most sophisticated robotic hands available today cannot make efficient stable grasp, lift, and dexterously manipulate objects.

Grasping control includes two major steps: object recognition, and coordinated finger motion planning. While the visual and haptic perception modalities complement each other, the touch plays a major role in the real-time control of the dexterous.

2.3.1. Vision

Over many decades, computer vision has been used for a multitude of industrial assembly and object manipulation applications. The video cameras are usually placed outside on the robotic hand. However, compact video camera developments allow being attached these cameras to the end-effector as *eye-in-hand* sensors. The Vision-based manipulation [38] is from a visual guide to model-based 3D object recognition.

Efficient vision-guided approaches have been developed for robotic object recognition and manipulation on the modern industrial assembly lines, to improve precision and productivity [39] [40].

Model-based 3D object recognition relies on the recovery of several relevant object-specific features such as edges; color, tags, and depth information from stereo vision [41] or structured light [42]. An eye-in-hand system [43] was also developed to provide smaller position uncertainties for more precise control.

However, although vision could be useful to approach the objects, the partial occlusion after reaching and grasping make tactile sensing more important a feature for successful dexterous in-hand manipulation [44].

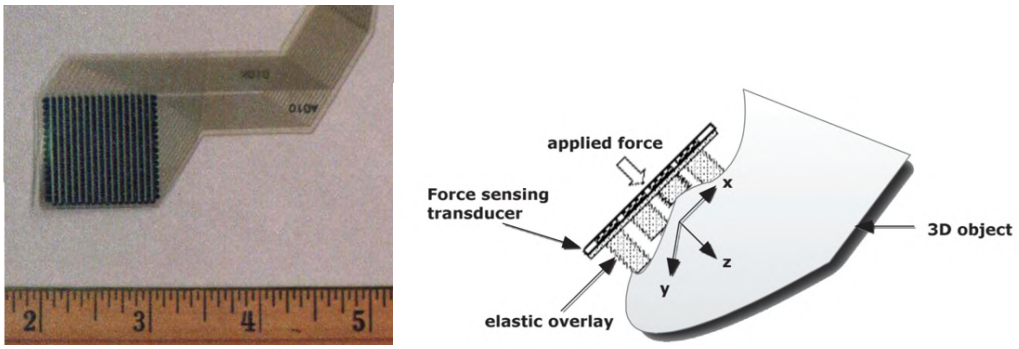
2.3.2. Tactile Sensors

As discussed in [1] the development of an efficient human-like haptic sensing capability for dexterous robotic hands requires mimicking the complicated human hand's touch sensing mechanisms during the dexterous manipulation of objects. Human touch

is the result of a complex investigatory object handling act involving two distinct sensing components: (i) the cutaneous information from tactile sensors about the topology, texture, contact force, and elasticity of the touched object surface, and (ii) the kinesthetic information about the position and velocity of the kinematic structure of the hand.

In unstructured environments, an object recognition ability in the early grasp phase is fundamental for the dexterous robotic hands needing to grasp objects as humans do. Humans developed an ability called "haptic glance" [45], which relies on the tactile sensing in order to perform a fast object recognition through non-exploratory manipulation.

Tactile sensors mimic the human cutaneous sensors found in the human fingertip area skin. They are in the first line of contact between the hand and the object allowing to gather action-related information about the touched object such as the normal and shear contact forces, slip, as well as hardness, friction, texture, temperature [46], [47].

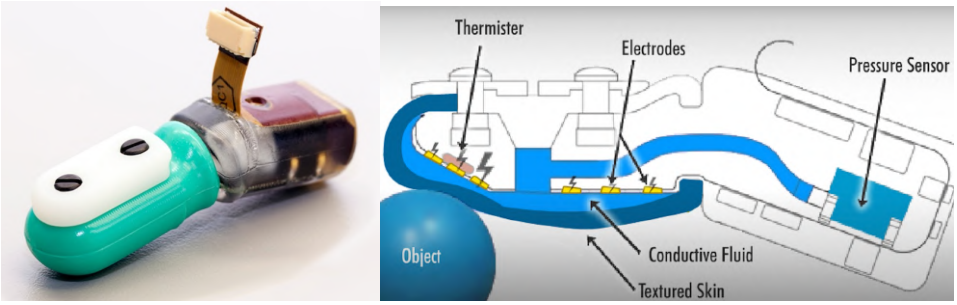


(1) *(2)*

Figure 2.10 (1) FSR array, and (2) Tactile sensor based on FSR array (from [48])

A popular tactile sensor technology uses a thick polymer film Force Sensing Resistor (FSR) array shown in *Figure* 2.10(1). It consists of a 16-by-16 matrix of FSR bodes spaced 1.58 mm apart on a 6.5 cm2 (1 square inch) area. The FSR nodes exhibit exponentially decreasing electrical resistance with applied normal force: the resistance changes by two orders of magnitude over a pressure range of 1 N/cm2 to 100 N/cm2. FSRs have a robust structure, small size, but relatively low accuracy. A custom-designed elastic *overlay* having a tab on top of each node of the *FSR* matrix as shown in *Figure* 2.10 (2) [48] allows the material to expand without any stress in the *x* and *y* directions making possible its compression in the *Z*-direction proportionally with the normal stress component. This tab configuration provides a *de facto* spatial sampling, which reduces the elastic overlay's blurring effect on the high 2D sampling resolution of the *FSR* transducer.

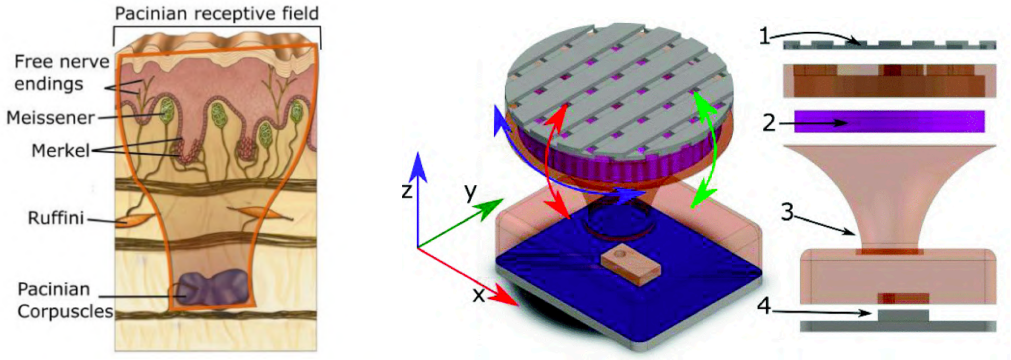
Other known pressure-sensitive tactile sensors use the *TeskanFlexiForce*™ pressure ink technology [49], or the *Quantum Tunnel Composites (QTC®)* technology [50].



(1) *(2)*

Figure 2.11 BioTac SP sensor from SynTouch. (from [52]).

Figure 2.11 shows the biology-inspired multimodal tactile-sensing *BioTac SP* (Single Phalanx) sensor [51]. The skin of the finger is compressed proportionally with the contact force, which in turn changes the thickness of a conductive fluid whose impedance is then measured by electrodes. When the fluid goes thin the impendence goes up. The sensor is also sensitive to vibrations that occur as the skin slide over the textured surface, vibrations that are conveyed through the fluid to a pressure transducer. The finger also has thermistors allowing it to measure variations in the temperature of the touched objects so it could be used to identify material properties of the objects [52].



(1) *(2)*

Figure 2.12 (1) Human skin mechanoreceptors (from [54]), and (2) Multi-modal Tactile Sensor module: 1 - tactile array; 2 - inertial measurement unit (IMU); 3 - cone compliant structure, and 4 - deep barometer (from [53]).

Figure 2.12 shows a biology-inspired multi-modal tactile sensor module recently developed by T.E. Alves de Oliveira in our lab at the University of Ottawa [53]. Its structure is inspired by the human skin mechanoreceptors [54]. As shown in *Figure* 2.12(2), the module consists of several layers: a 32-taxel tactile array, a MARG (magnetic, angular rate, and gravity), a passive compliant structure, and a barometer. The compliant structure consists of IMU and the barometer covered by polyurethane rubber. The bio-

inspired cone structure that conducts the forces to the deep pressure sensor helps to solve the tactile inverse problem [55].

2.4. Synergy

The robotic dexterous manipulation is an object-oriented act that requires not only specialized robotic hands with articulated fingers but also force, tactile, and kinesthetic sensors for the precise control of the forces and motions exerted on the manipulated object [45]. As fully autonomous robotic dexterous manipulation is impractical in changing and unstructured environments, a synergetic approach is to combine the low-level robot computer control with the higher-level perception and task planning abilities of a human operator [56].

Synergy comes from the Greek word ***συνεργία*** (synergia) that means "working together, joint work, working together, cooperation; assistance, help" according to [57].

Bernstein [58] introduced synergy as a worthy high-level technique based on a reduced number of parameters to control multi-DOF articulated hand structures. This approach is based on common patterns observed in the behavior of the human hand's muscles, joints, forces, actions, etc. Three types of synergy have been proposed to map the synergies from the human hands to the anthropomorphic robotic hands: **postural synergy**, **kinematic synergy**, and **dynamic synergy**.

Researchers have shown that a reduced set of representative postures known as *principal components* (PCs) could be used to describe the complete grasping process, defined as a **posture synergy** An early attempt was made by Napier in 1956 to describe

the two distinct movement patterns, “precision grip” and “power grip”, of the hand’s prehensile movements shown in *Figure* 2.13. It has been found that more than 80% of grasping operation could be described by only two principal components (PCs), PC1 for power grasping, and PC2 for precision grasping [59].

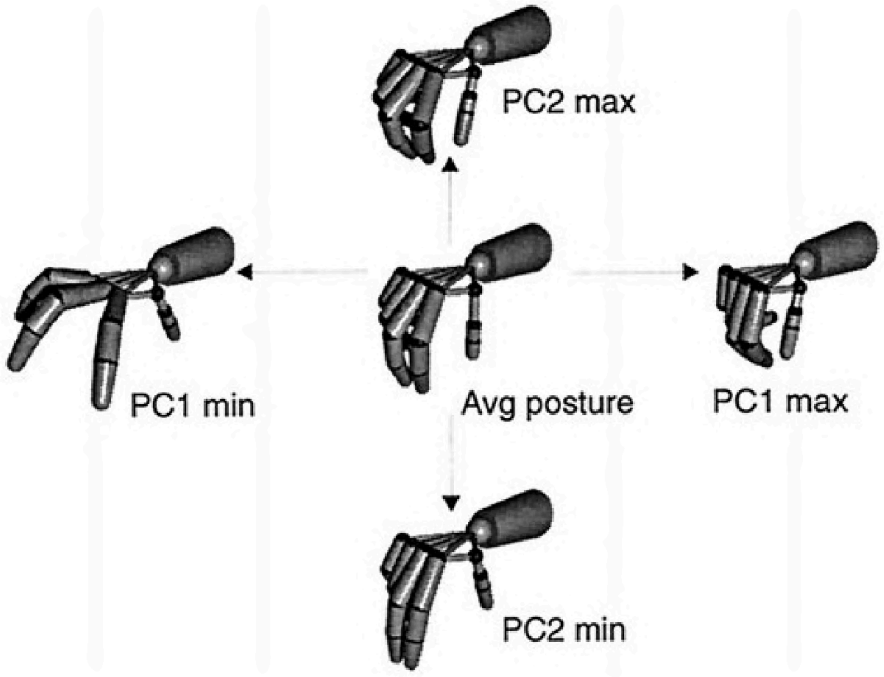


Figure 2.13 The principal components: PC1 for the power grasping and PC2 for precision grasping. (from [59]).

After Napier, other researchers such as Kamakura in the 1980s and Cutkosky & Howe in the 1990s used these two postural PCs to study which part of each of the hand’s fingers provided the force to hold the grasped object.

Kinematic synergies describe the stable correlations between the finger joint angles and their angular velocities [60]. when the hand performs multi-DOF operations such as multi-finger reach-and-grasp, or spatiotemporal coordination of the thumb and

index finger movements as shown in *Figure* 2.14 for a human hand [61].

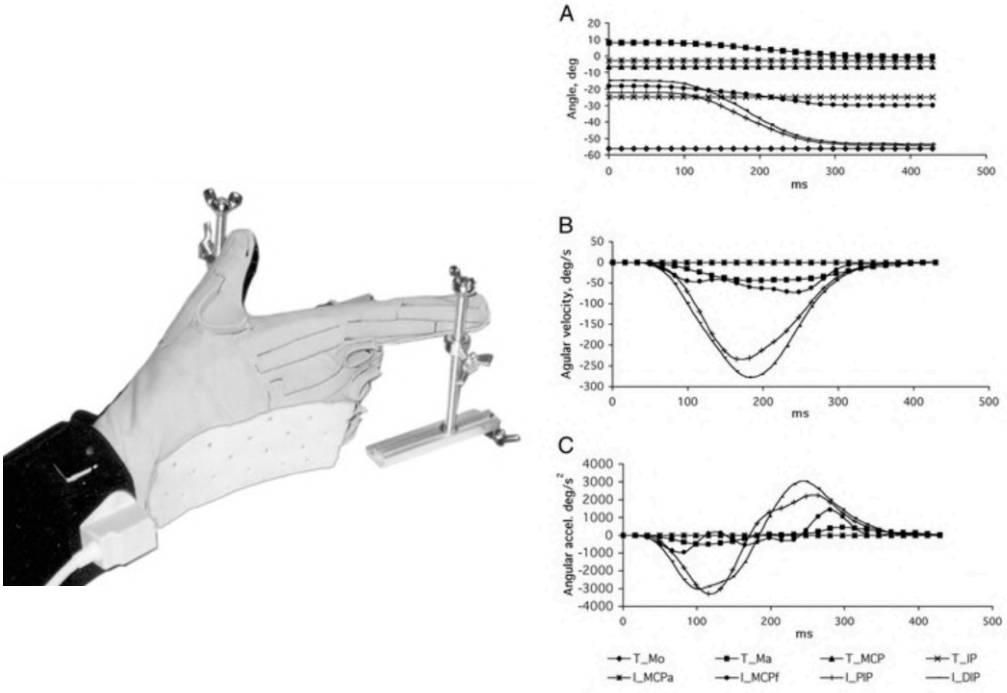


Figure 2.14 Human subject kinematics. A: thumb and index finger joint angles averaged over 20 trials of closing movements with normal velocity and normal aperture of subject S4(didn't show in the paper). B: corresponding angular velocities, and C: corresponding angular accelerations. (from [61]).

Dynamic synergies were defined as stable correlations between the hand's joint torques during precision grip movements [61]. It was found that dynamic synergies provide information over and above the information provided by the kinematic synergies. It was also found that PC1 and PC2 provide an adequate description of the dynamic synergies (as they did for the kinematic synergies).

It may be worth noting that in addition to the posture-, kinematic-, and dynamic-synergies, which are used in our thesis work , synergies were also proposed for the study of muscle activities [62].

To efficiently control an anthropomorphic multi-finger robotic hand using a reduced number of parameters similar with the synergistic actuation of the human hand we can map the synergies from the human hand to the robotic hand using one of the following techniques: ***joint-to-joint mapping***, ***Cartesian mapping***, and ***object-based mapping*** [63] shown in *Figure* 2.15.

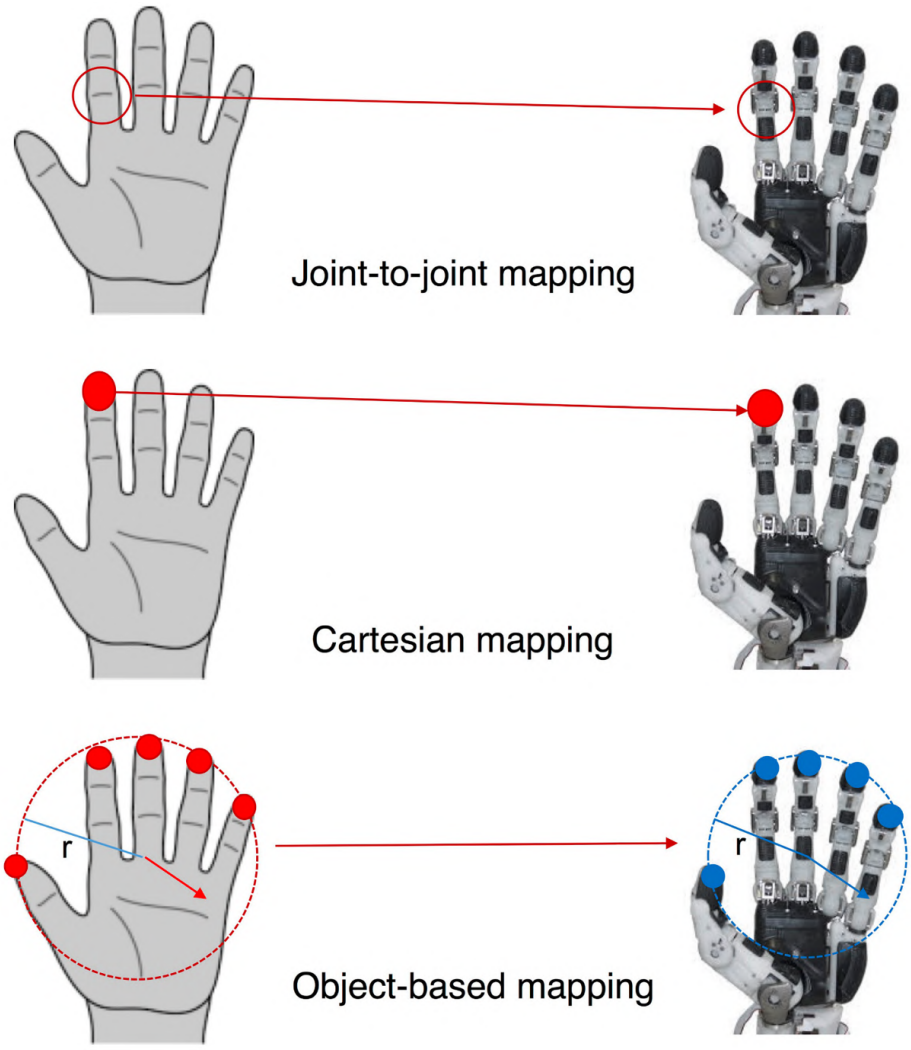


Figure 2.15 Schematic representation of the principal mapping techniques (from [63]).

2.5. *Fuzzy Logic Control*

The basic idea of a nature-inspired *Fuzzy Logic Control* (FLC) was introduced in 1972 by Prof. L.A. Zadeh [64].

The major benefit of the FLC is that the desired system behavior can be described with simple "if-then" relations based on very low-resolution models able to incorporate empirical engineering knowledge. Because of this FLCs have found many practical applications in the context of complex ill-defined processes that can be controlled by skilled human operators.

FLC can conveniently be used for robotic manipulation control applications allowing robotic hands to reach the objects and perform a stable grasp, balance the static posture and contact force, and perform in-hand manipulation [65], [66].

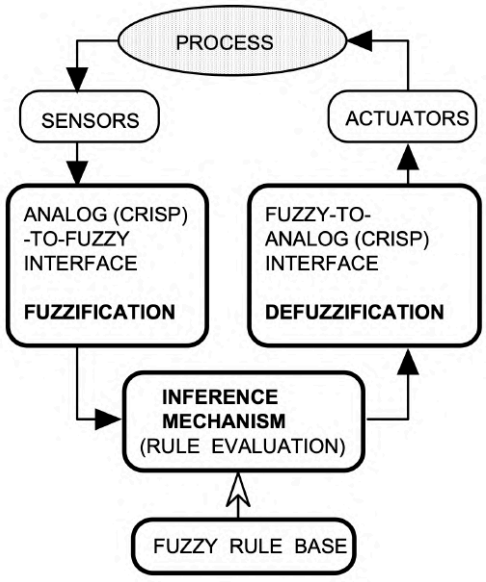


Figure 2.16 The structure of a fuzzy logic control system (from [67]).

Figure 2.16 illustrates the functional block diagram of a generic FLC system [67].

The *classic control* needs a detailed I/O function which maps high-resolution input variables to high-resolution output variables. Finding the mathematical expression for this detailed mapping relationship may be difficult, if not impossible, in many applications. On the other hand, the *fuzzy logic control* is based on an I/O function that maps very low-resolution input variables to very low-resolution output variables. As there usually, only 7 or 9 fuzzy quantization intervals are covering the input and output domains the I/O mapping relationship can be very easily expressed using an "if-then" inference mechanism.

2.6. Machine learning – Classifiers

The five data-driven classifiers [68] which will be used in Chapter 5 to organize each of the objects of interest into a class based on its specific attributes are: Support Vector Regression (SVR), Multilayer Perceptron (MLP), Random Forest, Extra Trees, and K-Nearest Neighbors (k-NN).

2.6.1. Support Vector Regression (SVR) Classifier

The SVR is the application in regression analysis using Support Vector Machine (SVM) [69], which is practical in texture classification [70], image classification [71], and text classification [72].

The SVMs construct hyperplanes or hyperplane sets in high-dimensional or infinite-dimensional spaces. The farther classification boundary from the nearest training

data point is better because it reduces the generalization error of the classifier.

2.6.2. Multilayer Perceptron (MLP) Classifier

Multilayer Perceptron is a typical feedforward neural network for approximate problems [73], which has at least one hidden layer with multiple neurons (nodes) connected (edges), configured by the transfer function, and connecting weight.

MLP is a supervised learning algorithm, which does a non-linear mapping of the input dataset and the output.

The perceptron is a linear classifier, its output y being a function of the inputs X_i by forming a linear combination with input weights W_i , bias b, and a non-linear activation function φ:

$$y = \varphi\left(\sum_{i}^{n} W_i x_i + b\right)$$

The MLP, illustrated in Figure 2.17, is a composed of layers of perception from the input layer set X_i to the output calculated as [74]:

$$f(x) = W_2 g\left(W_i{}^t X_i + b_1\right) + b_2$$

where W_2 are weights of the hidden layer, b_1, b_2 are the bias, and g is the activation function defined as $g(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$.

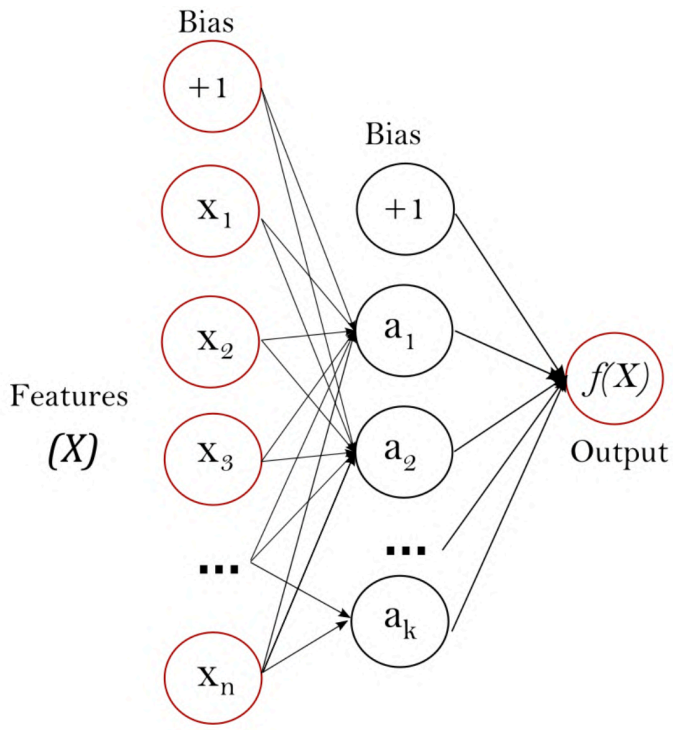


Figure 2.17 Multilayer Perceptron Neural Network with one hidden layer [74].

2.6.3. Random Forest Classifier

Random Forest is an algorithm that consists of a large number of individual decision trees that operate as an ensemble. Each individual tree in the random forest makes a class prediction as shown in *Figure* 2.18. The class with the most votes would become our final prediction. The deeper each tree is, the more complex the decision rules are and the fitter the model is [75].

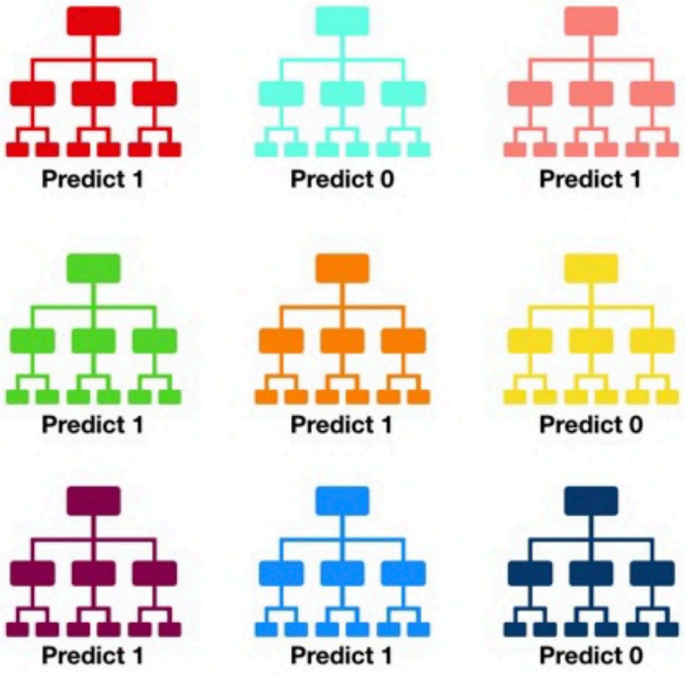


Figure 2.18 Simple Random Forest example

2.6.4. Extra Tree Classifier

Extra Trees classifier, also known as Extremely Randomized Trees, can often achieve as-good as or even better performance than the Random Forest algorithm. The difference lies in the fact that while the Random Forest method computes the best split combination, the Extra Trees technique selects a random value for the split and does not replace the samples. While the randomness of the Random Forest comes from bootstrapping the samples, the Extra Trees builds its randomness in the splitting process [76].

2.6.5. K-Nearest Neighbors (K-NN) Classifier

The K-Nearest Neighbor (K-NN) classifier uses an instance-based, “lazy”, non-parametric learning algorithm for classification and regression[77].

The classification is determined by the "majority vote" of its neighbors. The most common *k* neighbors determine the category assigned to the object. In k-NN regression, the output is the attribute value calculated as the average of the values of its *k* nearest neighbors.

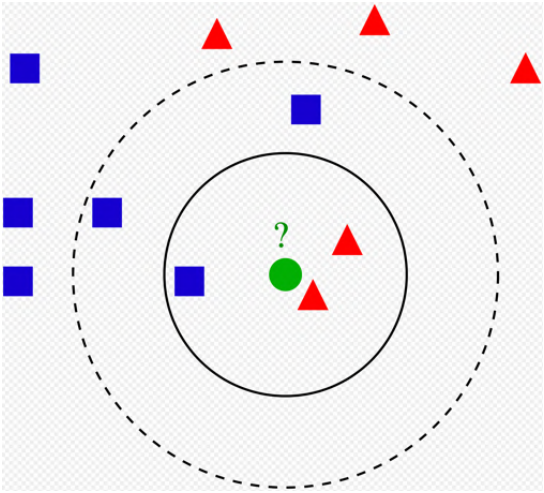


Figure 2.19 Simple K-NN classification example. (from [78])

An example of a K-NN classification is shown in *Figure* 2.19 [78]. If k = 3 (the solid-line circle neighborhood) then the green dot is assigned to the *red triangle* majority class. If k = 5 (the dashed-line circle neighborhood) then the green dot is assigned to the *blue square* majority class.

Chapter 3. System Overview

This chapter presents the experimental system, which includes hardware and software subsystems, used to teach a dexterous robotic hand to perform grasping and dexterous operations. The teaching process requires five elements: a clear teaching topic, a good teacher, a robotic hand, and a test unit.

The two system architecture components are discussed in section 3.1 Hardware and section 3.2 Software. The hardware subsystem consists of four major components: the *CyberTouch* glove, the novel *Ring Ada* robotic hand which was upgraded in our lab, the *Leap Motion* depth camera, and a computer controller as the center node. The software subsystem consists of the Robot Operating System (ROS) framework, the software development kit for Leap Motion Controller, the custom-developed drivers for the *Ring Ada* hand and the *CyberTouch* data glove, and a C++ and Python software which we developed to provide the learning method for the grasping and teleoperation. Section 3.3 discusses the limitations of the system.

3.1. Hardware:

3.1.1 CyberTouch Glove

A good HCI should provide instinctive control and reliable task-related feedback. We choose the interface *CyberTouch* data glove [9] to record human operator hand's real-time bone-joint angles and to provide linear vibration feedback on both fingers and

palm, shown in *Figure* 3.1.

The *CyberTouch* glove has 18 embedded bendable, and flexible sensors. Pairs of these sensors are positioned between the joints of each finger: one sensor is positioned between the intermediate phalange and proximal phalange, and the other between the proximal phalange and metacarpal joint.

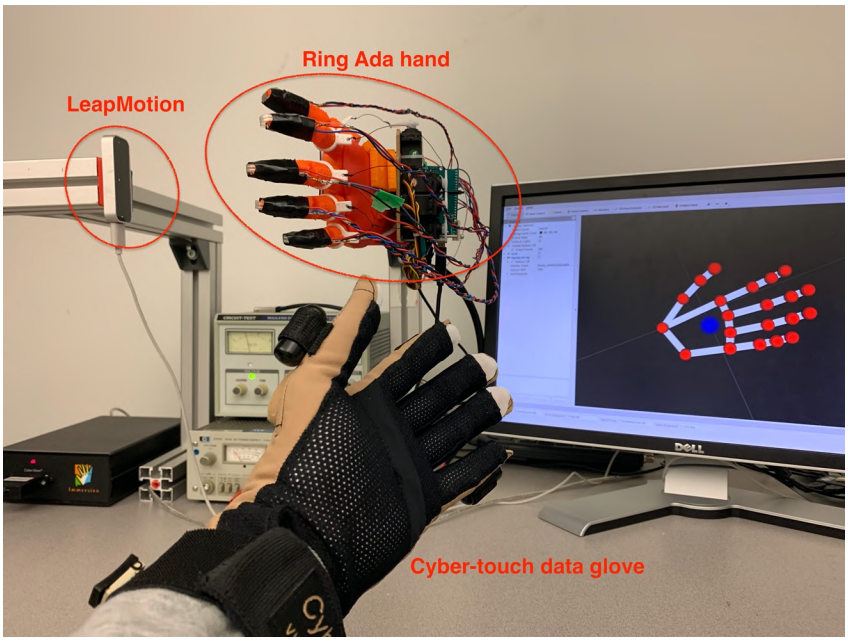
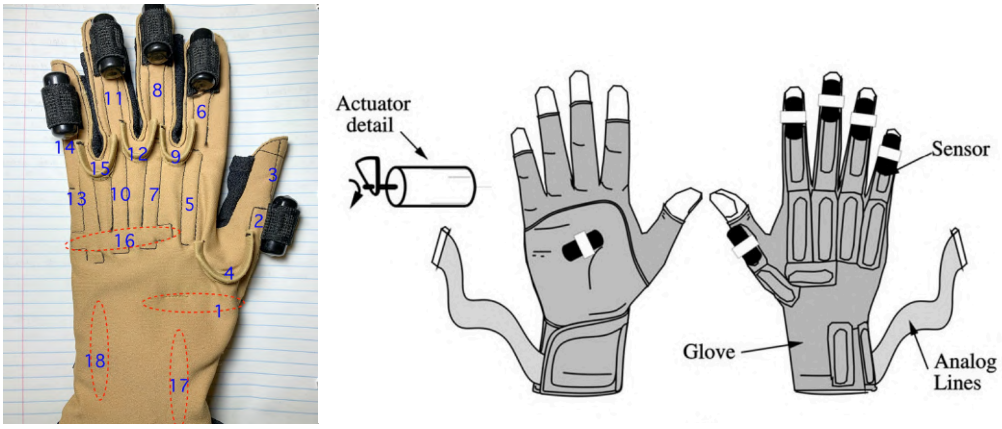


Figure 3.1 Main components of the experimental setup.

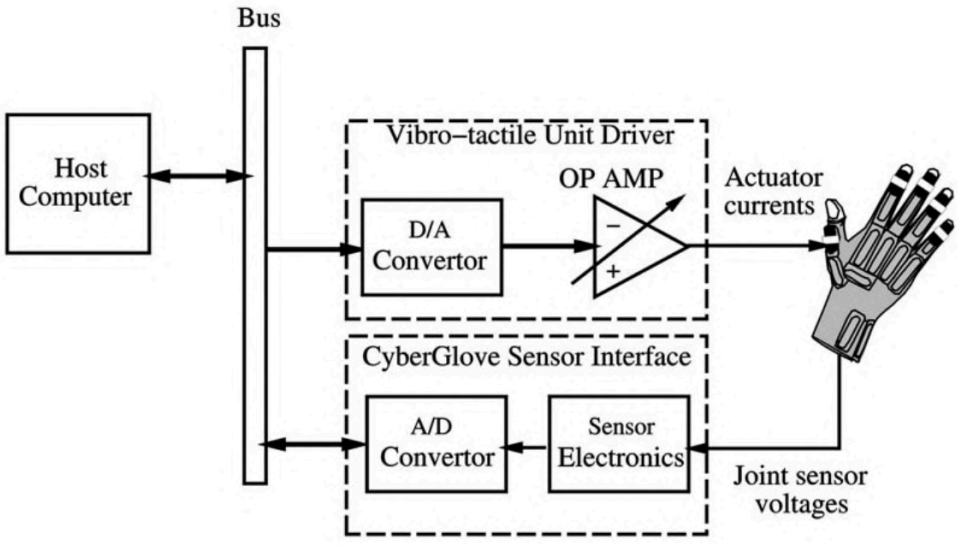
As shown in *Figure* 3.2 (1), there are 18 joint sensors and 6 actuators built in the *CyberTouch* glove, which are working together to provide the haptic feedback to the human operator's hand, [79]. The red circles in this picture are indicating the sensors inside of the glove structure. One vibrating actuator is placed on each fingertip of the glove and the 6th one is placed in the glove's palm.

As shown in *Figure* 3.2 (2), the data to and from the sensors and actuators are

transferred to the host computer using a Serial-to-USB protocol. In our case, the joint angle sensor data (going through the A/D converter) has an 8-bit format. The build-in vibrating actuators positioned at each tip of the glove and the center palm. These actuators provide linear intensity vibrations controlled by 8-bit digital signals coming from the host computer via the D/A converters.



(1)



(2)

Figure 3.2 The CyberTouch glove and the control interface (from [79]).

3.1.2 Ring Ada Hand

The *Ring Ada* robotic hand that we developed is an upgraded hand design, based on the open-source *Ada* hand platform [10]. The hand, shown in *Figure* 3.1, has a 9 DOFs dexterous ability and uses tactile sensing for grasp control. We aim to combine the controlling-synergy and haptic-feedback to achieve precise stable grasping using the data extracted from the teleoperator's haptic control system.

Open-source Ada Hand

The open-source *Ada* robotic hand shown in *Figure* 3.3 [10] is a tendon-driven anthropomorphic 3D printed elastic robotic hand provided by *Open Bionics*, derived from the Open Hand project [80].

It is an underactuated hand that has 5 DOFs (*Figure* 3.3). Based on the performance of its linear motors, it can achieve relatively fast closing of each finger and make some basic grasping movements. Because of its human-like shape, it has already been used as a prosthetic hand in many medical studies. As all its components are compact and located in the printed palm, it is easier to mount on a robotic arm and also quite convenient to modify.

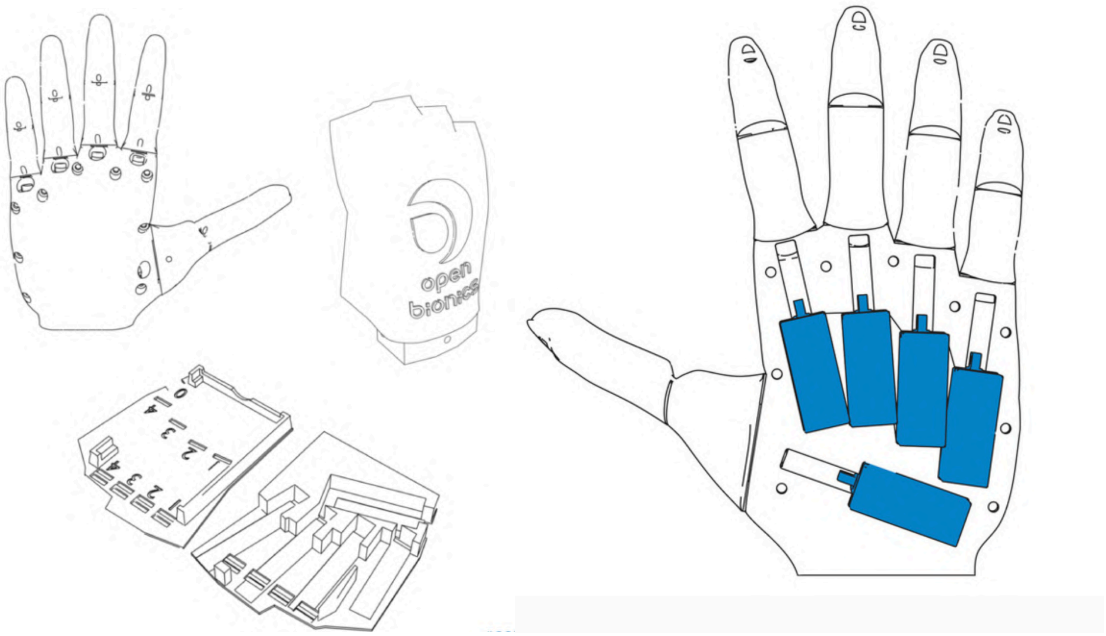


Figure 3.3 Ada hand structure and its linear motor layout inside of palm (from [10]).

In order to assemble our Ada hand platform, we 3D printed the back cover, a PCB tray, the upper and lower of the hand with hard PLA material. The palm was 3D printed with Ninja-Flex elastic material which provides good resistance and ability to deform at the same time.

Each finger is powered by a PQ12 linear actuator placed in the palm as shown in *Figure* 3.3. The actuator linear motor operates with a 12V input, has internal position feedback, and has a usable stroke length of nearly 18 mm. Each linear motor moves the finger through a tight fiber tendon linked to the hole positioned at the backside of the distal phalange as shown in *Figure* 3.3, which makes the hand able to completely close the finger.

We use the PCB controller board developed in our BioIn Robotics lab, shown in

Figure 3.4. This controller is driven by an Arduino compatible *Teensy* microcontroller in order to save space without compromising the performance. The microcontroller acts as a ROS node that operates the linear motors after receiving commands from ROS '/position' topic and publishes the '/feedback' topic to reflect the real-time position at the same time. More details on this will be covered in Chapter 3.2.

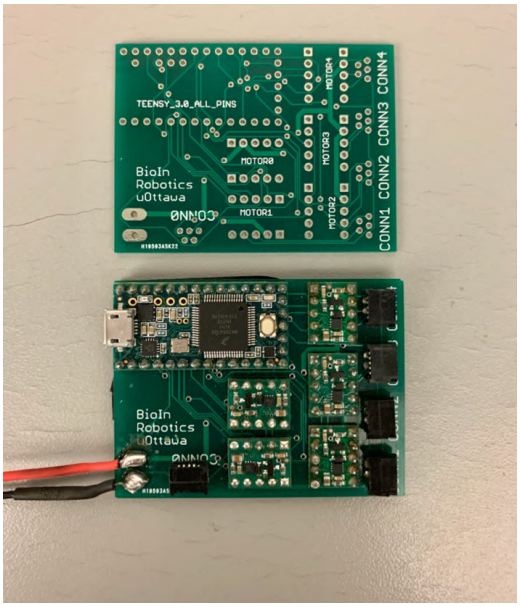


Figure 3.4 Custom designed PCB controller.

Upgraded hand designs

All dextrous robotic hand designs have a common feature: the proximal phalange is controlled separately through either joint motors or by tendons.

The design of the Ada robotic hand used as the basic platform for our experiments has the ability to be easily modified in order to perform better grasping operations. We augmented *Ada* hand's dexterity by adding another DOF to each finger through a simple 'Ring' shown in *Figure* 3.5-left. The Rings were 3D printed by rigid

PLA material. Each ring is attached at the middle of the proximal phalange for each finger, as shown in *Figure* 3.5-right.

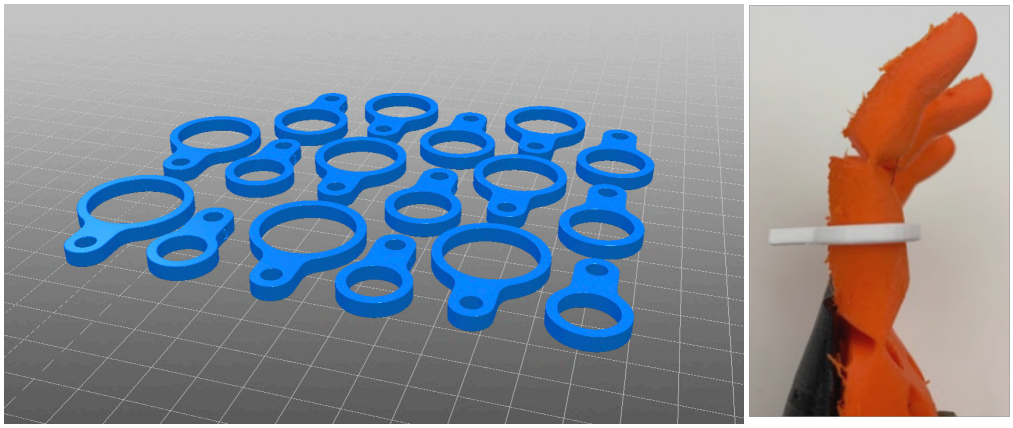


Figure 3.5 'Ring' models (left side), and how the ring is attached to the finger (right side).

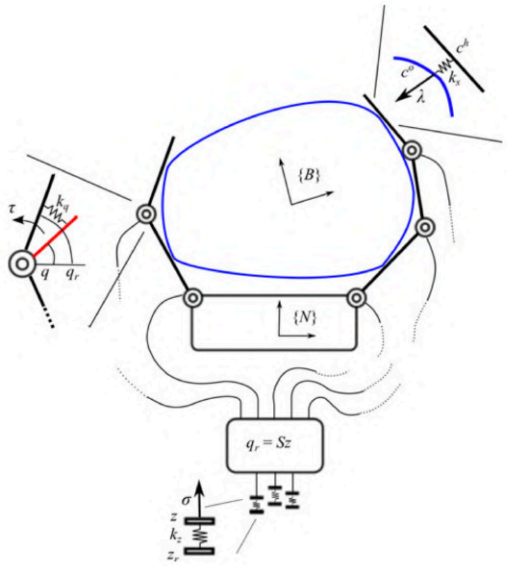


Figure 3.6 The design of the Motion and Force Controllable precision grasper. (from [81]).

The 'Ring' function was inspired by [81] which presented a multi-tendon driven two-finger grasper, shown as *Figure* 3.6. The idea was to restrict the closing joint's angle in order to increase the fingertip torque for a better grasping ability. This 'restrict' ability

has been adopted as an important feature throughout all experiments of the *Ring Ada* robotic hand.

The small hole at the top of the 'Ring' is designed to link to a servo motor through a relatively firm tendon, which provides restriction for the proximal phalange. This adds a natural claw-like behaviour which is seen as a prototype in *Figure* 3.8 (2). 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. Reflecting on the role of this 'Ring', we chose to name the new robotic hand as the *Ring Ada* hand.

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

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

As long as g^p is controlled by the servo motor, the gap between the intermediate and proximal phalanges g^i follows the three situations shown on the right side of *Figure*.3.7, 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 the stiffness to the intermediate joint. It can be confidently assumed that the proximal joint's resistance is lower than the intermediate joint, and therefore always has higher priority to be pulled by the linear motor, which simplifies the control.

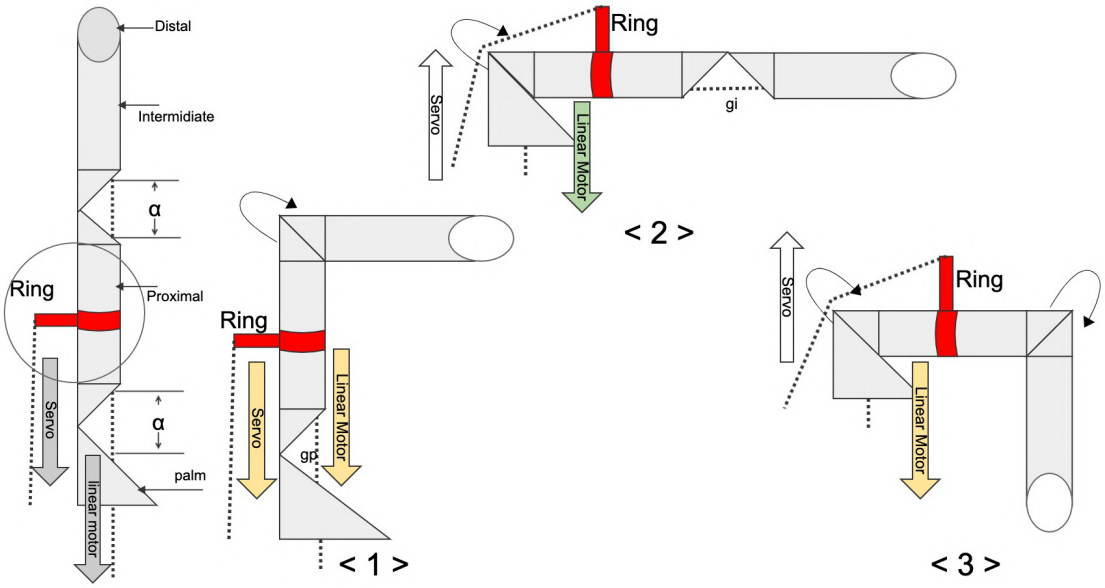
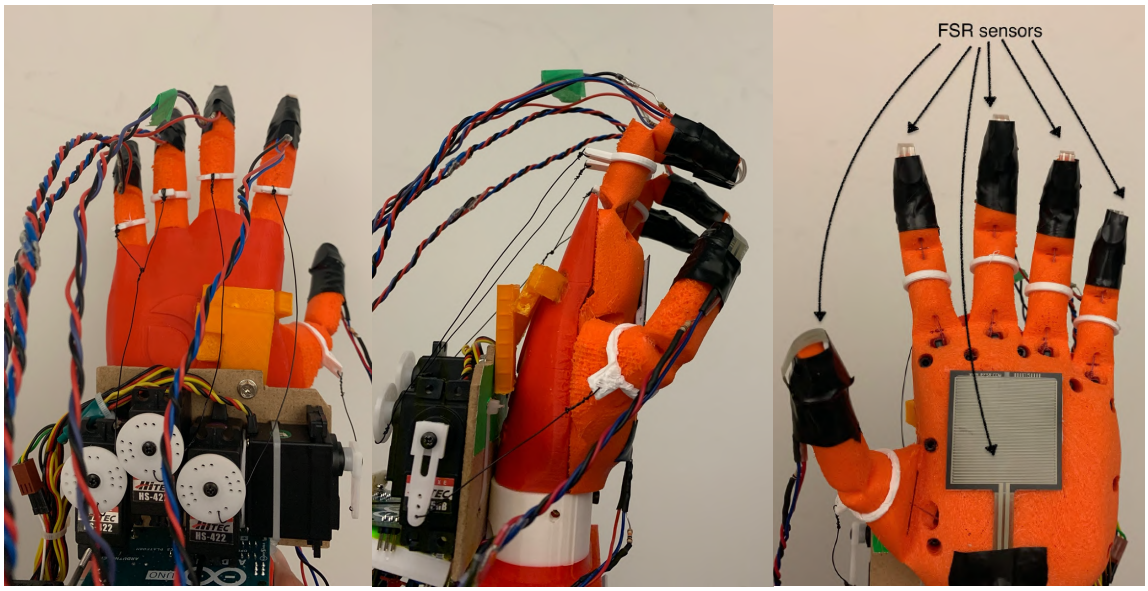


Figure 3.7 The "ring" which adds another DOF for the finger (left side). Typical finger movement (right side).

Figure 3.8 illustrates the experimental 9-DOF *Ring Ada* hand setup. Copper strings act as tendons linked to the servo motors. Due to the string's rigid character, adding some stability for the joint control is required in order to obtain a more stable movement during not only the pull movement but also the push. FSR sensors are attached to the distal phalanges and on the palm, as shown in *Figure* 3.8 (3), on positions corresponding to the positions of the actuators on the *CyberTouch* glove. This provides

haptic feedback information for the human operator equipped with the *CyberTouch* glove.



(2) *(3)*

Figure 3.8 (1) the back of the Ring Ada hand, (2) the side look of the Ring Ada with 'Claw' behavior achieved. (3) the palm of the Ring Ada with FSR pressure sensors on each fingertip and on the palm.

According to the manipulation taxonomy graph from [82] and the grasp taxonomy given in [83], there is not a special situation when the pinky and the ring finger are moving separately. Using one servo motor to control the proximal part for the ring and pinky together is acceptable. However, this may limit the hand's use for hand gesture applications such as sign language.

Role of the thumb

Based on the taxonomy of grasping [83] and manipulation classification [84], it

has been found that the thumb has a major role to play in the majority of hand movements that involve grasping. According to [85] the loss of the thumb corresponds to the loss of 40% of the hand performance. Based on these, it is quite reasonable to conclude that an evaluation of the reachability space of the thumb is essential when analyzing any new robotic hand design.

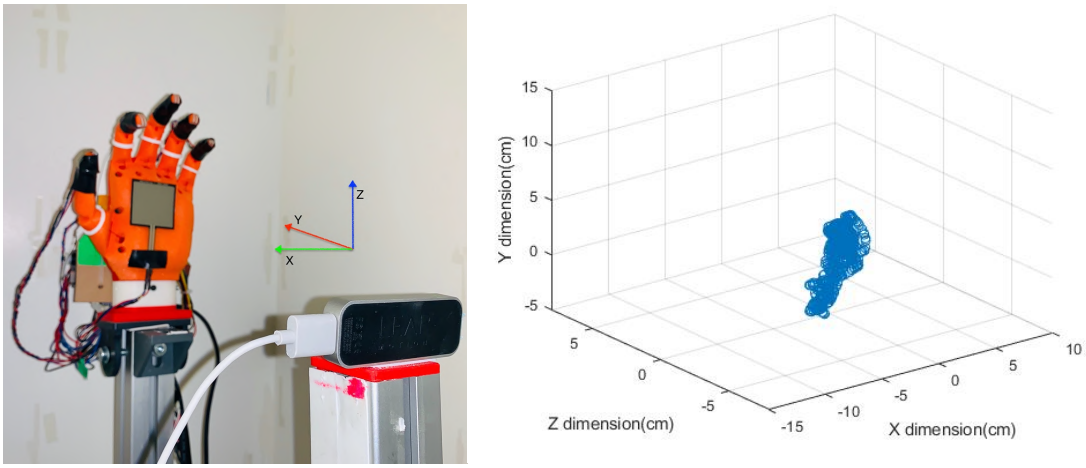


Figure 3.9 Leap Motion depth camera setup for measuring Ring Ada hand's thumb fingertip positions (left side), and the 3D diagram of Ring Ada's thumb reachability space (right side).

Figure 3.9 (left side) shows our experiment setup using a *Leap Motion 3D* camera to find the reachability space of *Ring Ada* hand's thumb. *Figure* 3.9 (right side) illustrates the 3D reachability space recovered from a 900 thumb fingertip position data set collected during the experiments.

The comparison with the famous iCub hand in the diagram consequence demonstrated As a comparison, *Figure* 3.10 (left side) shows the thumb reachability diagrams that we recovered while tracking the *Ring Ada* hand thumb, and *Figure* 3.10 (right side) shows the similar thumb reachability diagrams for the iCub hand used in Giuseppe's

research [31]. It is worth to mention that the iCub hand thumb diagrams were calculated using forward kinematics equations [86], while our *Ring Ada* data were obtained directly while experimentally tracking the thumb fingertip movements using the Leap Motion camera.

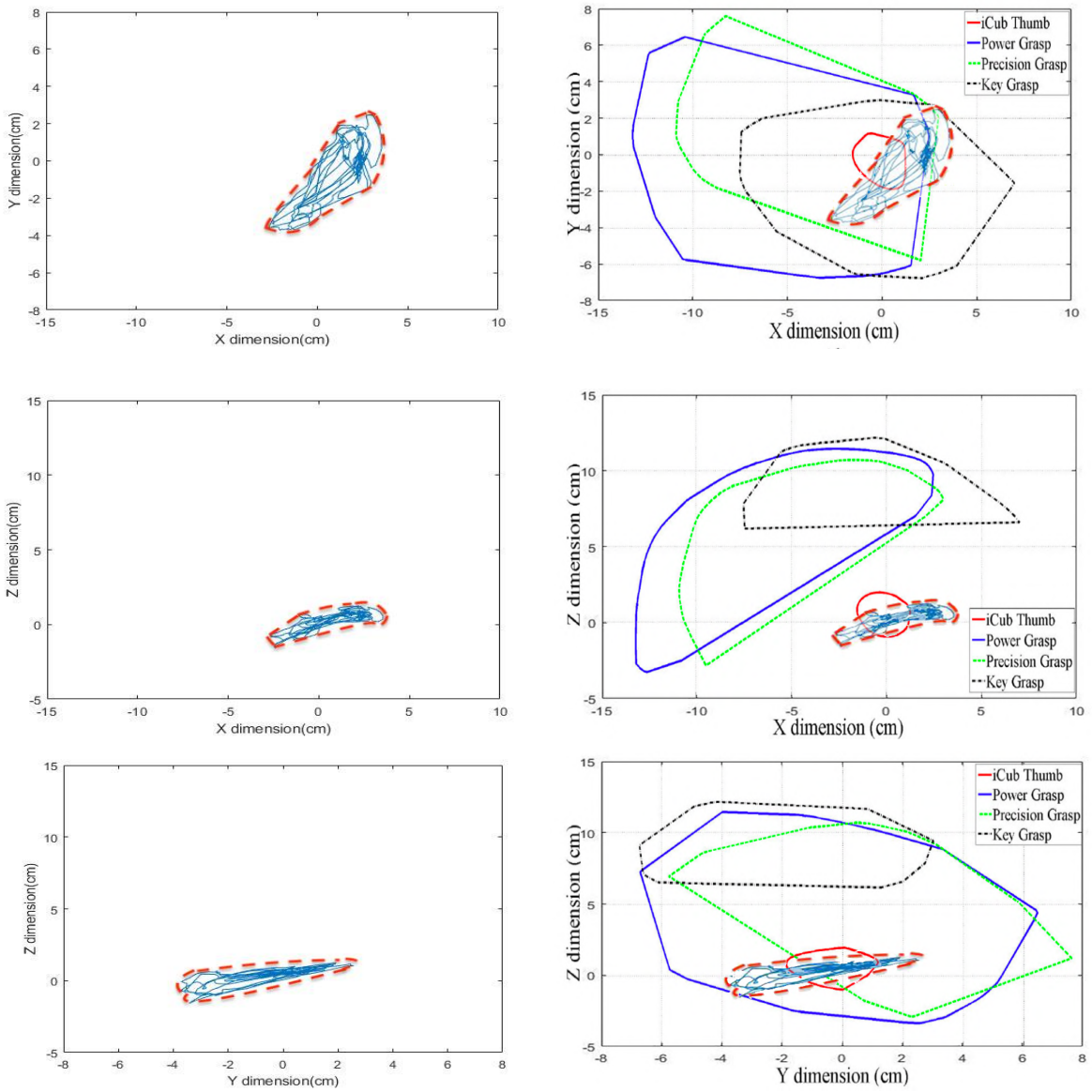


Figure 3.10 The Ring Ada robotic hand thumb reachability in each dimension (continuous blue line was generated by real-time data; red dash line was linked from the peak value point in each dimension) compare to the human thumb and the iCub hand (from [31]).

Based on this comparison, it may be concluded that from the thumb reachability point of view our novel *Ring Ada* hand is significantly better than the *iCub* hand.

3.2. Software

Figure 3.11 gives an overview diagram of the software system used in our research. There are three major components: the *CyberTouch*, the ROS nodes pool, and *Ring-Ada* robotic hand. All three systems communicate through USB serial connections: USB to RS232 serial protocol for *CyberTouch* interface [9], ROS-serial Arduino rules following the ROS standard [87] for the two microcontrollers used in the *Ring Ada* hand.

The continuous lines in *Figure* 3.11 are input data going to the next module, and the dashed lines are feedbacks from the modules. The ROS node in the microcontroller is used to *Figure* out the motor commands, which come from a synergy execution node in the system. The synergy node has been feed with a human operator's gesture information and *Leap Motion* data. And for the feedback, the haptic actuator is driven by two sources: the FSR sensor data and the position data from the linear motors.

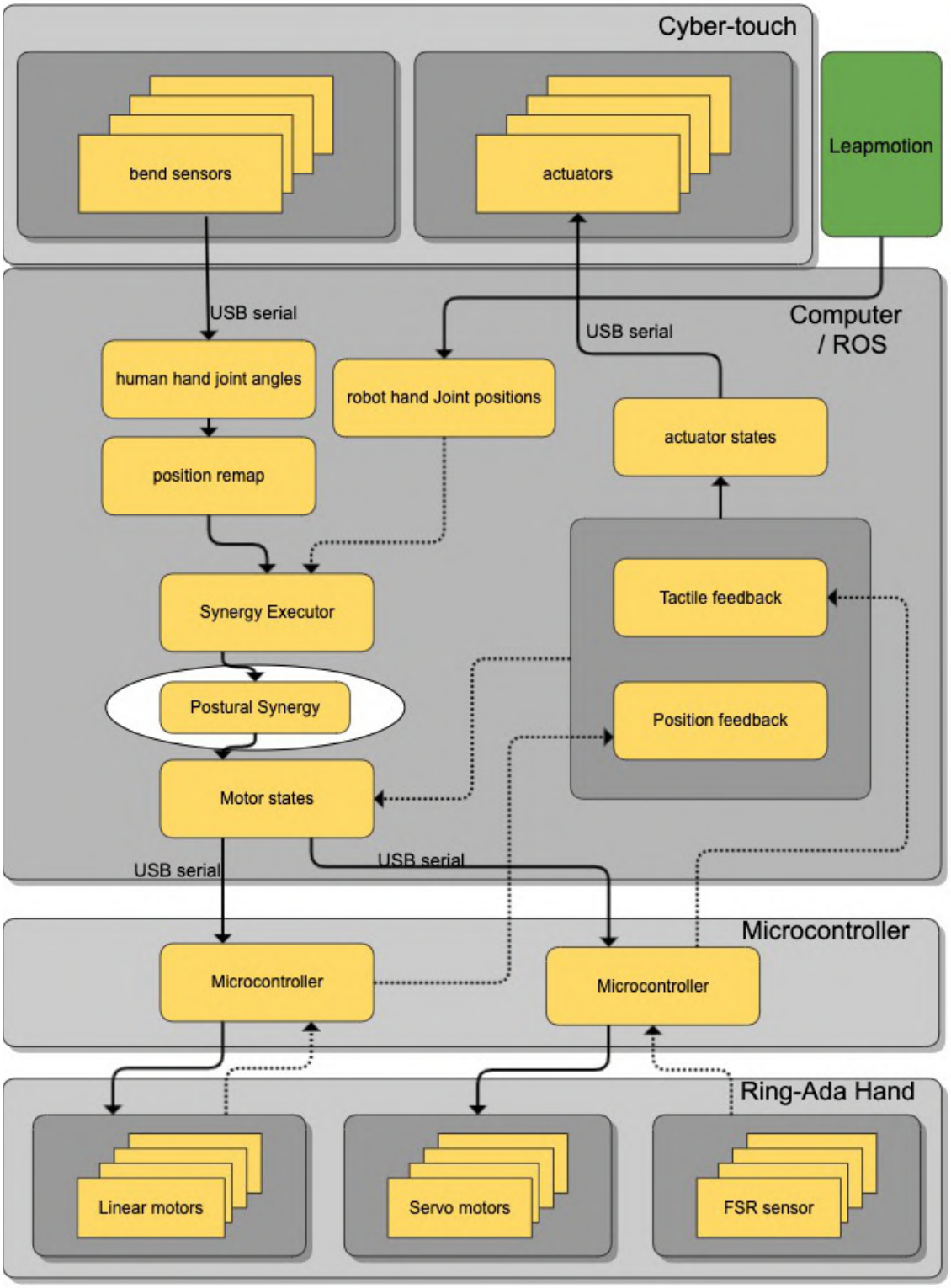


Figure 3.11 System overviews for the Ring Ada experimental setup.

3.2.1. Robot Operating System (ROS)

The open-source Robot Operating System (ROS) was designed for robot software development and inter-robot's communication [88]. ROS runs as a meta-level operating system that runs on a Linux operating system core. This system includes abstract hardware descriptions, low-level driver management, common function execution, inter-program messaging, and program distribution package management. The most important aspect is that ROS also provides great compatibility with large-scale software, such as an emerging device SDKs, new algorithms for the learning, controlling, and analysis of all kinds of robotic research. The user-friendly communication structure it provides allows implementing a P2P network between modules, including Remote Procedure Call (RPC) communication and Topic-Based asynchronous data streaming and storage on a parameter server.

Packages are the basic components for organizing the software in the ROS system. A package might contain ROS ***nodes***, ROS-independent libraries, datasets, configuration files, third-party software, or anything else that logically constitutes a useful module [89]. The package is designed to provide easy management for reusable software and keep system use minimal at the same time.

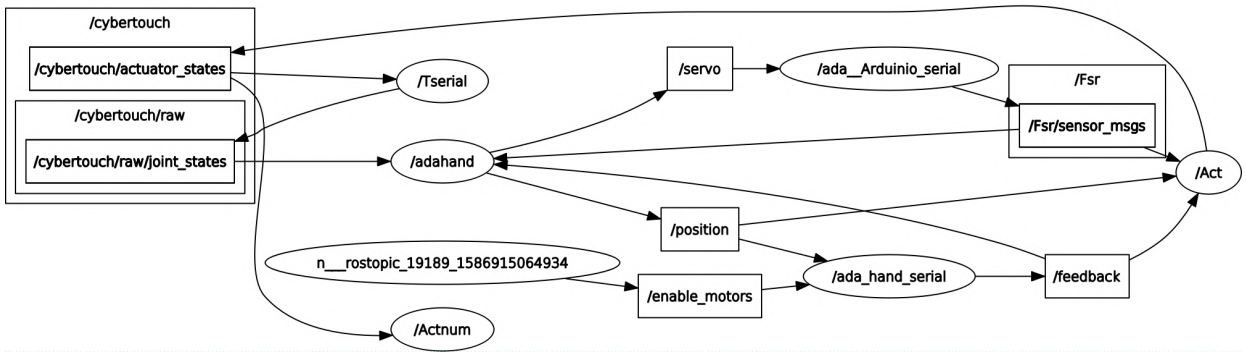


Figure 3.12 System data flow diagram.

The ellipse shapes in *Figure* 3.10 show the active ROS nodes during the interacting *CyberTouch* & *Ring Ada* grasping teleoperation.

The ***Node*** is the basic unit for data processing, a node communicates with other nodes through *subscribing* and *publishing* the ***Topics***, with data structured by ***Messages*** [90].

The named buses labeled for a node to exchange messages, *Topics,* have anonymous publish/subscribe semantics, which decouples the production of information from its users [91].

The real-time *Topics* list in this teleoperated grasping system is shown in *Figure* 3.13.

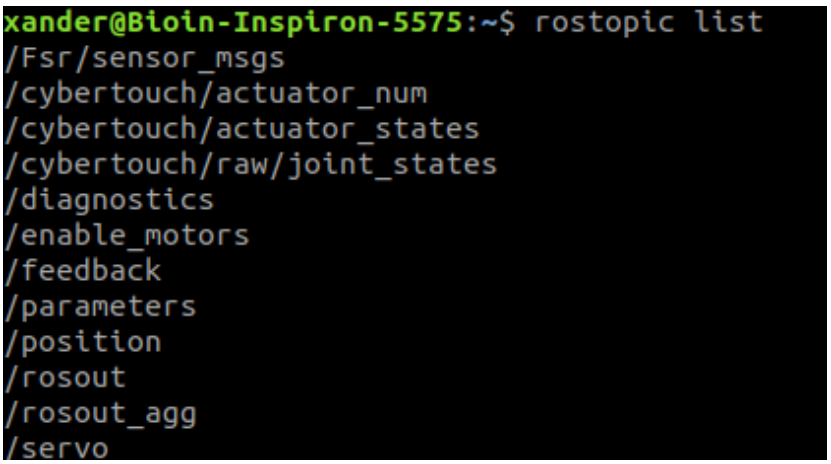


Figure 3.13 The real-time topic list when running the experimental setup.

The publish/subscribe is a very flexible communication paradigm, but its many-to-many one-way transport is not appropriate for RPC request/reply interactions, which are often required in a distributed system. Request/reply is done via a *Service*, which is defined by a pair of messages: one for the request and one for the reply [92].

The *CyberTouch* data glove uses an RS-232C 9-pin connector with a USB port converter for the serial connection to the computer. The converter is driven by a PL-2303 USB to serial bridge controller chip [93], shown in *Figure* 3.14. The two large on-chip buffers accommodate data flow from two different buses. The USB bulk-type data is adopted for maximum data transfer. An automatic handshake is supported by the serial port. A much higher baud rate can be achieved compared with the legacy UART FTDI controller. For instance, in our case, the *CyberTouch* glove serial output runs at 115,200 Hz.

Figure 3.14 RS232c converter linked to the CyberTouch glove.

The *Ring Ada* communication through USB cables, driven by the ROS serial package, is installed on the Arduino IDE. Both microcontrollers, the Teensy and Arduino board, used in our system act as a ROS node running independently. The data transferred into the ROS master are preprocessed inside the microcontrollers. The data flow is shown in *Figure* 3.12.

3.2.2. Leap Motion Controller SDK

The Leap Motion controller is connected via a USB 3.0 bus to the computer running the "Leap Service" software [94] which provides a library of functions that include the hand's structure, tracking data, and the interface settings.

The 'Leap Service' sends the tracking data to the foreground application by default or sends them to the background application by request. The background configuration settings are inherited from the foreground applications.

The SDK provides two APIs (Application Programming Interfaces), Native Application Interface (NAI), and a WebSocket Interface. In our system, the foreground application continuously receiving hand movement tracking information from the device through only NAI. The main classes used in the experiment provided by NAI consists of a controller, Listener, Frame, Hand, Finger, and Bone.

3.3. Limitations

Although the *CyberTouch* glove's flex-sensor readings have an under 1-degree resolution, the overall repeatability of the system has only a 3-degree accuracy. This is due to the different lengths, width, shapes even habits of operators' hands while using the glove. To deal with this problem, the glove is calibrated and configured in real-time for the flex sensor data. The *CyberTouch* data extractor node shown in *Figure* 3.12 detects the maximum angle and minimum angle in real-time and remaps them as 0 ~ 255 joint angle range, which is then normalized 0.0 ~ 1.0 into a standard communication format.

The actuators on the *CyberTouch* glove have some issues as well. As shown in *Figure* 3.15 the actuator mounted on the palm's center can lose touch with the human operator's hand in some cases. In our case the actuator only acts as feedback, which is driven by the robotic hand's tactile, there is not any problem as long as a contact happens while grasping an object.

The thickness of the actuator shell is similar to that of the finger itself, which may affect the dexterity of the gloved operator, especially when power grasping some objects.

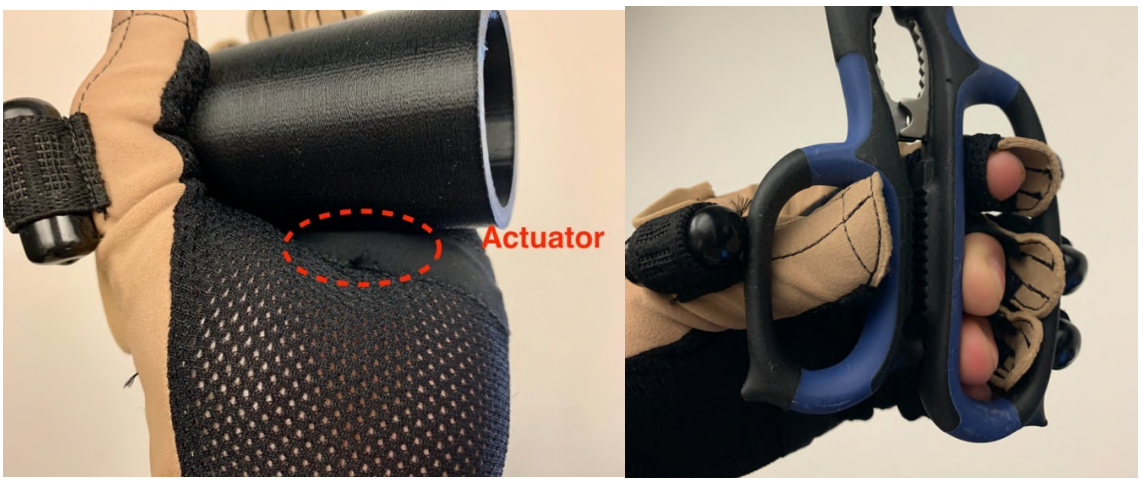


Figure 3.15 Grasping limitations of the CyberTouch glove.

The dexterity of the pinky and the ring finger could be increase by abandoning the 'ring and pinky linkage' so that the *Ring Ada* hand could also be used for American Sign Language (ASL) applications [95], [96]. Such a trade-off for a compact design is viable.

The *Ring Ada* hand has quite a few inaccurate elements: the soft joints, the stretchable tendons, the tolerance of the linear and servo motors, and the attaching point shift of the rings on the robotic finger to name some. It has been noted that the soft joints of the *Ring Ada* hand did bend far too much during grasping, to the point that sometimes the same input commands of the motors would end up with significant differences in final finger positions. The flaw is too big to be just ignored, which highlights the importance of the tactile feedback from the FSR sensors. In the next chapter, the performance of the *Ring Ada* hand will be further discussed.

Chapter 4. Teleoperated Grasping

The teleoperated grasping of different objects using the *Ring Ada* robotic hand essentially is a closed-loop control process as shown in *Figure* 4.1. The human operator could adjust based on the feedback from the CyberTouch glove to 'close the loop'.

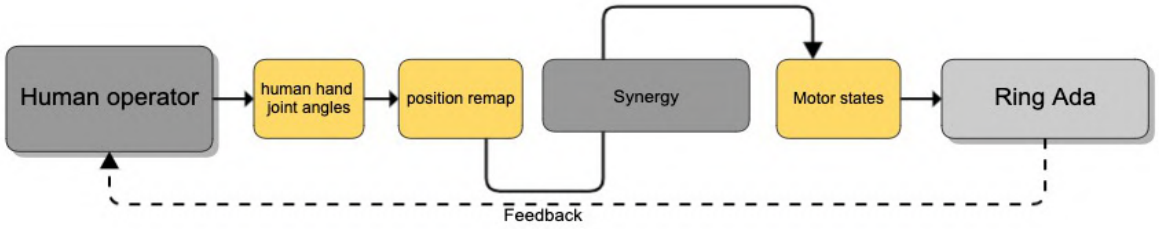


Figure 4.1 The teleoperation of the Ring Ada robotic hand.

As discussed in Chapter 3, the flex sensors of the '*CyberTouch* glove have instrumental errors and the *Ring Ada* hand has limitations of its finger movements. Due to its reduced thumb movement performance, the spatial coverage of the *Ring Ada* is also quite limited when compared with the teleoperator human hand. The precision of the teleoperation is also affected by the *Ring Ada* hand's linear motors' errors and t elasticity of the tendons. All these considerations make clear the need for the development of a suitable synergetic controller for the teleoperated grasping operations.

4.1. Synergies

As mentioned in the literature review section, there are three main types of synergy approaches: kinematic synergy, postural synergy, and dynamic synergy. In order to increase the capability of the hand and to keep the teleoperation control simple enough, we are using this synergetic approach for the control of the *Ring Ada* robotic hand.

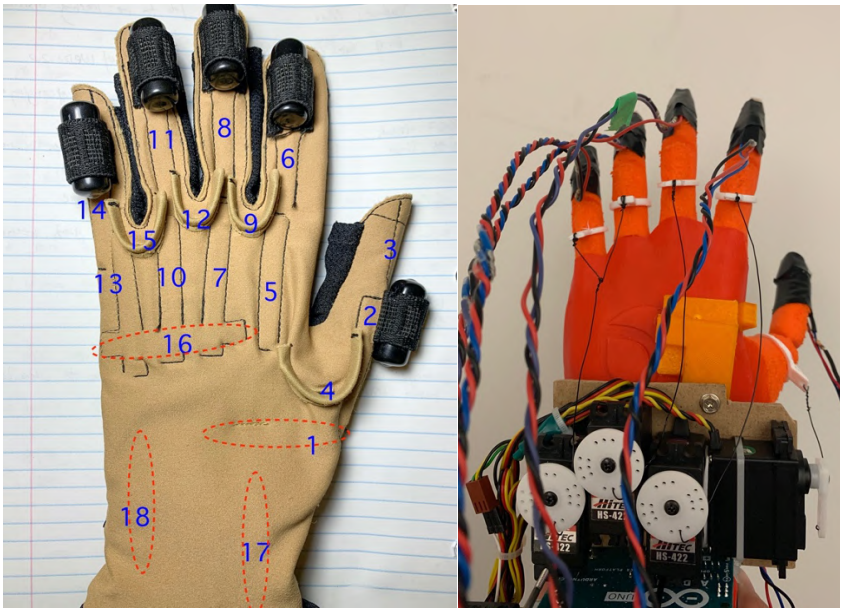


Figure 4.2 The back side view of the CyberTouch glove, and the Ring Ada robotic hand.

4.1.1. Kinematic Synergy

We use kinematic synergy to perform the joint-to-joint remapping of the human operator's movements to the servo motors pulling the *Ring Ada* hand proximal phalanges using the *CyberTouch* glove flex sensor information.

In the case of the thumb, the glove uses three sensors to detect the thumb's movement, namely the sensors 1, 2, and 3 shown in the left side image of *Figure* 4.2. The glove sensor 1 will contribute 75% and the sensor 2 will contribute 25%, working together for the first 65% of the hand's linear motor abduction, and sensor 3 will contribute the other 35% for the hand's thumb linear motor extension.

We know from Chapter 3 that if a servo motor actuates a proximal phalange that holds the gap g^P wide open, the linear motor would hit the limit. The maximum linear positions for the *Ring Ada* hand fingers are shown in Table 4.1.

Joint	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* $g^p = \alpha$.

In our real-time implementation, the teleoperator puts on the glove before booting the program in order to prevent overflow readout (the sensor would have some drastic deformation when wearing *CyberTouch* glove) and causes unprecise configuration. After starting the program, the operator performs three times opening and closing the hand gently in order to set the reference data of the sensor range. The remapping would dynamically calculate the proportion of flex sensor deforming according to Table 4.1 and sends a position command to linear and servo motors through the ROS topic.

4.1.2. Postural Synergy

The second synergy controller is the postural synergy which using two predominant postural synergies mapping fingers from 2 DOFs. Postural synergy is based on typical hand gestures. The idea is to define transformations (coverage) from one gesture to another, each transformation is considered as a *principal component* (PC). Combining different PC_s will then allow the robotic hand to achieve some level dexterous movement, an approach that simplifies the control design.

The first postural synergy research published in 1998 [59] uses two basic postures: PC_1 (the horizontal) and PC_2 (the vertical), which could adequately cover a multitude of practical applications. Due to our *Ring Ada* hand's limitations, combining two simple postures, shown in *Figure* 4.3, provide sufficient postural synergy for our teleoperation application.

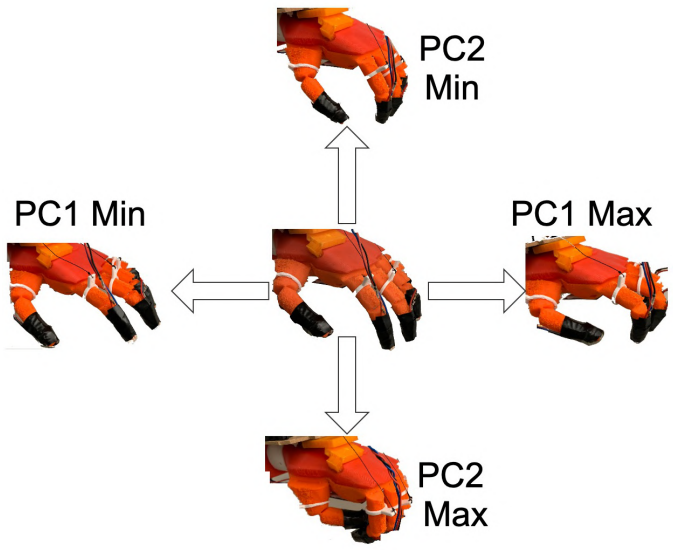


Figure 4.3 The minimum and the maximum value for the horizontal posture components PC1 and the vertical posture components PC2 for the Ring Ada hand.

We are also using the MATLAB synergy toolbox named SynGrasp [97] which provides: "Analysis of fully or underactuated robotic hands with compliance. Compliance can be modeled at contact points in the joints or in the actuation system, including transmission". While using a human hand shape with five fingers, it only actuates nine DOFs of the *Ring Ada* hand as shown in *Figure* 4.4.

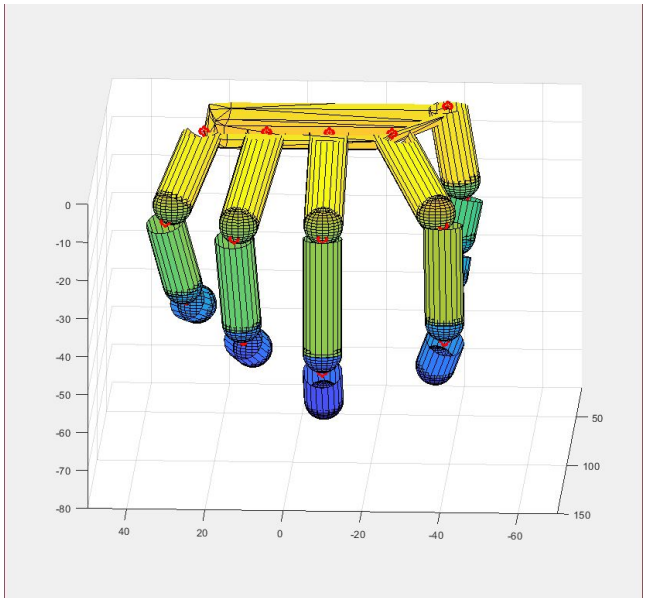


Figure 4.4 The robotic hand mode demonstration in SynGrasp [97].

Kinematic Pattern of the Two Predominant Synergies.

Figure 4.5 and *Figure* 4.6 show the synergies for the horizontal posture components PC1 and respectively the vertical posture components PC2 for the *Ring Ada* hand using the polar coordinates as suggested by Fanny's work [98], who followed the inspiration of the general motorize control synergy graph. The 'zero' point in these polar coordinate diagrams corresponds to the natural position of the Ada-hand. The two graphs representing the PC_1 *and* PC_2 vectors allow identifying the nine actuated joints whose

rotations are involved in each action.

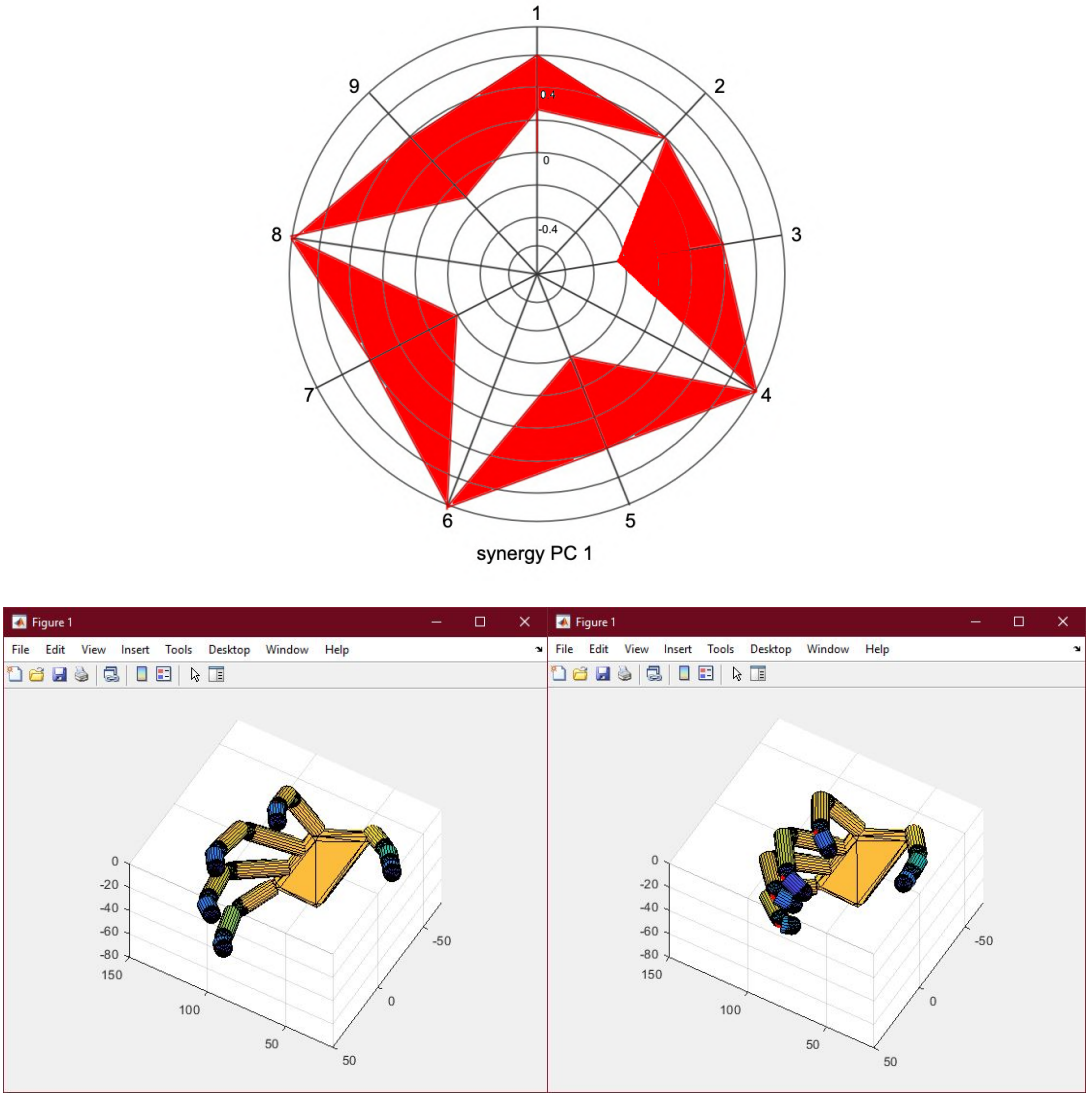


Figure 4.5 (up) The horizontal synergy PC1, 1,2 for the thumb; 3,4 for the index; 5,6 for the middle finger, 7,8 for the ring finger; also 8,9 for the pinky; (down) The minimum and the maximum PC1 present in simulation through SynGrasp.

As shown in *Figure* 4.5 for the horizontal posture (PC_1), the hand performs the gesture from 'flat' hand to 'claw' shape with the proximal phalange controlled by servo. As shown in the polar coordinates graph, the 2, 4, 6, 8 actuated joints stay far away from the natural point. The 'Claw' behavior moving the distal phalange, which dominates the

angle between distal and proximal, is the most important movement in the grasp phase, as discussed in Chapter 3. The simulation of the movements is shown in *Figure* 4.5 (down).

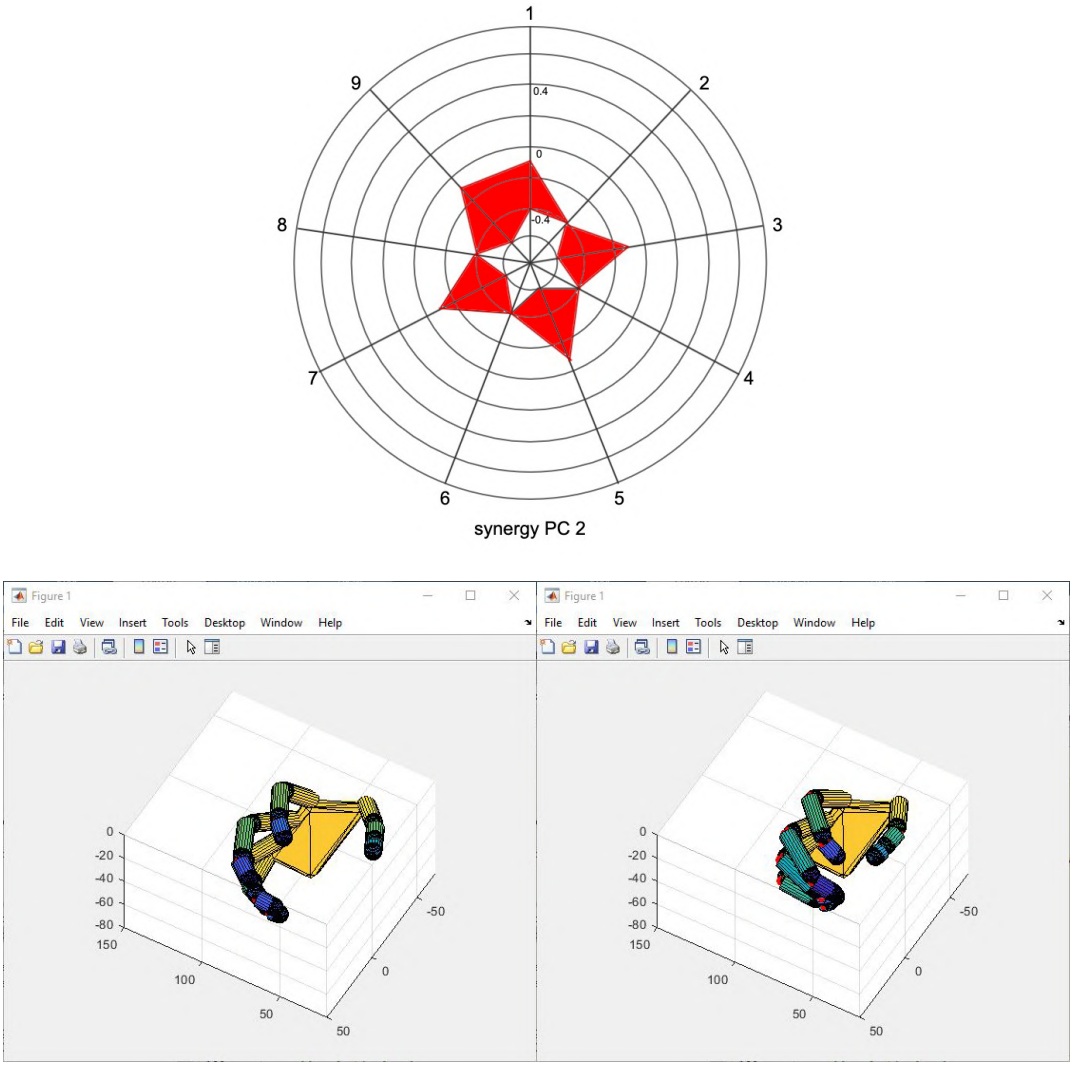


Figure 4.6 (up) The vertical synergy PC2, 1,2 for the thumb; 3,4 for the index; 5,6 for the middle finger, 7,8 for the ring finger; also 8,9 for the pinky; down) The minimum and the maximum PC2 present in simulation through SynGrasp.

Figure 4.6 shows the vertical posture (PC_2) transfer from the 'fist' gesture to a gesture that relaxes all fingertips which are pointing forward. In this synergy, we could see

how the fingers are 'closing' towards the palm from the natural point. The simulation of the movements is shown in *Figure* 4.6 (down).

Synergetic Control

The main idea of this synergetic approach is to manage to have the robotic hand grasp different object sizes inside of the palm. We can see the linear motors' adduction, the servo releasing towards the minimum, to free the closed hand.

	Thumb	***Index***	***Middle***	***Ring***	***pinky***
N_s	*140° (143°)*	*60° (58°)*	*60° (60°)*	*60° (70°)*	
N_l	***0.4(0.44)***	***0.6(0.59)***	***0.6(0.53)***	***0.6(0.63)***	***0.6(0.64)***

Table 4.2 The natural point of the Ring Ada hand.

In the beginning, a "zero" should be found in this synergy control, but there is no such a 'natural position'. A clever way to reach this was to extracting this state from the *CyberTouch* data glove while the operator relaxes the hand as shown in Table 4.2.

After the "zero"" has been found, we could normalize servo actuators by remapping on the kinematic pattern. The resulting relations between the servo motor parameters $V_s;\ V_l$ and the corresponding points on the pattern $P_s; P_l$ are:

for the *thumb*:

$$V_{sT} = 10° * P_s + 140° \ (-0.4 < P_s < 0.4)$$

$$V_{lT} = P_l + 0.4 \qquad (-0.4 < P_l < 0.6)$$

For other fingers:

$$V_s = 15° * P_s + 60° \ (-0.4 < P_s < 0.8)$$

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

The PC_1 and PC_2 pattern values are defined as P_{s1}, P_{l1}; P_{s2}, P_{l2}. The execution of the grasping is controlled by the weight ϵ between the two PCs, and *min* to *max* value during execution of two *PCs* is from 0 to 1. The resulting motor control voltages are calculated as:

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

where ϵ is the proportion between sensor 1;2 and sensor 3;4, as illustrated in *Figure* 4.7.

As could be seen the ϵ value controls the gesture of the *Ring Ada* hand. We use the *CyberTouch* glove to calculate the ϵ value and real-time *P* values.

Figure 4.7 Typical gestures illustrating the postural synergetic control.

4.2. Tactile Sensors and Sensory Feedback

Figure 4.8 shows the vibrating tactile sensory feedback actuator on the *Cyber-Touch* glove and the force sensor resistor (FSR) tactile sensors on the *Ring Ada* robotic hand. One round FSR is attached to the tip of each distal phalange of the robotic hand, and a large square FSR sensor is placed on the palm sensing if any object is in contact or not with the robotic hand.

Figure 4.8 (up) Vibrating feedback actuator on CyberTouch data glove. (down) FSR sensor 402, and FSR 406.

We are using *Interlink Electronics FSR™* 400 series [99] sensors set to measure pressure (Sensing range around 0.2N ~ 2N). The resulting FSR pressure readings are used to control the vibration levels of the feedback actuators on the data glove, following a simple linear correspondence.

As discussed in Chapter 3 in the software segment, the haptic actuator for the Cyber-glove driven by two sources: the tactile feedback and the position feedback. The major reason for using the position feedback is that the FSR is only working in the limited sensing area.

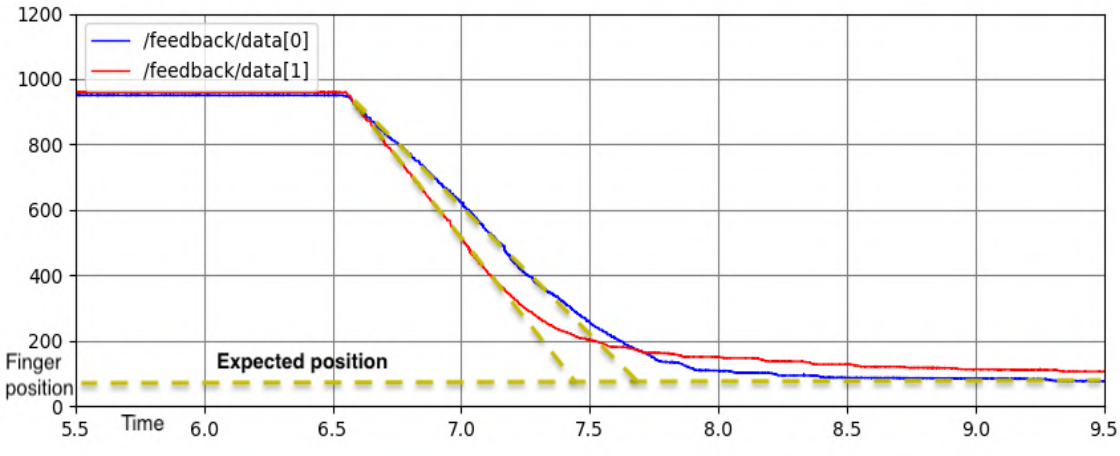


Figure 4.9 Feedback data of thumb and index collected during the example grasp operation.

The position feedback is calculated as the difference between the motor command l^ind the actual position $f^i$of the linear motor as shown in *Figure* 4.9 shows the expected position l^i starts sperate with the f^i, which the curve slope changed significantly when the finger was facing the obstacle. The actuator provides a stronger haptic-feedback v^l when the difference $\bar{d}$ become wider. In the end, the system reaches a stable grasp without any further movement. The v^l with $\bar{d}$ are following the linear positive correlation. Finally, the $v = \max(v^l, v^{FSR})$ becomes the final output command received by the actuator.

We have used a set of 3D printed rectangular and cylindrical objects for the experimental grasping test in our laboratory [100] shown in *Figure* 4.10. We used three object dimensions $50mm, 70mm, 90mm$.

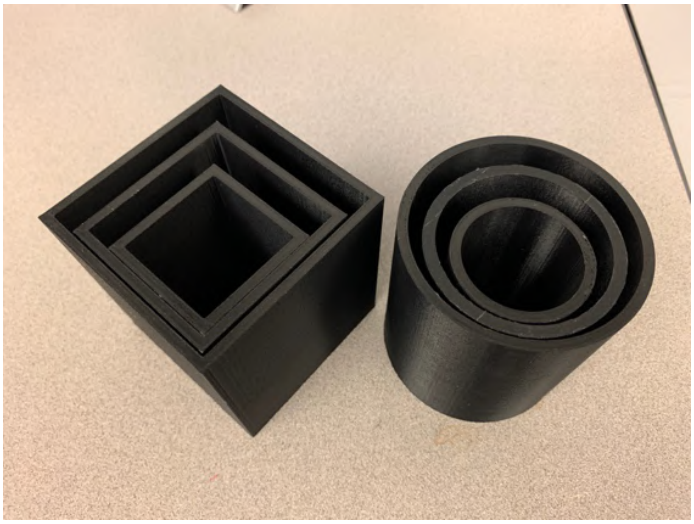


Figure 4.10 The objects used in experiments

Before starting the experiment, an object is positioned within the *Ring Ada* hand's reach, near the palm. The human operator then uses the teleoperated *CyberTouch* glove to close the hand slowly. Using the glove feedback, the operator could recognize which finger of the robotic hand is facing the obstacle, reflecting the potential stable grasp.

Experimental Results

Sensor data generated during grasping, extracted from ROS nodes, were used to monitor the real-time finger interactions.

One of the most important characteristics of the haptic feedback is the time lag, which can be due because (1) of the limited computing power, (2) the serial RS232c USB converter cannot handle high baud rate full-duplex channels, and (3), the data converters in the *CyberTouch* drive box are slow to react to serial data changes. As a result, nearly a 0.5-second delay would appear at every new feedback update from the *Ring*

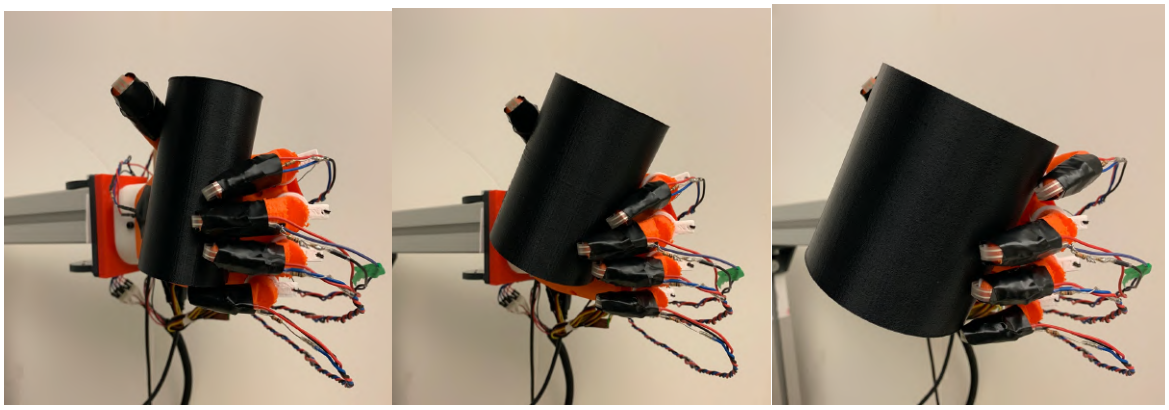
Ada robotic hand to the *CyberTouch* glove.

Figure 4.11 illustrates the grasping of cylindrical objects of three diameters. [100].

As shown by the experimental results depicted in *Figure* 4.13, chart (1), when *Ring Ada* hand is closing the reaction of each actuator is fairly equal and stable, all fingers trigger the same level except the thumb. Due to its small size, the $\phi 50mm$ cylinder could be handled through a power grasp which means that every finger's distal parts touch the object. However, the thumbs distal could not fully close, that is the reason for the red lines not present in this chart.

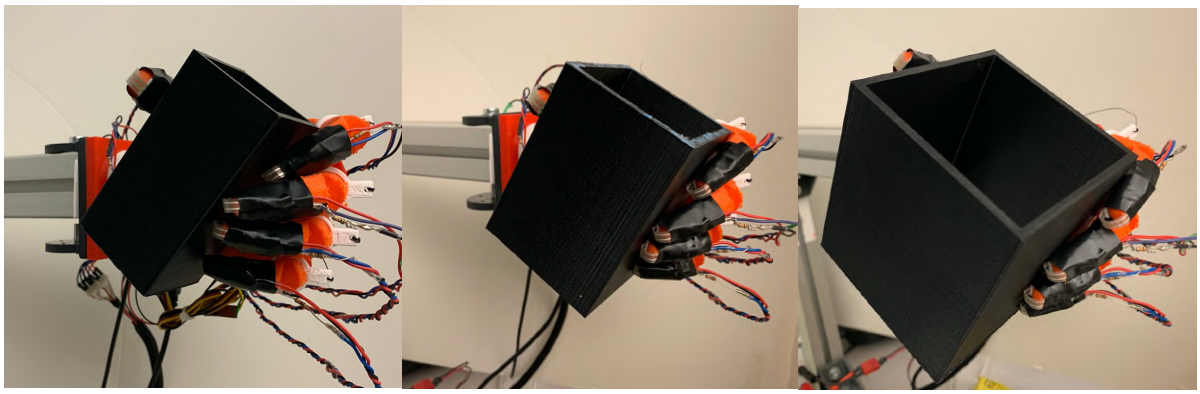
While grasping the middle size $\phi 70mm$ cylinder the hand starts to work harder. As illustrated by the chart (2) of *Figure* 4.13, the thumb reaches a relatively stable situation after holding the object. Following the thumb, the middle and the ring fingers touched then the palm. The applied pressure endures a stable grasp. As the hand's joints use a Ninja-Flex soft material, the torque limitation of the linear motors is apparent in some trembles in the chart during the manipulation.

Grasping the biggest diameter cylinder was much more difficult for our robotic hand. As shown in the chart (3) in *Figure* 4.13, only the thumb and ring finger feedbacks have been received. The robotic hand could hold the cylinder only if touching it gently.



[1] *[2]* *[3]*

Figure 4.11 Stable grasping of cylindrical objects having different diameters.



[1] *[2]* *[3]*

Figure 4.12 Stable grasping of rectangular objects of different sizes.

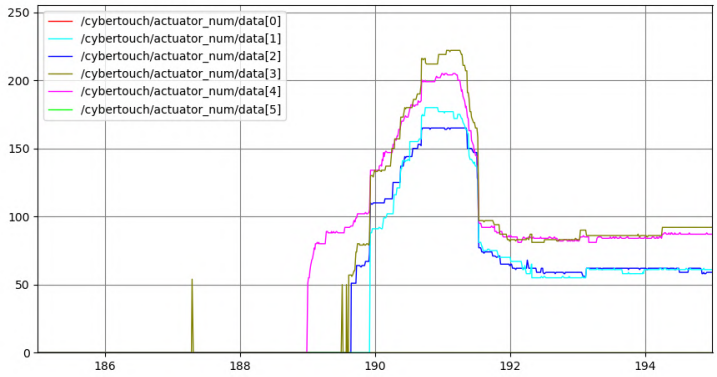
Figure 4.12 illustrates the grasping of rectangular objects of three sizes. [100].

When compared with the cylindrical shape, the rectangular object edges provide less contact area, which will require special approaching poses.

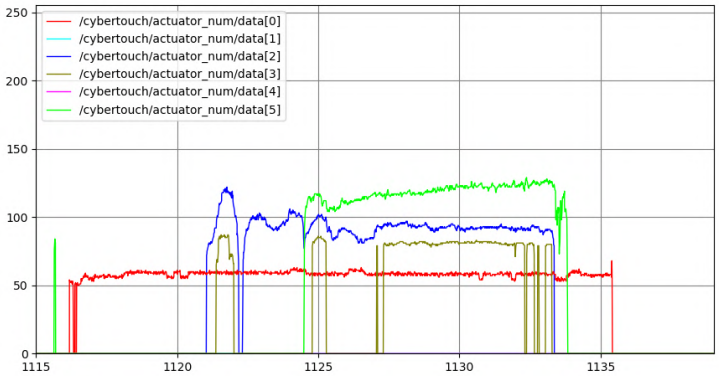
As shown in *Figure* 4.14. (1), grasping of smaller, $50mm\ size$, rectangular object was similar to grasping the smaller cylindrical object. The *Ring Ada* hand could also

achieve a stable power grasp if the thumb and fingers reached the object edges at the same time. The only finger behavior different from the cylindrical object is that this time the thumb FSR sensor is triggered as the distal just contacts the object edge.

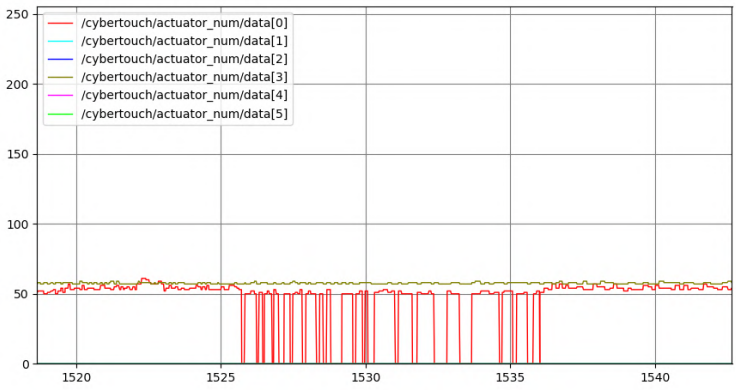
In the case of larger cubes, the recorded data output is shown in *Figure* 4.14. (2) and (3) display a significant output due to the feedback causing drastic vibrations in the glove. This behavior comes from the FSR and linear motor feedback generated at the same time. These data are really valuable for human operators and indicate that the hand is losing its grip of the object. The data from the in-hand sliding movement end with a drop of the object. We find that the stable grasping in *Figure* 4.12 [2] and *Figure* 4.12 [3] has approximately only a 10% probability of this happening. In most of the cases, the pinky finger is supporting the bottom of the object, stopping the further sliding of the cube.



(1)

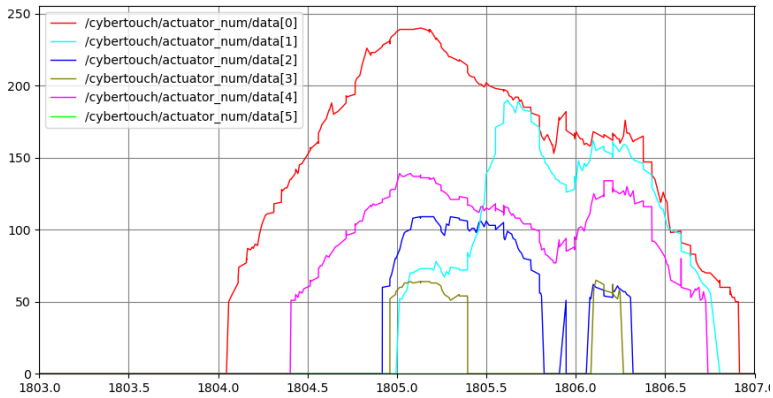


(2)

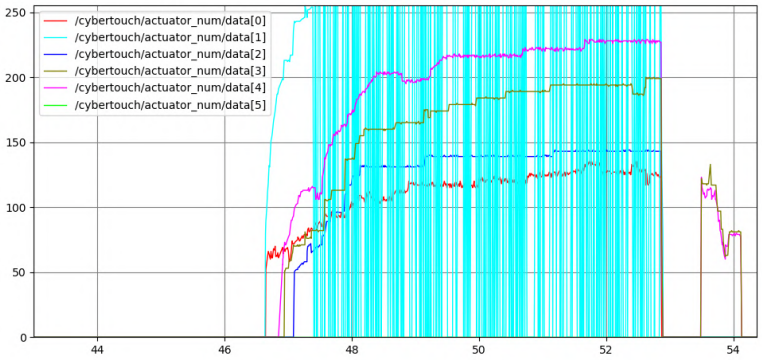


(3)

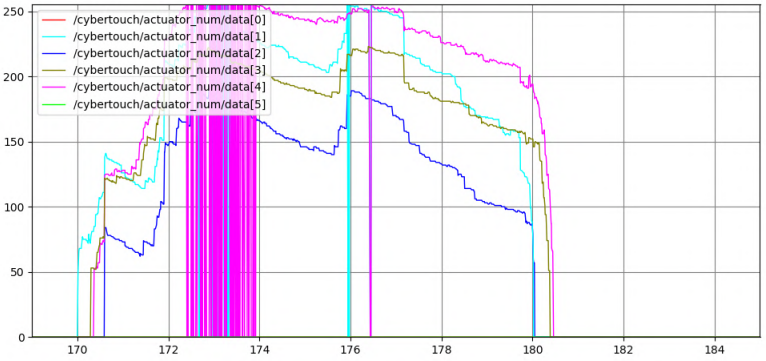
Figure 4.13 Raw data from the actuator input when grasping cylindrical objects with three different diameters (1) ϕ50mm, (2) ϕ70mm, and (3) ϕ90mm. Data [1]: index; Data [2]: meddle; Data [3]: ring; Data [4]: pinky; Data [5]: palm.



(1)



(2)



(3)

Figure 4.14 Raw data from the actuator input when grasping rectangular objects with three different diameters: (1) 50*mm, (2)* 70*mm, and (3)* 90*mm sizes. Data [1]: index; Data [2]: meddle; Data [3]: ring; Data [4]: pinky; Data [5]: palm.*

4.3. Conclusion

The experimental studies presented in this chapter validate the performance of the grasping by tactile-enabled multi-finger robotic hands remotely controlled by a human operator using a haptic-feedback data glove. The proposed research approach will contribute scientific knowledge to the development of a new generation of intelligent robots with advanced, human-like, tactile perception capabilities, enabling them to perform complex grasping or in-hand telemanipulation operations under poor or nonexistent visibility conditions, such as underwater, in space, in hazardous or high-risk security operational environments like nuclear power stations, highly infectious hospital rooms, war zones or in robotic surgery where touch feeling is of paramount importance.

Chapter 5. Grasp Control Skills

While machine vision could easily separate the object basic shapes, the recovery of the actual size and details of the objects still remains a challenging problem that could be difficult to solve due to the uncontrollable light conditions, background, and video camera calibration problems. Grasping objects by human-like robotic hands equipped with haptic sensors that provide a "haptic glance" could offer a valuable alternative allowing for blind object recognition.

As shown in the previous chapter, our novel *Ring Ada* robotic hand equipped with kinematic position sensors and touch sensors has proved that it is able to efficiently grasp different lightweight objects through teleoperation.

How the robotic hand can stable grasp the object is an essential step for object recognition. When using a multi-DOF robotic hand like *Ring Ada*, grasp planning is a high-dimensional problem. On top of that, the object's shape and material properties have to be taken into account, which adds another layer of complexity to the planning problem [101]. We need to specify information including but not-limited-to object's definition and location, robotic hand's location, grasping strategy selection, 3D spatial motion planning, etc.

Controlling a multi-DOF human-like dexterous robotic hand like *Ring Ada* so it efficiently performs object grasping and recognition operations is a highly complex problem, which cannot usually be solved by traditional analytic methods. To perform

more naturally, like the humans do, such complex operations the robotic hands should mimic the grasping skills of the humans. This chapter will present an intelligent fuzzy logic controller that allows the *Ring Ada* hand to learn from the human teleoperator such grasping skills.

5.1. Approaching

Before proceeding with the actual grasping, we need to solve the problem of how to approach the object.

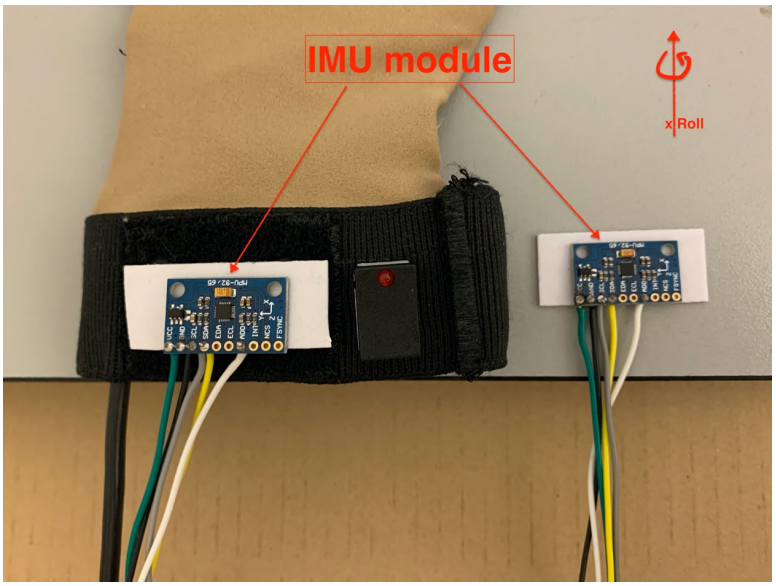


Figure 5.1 Measuring operator's wrist approaching angle

Figure 5.1 shows the experimental setup, which uses two inertial MPU6500 sensor modules [102] to measure the human operator's wrist angle relative to the table surface. One sensor is mounted on the table setting the reference frame, and another sensor is set on the back of the *CyberTouch* glove. Since the human wrist and the palm are tightly

connected by a group of ligaments [103], we could confidently use the wrist angle to represent the palm orientation.

In addition to the standard object sets used in the previous chapter, we are also using a few normal-sized objects fit inside of the Ring Ada hand: paper cups and tape rolls, as shown in *Figure* 5.2.

Figure 5.2 The objects used during the grasping experiments.

The teleoperator performed 30 times approaching and grasping demonstrations wearing the glove. In our case, we only detect the angle difference along the X-axis of the sensor, in other words, the 'Roll' angle. While the sensor baud rate set as 9600 baud, we are only extracting 1 sample per second in order to minimize the junk data. Because of the angle detection is based on the magnetic field, we put the two sensors as parallel as possible, and all the human hand movements happened at the top of the reference sensor on the table.

For a more convenient grasping, we use the object supporting platforms as shown in *Figure 5.3*.

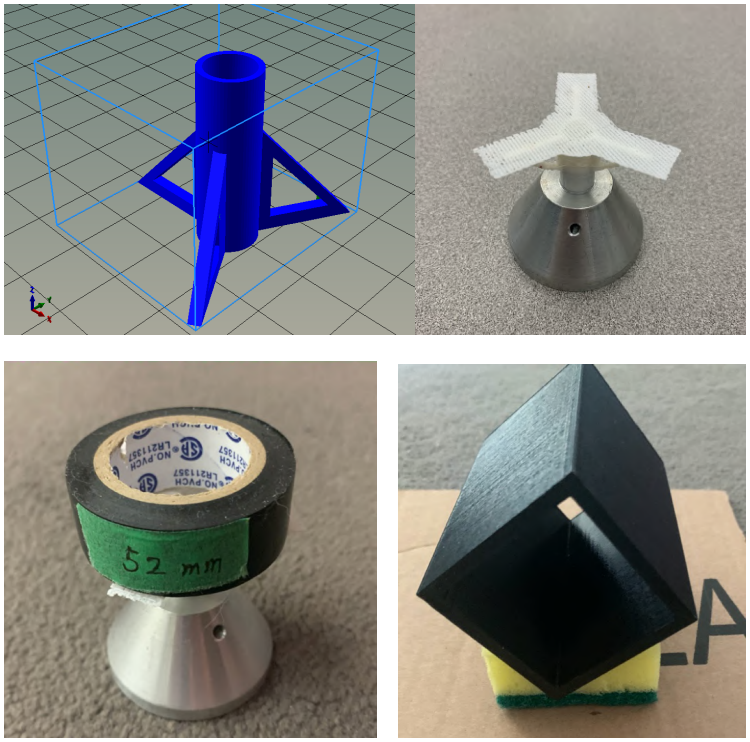


Figure 5.3 The object supports for grasping tasks

Figure 5.4 shows the experimental results representing the human wrist approaching angles collected by two of the IMU sensors while repeating 30 times the grasping for each object. Please note that we are only looking for the wrist angle when the grasping contact is happening. The diagram on the left side of each figure represents the 1Hz raw data picked from the 9.6kHz sensor output during the approaching cycles. The horizontal axis is the time in seconds, the vertical axis is the wrist angle. The diagram on the right side of each figure represents the angle data after applying a threshold as shown in Table 5.1, which keeps the peak angle only when reaching stable grasping.

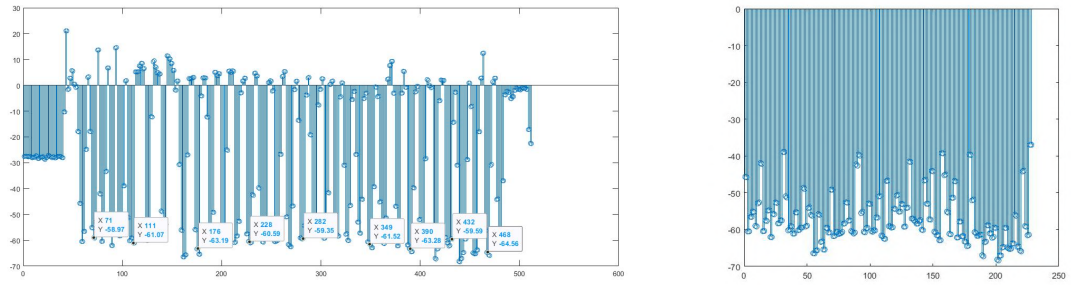


Figure 5.4 (1) Approach angle when grasping paper cups

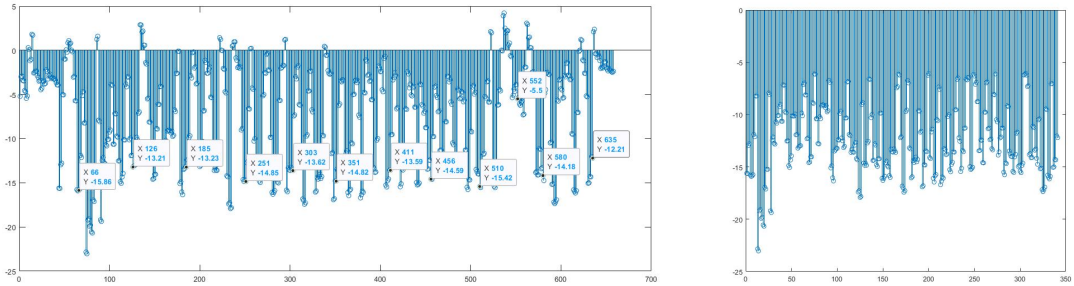


Figure 5.4 (2) Approach angle when grasping tape rolls

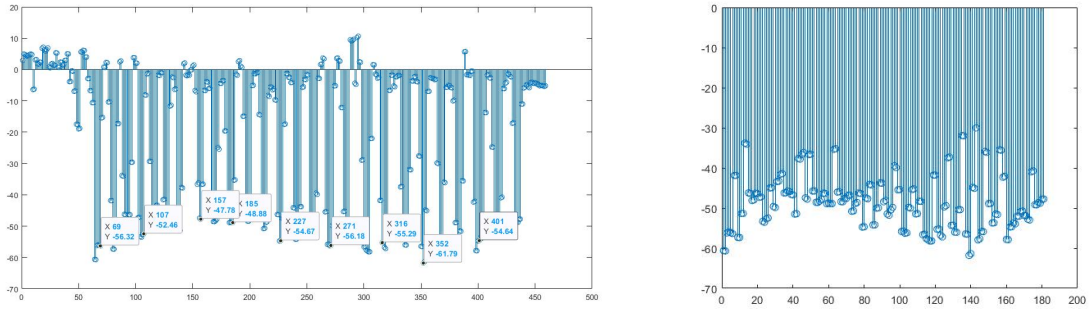


Figure 5.4 (3) Approach angle when grasping 30 mm cylinders

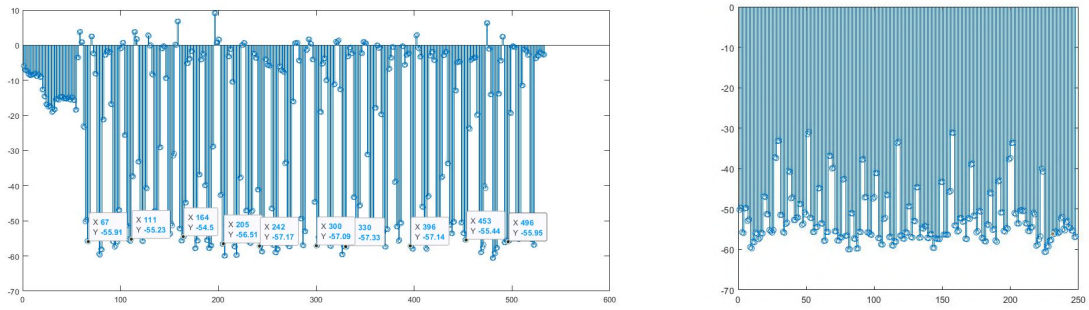


Figure 5.4 (4) Approach angle when grasping 70 mm cylinders

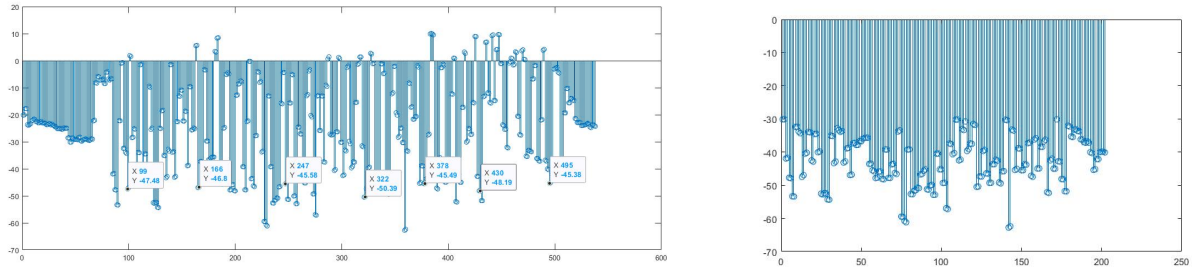


Figure 5.4 (5) Approach angle when grasping 30 mm rectangular objects (base)

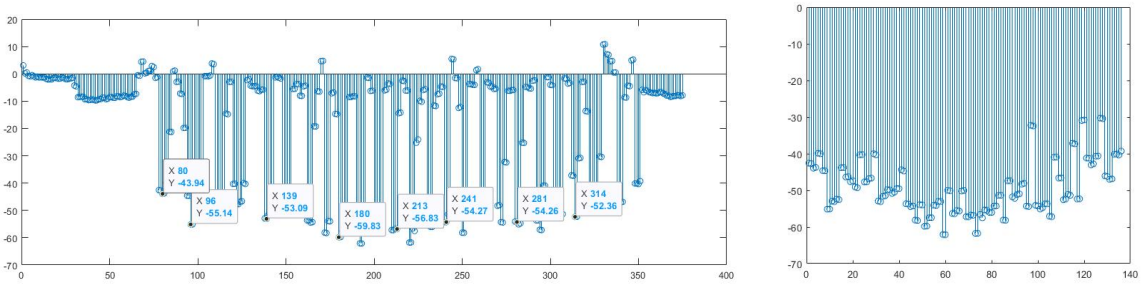


Figure 5.4 (6) Approach angle when grasping 30 mm rectangular objects (flat surface)

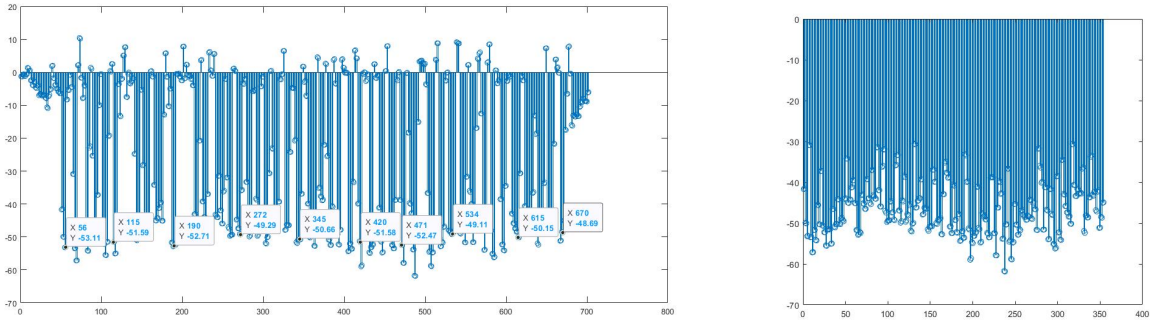


Figure 5.4 (7) Approach angle when grasping 70 mm rectangular objects (base)

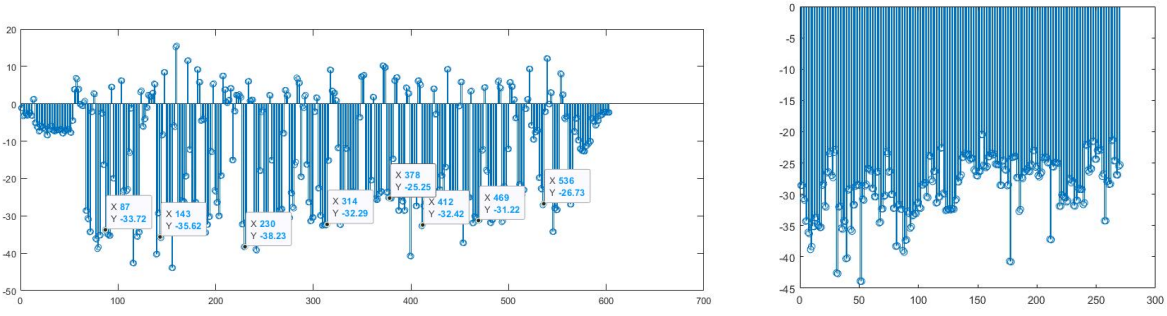


Figure 5.4 (8) Approach angle when grasping 70 mm rectangular objects (flat surface)

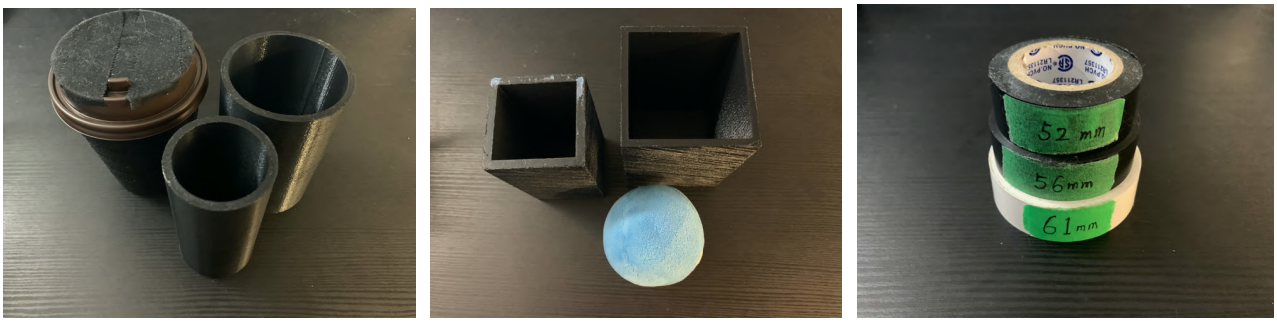
Object	*Threshold*	*Mode Value*
R30mm Flat	*Roll > 30°*	*62.12°*
R30mm Base	*Roll > 30°*	*46.95°*
R70mm Flat	*Roll > 20°*	*32.37°*
R70mm Base	*Roll > 30°*	*46.22°*
tapes	*Roll > 6°*	*14.29°*
Cup	*Roll > 35°*	*60.58°*
C30 stand	*Roll > 30°*	*56.18°*
C70 stand	*Roll > 30°*	*55.66°*

Table 5.1 Grasping approach angle of the palm

The experiments have shown that the human operator's hand only need to adjust a small angle to achieve a good grasping position for each standard object. As the human hand thumb carpometacarpal (CMC) joint could adduct more than 55°, while in the case of the *Ring Ada* hand the similar first joint could only adduct around 20°. *Figure* 5.5 shows a comparison of the twist angles of the human operator's hand and the *Ring Ada* hand when naturally position the thumb and the pinky at the same level. The human palm twisted around 35° to the reference surface, while the *Ring Ada palm needed to* twist to 15° for compromising the thumb's limitation. So, to reach similar contact positions when performing grasping operations, the *Ring Ada* hand's approach angle should be 20° smaller than that of the teleoperator's hand.

Figure 5.5 Human hand and Ring Ada hand thumb positions comparison

Experiments carried out on the set of objects shown in *Figure* 5.2 led to the conclusion that these objects could be split into three groups based on the approaching angles of the *Ring Ada* hand as follows: 60° for the 'Set 1' consisting of the cylinders and the paper cup; 15° for the 'Set 2' consisting of rectangular objects and the ball, and 10° for the 'Set 3' of all tape rolls, as shown in *Figure* 5.6.



(1) Set 1. Cylindrical object set. (2) Set 2. Rectangular object set. (3) Set 3. Tape roll set.

Figure 5.6 Object sets requiring similar grasping approach angles

5.2. Synergistic Control System.

Inspired by [100], we developed a synergistic control system based on the flow chart shown in *Figure* 5.7. The central block is the Fuzzy logic control module which handles the finger movements and uses FSR sensor data to provide an efficient postural synergy grasp control for the *Ring Ada* hand.

Essential data coming from the *Fuzzy logic control* module are also used for the object classification applications using the *Ring Ada* robotic hand.

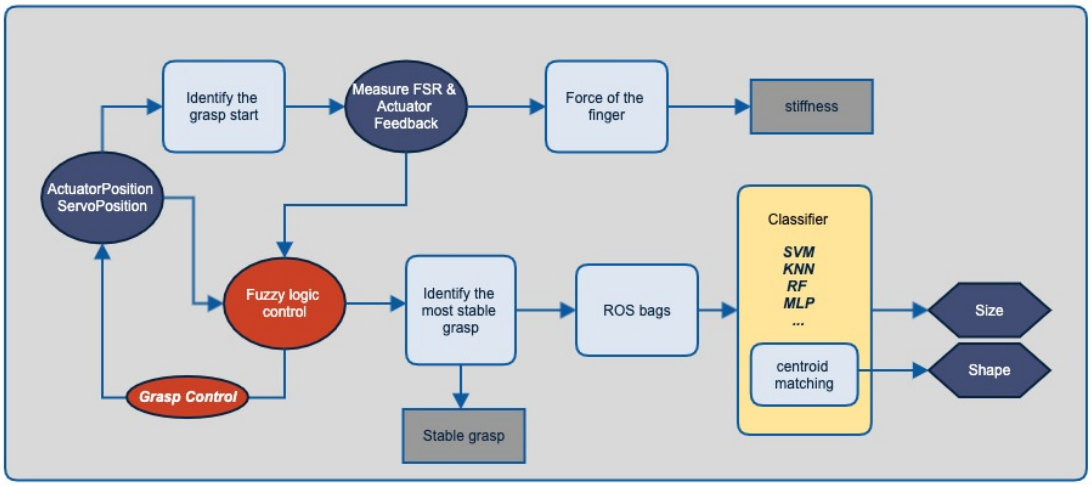


Figure 5.7 Control system flow chart.

Figure 5.8 shows the experimental set up used in this chapter, allowing the *Ring Ada* hand to draw near the object on a linear platform under a 'natural' approaching angle as previously discussed in section 5.1 in order to perform a stable grasp.

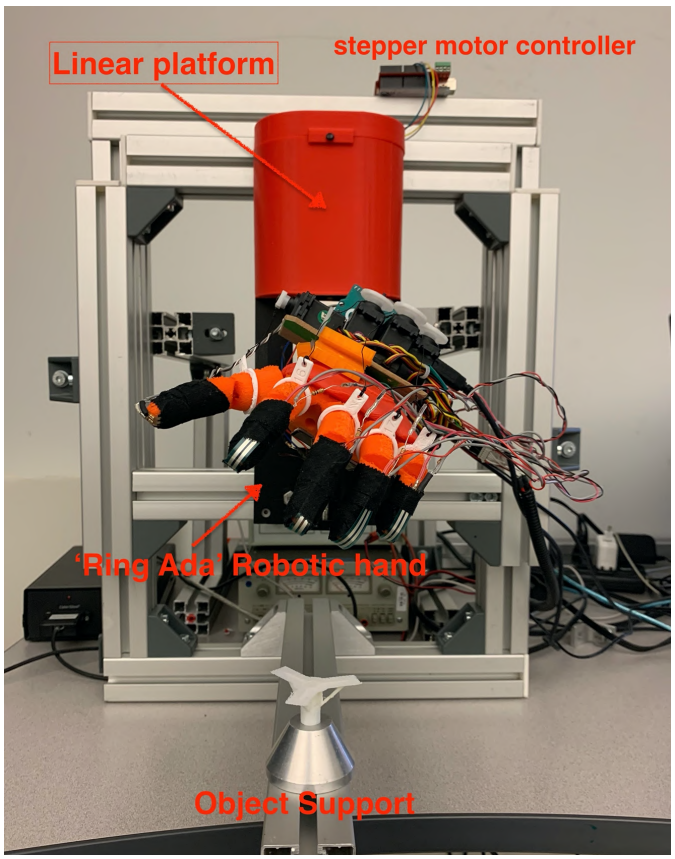


Figure 5.8 Experimental platform.

5.2.1. Fuzzy Logic Controller

The autonomous fuzzy grasping controller must provide a consistent grasp force while handling objects of different sizes.

Reference [65] has presented a double actuated fuzzy tactile controller based on pressure and micro-vibration outputs from BioTac© sensors[51]. The only control function of the finger feedbacks was to stop the motors from over pulling the finger close. With each finger actuated independently, more possibilities emerge for a more complex grasp controller [104] using postural synergy as a supervisor. It harmonizes finger movement within the synergy pattern in order to prevent undesirable behaviors.

As discussed in section 4.2.1, we could use both the FSR sensory feedback and the linear actuator feedback to control the underactuated fingers of the *Ring Ada* hand.

Figure 5.9 shows the Open-loop Fuzzy logic control flowchart for each hand finger, which is based on the finger status and sensor values in order to provide the proper action for each motor: “Open”, “Stay” or “Close”.

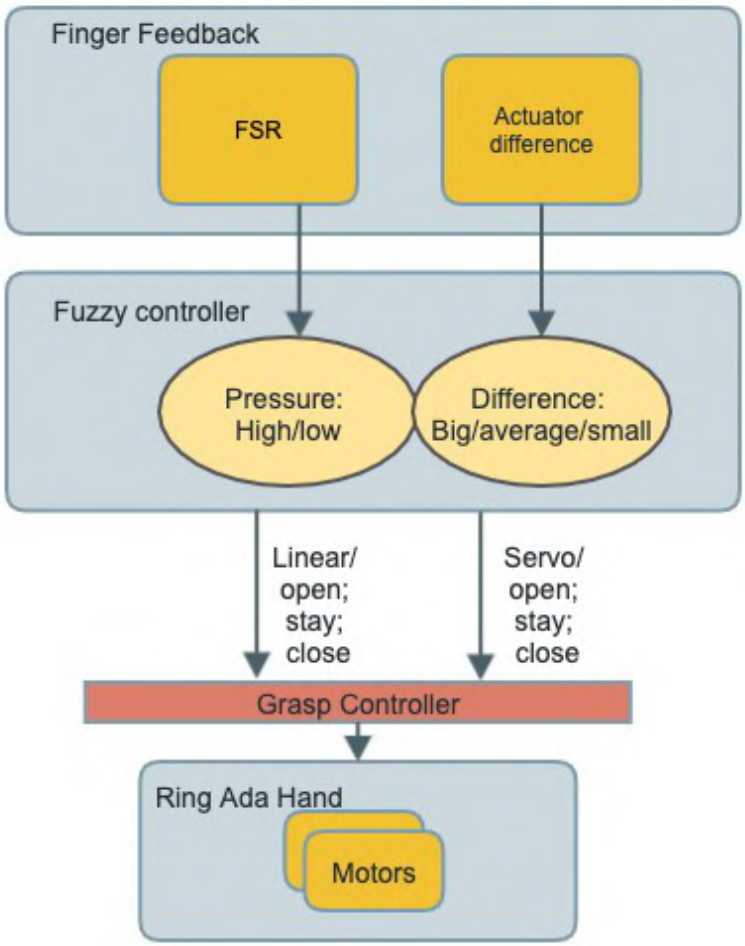


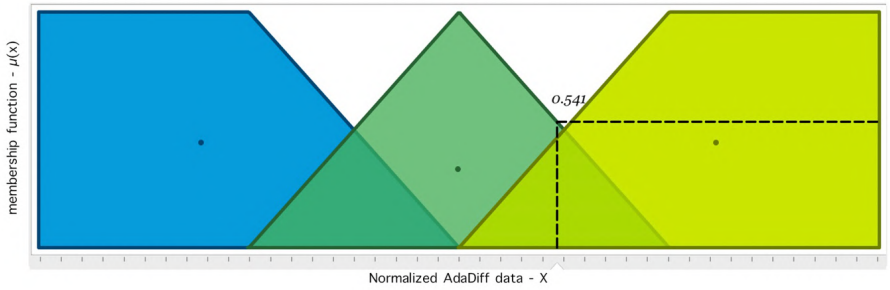
Figure 5.9 Motor fuzzy controller flowchart.

The fuzzy logic controller using a *Mamdani* and *Sugeno fuzzy-inference* engine with two sets of fuzzy rulebooks shown in Table 5.2. As discussed in Chapter 4.2, the FSR senor data lack of consistency and coherence, we addressed this issue in the 1st rulebook shown in *Table 5.2 (1)*. The servo motors' fuzzy control rulebook is shown in *Table 5.2 (2)*. It uses linear position output from rulebook one as input parameter,

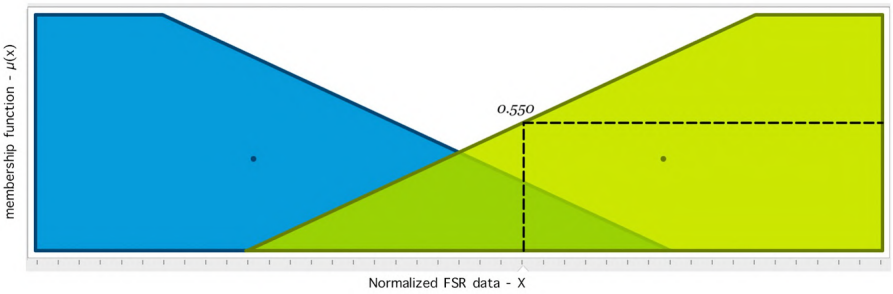
For each finger, two fuzzy input sets are defined in order to describe the finger status. The first one is 'AdaDiff' describing the difference between finger linear motor feedback, using 'Small', 'Average', and 'Big' fuzzy sets having trapezoidal and triangular shapes, shown in *Figure 5.10 (1)*. The second fuzzy set is the FSR input data, defined by the 'Low' and 'High' fuzzy sets shown in *Figure 5.10 (2)*. The fuzzy sets for the linear motor output and the servo motor output are shown in *Figure 5.10 (3)* and *Figure 5.10 (4)* respectively.

In the specific case of the example shown in *Figure* 5.10, the normalized AdaDiff membership function $\mu(x) = 0.000/small + 0.532/average + 0.468/big$, *resulting in* $\mu(x) = 0.541$. Similarly, the FSR data activate as $\mu(x) = 0.290/low + 0.543/high$, generate the result $\mu(x) = 0.550$.

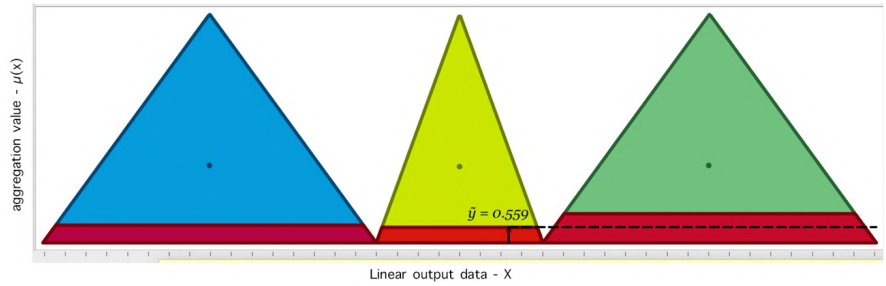
After the defuzzification for the 'first-stage' linear output, the output is shown as *Figure* 5.10(3)(4). The red overlap shapes the *Mamdani* Fuzzy-interface aggregation results. And the centroid output in this case of the fuzzy controller is $\tilde{y} = 0.559$, in the 'Stay' zone.



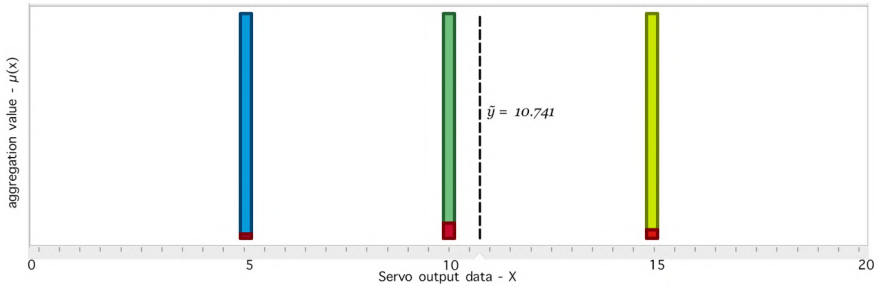
(1) Input sets of AdaDiff.



(2) Input sets of FSR sensor



(3) Linear Actuator position output sets



(4) Servo motor position output sets

Figure 5.10 Fuzzy logic controller's inputs and outputs.

We are using a singleton output membership function for the servo output of the *Sugeno* fuzzy-interface engine. The final output in this example is: $\tilde{y}$ = 0.020/close + 0.068/stay + 0.038/open (retaining only three decimal places), which yields $\tilde{y}$ = 10.741, which is in the 'Stay' zone, resulting in a standard stable grasp configuration for the *Ring Ada* hand.

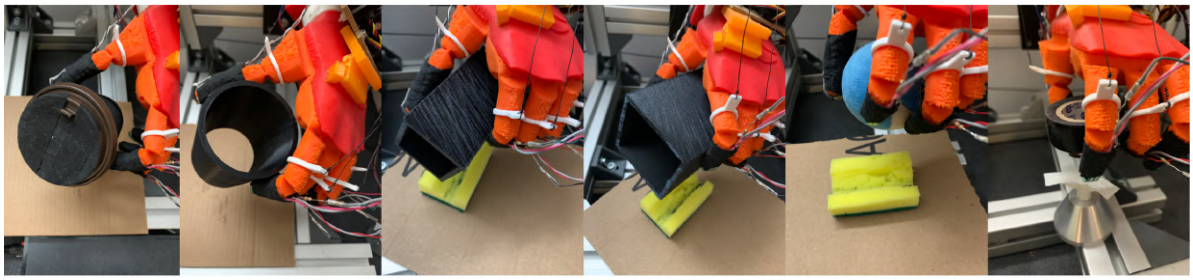
FSR / ***AdaDiff***	***Low***	***High***
Small	*close*	*none*
Average	*0.5*close*	*0.1*open*
Big	*0.5*stay*	*0.5*open*

Table (1), Stage one, Finger Linear actuator rulebook.

Linear output / ***AdaDiff***	***Close***	***Stay***	***Open***	***Special: FSR at high***
Small	*0.4* close*	*0.3*close*	*0.1*open*	*0.1*open*
Average	*0.5*close*	*0.2*close*	*0.1*open*	*0.1*open*
Big	*stay*	*stay*	*0.1*open*	*0.1*open*

Table (2), Stage two, Finger Servo motor rulebook.

Table 5.2 Fuzzy logic control rulebooks.



(1) *(2)* *(3)* *(4)* *(5)* *(6)*

Figure 5.11 Success grasping executions for (1) paper cup; (2) 70 mm cylinder; (3) 30 mm rectangular object; (4) 70 mm rectangular object; (5) soft ball; and (6) tape roll.

Figure 5.11 illustrates successful grasping operations performed by the *Ring Ada* hand on the set of objects shown in *Figure* 5.2.

5.2.2. Methodology and Data Acquisition

Figure 5.12 shows the experimental results obtained during grasping of a 30 mm rectangular objects from the 'Set 2' in *Figure* 5.6 (2), illustrating the four phases of the grasping process as further detailed.

The ***first phase*** is approaching, (before the red line in the diagrams of *Figure* 5.12), when the experimental platform lowers the *Ring Ada* hand to the right position for grasping, and the hand starts to close simultaneously controlled by the fuzzy controller.

During the ***second phase,*** the FSR sensor at the robotic hand's palm detects a first contact with the object surface as it could be seen in the *FSR/sensor_msg/data [0]* signal and the *cybertouch/actuator_num/data* feedback output [5], using the same data structure as the synergy based grasping teleoperation.

The ***third phase*** is stable grasping (between the green and orange line in *Figure* 5.12) when AdaDiff is 'Big' and all the FSR reads are 'Low'. Both the linear actuators and the servos are holding in 'stay'. After confirming that the hand movement is steady, the operator raises the platform while the hand is 'stable grasping' the object.

During the ***fourth phase,*** the *Ring Ada* hand is reset back to the initial position. It may be worth mentioning that the reference frame difference because of the various baud rates from different data ports [105].

The severe feedback is shown in *Figure* 5.12 (4) appears because of the 'big' AdaDiff input, reflected as a random alerting readout as discussed in Chapter 4.3.

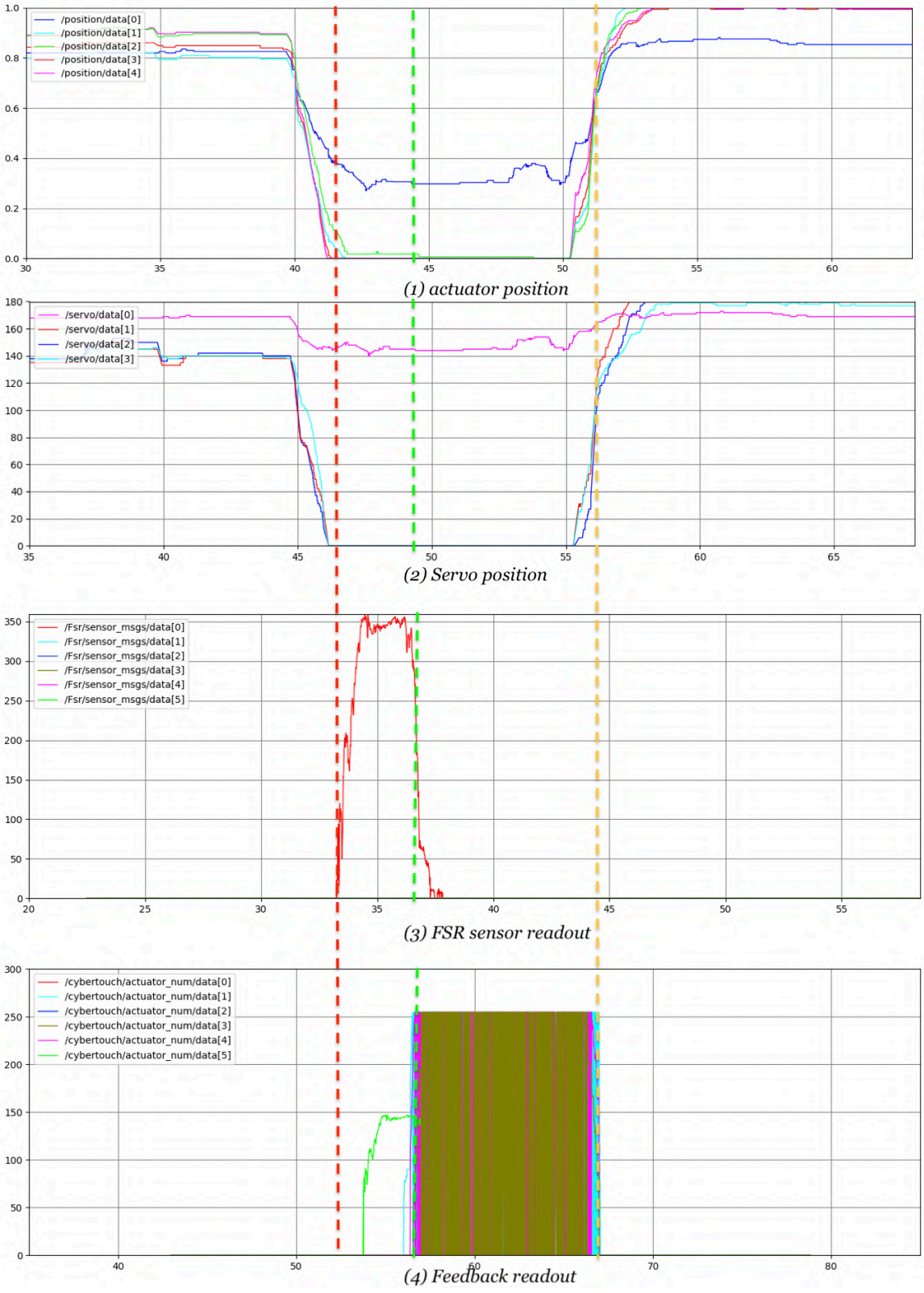


Figure 5.12 Fuzzy controller grasping data demonstration.

5.3. Classification

For each of the objects in each set, we performed 30 repeated experiments collecting data only when a stable grasping was successfully achieved. To prevent overfitting, the models we used separate data sets for further training and testing phases of the object classification.

We defined each data as a feature vector including 5 linear motor data and 5 servo position (pinky and ring finger repeat) data as: $[l_1, S_1, l_2, S_2, l_3, S_3, l_4, S_4, l_5, S_4]$, all data being normalized to 0~1. Each experiment has an average of 5 times grasping for the object sets labeled as 0~3 in Rosbag files. Training randomly picks up 4 bags of data in 5 grasping for each object from 21 groups of experiments. The remaining 9 groups of data are used for testing (only once). The training/ testing ratio was 70% / 30%.

Five classifiers were used for supervised learning: (i) Epsilon-Support Vector Regression (SVR in SVM) classifier, (ii) Multi-Layer Perceptron classifier (MLP), (iii) Random Forest classifier, (iv) Extra Tree classifier, and (v) KNeighbors classifier (implement k-NN).

The classifiers used the following (empirically chosen and are most likely not optimal) settings:

(1) SVR: kernel = 'poly'; regularization parameter: C = 5; Degree = 3;

(2) MLP: hidden layers = 10; 20 tuple each layer; Max iteration of 300;

(3) Random Forest: 200 estimators; max depth of 10;

(4) Extra Tree: 200 estimators; no max depth size;

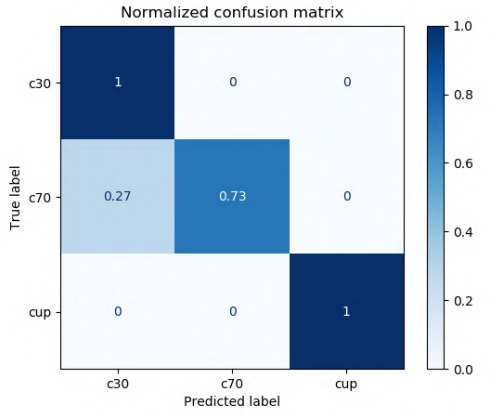
(5) KNN: 3 neighbors with leaf size of 20.

Table 5.3 shows the correct classification accuracy scores obtained by the five classifiers which were used. The best classifier for each object sets respectively was the Extra tree; *MLP* and *MLP*. The most challenging object set was 'Set 3' (the tape rolls).

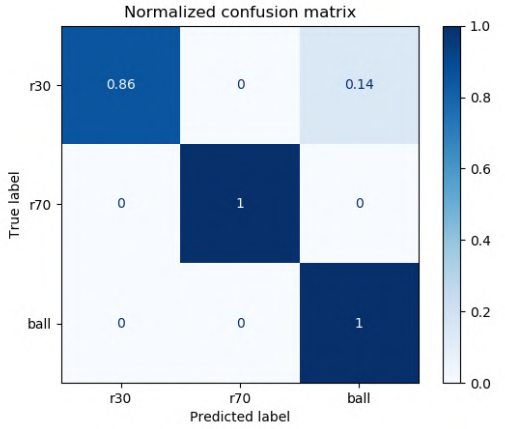
Classifier / ***Object Sets***	***SVR***	***MLP***	***Random Forest***	***Extra Tree***	***KNN***
'Set 1' in Figure 5.6	*0.891*	*0.861*	*0.891*	*0.918*	*0.917*
'Set 2' in Figure 5.6	*0.913*	*0.949*	*0.891*	*0.892*	*0.845*
'Set 3' in Figure 5.6	*0.464*	*0.80*	*0.539*	*0.543*	*0.437*

Table 5.3 Classification accuracy scores per object-set and classifier

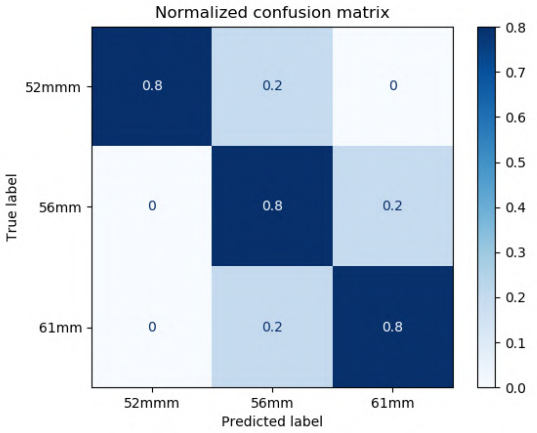
Figure 5.13 shows the confusion matrix of the prediction for each of the three object sets used in this experiment: the 'Set 1' consisting of cylinders and the paper cup, the 'Set 2' consisting of rectangular objects and the ball, and the 'Set 3' of tape rolls (as shown in *Figure* 5.6).



Extra Tree confusion matrix for Set1 (cylindrical objects and paper cup).



MLP confusion matrix for Set 2 (rectangular objects and ball).



(3) MLP confusion matrix for Set 3 (tape rolls).

Figure 5.13 Confusion Matrices for the three object sets.

Discussion

Based on the accuracy scores shown in *Figure* 5.13 and considering the limitations of the *Ring Ada* robotic hand, it can be concluded that the classification results are very good.

However, it should be noted that the experiments have resulted in a rather few successful stable grasps, which only covered 30% ~ 40% of all grasping attempts. The most challenging object set was that of the tape rolls because of their lightweight and slippery surface.

The thumb proved to be difficult to hold in an opposite position facing the other fingers' pressure. Since the thumb position is the most influencing data in the classification process, the low scores obtained for the SVR, Random Forest, Extra Tree, and KNN classifiers in the case of the 'Set 3' (the tape rolls) are explainable.

It is also easy to understand why the 70mm cylinder was recognized as a 30mm one. This confusion was caused by the elastic compliance of the robotic fingers while the linear actuator pulls them towards zero (the adduction fist gesture) when grasping any of these two cylinders.

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 and a biology-inspired synergistic real-time control system for teleoperated grasping operations using a *CyberTouch* Human Computer Interface (HCI) glove.

An intelligent fuzzy logic controller module was developed to efficiently control the underactuated *Ring Ada* robotic hand during grasping. A classification system is ready to further assist the human teleoperator during remote object grasping and manipulation operations.

Experiments have convincingly demonstrated that our *Ring Ada* robotic hand equipped with kinematic position sensors and touch sensors is able to efficiently grasp different lightweight objects through teleoperation.

The results of the work presented in this thesis are published in four international conference papers.

6.2. Future Work

Future improvements could be made by taking inspiration from the multimodal

human somatosensory system used during the dexterous object grasping and in-hand manipulation and develop a new generation of anthropomorphic robotic hands that will further reduce the gap between the robotic hands and the human hands.

A new *"Ring Ada - version 2"* prototype could be developed in the future by incorporating miniaturized multi-modal bio-inspired cutaneous tactile sensors recently developed by other graduate students working in our laboratory at the University of Ottawa. A new hand would replace the distal phalange with a 3D printed mount for new sensors, and the palm structure would be redesigned for bio-inspired tactile sensors, a new prototype anthropomorphic robotic hand as shown in *Figure* 6.1.

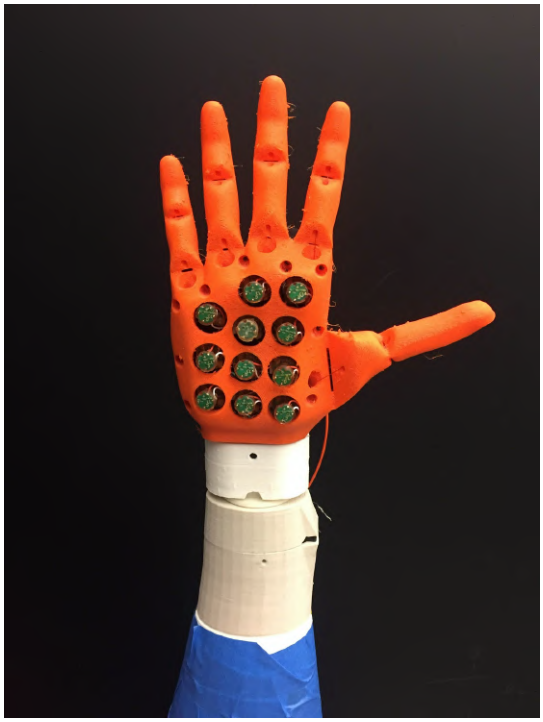


Figure 6.1 Prototype of multi-modal bio-inspired tactile sensor to equip Ada hand's palm.

The haptic feedback performance of the *CyberTouch* HCI glove needs to be further augmented by developing new vibratory tactile feedback modules to be attached to the glove's fingertips and palm so they map 1:1 the tactile information provided by the cutaneous sensors of the future *"Ring Ada - version 2"* hand. This would allow human teleoperators to take advantage of the object classification technique developed in Section 5.2.3.

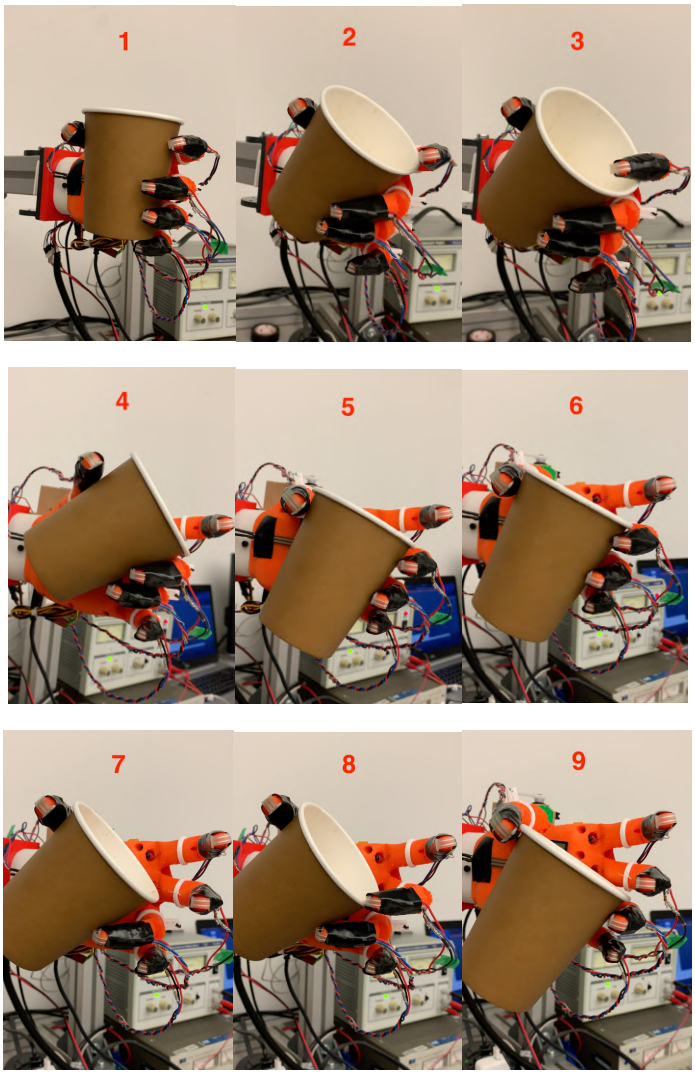


Figure 6.2 Dynamic grasp of a soft paper cup using Ring Ada robotic hand.

Finding the best object contact points is essential for a successful object grasping by an anthropomorphic robotic hand, as illustrated in *Figure* 6.2

Another future work direction would be upgrading the robotic hand's dexterity and refining the dynamic grasp control for dexterous manipulation [106], This relies on a fast-responsive synergy and sensitive feedback to allow reconfiguring the control parameters in the middle of the grasping operations,

Due to the infinite number of objects to be grasped in real-life applications, finding optimized grasping techniques is still an unfinished research challenge [107], [108].

References:

[1] T. E. A. de Oliveira, A.-M. Cretu, V. P. da Fonseca, and E. M. Petriu, "Touch sensing for humanoid robots," *IEEE Instrum. Meas. Mag.*, vol. 18, no. 5, pp. 13–19, Oct. 2015.

[2] J. H. He and J. W. Zhang, "In-hand haptic perception in dexterous manipulations," *Sci. China Inf. Sci.*, vol. 57, no. 12, pp. 1–11, Dec. 2014.

[3] H. Hanafusa and H. Asada, "A Robot Hand with Elastic Fingers and Its Application to Assembly Process," *IFAC Proc. Vol.*, 1977.

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

[5] A. M. Okamura and M. R. Cutkosky, "Feature detection for haptic exploration with robotic fingers," *Int. J. Rob. Res.*, vol. 20, no. 12, pp. 925–938, 2001.

[6] E. M. Petriu, "Haptic Sensors & Interfaces for Robotic Telemanipulation," 2006. [Online]. Available: http://www.site.uottawa.ca/~petriu/HapticDexterTelemanip-2018a.pdf. [Accessed: 27-Jul-2020].

[7] E. Al-Gallaf, A. Allen, and K. Warwick, "A survey of multi-fingered robot hands: Issues and grasping achievements," *Mechatronics*, 1993.

[8] R. R. Ma and A. M. Dollar, "On dexterity and dexterous manipulation," in *IEEE 15th International Conference on Advanced Robotics: New Boundaries for Robotics, ICAR 2011*, 2011, pp. 1–7.

[9] "CyberTouch — CyberGlove Systems LLC." [Online]. Available: http://www.cyberglovesystems.com/cybertouch. [Accessed: 17-Feb-2019].

[10] "Ada V1.1 Assembly Instructions — Open Bionics." [Online]. Available:

https://openbionicslabs.com/obtutorials/ada-v1-assembly. [Accessed: 12-Feb-2019].

[11] 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," in *Medical Measurements and Applications, MeMeA 2019 - Symposium Proceedings*, 2019.

[12] B. M. Rocha Lima *et al.*, "Heart Rate Detection Using a Miniaturized Multimodal Tactile Sensor," in *Medical Measurements and Applications, MeMeA 2019 - Symposium Proceedings*, 2019.

[13] A. Saikia *et al.*, "Recent advancements in prosthetic hand technology," *Journal of Medical Engineering and Technology*, vol. 40, no. 5. Taylor & Francis, pp. 255–264, 03-Jul-2016.

[14] "Hand & Wrist Anatomy | Dr Paul Jarrett, Hand, Wrist & Shoulder Surgeon." [Online]. Available: https://pauljarrett.info/treatments/hand-wrist/hand-wrist-anatomy/. [Accessed: 12-Dec-2019].

[15] "Phalanx bone - Wikipedia." [Online]. Available: https://en.wikipedia.org/wiki/Phalanx_bone. [Accessed: 12-Dec-2019].

[16] N. A. Davidoff and A. Freivalds, "A graphic model of the human hand using CATIA," *Int. J. Ind. Ergon.*, vol. 12, no. 4, pp. 255–264, Dec. 1993.

[17] E. Peña-Pitarch and Others, "Virtual Human Hand: Grasping Strategy and Simulation," *October*, no. October, pp. 1–144, 2007.

[18] A. K. Palmer, F. W. Werner, D. Murphy, and R. Glisson, "Functional wrist motion: A biomechanical study," *J. Hand Surg. Am.*, vol. 10, no. 1, pp. 39–46, 1985.

[19] S. Nylén, C. Sollerman, D. Haffajee, and L. Ekelund, "Swanson implant arthroplasty of the wrist in rheumatoid arthritis," *J. Hand Surg. Am.*, vol. 9, no. 3, pp. 295–299, 1984.

[20] R. H. Brumfield and J. A. Champoux, “A biomechanical study of normal functional wrist motion,” *Clin. Orthop. Relat. Res.*, vol. NO. 187, pp. 23–25, 1984.

[21] D. E. T. Shepherd and A. J. Johnstone, “Design considerations for a wrist implant,” *Medical Engineering and Physics*, vol. 24, no. 10. pp. 641–650, 2002.

[22] S. Jacobsen, E. Iversen, D. Knutti, R. Johnson, and K. Biggers, “Design of the Utah/M.I.T. Dextrous Hand,” in *1986 IEEE International Conference on Robotics and Automation*, 2005, pp. 1520–1532.

[23] K. Suzumori and A. A. Faudzi, “Trends in hydraulic actuators and components in legged and tough robots: a review,” *Adv. Robot.*, vol. 32, no. 9, pp. 458–476, May 2018.

[24] S. C. Jacobsen, H. Ko, E. K. Iversen, and C. C. Davis, “Control Strategies for Tendon-Driven Manipulators,” *IEEE Control Syst. Mag.*, 1990.

[25] T. Okada and S. Tsuchiya, “On a versatile finger system,” in *7th International Symposium of Industrial Robots*, 1977, pp. 345–352.

[26] Y. Nakano, M. Fujie, and Y. Hosada, “Hitachi’s robot hand,” *Robot. Age 6.7*, pp. 18–20, 1984.

[27] S. C. Jacobsen, J. E. Wood, D. F. Knutti, and K. B. Biggers, “The UTAH/M.I.T. Dextrous Hand: Work in Progress,” *Int. J. Rob. Res.*, vol. 3, no. 4, pp. 21–50, Dec. 1984.

[28] “Robots - Utah/M.I.T. Dexterous Hand closeup | 102693567 | Computer History Museum.” [Online]. Available: https://www.computerhistory.org/collections/catalog/102693567. [Accessed: 22-Apr-2020].

[29] S. R. Company, “Shadow dexterous hand C5 technical specification,” 01-Jan-2008. [Online]. Available: http://www.shadowrobot.com/downloads/shadow_dextrous_hand_technical_

specification_C6P6.pdf. [Accessed: 22-Apr-2020].

[30] “Shadow Hand - ROBOTS: Your Guide to the World of Robotics.” [Online]. Available: https://robots.ieee.org/robots/shadow/. [Accessed: 22-Apr-2020].

[31] 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 Trans. Syst. Man, Cybern. Syst.*, vol. 47, no. 7, pp. 1061–1070, Jul. 2017.

[32] “CyberGlove II — CyberGlove Systems LLC.” [Online]. Available: http://www.cyberglovesystems.com/cyberglove-ii. [Accessed: 22-Apr-2020].

[33] 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 Trans. Mechatronics*, 2017.

[34] 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,” in *IEEE International Conference on Intelligent Robots and Systems*, 2012.

[35] V. Prado, T. E. A. De Oliveira, K. Eyre, and E. M. Petriu, “Stable Grasping and Object Reorientation with a Three-Fingered Robotic Hand,” in *2017 IEEE International Symposium on Robotics and Intelligent Sensors (IRIS)*, 2017, pp. 5–7.

[36] V. Prado da Fonseca, T. E. Alves de Oliveira, and E. M. Petriu, “Estimating the Orientation of Objects from Tactile Sensing Data Using Machine Learning Methods and Visual Frames of Reference,” *Sensors (Basel).*, 2019.

[37] B. Massa, S. Roccella, M. C. Carrozza, and P. Dario, “Design and development of an underactuated prosthetic hand,” in *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292)*, vol. 4, pp. 3374–3379.

[38] B. Calli and A. M. Dollar, “Vision-based model predictive control for within-hand precision manipulation with underactuated grippers,” in *Proceedings - IEEE International Conference on Robotics and Automation*, 2017.

[39] J. Miura and K. Ikeuchi, “Task-oriented generation of visual sensing strategies in assembly tasks,” *IEEE Trans. Pattern Anal. Mach. Intell.*, 1998.

[40] T. H. Park, H. J. Kim, and N. Kim, “Path planning of automated optical inspection machines for PCB assembly systems,” *Int. J. Control. Autom. Syst.*, 2006.

[41] B. D. Lucas and T. Kanade, “An Iterative Image Registration Technique with an Application to Stereo Vision,” in *From Proceedings of Imaging Understanding Workshop*, 1981, pp. 121–130.

[42] P. Lavoie, “3D object reconstruction using structured light and two 2D images.,” Master Thesis. University of Ottawa, Ottawa,ON,Canada, 1996.

[43] B. J. Nelson, N. P. Papanikolopoulos, and P. K. Khosla, “Robotic visual servoing and robotic assembly tasks,” *IEEE Robot. Autom. Mag.*, vol. 3, no. 2, pp. 23–31, Jun. 1996.

[44] T. E. Alves de Oliveira, E. Petriu, and A.-M. Cretu, “Multimodal bio-inspired tactile sensing module,” *IEEE Sens. J.*, vol. 17, no. 11, pp. 1–1, 2017.

[45] S. J. Lederman and R. L. Klatzky, “Extracting object properties through haptic exploration,” *Acta Psychol. (Amst).*, 1993.

[46] H. I. Christensen and G. D. Hager, “Sensing and Estimation,” in *Springer Handbook of Robotics*, 2008.

[47] R. S. Dahiya, P. Mittendorfer, M. Valle, G. Cheng, and V. J. Lumelsky, “Directions toward effective utilization of tactile skin: A review,” *IEEE Sensors Journal*, vol. 13, no. 11. pp. 4121–4138, 2013.

[48] 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," in *WISP 2015 - IEEE International Symposium on Intelligent Signal Processing, Proceedings*, 2015.

[49] "Embedded Force Sensors | Tekscan." [Online]. Available: https://www.tekscan.com/products-solutions/embedded-force-sensors. [Accessed: 22-Jul-2020].

[50] "Peratech - Standard Products." [Online]. Available: https://www.peratech.com/standard-products/. [Accessed: 22-Jul-2020].

[51] "BioTac® Product Manual." BioTac, 2018.

[52] M. Borzage, "Startup Spotlight: SynTouch Seeks to Enable Robots With 'Machine Touch' - IEEE Spectrum." [Online]. Available: https://spectrum.ieee.org/automaton/robotics/robotics-hardware/startup-spotlight-syntouch. [Accessed: 22-Jul-2020].

[53] T. E. Alves De Oliveira, A. M. Cretu, and E. M. Petriu, "Multimodal Bio-Inspired Tactile Sensing Module," *IEEE Sens. J.*, 2017.

[54] L. W. Dale Purves, George Augustine, David Fitzpatrick, William Hall, Anthony-Samuel Lamantia, *Neuroscience. 5th Edition*. 2012.

[55] W. C. Nowlin, "Experimental results on Bayesian algorithms for interpreting compliant tactile sensing data," in *Proceedings - IEEE International Conference on Robotics and Automation*, 1991.

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

[57] "synergy | Origin and meaning of synergy by Online Etymology Dictionary." [Online]. Available: https://www.etymonline.com/word/synergy. [Accessed: 01-Apr-2019].

[58] B. Bernshtein and 1896-1966 Bernshtein, N. A. (Nikolai Aleksandrovich), *The co-ordination and regulation of movements*, 1st English ed. New York, 1967.

[59] M. Santello, M. Flanders, and J. F. Soechting, “Postural hand synergies for tool use,” *J. Neurosci.*, vol. 18, no. 23, pp. 10105–10115, 1998.

[60] 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.

[61] I. V. Grinyagin, E. V. Biryukova, and M. A. Maier, “Kinematic and dynamic synergies of human precision-grip movements,” *J. Neurophysiol.*, vol. 94, no. 4, pp. 2284–2294, Oct. 2005.

[62] A. D’Avella, A. Portone, L. Fernandez, and F. Lacquaniti, “Control of fast-reaching movements by muscle synergy combinations,” *J. Neurosci.*, vol. 26, no. 30, pp. 7791–7810, Jul. 2006.

[63] G. Salvietti, “Replicating human hand synergies onto robotic hands: A review on software and hardware strategies,” *Front. Neurorobot.*, 2018.

[64] L. A. Zadeh, “A Rationale for Fuzzy Control,” *J. Dyn. Syst. Meas. Control. Trans. ASME*, 1972.

[65] V. Ciobanu and N. Popescu, “Tactile controller using fuzzy logic for robot inhand manipulation,” in *2015 19th International Conference on System Theory, Control and Computing, ICSTCC 2015 - Joint Conference SINTES 19, SACCS 15, SIMSIS 19*, 2015.

[66] D. Petković, M. Issa, N. D. Pavlović, L. Zentner, and Ž. Ćojbašić, “Adaptive neuro fuzzy controller for adaptive compliant robotic gripper,” *Expert Syst. Appl.*, 2012.

[67] E. M. Petriu, “Fuzzy Systems for Control Applications.” [Online]. Available: http://www.site.uottawa.ca/~petriu/. [Accessed: 29-Jul-2020].

[68] R. Osswald and W. Petersen, “A Logical Approach to Data-Driven Classification,” 2003, pp. 267–281.

[69] Y. Wu and V. N. Vapnik, *Statistical Learning Theory*, vol. 41, no. 4. 1999.

[70] S. Li, J. T. Kwok, H. Zhu, and Y. Wang, "Texture classification using the support vector machines," *Pattern Recognit.*, 2003.

[71] C. F. Tsai, "Training support vector machines based on stacked generalization for image classification," *Neurocomputing*, 2005.

[72] W. Zhang, T. Yoshida, and X. Tang, "Text classification based on multi-word with support vector machine," *Knowledge-Based Syst.*, 2008.

[73] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Networks*, 1989.

[74] "1.17. Neural network models (supervised) — scikit-learn 0.23.1 documentation." [Online]. Available: https://scikit-learn.org/stable/modules/neural_networks_supervised.html. [Accessed: 22-Jul-2020].

[75] "1.10. Decision Trees — scikit-learn 0.23.1 documentation." [Online]. Available: https://scikit-learn.org/stable/modules/tree.html. [Accessed: 22-Jul-2020].

[76] P. Geurts, D. Ernst, and L. Wehenkel, "Extremely randomized trees," *Mach. Learn.*, 2006.

[77] N. S. Altman, "An introduction to kernel and nearest-neighbor nonparametric regression," *Am. Stat.*, vol. 46, no. 3, pp. 175–185, 1992.

[78] "File:KnnClassification.svg - Wikimedia Commons." [Online]. Available: https://commons.wikimedia.org/wiki/File:KnnClassification.svg. [Accessed: 22-Jul-2020].

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

[80] "The Open Hand Project - Home." [Online]. Available:

http://www.openhandproject.org/. [Accessed: 21-Nov-2019].

[81] D. Prattichizzo, M. Malvezzi, M. Gabiccini, and A. Bicchi, “On motion and force controllability of precision grasps with hands actuated by soft synergies,” *IEEE Trans. Robot.*, vol. 29, no. 6, pp. 1440–1456, 2013.

[82] I. M. Bullock, R. R. Ma, and A. M. Dollar, “A Hand-Centric Classification of Human and Robot Dexterous Manipulation,” *IEEE Trans. Haptics*, vol. 6, no. 2, pp. 129–144, Apr. 2013.

[83] 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.

[84] I. M. Bullock and A. M. Dollar, “Classifying human manipulation behavior,” in *IEEE International Conference on Rehabilitation Robotics*, 2011.

[85] “Guides to the Evaluation of Permanent Impairment,” *JAMA J. Am. Med. Assoc.*, 1971.

[86] A. Parmiggiani *et al.*, “The design of the iCub humanoid robot,” *Int. J. Humanoid Robot.*, vol. 9, no. 4, Dec. 2012.

[87] “rosserial_arduino/Tutorials - ROS Wiki.” [Online]. Available: http://wiki.ros.org/rosserial_arduino/Tutorials. [Accessed: 02-Jan-2020].

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

[89] “Packages - ROS Wiki.” [Online]. Available: http://wiki.ros.org/Packages. [Accessed: 28-Dec-2019].

[90] “sensor_msgs - ROS Wiki.” [Online]. Available: http://wiki.ros.org/sensor_msgs. [Accessed: 19-Apr-2020].

[91] “Topics - ROS Wiki.” [Online]. Available: http://wiki.ros.org/Topics. [Accessed:

04-Mar-2020].

[92] “Services - ROS Wiki.” [Online]. Available: http://wiki.ros.org/Services. [Accessed: 04-Mar-2020].

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

[94] “System Architecture — Leap Motion C++ SDK v3.2 Beta documentation.” [Online]. Available: https://developer-archive.leapmotion.com/documentation/cpp/devguide/Leap_Architecture.html. [Accessed: 06-Jan-2020].

[95] 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,” in *CIVEMSA 2018 - 2018 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications, Proceedings*, 2018.

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

[97] M. Malvezzi, G. Gioioso, G. Salvietti, and D. Prattichizzo, “SynGrasp: A MATLAB toolbox for underactuated and compliant hands,” *IEEE Robot. Autom. Mag.*, vol. 22, no. 4, pp. 52–68, 2015.

[98] F. Ficuciello, G. Palli, C. Melchiorri, and B. Siciliano, “Postural synergies and neural network for autonomous grasping: A tool for dextrous prosthetic and robotic hands,” in *Biosystems and Biorobotics*, vol. 1, Springer International Publishing, 2013, pp. 467–480.

[99] “FSR 400 Series.” [Online]. Available: https://www.interlinkelectronics.com/fsr-400-series. [Accessed: 20-Apr-2020].

[100] 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 Trans. Haptics*, vol. 9, no. 2, pp. 207–220, Apr. 2016.

[101] M. Przybylski, N. Vahrenkamp, T. Asfour, and R. Dillmann, "Grasp and motion planning for humanoid robots," in *Mechanisms and Machine Science*, vol. 10, Springer Netherlands, 2012, pp. 329–359.

[102] "MPU-6500 | TDK." [Online]. Available: https://invensense.tdk.com/products/motion-tracking/6-axis/mpu-6500/. [Accessed: 16-Jul-2020].

[103] "Anatomy of Hand & Wrist: Bones, Muscles, Tendons, Nerves, Pictures." [Online]. Available: https://www.healthpages.org/anatomy-function/anatomy-hand-wrist/. [Accessed: 18-Feb-2020].

[104] C. Mehring *et al.*, "Augmented manipulation ability in humans with six-fingered hands," *Nat. Commun.*, 2019.

[105] "rosbag - ROS Wiki." [Online]. Available: http://wiki.ros.org/rosbag. [Accessed: 13-Jul-2020].

[106] A. M. Okamura, N. Smaby, and M. R. Cutkosky, "Overview of dexterous manipulation," in *Proceedings - IEEE International Conference on Robotics and Automation*, 2000.

[107] V. Rakesh, U. Sharma, S. Murugan, S. Venugopal, and T. Asokan, "Optimizing force closure grasps on 3D objects using a modified genetic algorithm," *Soft Comput.*, 2018.

[108] M. T. Ciocarlie and P. K. Allen, "Hand Posture Subspaces for Dexterous Robotic Grasping," *Int. J. Rob. Res.*, vol. 28, no. 7, pp. 851–867, Jul. 2009.