SMART HAND

Mid Term Report

For Final Year Project

Due 10th January 2017

Stephanie Stoll

Department of Electronic Engineering

Faculty of Engineering and Physical Sciences

University of Surrey

Abstract

We introduce the Smart Hand Project, which aims to develop a hardware and software system to facilitate semi-automatic grasping of objects. A hardware prototype as well as a low-level control API were developed that allows the hand to open and close. The hand prototype is of a five-finger anthropomorphic design to allow a range of objects to be grasped. A high level control algorithm using a Convolutional Neural Network is being developed. The algorithm's aim is to servo the motors of the hand by evaluating the success or failure of a grasping strategy using visual as well as tactile sensory input. We give an outlook on future work which will include the collection of a dataset to train the high level control algorithm, as well as the structure of the final report to this project.

Contents

Abstract 2

Contents 3

1. Introduction 4

2. Literature Review 6

2.1. Dexterous Hands – Different Approaches for Different Applications 6

2.2. Dexterous Hands in the Open-Source Community, in Academia, and the Industry 9

2.3. Sensory Feedback Control Systems 12

2.4. Teaching Grasping Methods using Neural Networks 13

Convolutional Neural Networks 13

Using Neural Networks for Teaching Grasping Methods 14

2.5. Summary 15

3. Development to Date 17

3.1. System Overview 17

3.2. Hardware Prototype 18

Design and Assembly of Hand Chassis 18

Electronic Hardware 22

3.3. Low Level Control 23

3.4. High Level Control 24

4. Future Work 25

4.1. Planned Deliverables 25

4.2. Project Plan 27

4.3. Estimated Cost 28

4.4. Final Report Structure 29

5. Summary 30

6. Conclusions 30

7. References 31

Appendix A 35

Appendix B 39

1. Introduction

The aim of this project is to develop a hardware and software system capable of semi-automatic grasping of objects. For this a prototype hand called “Smart Hand” is being developed. The central idea is that by giving the hand the ability to determine the best grasping strategy for an object, the human control signal can be simplified significantly. This report documents the current state of the Smart Hand, in the context of an undergraduate Bachelor of Engineering project. The motivation for this work grew out of the advancements in recent years in the field of robotic hands and electronic hand prosthetics. Whilst research into robotic grasping has been undertaken for at least the last 50 years [1] it is only in the last two decades that robotic grippers and hands have reached the dexterity needed to fulfil complex and diverse tasks. As this technology advanced it has moved from industrial applications into the medical field, opening up the possibility for electronic, and sometimes sensing prosthetics that can offer varying degrees of ability. However, any sophisticated solutions still come at a high development and production cost. To make robotic hands and prosthetics affordable a growing number of hobbyists and engineers have made it their mission to take findings from academia and the industry and use them to develop low-cost, functional robotic hands and prosthetics. Especially with the advent of consumer 3D printing, low-cost PCB manufacture and off-the-shelf components this has become more and more feasible. However, most affordable hands still need to be manually controlled, either by a PC interface, a controller, or muscle sensors, the latter only giving rough control. This project’s aim is to develop a low-cost, but effective robotic hand prototype, that has the intelligence to make its own grasp choices depending on the object of interest. For this a number of objectives have to be met:

1. A working hand prototype needs to be developed to run experiments, collect data and showcase results.

2. A low-level API needs to be written that interfaces with the underlying hardware in an accessible way and is generic enough to be reusable for future work.

3. A high-level control algorithm needs to be developed that returns a grasping strategy to be executed by the prototype hand.

There are two main constraints in meeting those goals. Firstly, this project spans over one academic year only, with about ten work hours per week. An efficient work plan needs to be in place to ensure maximum work can be achieved in this time frame. Secondly, budget constraints could curb development, as this project relies on a working hardware prototype, whose design and manufacture goes through multiple renditions, accumulating cost.

This report is organised as follows: A review of developments in the field of robotic hands is presented in chapter 2. Chapter 3 presents the current state of the project, whilst chapter 4 gives an outlook on future work, including a project plan and cost estimation. Chapter 5 is a concise summary of all findings, and chapter 6 is a list of references. Appendix A shows the hand prototype in its current assembly and grasping some exemplary objects. Gantt Charts for semester one and two can be found in Appendix B.

2. Literature Review

The study of robotic hands is a vast and multifaceted field, spanning from mechanics, and mathematics, to control theory, machine learning, to artificial intelligence and bio-engineering. From the first industrial robotic arm in 1961 [1] to more recent developments such as anatomically accurate anthropomorphic robotic hands [2] and Google's big data and deep learning based robotic grippers [3] the field has a long history across different areas of interest. Robotic hands and grippers are found in industrial production lines, surgical and prognostic medical procedures and prosthetics, just to name a few. Due to the vastness of the field, when discussing robotic hands this review is focussing on dexterous hands only, specifically the definition given in [4], where manipulative dexterity is defined as "*the capability of the hand to manipulate objects so as to relocate them arbitrarily for the purpose of the task*". Sections 2.1 and 2.2 will introduce dexterous hands in different applications and from different sources. Section 2.3 will discuss sensor-based control systems, whilst section 2.4 will focus on ways to use machine learning to teach robotic hands to grasp different objects. Section 2.5 will summarise findings and conclusions.

2.1. Dexterous Hands – Different Approaches for Different Applications

For the design of dexterous robotic hands there are two schools of thought: building a hand that resembles the design of the human hand and can have a broad range of applications (anthropomorphic), or designing an application-specific gripper, that is tuned to a certain task and environment (minimalistic) [5], [6]. Which way to choose depends on the hand's application and purpose, e.g. in prosthetics robotic hands mostly resemble human hands; for cosmetic reasons, but also to exploit the natural design created by years of evolution, making them versatile tools that can assist in many daily tasks. Humanoid robot hands tend to have anthropomorphic features for similar reasons. Anthropomorphic robotic hands also have the advantage of being more intuitive to control if operated by a human [7]. Minimalistic hands tend to be good at one specific tasks, for example displacing an object in a production line, or grasping specific items. Their purpose-built design tends to be simpler, as it does not have to account for a variety of use cases or resemble a human hand.

Name	**Research Institute**	**Year**	**No. of Fingers**
Stanford/JPL Hand	Stanford University	1983	3
Utah-MIT Hand	Utah University	1985	4
Gifu Hand	Gifu University	1999	5
DLR II Hand	DLR-German Aerospace Center	2001	4
Shadow Hand	Shadow Robot Company Ltd.	2002	5
Thing Hand	University of Florida	2002	4
Modular Prosthetic Limb (MPL)	John Hopkins University	2011	5
Biomimetic Robotic Hand	University of Washington	2016	5

Table 2-1: Some important dexterous robotic hands and their number of fingers [7], [8], [2]. See section 2.2 for a more in-depth description of some of these hands.

Table 2-1 shows some significant dexterous robotic hands developed over the last 33 years. While we will look at the design choices for some of these hands in more detail in the following section, the focus here lies on the number of fingers chosen to implement each hand. All hands apart from the Stanford/JPL hand employ a four to five finger design, which suggests anthropomorphic features. The volume of data in this table is by no means large enough to deduct a global trend, however [7] suggested in 2005 that 50% of dexterous robotic hands show a five finger design (28% for four, 22% for three).

Why a four to five finger design seems favourable when designing multi-purpose, dexterous hands becomes apparent when looking at different grasp types the human hand is able to perform (see Figure 2-1).

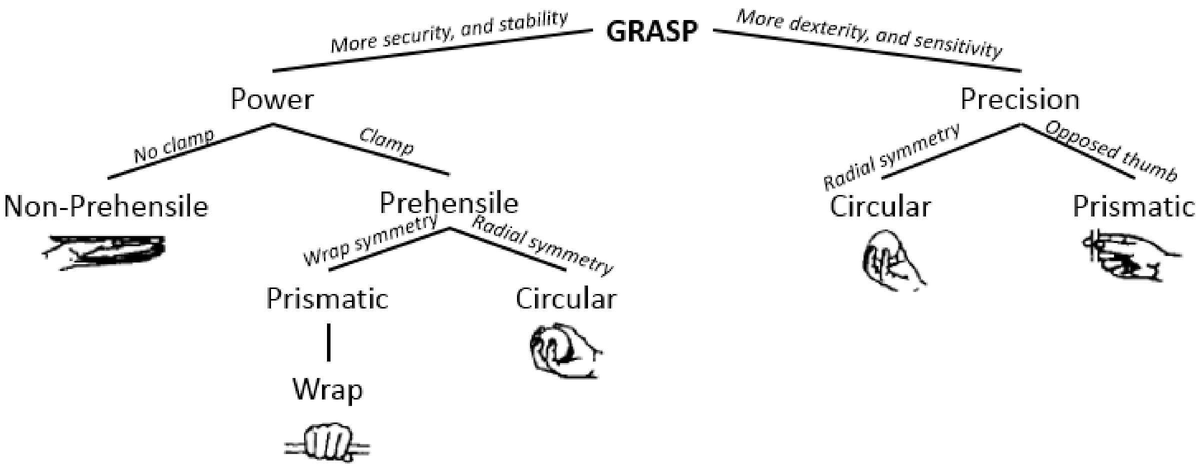

Figure 2-1: Simplified diagram of different grasps (adapted from [9]) . Each grasp method is represented by one exemplarily variation.

In both power- and precision-prismatic grasps the use of an opposed thumb seems of great importance. Additionally, using more fingers on the other side of the object provides more stability. For circular grasps it is necessary to radially wrap around the object, which requires at least three fingers ("tripod grasp"), where again more fingers provide more support [9].

The idea that hands with an anthropomorphic design perform well for doing anthropomorphic tasks seems somewhat intuitive. According to [10] "*the human hand became generalized rather than specialized to a particular environment*", making it the perfect multi-purpose tool. Anthropomorphic robot hand design strives to re-engineer the human hand's capabilities by using the human hand itself as a model. Whilst this is a relatively straight-forward design concept, its actual implementation often becomes incredibly involved [7], [5]. The human hand has 22 degrees-of-freedom [11]. While there are anthropomorphic robotic hands that can reach that level of dexterity (e.g. the Shadow Hand [12]), their development involves a significant amount of time, resources, and expertise.

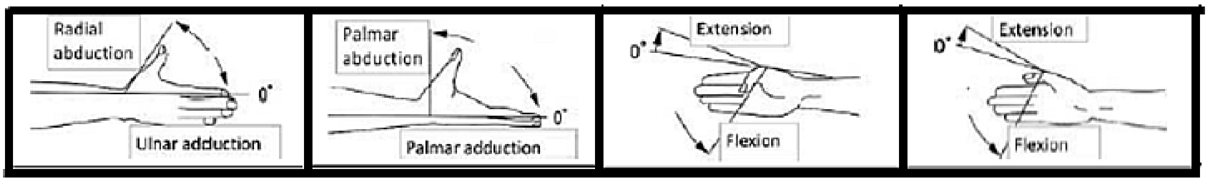

Figure 2-2: Possible movement of the human thumb (adapted from [10]).

To illustrate the complexity of the human hand it is worth taking a closer look at the thumb. As shown in Figure 2-2 the human thumb is capable of a number of different motions. It is the capabilities of the thumb that sets the human hand apart from those of primates [10]. Implementing a highly dexterous robotic thumb requires significant mechanical skill [13], and often not all aspects of the human thumb can be replicated. Some examples of thumb implementations are [13] (Gifu Hand), [11] (Utah-MIT Hand), and [14] (DLR-II Hand).

The mechanical complexity and associated cost of most anthropomorphic dexterous robotic hands make them unsuitable for mass use in the manufacturing industry, where cheaper, lighter and scalable solutions are needed. Two- and three-finger robotic grippers aim to fit those requirements [6], [15]. There are two main flavours of two- and three-finger grippers: parallel and adaptive [15] (see Figure 2-3).

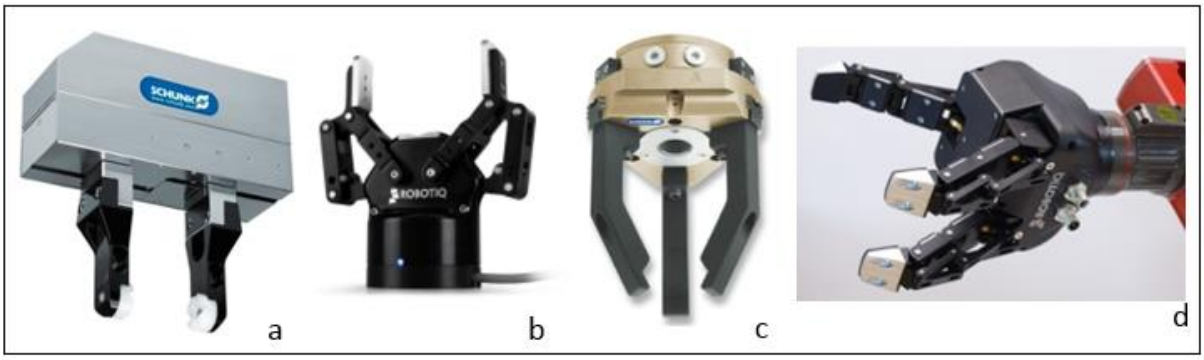

Figure 2-3: Different Gripper Types. (a) parallel two-finger gripper, (b) two-finger adaptive gripper, (c) three-finger parallel gripper, (d) three-finger adaptive gripper. (Adapted from [15]).

Parallel two-finger grippers have two stiff fingers or plates that move towards and in parallel to each other, which allows them to perform a pinch grasp. Due to their relatively simple design they exist in various sizes and different materials. However, their inadaptability makes them unsuitable for tasks that require handling different objects. In this case on might opt for an adaptive gripper, whose fingers can shape itself around an object, making them a more versatile solution. Three-fingered parallel grippers, have stiff fingers that move towards their shared mid-point, which makes them automatically self-centring. However, much like the two-fingered variant, they are unsuitable when handling a range of objects. The three-fingered adaptive gripper provides a much better compromise between flexibility and reliability. It can adapt to an object's geometry, whilst still being able to carry relatively large payloads [15]. The Stanford/JPL hand [16] is a good example of this type of gripper. There are other variations of these basic robotic grippers that aim to achieve high dexterity using only two to three fingers; e.g.

the DxGrip-II [6] uses rotating pads in its two grippers to manipulate an object's position by rotation. More recently, Velvet Fingers have been introduced whose active surfaces enable them to apply different levels of friction and tangential thrust to an object [17].

For this project an anthropomorphic hand design was chosen. This decision was made based on a number of different reasons: Firstly, because one of the hand's requirements is to be able to grasp a range of different objects. A two- or three-finger gripper could have been developed to fulfil this task, but would have needed significant time and mechanical skill to design. Albeit some time was spent on developing the anthropomorphic robotic hand, its design was a lot more intuitive. Secondly, an anthropomorphic design adheres to our original motivation of investigating ways to build a low-cost, functional prosthetic. Additionally, there are a lot of instructions and open source materials available online that specialise on building anthropomorphic hands for robotics, animatronics, and prosthetics. Using those open source projects as inspiration also proved useful as they often are subject to budget and time constraints similar to this project. The following section will introduce some open source, low budget projects and compare them to anthropomorphic robotic hands in academia and industry.

2.2. Dexterous Hands in the Open-Source Community, in Academia, and the Industry

Budget, time, and expertise determine the scope and outcome of any project. Robotic hand projects are no exception. The money available determines the materials that can be used to build the hand, for example it is less costly to use plastic rather than metal, as plastic is a cheaper raw material and more affordable in manufacturing. The downsides are stability and fragility. The budget also affects the number and types of sensors which can be built into the hand, as well as their quality. Hence, money has a direct influence on the durability and ability of any robotic hand. Time and expertise also affect the ability and sophistication of a hand. A dedicated research team comes with a lot more man hours and skills than one person alone (like a hobbyist or engineering student). As this is a relatively low budget, one person project, it is useful to compare some of the most advanced hands in academia and the industry to a selection of hobbyist and open source projects to get a better understanding of what is possible to achieve. This comparison includes different types of actuation and transmission, as well as types and number of sensors.

The internet has made it easier than ever for researchers, and hobbyists to share information and present their achievements to the world. This enables hobbyists from different backgrounds to take inspiration from academic and industrial research and adapt it to suit their real-world needs, or come up with less costly solutions. There are countless platforms available to an ever-growing community of like-minded engineers, students, artists, and hobbyists. Some examples are sites such as Hackaday [18] (electronics), Instructables [19] (electronics, arts, crafts, foods), and Thingiverse [20] (anything 3D printable). A quick search for "robotic" "hand" gives hundreds of results across these sites. Many projects are driven by the demand for affordable and useful hand and arm prosthetics. One very successful prosthetics project is Open Bionics which has moved on to become an award-winning start up [21]. Their Ada Hand is a 3D-printed robotic hand with 5 DOF, using linear actuators and tendons as transmission. It can be controlled with a PC or Mac via USB, or using muscle sensors [22]. Another notable project is Gael Langevin's InMoov, an open-source 3D-printed life size robot. It is completely re-printable on a home 3D printer and all files are downloadable for free [23]. The InMoov hand uses five servos as actuators and tendons as transmission and is controlled using an Arduino microcontroller. There are self-made capacitive sensors in each fingertip [24]. Both the Ada Hand and the InMoov Hand are used by researchers and hobbyists, either as-is or as a platform for further development, at a fraction of the cost of a more sophisticated hand.

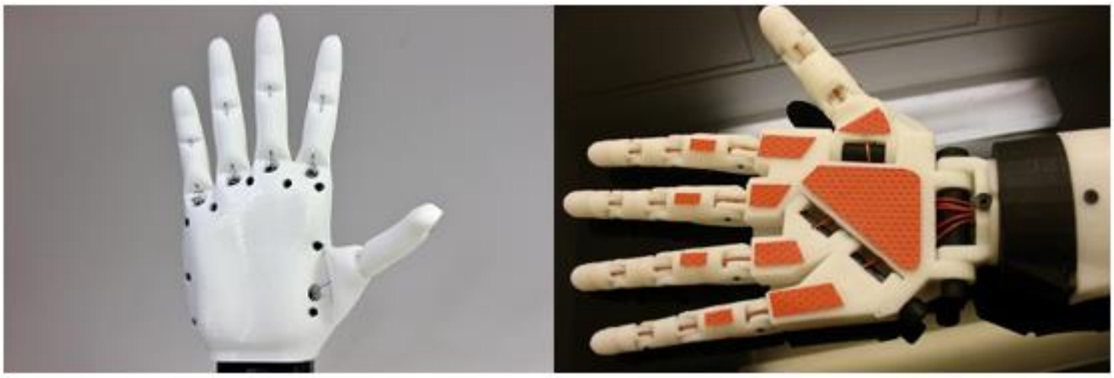

Figure 2-4: Open Bionics Ada Hand (left) [22], and InMoov Hand (right) [24].

In 2.1 it already became apparent that there has been a growing interest in developing reliable dexterous anthropomorphic robotic hands over the last 30 years. Many books (e.g. [25]) and papers (e.g. [4], [5], [7]) have been written depicting the history of robotic hands. Due to the number of different hands being developed over time it would be impossible to discuss all important hands in the context of this report. Hence, our discussion will focus on the Gifu Hand and the DLR II, as they are both still being further developed, and referenced across literature.

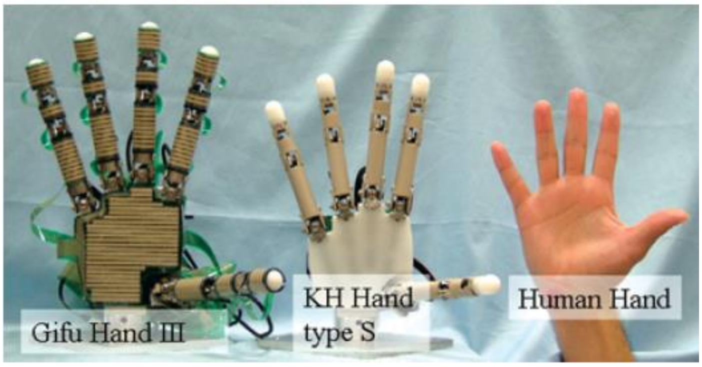

Figure 2-5: The Gifu Hand Series [13].

The Gifu Hand series has been in development since 1996 to study grasping and dexterous manipulation. It is designed to resemble a large human hand, each of the four fingers has three DOF, whilst the opposable thumb has four DOF. The four fingers are actuated by three servomotors each, the thumb by four servomotors, all equipped with a rotary encoder. Gears and links are used for transmission to avoid error introduced by the elasticity of tendons. Force sensors in the fingertips and distributed tactile sensors across the fingers and palm help grasp and manipulate objects [13]. The latest models in the Gifu series are shown in Figure 2-5.

The DLR-II Hand was first introduced in 2001 as the DLR-I's successor. It has three fingers and a thumb. Each finger has three DOF, while the thumb again has four DOF, making a total of 13 DOF. Brushless DC-motors are used as actuators, with gears providing transmission in the joints. A differential bevel gear joint at the base of each finger enables extension as well as flexion of fingers. There is a position, motor speed, and torque sensor in each joint, and a six-dimensional force/torque sensor in each fingertip. Additionally, every finger holds six temperature sensors. There is a serial ADC in each link, converting sensor signals into digital data before transferring it to the communication controller, which then sends data to an external PC. All electronics are fully integrated into the hand and printed on flexible circuit board [14].

Comparing the Ada and the InMoov Hand with the Gifu and the DLR-II Hand it becomes apparent that for the latter two a great amount of time and money has been spent on independently actuating joints, rather than using tendons to transmit force from a remote actuator across the finger joints. This has the advantage of providing position and bend information for each finger, as well as allowing accurate control over each link. Avoiding the use of tendons also eliminates error which is otherwise introduced by slight elasticity of the tendons and backlash introduced by the Bowden system. There is also a significant increase of sensors integrated into the hand, compared to the open source examples.

Figure 2-6: The Shadow Dexterous Hand [26].

One of the most advanced dexterous robotic hands available today is the Shadow Dexterous Hand by the Shadow Robot Company. Its current version boasts 20 DOF and a total of 24 joints, as well as 129 sensors including absolute position and force sensors as well as touch sensors. The hand is available with electric actuation and tendon-driven or with pneumatic actuation using air muscles [26].

From comparing open source, academic, and industrial hands it appears that for every bit of added dexterity more budget is needed to pay for an increased number of motors, integrated circuit boards, material such as metal and flexible PCBs, as well as highly sensitive sensors. Although no pricing has been made available for the Shadow Dexterous Hand it was estimated at €90,000 when it was first available for public sale in 2005 [27], [28]. Apart from material cost, all academic and industrial hands needed years of developing time to get to this stage of sophistication, requiring expertise in mechanical design, electronics, miniaturisation and programming. For this project it would be unrealistic to aim to design a hand as developed as the Shadow, Gifu, or DLR Hand. But even Open Bionics' Ada Hand costs £569 excl. VAT [22], which exceeds the budget of this project. Hence, it was initially investigated to use the InMoov Hand as a cost-effective solution. It would be re-printable with the facilities available to this project, and there are detailed instructions available for its assembly online. However, the original InMoov Hand proved unsuitable for this project without some serious alterations in order to fit more sensors onto the fingers and the palm. Eventually, the choice was made to design a 3D-printable anthropomorphic robotic hand incorporating lessons learned from other hand projects such as InMoov. This provides the required amount of flexibility for making design choices and guarantees that underlying mechanical and electrical principles are understood as they are integral to the design process.

2.3. Sensory Feedback Control Systems

Sensory feedback can be used to control a robotic hand by providing information about its' environment, position and any object it is interacting with. There are many different sensors available to be used in robotics, however, this brief review will focus on tactile and visual sensors only.

Tactile sensors, such as capacitive [29] or resistive force sensors [30] give the hand dexterity by providing information about whether the hand is touching an object and with what force. Flex sensors can be used to measure flexing or bending of a finger as it has actually been done in the Nintendo Power Glove [31]. Stretch sensors can be used to measure stretch, displacement and force by increasing its resistance when stretched [32]. When using tendons for transmission and servos as actuators a potentiometer on the servo can give positional feedback [33] which in turn can provide information about whether a finger is closing around or touching an object.

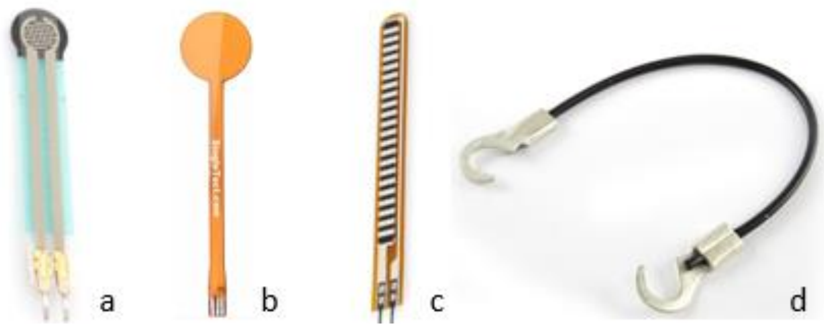

Figure 2-7: Tactile Sensors. a) force-sensitive resistor, b) capacitive force sensor, c) flex sensor, d) stretch sensor. Images taken from [35].

Sensory feedback can also be visual, in which case cameras are used as sensors. These can be end-effector mounted, e.g. onto the hand, or fixed in the workspace. A system of cameras is also possible. Correct coordinate transformations between object space, hand space and camera spaces are essential for this type of feedback [34]. Video frames can be used to extract information such as positions and velocities, or to extract image features and use those for object recognition or visual servoing[1].

Apart from actuators, sensors are the most expensive part of this project. The challenge therefore is to achieve maximum control over the hand, using a minimum of sensors. As well as the number of sensors, cutbacks are also made in the quality of the sensors. The current setup for this project uses two resistive force sensors in each fingertip (apart from the little finger) and a small camera in the palm of the hand. Potentiometers provide feedback from the servo motors. The force sensors are resistive rather than capacitive meaning they each only cost about a quarter of a capacitive equivalent (see e.g. [35]). The drawback is that resistive force sensors are a lot less accurate than the capacitive alternative. For a detailed description of the sensor configuration in this project see section 3.

2.4. Teaching Grasping Methods using Neural Networks

More research into the field of deep learning, and specifically Artificial Neural Networks is needed for this project. Hence this review into the usage of Artificial Neural Networks for teaching grasping methods is kept brief.

Convolutional Neural Networks

There has been an increasing trend to use Artificial Neural Networks (ANNs) for solving problems in a variety of fields, such as classification, compression and other data processing, control, and robotics. They are loosely based on the human nervous system, in the way it consists of a high number of simple, yet highly interconnected units (neurons) working together to solve problems [36]. ANNs first gained popularity in the late 1980s/ early 1990s, however, their use cases were limited due to the relatively small computational resources at the time. In the last few years ANNs have experienced a renaissance as computational power has increased dramatically. Today they are an integral part in solving many data based problems.

Before investigating methods used to train robotic hands to grasp objects, it is necessary to look at convolutional neural networks (CNN), as they are the type of neural network most commonly used for computer vision problems [37]. A simplified representation of a CNN architecture called LeNet [38] is given in Figure 2-8.

[1] "The task in visual servoing is to use visual information to control the pose of the robot's end effector relative to a target object or a set of target features." [34]

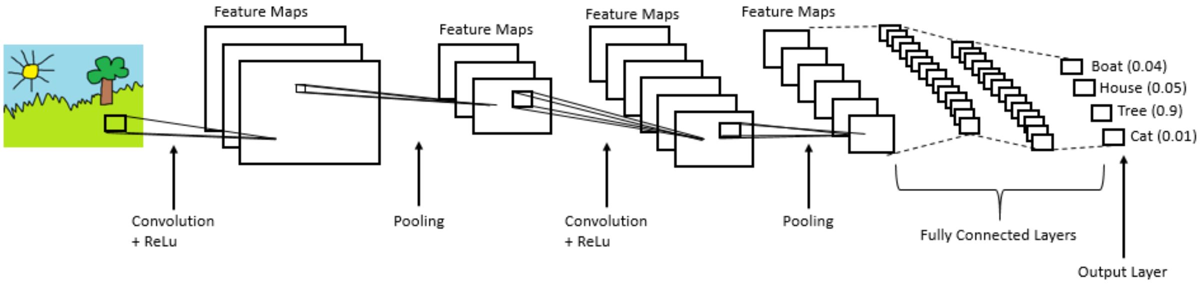

Figure 2-8: Simplified Representation of a CNN architecture based on the LeNet architecture (adapted from [37] and [39]).

A CNN can have different architectures but always includes the following steps:

1. Convolution

2. Non-Linearity (Rectified Linear Unit)

3. Pooling (Sub Sampling)

4. Classification (Fully Connected Layer)

In the convolution stage the input image is convolved with a range of filters. The filters are adjusted in training. The resulting feature maps are rectified by setting all negative pixel values to zero using the Rectified Linear Unit (ReLu) operation [39]. In the pooling or sub-sampling step the dimensionality of the rectified feature maps is reduced whilst retaining important information. This makes the feature maps less computationally expensive and also less susceptible to scale and translation. Convolution, rectification, and pooling steps can be repeated multiple times, extracting features from an input image. In the fully connected layer the extracted features are used for classification ([37], [39]).

Using Neural Networks for Teaching Grasping Methods

There are two papers we briefly want to look at, the first one by the Artificial Intelligence Laboratory of the University of Zurich on an adaptive learning mechanism for teaching grasping to a robotic hand [40]. The second paper is a recent publication by Google describing an algorithm to teach robotic grippers hand-eye coordination using CNNs [3].

Gómez et al. propose a Neural Network that adapts to the capabilities of a robotic hand, allowing different sensory information to be used as input. This information is then used to calculate motor activities to actuate the hand (see Figure 2-9). In theory this means the control algorithm can work for robotic hands with varying capabilities [40]. Note that in the setup described no visual information is used as input to the network.

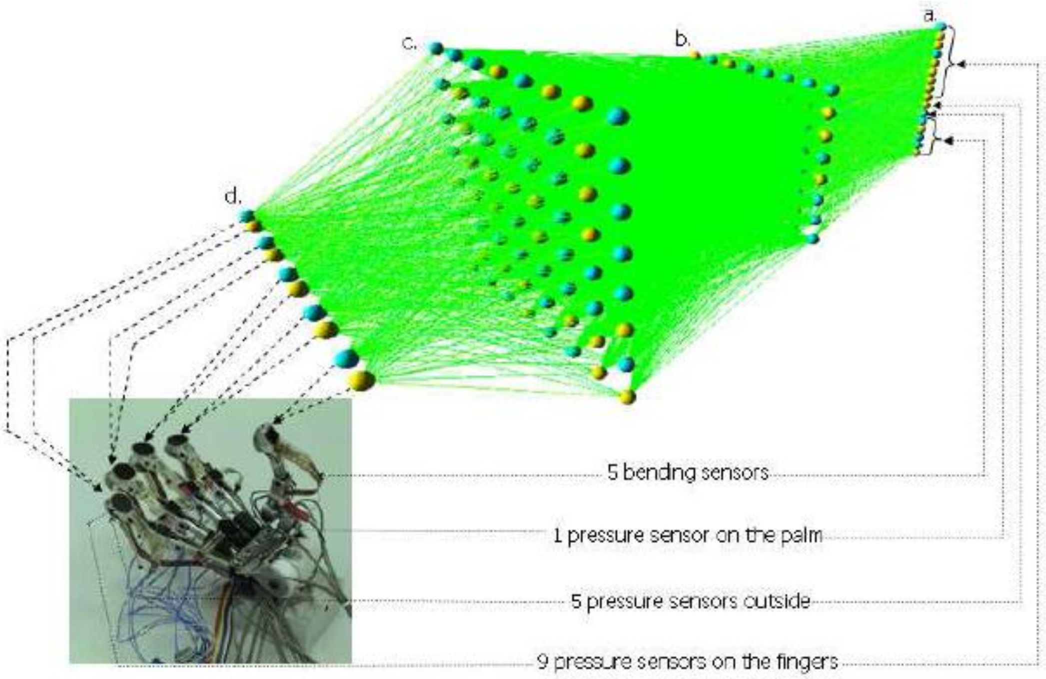

Figure 2-9: The Neural Network proposed by the Artificial Intelligence Laboratory at the University of Zurich. (a) sensorField, (b) hiddenField, (c) motorField, (d) motorActivities [40].

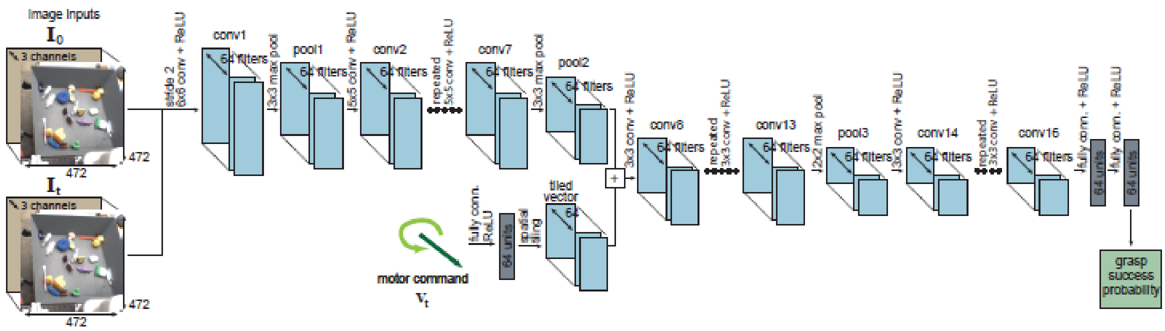

Figure 2-10: Google's CNN-based grasp predictor [3].

Levine et al. on the other hand use visual input only. Monocular images are fed into a large CNN that then predicts the probability of a successful grasp for a given motor command based on the visual information (see Figure 2-10). To train the CNN a dataset of over 800,000 grasp attempts from 6 to 14 different robots was collated over two months [3].

2.5. Summary

The differences between anthropomorphic and minimalistic robotic hand design were discussed. An anthropomorphic design was chosen for this project, as it provides more flexibility for grasping a variety of objects, and its design aesthetic is more suitable for human prosthetics than a two- or three-finger gripper. There are also a number of open source and hobbyist examples of low-cost anthropomorphic robotic hands available online.

Different robotic hands from the open source community, academia, and industry were introduced and their functionalities compared. The choice was made to design a 3D-printable hand as it is a flexible and low-cost way to create a hand prototype.

Next, sensory control was examined, focussing on tactile and visual sensors. It was proposed that combining tactile and visual sensory information could prove useful in grasp prediction.

Finally, a brief overview was given on the usage of Neural Networks for teaching grasping methods to a robotic hand.

3. Development to Date

To date a prototype hand has been designed and manufactured. The hand consists of a number of different 3D printed PLA plastic parts that are glued or bolted together. A Bowden system is used to transmit force from five servo motors to all five fingers. There are force sensors in each fingertip (apart from the little finger) and the thumb. The current servo motors do not yet have a potentiometer read out, but an appropriate replacement has been ordered. A camera will be permanently installed into the palm to act as a visual sensor. An Arduino Uno is currently used for controlling the servos and collecting sensory information. Once the correct servos have arrived this will be replaced with an Arduino Mega to provide sufficient analogue input pins. A PC is used for implementing high level control. See the following sections for a more detailed description of the system, hardware, low-level-, and high-level-control.

3.1. System Overview

Figure 3-1 is a simplified system diagram. Not yet implemented parts are shown in grey. Each force sensor module (FS 1+2 to FS 7+8) is connected from the fingertips to an Arduino. All five servos are controlled via the Arduino's PWM pins. The servo's potentiometer values are read and processed. The Arduino sends information to the high level controller which will be responsible for the choice of grasp to be used on an object.

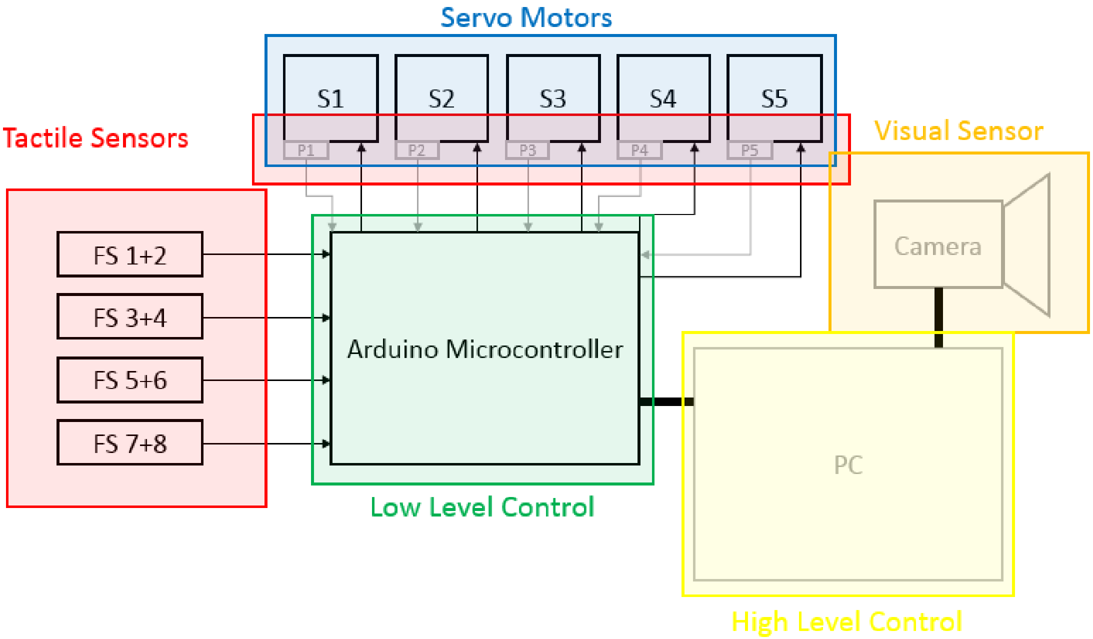

Figure 3-1: Simplified System Diagram. Low Level Controller controls servos and receives information from force sensors and servo potentiometers. Sensor data and camera frames are fed into High Level Controller.

3.2. Hardware Prototype

Design and Assembly of Hand Chassis

All parts for the hand were designed in Autodesk Fusion 360 [41] and printed in PLA plastic. All fingers consist of the same base parts (see Figure 3-2), only the base of the male and female part changes size according to the length of the specific phalanx. Phalanges are connected with joints rotating around 2mm metal rod. The finger assembly was tested on a printed finger base as can be seen in Figure 3-3. Next, a palm was designed with five male finger parts as the base for each finger and the thumb. Inside the palm there are guides for the Bowden system and space for the camera. Figure 3-4 shows the described palm design and the first fully printed and assembled hand.

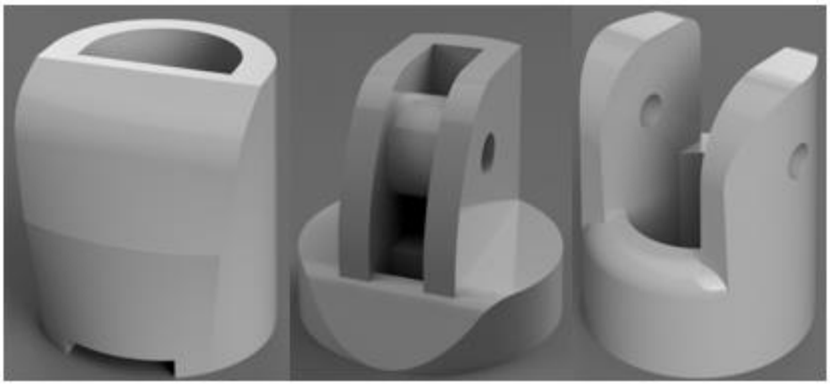

Figure 3-2: Finger Parts. Fingertip (left), Male Part (middle), Female Part (right).

Figure 3-3: Testing the finger design on a printed base.

Once the first hand was printed it needed to be actuated using five servo motors. The five servos are housed in a servo bed with guides which lead the tendons from the palm to the servos. The tendons are mounted onto each servo horn using two servo rings of different sizes, as it was found this gives the maximum turn with minimum backlash. The palm design needed to be altered at the wrist to allow for two M3 bolts to connect the palm to the servo bed. The current palm and servo bed design as well as their assembly can be seen in Figure 3-5, the servo rings used to connect each tendon to its servo is shown in Figure 3-6.

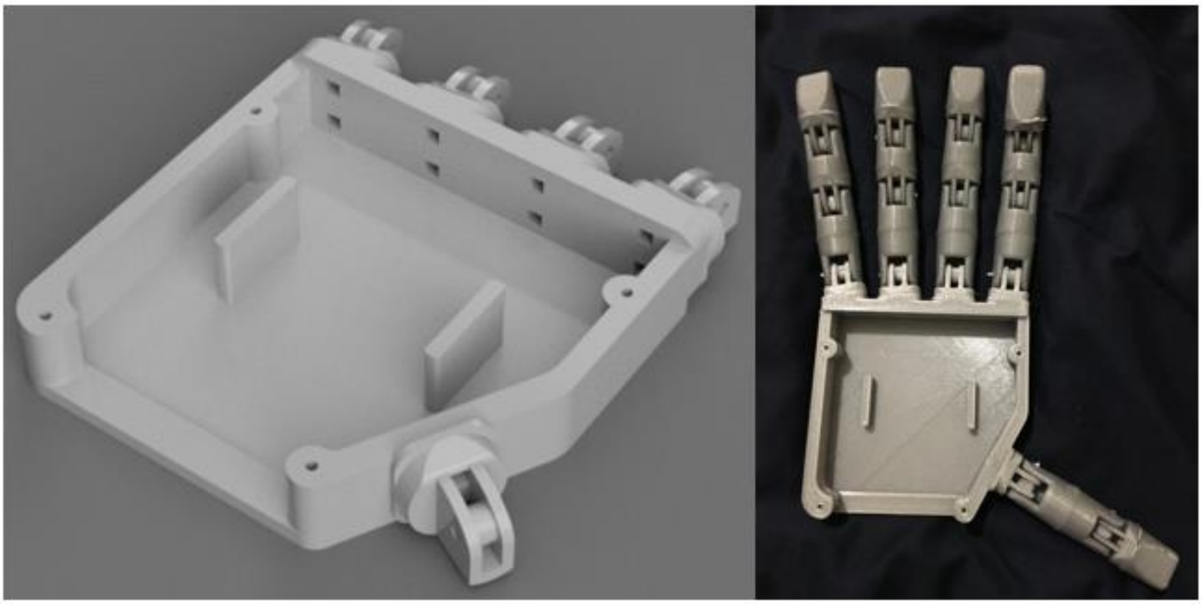

Figure 3-4: Initial Palm Design with Guides for Bowden System and Space for Camera (left). The First Printed and Assembled Hand (right).

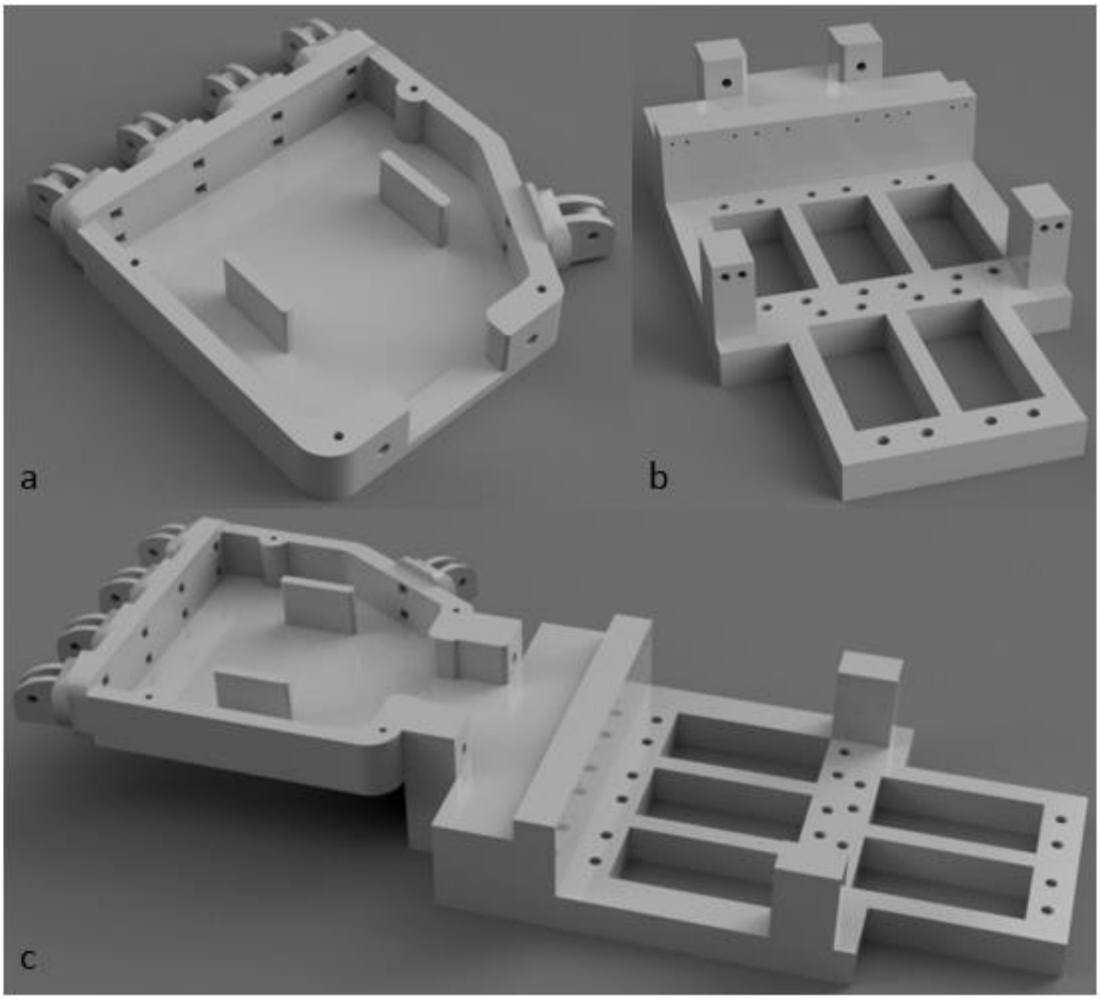

Figure 3-5: Current Palm (a) and Servo Bed (b) Design, and Assembly of Both (c).

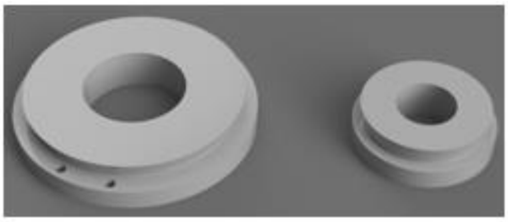

Figure 3-6: Servo Rings to Mount Tendons onto Servos.

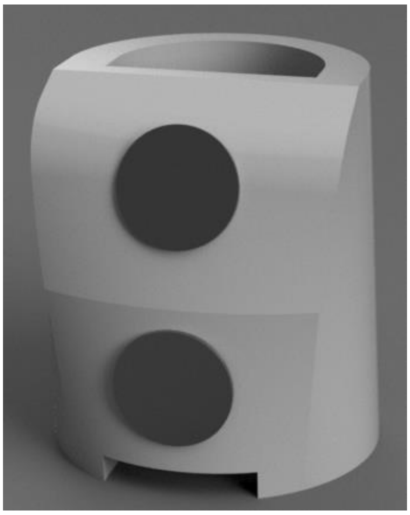

Figure 3-7: Position of Force Sensors on each Fingertip.

To complete the actuation and transmission part of the hand, two tendons were run through each finger, for extension and contraction respectively. The tendons run out of each finger into a Bowden system, through the palm and out through the wrist into the servo bed to their respective servo motors (see Figure 3-8).

After assembling and testing the actuation and transmission system the force sensors were placed onto the fingertips. Due to the cost of the sensors, their mounting is temporary using PVC insulation tape, until the sensors have been thoroughly tested in their capability of assisting the control algorithm. The current sensor positions can be seen in Figure 3-7. The sensors are covered by a layer of PVC tape. The whole fingertip is covered by a soft silicon coating. This gives the fingertips more grip and dexterity. The rest of the fingers are covered in a layer of PCV tape, to provide required stiffness (see Figure 3-8). This is a temporary solution and the tape will be replaced by a heat shrink coating.

The camera has not been permanently mounted inside the palm yet, as the visual information is directly fed into the high-level control algorithm, which has not been written yet.

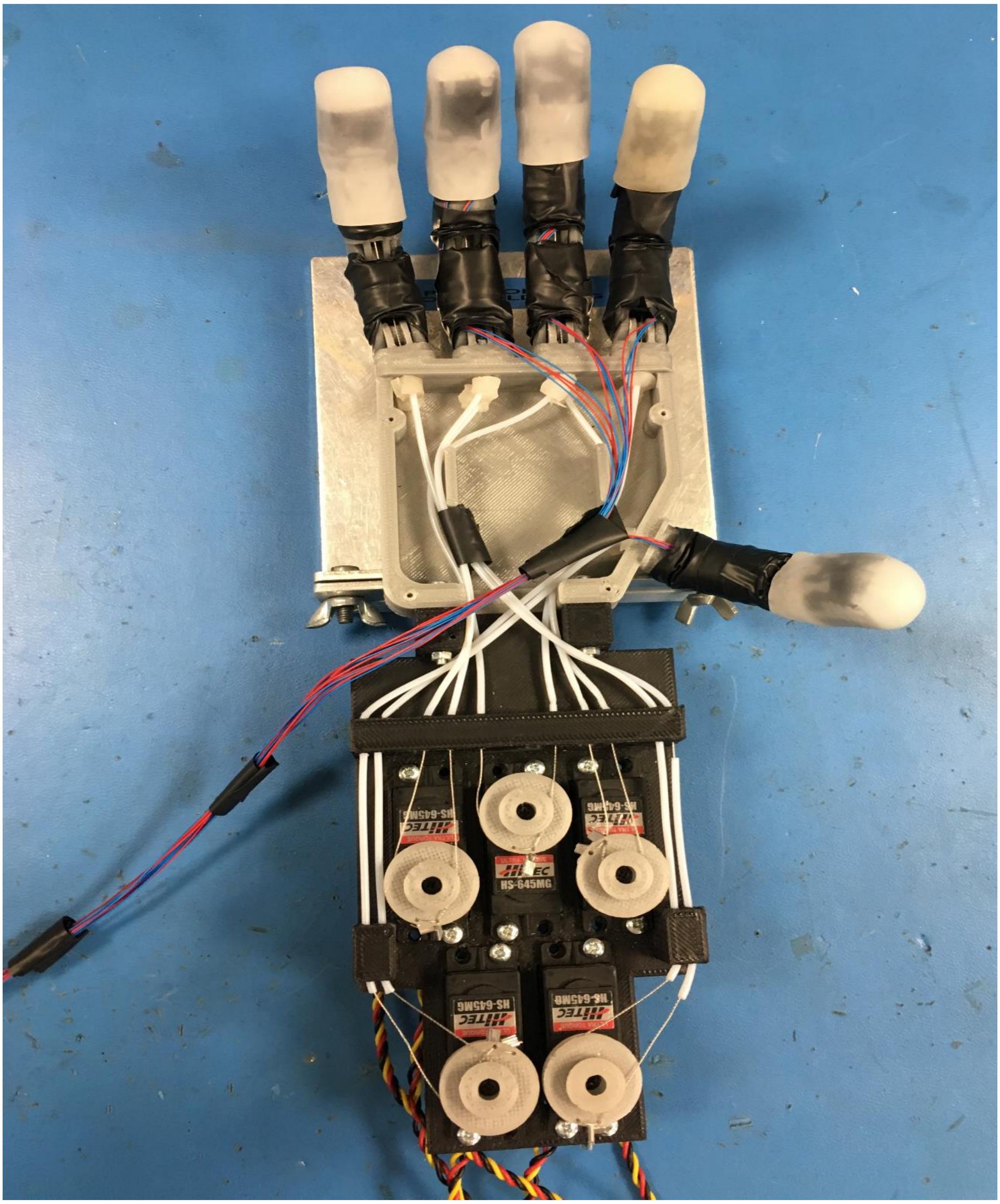

Figure 3-8: Current Assembly of the Hand Prototype (missing palm cover to show Bowden system and space for camera).

Electronic Hardware

System Overview

The electronic circuitry connecting all servos and sensors to the Arduino Uno microcontroller is currently on breadboard but is being designed into an arduino-compatible shield. Figure 3-9 shows the current setup. All force sensor units are directly mapped to analog pins on the arduino through a voltage divider. Each force sensor unit consists of the two force sensitive resistors per fingetip in parallel. The servos are driven by digital pins 5,6,9,10,11. The servos are powered with 6 Volts, hence a 5 Volt voltage regulator is needed to assure the Arduino is powered with 5 Volts. It was found that voltage spikes can upset the servos, hence a smoothing capacitor was added between power and ground.

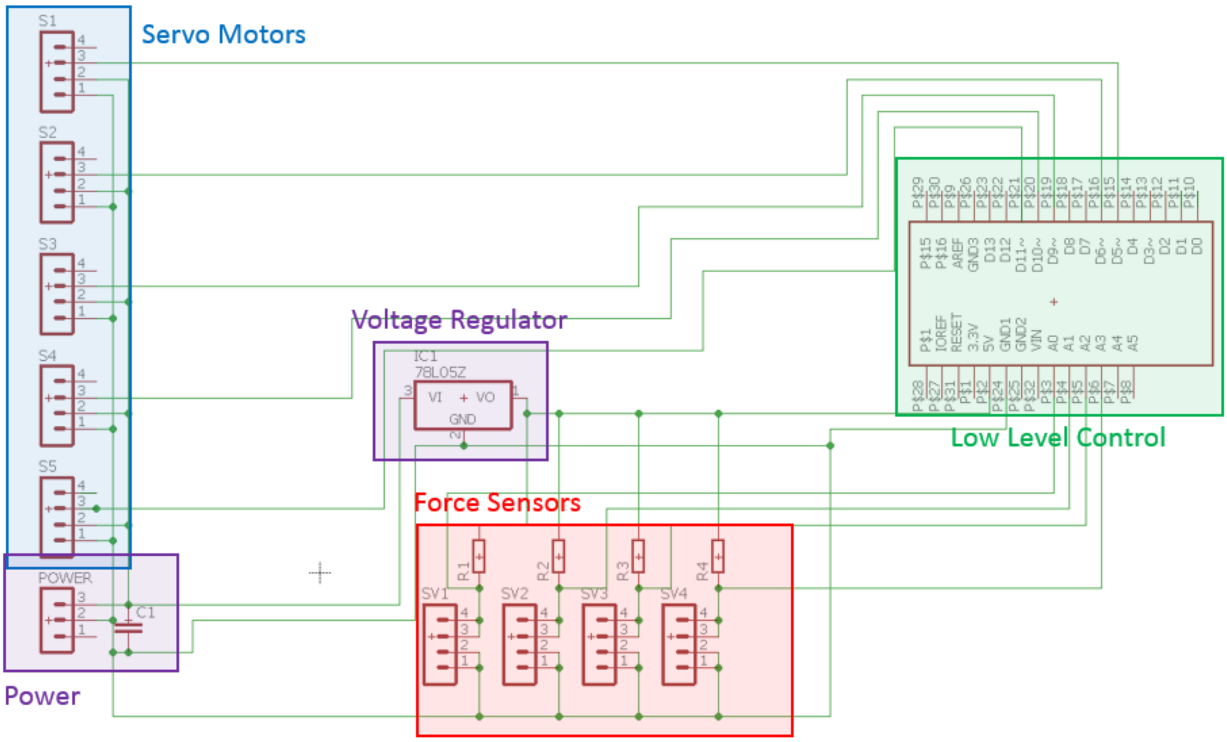

Figure 3-9: Electronic Circuitry of the current Prototype.

The Microcontroller

The current microcontroller board used to implement the low level control is the Arduino Uno, based on the ATmega328P microcontroller. It has 14 digital input/output pins, of which pins 3, 5, 6, 9, 10, and 11 can be used as PWM pins. Six analogue pins provide additional I/O. The microcontroller can be programmed via a USB connection [42]. This board is sufficient for the current setup. However, once the current servo motors are replaced with servo motors providing analogue feedback, the Arduino Uno will be replace by an Arduino Mega which is based on the ATmega1280. The Mega has 16 analogue I/O pins, which suffices for connecting the servo feedback, and force sensors [43].

3.3. Low Level Control

A low level control API was written based on the Arduino Servo library. In order to make code reusable in the future a modular approach was chosen. The API consists of two classes, Finger and Hand (see Figure 3-10 for a UML diagram). A Hand is created by giving the number of fingers (with a maximum of five). The constructor of Hand then creates the number of fingers specified and puts them into the fingers array. Hand then requires the pin numbers each finger is connected to on the Arduino, as well as the minimum and maximum servo values for each. This is necessary to assure that no servo is overdriven, snapping a tendon. These values are passed on to the appropriate fingers. A threshold for the force sensors can be set, which is calculated in the high level control. As the high level control software has not been written yet, the threshold step is for now ignored. The hand can then be opened and closed by calling execute_grasp() and release_grasp(). This calls the open and close functions of each finger, and their individual threshold checks. In testing, the hand is capable of opening and closing repeatedly and no bugs or issues have been encountered so far.

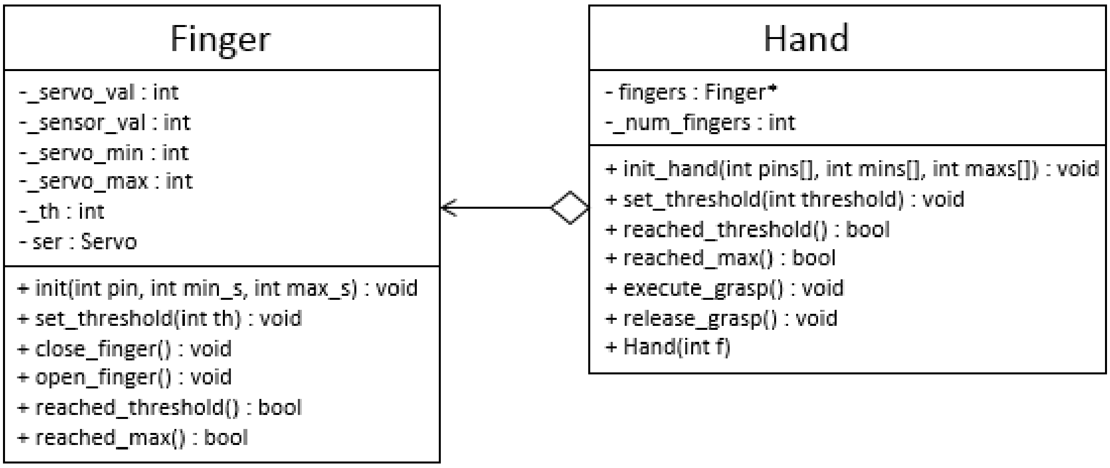

Figure 3-10: UML diagram of classes Finger and Hand, used in low level API.

3.4. High Level Control

More research into the field of Deep Learning and Neural Networks is needed to develop a specific algorithm to solve the problem of providing control to the robotic hand. The main idea at this stage is to train a Convolutional Neural Network to be able to predict the success or failure of a grasp attempt, and changing the motor settings accordingly. This would require the visual information captured by the palm camera. Additionally, we want to incorporate the force sensor and potentiometer information to add depth to the training dataset. The sensory data from the force sensor and servo potentiometers will be fed into an Artificial Neural Network, whose results will be added to the CNN. The CNN will also receive the current motor commands. Ideally, the CNN's prediction of success or failure will be fed back to the servo motors to adjust their position to yield a better chance of success. This would result in closed-loop iterative training. However, due to time constraints this might not be implemented, in which case the system will work based on single batch learning. See Figure 3-11 for a visual representation of both concepts.

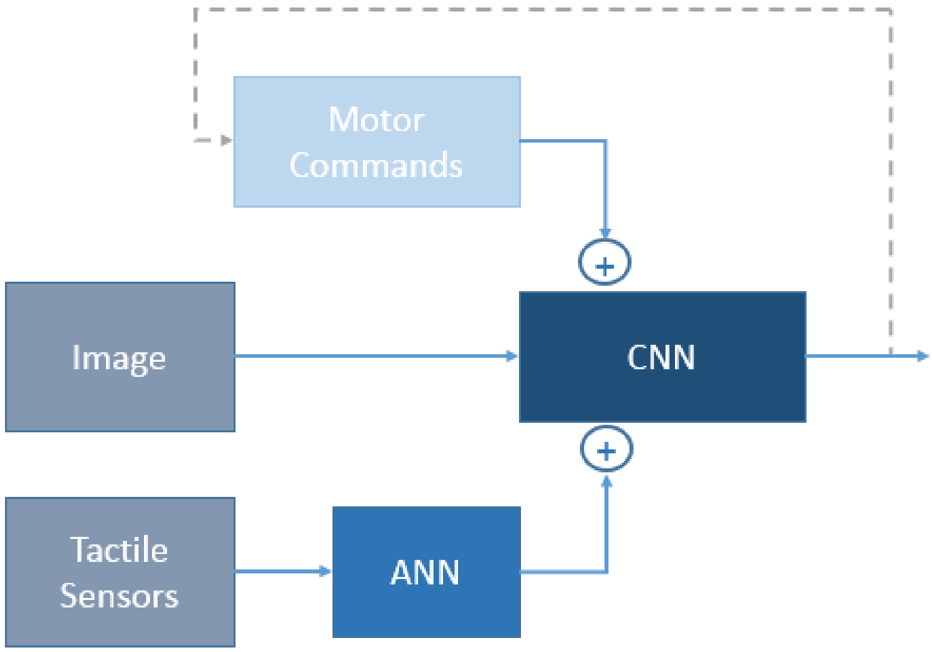

Figure 3-11: Proposed High Level Control System. Ideally the output of the system can be fed back to adjust the motor command, to implement closed-loop iterative training. If this cannot be implemented in time, the system will rely on single batch learning.

4. Future Work

4.1. Planned Deliverables

In order to meet the objectives of this project the following deliverables are deemed necessary.

1. A hand prototype that is flexible, controllable, and durable.

   - A modular, easily re-printable body, which can be used in experiments, data collection, and demonstrations.

   - Modular, clean, and compact electronic hardware that is safe and does not hinder the operation of the hand, e.g. no wires that could get caught in a moving finger.

2. A low-level API that interfaces with the underlying hardware.

   - Needs to abstract away the complexities of the underlying hardware in order to be east to use.

   - Needs to be thoroughly tested to assure the correct and safe working of the hand.

   - Needs to be generic enough to be reusable in future work.

3. A high-level control algorithm.

   - Harnesses recent developments in computer vision and image processing.

   - Employs a convolutional neural network to identify objects and develop a grasping strategy.

   - A dataset needs to be generated for the training phase.

4. Sufficient documentation to make work reproducible.

The first two objectives have already been met, and will only see some minor additions and changes. A significant step has been taken towards deliverable four, by writing this report and documenting progress on the hardware prototype through photos and videos. The focus from now on lies in working towards developing and implementing the high level control of the hand.

Future practical steps are to change the current servos with the ordered analogue feedback servos, and integrate the servo potentiometers into the electronic circuitry. This will also involve changing the Arduino Uno Board to an Arduino Mega. Once the circuit schematic is finalised a shield can be designed to fit the Arduino Mega Board. The shield will make the design more compact and safer. The force sensors and camera can be mounted permanently once the electronic hardware design is finalised and tested with a high level and low level control system. For this the high level control system needs to be designed and implemented, and the low level control API rewritten to include the servo potentiometers and interface with the high level control. The PVC tape providing necessary stiffness to the fingers will be replaced with het shrink tubing. The Neural Network in the high level control will be trained with a dataset that needs to be collated.

4.2. Project Plan

Detailed Gantt Charts have been created near the beginning of the project for both semesters. Due to their size, they can be found in Appendix B for readability. They take into account revision and exam times, in which work will proceed more slowly. Holidays are also accounted for, e.g. the Easter break, in which a lot of the dissertation write up will take place. Each deadline is clearly marked and tasks are broken down into subtasks. Despite a detailed plan, a project is never risk-free. A breakdown of the main risks to this project is given in Table 4-1 including their possible impact on the project and strategies to avoid the problem or deflect its consequences.

Risk Description	**Consequences**	**Impact**	**Precautionary Measures**
Hardware might break or encounter severe malfunctions.	Time will be taken up to fix/debug, which will delay rest of the project. In worst case no working prototype in time for data acquisition.	Severe.	Start development of hardware as early as possible, and test in parallel. Make design of body and electronics modular, so broken parts can be replaced quickly. Use resources readily available in undergraduate laboratories, such as Arduinos, and 3D printer.
Lead times for components and PCB.	Might delay the project, as prototype cannot be finished without all parts.	Moderate.	Work with alternatives, e.g. test with available servos, until correct servos arrive, use breadboard and jumper wires until Arduino Shield available.
Data acquisition for training of CNN.	Not enough data might be collected in time to produce any valuable results for dissertation.	Severe.	Start development time early to allow a lot of time dedicated to training. Automate as much as possible.
Might go over budget.	Prototype might not be finished as budget is exceeded.	Low.	Prototype is almost finished at this point and most parts paid for. If more cost occurs, alternative means of funding need to be found.

Table 4-1: Risks to Project, their Impacts and Consequences, and Measures to avoid them.

4.3. Estimated Cost

A breakdown of the estimated cost for this project in terms of equipment and materials can be found in Table 4-2. A lot of the development cost can be absorbed by using materials are already available, or are free to use in EEUG laboratories. However, there is still significant cost associated with the servomotors, sensors, and the Arduino Mega Board.

Equipment and Materials	**Estimated Price**	**Source**
Arduino Uno Board	£17	EEUG labs (free of charge)
Arduino Mega Board	£31	Robotshop UK
PCB for Shield	~£6	EEUG labs
10x Force Sensors	£47.71	Rapid
6x Servos with analogue feedback	~£75	Adafruit
Camera (webcam)	£7.99	Amazon
3D-printed parts	£15	EEUG labs
Tendons (0.5mm steel wire 5m)	£11.25	Prime miniatures
Metal Ferrules 0.5m 75pcs	£6	Prime miniatures
Silicon Fingertip Coating (silicon toe cap protectors) 10 pcs	£5	Amazon
PVC Tape	~£1	EEUG Labs (free of charge)
Heat Shrink Tubing	~£2	EEUG Labs (free of charge)
2mm Metal Rod	~£4	Already had
Superglue	~£2	EEUG labs
Epoxy	~£6	Screwfix/EEUG labs
RCL components	~£5	EEUG Labs (free of charge)
6x No Feedback Servos	~£100	Already had

Table 4-2: Breakdown of Estimated Cost for Equipment and Materials.

4.4. Final Report Structure

The structure for the final report is similar to this one. It introduces the project, and provides a literature review before presenting the methodology used. Instead of introducing future works, a discussion of results will take place. A conclusion reiterates key findings, followed by a bibliography and additional material in appendices. See below for details.

- Title Page
- Abstract
- Acknowledgements
- Introduction
- Literature Review
  - More information on machine learning in robotics.
- Methodology
  - Explain and justify development to date, research methods, data collection, experiments.
- Results and Discussion
  - The state of the hand and its abilities.
  - Interpretation and evaluation of results and research.
- Conclusion
- (Future Work)
  - Give an outlook on what could be done in the future building on the research undertaken in this project.
- References and Bibliography
- Appendices

5. Summary

This report introduces the Smart Hand Project, whose aim is to develop a software and hardware system for semi-automatic grasping of objects. The hardware in the form of a working robotic hand prototype, will be used to collect grasping data and run experiments, training a software control algorithm to make grasp choices.

A literature review discusses the differences between anthropomorphic and minimalistic hand design, and presents different exemplary hands from different backgrounds, namely the open source community, academia, and industry. Different types of sensors are discussed before briefly exploring the usage of Neural Networks in training methods for robotic hands.

The current state of development of the robotic hand prototype and its control is presented. Albeit some aspects of the hardware still need to be changed or implemented, it development is advanced enough to start work on designing and implementing a high level control algorithm. This algorithm is still in its early planning stage. Currently the use of a Convolutions Neural Network in conjunction with an Artificial Neural Network is favoured.

Future work is described, using a detailed project plan as a guideline. A detailed risk assessment points out potential problems, such as hardware failure and component lead times. Strategies to avoid these problems or deflect their impact are also presented. The cost of the project's materials and equipment is estimated and a structure for the final report is laid out.

To summarise, the hardware prototype is functional and close to full completion. Work on a high level control algorithm has begun. The aim is to start collating data and to run experiments using the hand prototype in the near future. Results will be presented in a final report.

6. Conclusions

The Smart Hand Project has thus far produced a working prototype that is close to completion, which measures up the expectations laid out in the project plan. It is now important to design and implement a high level control algorithm and start the collation of a dataset for experiments and training of the hand. A more thorough understanding of Neural Networks and Deep Learning is needed to achieve flexible and efficient high level control. However, the project so far is on track of its planned completion in May 2017.

7. References

[1] R. I. Association, “Unimate - The First Industrial Robot,” 2016. [Online]. Available: http://www.robotics.org/joseph-engelberger/unimate.cfm. [Accessed 25 11 2016].

[2] E. T. Zhe Xu, “Design of a Highly Biomimetic Anthropomorphic Robotic Hand towards Artifical Limb Regeneration,” University of Washington, 2016.

[3] P. P. A. K. D. Q. Sergey Levine, “Learning Hand-Eye Coordination for Robotic Grasping with Deep Learning and Large-Scale Data Collection,” Google, 2016.

[4] A. Bicchi, “Hands for Dexterous Maniplulation and Robust Grasping: A Difficult Road Toward Simplicity,” *IEEE Transactions on Robotics and Automation,* vol. 16, no. 6, pp. 652-662, 2000.

[5] H. Bos, “Evolution of Robotic Hands,” University of Twente, Twente, 2010.

[6] A. M. Antonio Bicchi, “Dexterous Grippers: Putting Nonholonomy toWork for Fine Manipulation,” *The International Journal of Robotics Research,* vol. 21, no. 5-6, pp. 427-442, 2002.

[7] M. A. a. R. P. D. Alba, “An Introductory Revisionto Humanoid Robot Hands,” in *Climbing and Walking Robots*, Berlin Heidelberg, Springer, 2005, pp. 701-712.

[8] e. a. M.S. Johannes, “An Overview of the Developmental Process,” *Johns Hopkins APL Technical Digest,* vol. 30, no. 3, pp. 2017-216, 2011.

[9] M. R. Cutovksy, “On Grasp Choice, Grasp Models, and the Design of Hands for Manufacturing Tasks,” *IEEE Transactions on Robotics and Automation,* vol. 5, no. 3, pp. 269-279, 1989.

[10] H. Kawasaki, “The Human Hand and the Robotic Hand,” in *Robot Hands and Mult-Fingered Haptic Interfaces - Fundamentals and Applications*, World Scientific, 2015, pp. 3-27.

[11] S. Narasimhan, “Dexterous Robotic Hands: Kinematics and Control,” MIT Artificial Intelligence Laboratory, Boston, 1988.

[12] S. R. Company, “Design of a Dextrous Hand for advanced CLAWAR,” Shadow Robot Company, London, 2003.

[13] T. E. &. H. K. Tetsuya Mouri, “Review of Gifu Hand and Its Application,” *Mechanics Based Design of Structures and Machines,* vol. 39, no. 2, pp. 210-228, 2011.

[14] "DLR-Hand II: Next Generation of a Dextrous Robot Hand," German Aerospace Research Center (DLR), Wessling, 2001.

[15] M. Bélanger-Barrette, "blog.robotiq.com: Why Use a Robot Gripper with 3 Fingers?," Robotiq, 2 July 2015. [Online]. Available: http://blog.robotiq.com/why-use-a-robot-gripper-with-3-fingers. [Accessed 3 Deecember 2016].

[16] G. P. S. Victor J. Johnson, "Kinematic and Dynamic Analyses of the Stanford/ JPL Robot Hand," Sandia National Laboratories, Albuquerque, Livermore, 1987.

[17] e. a. Vinicio Tincani, "Velvet Fingers: A Dexterous Gripper with Active Surfaces," 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, Vilamoura, Algarve, Portugal, 2012.

[18] M. Szczys, "hackaday.io," Hackaday, 2016. [Online]. Available: https://hackaday.io/. [Accessed 30 12 2016].

[19] E. J. Wilhelm, "Instructables," Instructables (Squid Labs), 2016. [Online]. Available: http://www.instructables.com/. [Accessed 30 12 2016].

[20] MakerBot, "Thingiverse," MakerBot, 2016. [Online]. Available: http://www.thingiverse.com/. [Accessed 30 12 2016].

[21] "Open Bionics," Open Bionics, 2016. [Online]. Available: https://www.openbionics.com/about/. [Accessed 30 12 2016].

[22] O. Bionics, "Open Bionics: Ada Hand Kit," Open Bionics, 2016. [Online]. Available: https://www.openbionics.com/shop/ada. [Accessed 30 12 2016].

[23] G. Langevin, "InMoov," InMoov, 2016. [Online]. Available: https://inmoov.fr/. [Accessed 30 12 2016].

[24] G. Langevin, "InMoov: Hand and Forearm," InMoov, 2016. [Online]. Available: https://inmoov.fr/hand-and-forarm/. [Accessed 30 12 2016].

[25] H. Kawasaki, Robot Hands and Multi-Fingered Haptic Interfaces, Gifu: World Scientific Publishing Co Pte Ltd, 2015.

[26] S. R. Company, "Dexterous Hand," Shadow Robot Company, 2016. [Online]. Available: https://www.shadowrobot.com/products/dexterous-hand/. [Accessed 30 12 2016].

[27] "Shadow's Dexterous Hand | Hackaday," Hackday, 24 5 2005. [Online]. Available: http://hackaday.com/2005/05/24/shadows-dexterous-hand/. [Accessed 31 12 2016].

[28] C. Willis, "Android Hands," [Online]. Available: http://www.androidworld.com/prod76.htm. [Accessed 1 1 2017].

[29] "Single Tact: The Mirco Force Sensor," Pressure Profile Systems, Inc., 2016. [Online]. Available: https://www.singletact.com/. [Accessed 1 1 2017].

[30] I. Electronics, *FSR 400 Data Sheet,* http://cdn.sparkfun.com/datasheets/Sensors/ForceFlex/2010-10-26-DataSheet-FSR400-Layout2.pdf.

[31] Spectrasymbol, *Flex Sensor Datasheet,* Salt Lake Cuty: Spectrasymbol.

[32] I. S. I. Inc., *Stretch Sensor Datasheet,* Staten Island NY: http://www.robotshop.com/media/files/pdf2/stretch.pdf.

[33] B. Earl, "About Servos and Feedback," Adafruit, 4 5 2015. [Online]. Available: https://learn.adafruit.com/analog-feedback-servos/about-servos-and-feedback. [Accessed 1 1 2017].

[34] e. a. Seth Hutchinson, "A Tutorial on Visual Servo Control," *IEEE Transactions on Roborics and Automation,* vol. 12, no. 5, pp. 651-670, 1996.

[35] "RobotShop," RobotShop inc., 2016. [Online]. Available: http://www.robotshop.com/en/. [Accessed 1 1 2017].

[36] D. S. Christos Stergiou, "Neural Networks," 1996.

[37] R. Fergus, "Neural Networks," New York University, 2015.

[38] e. a. Yann LeCun, "Convolutional Networks for Images, Speech, and Time-Series," in *The handbook of brain theory and neural networks,* Cambridge, MA, USA, MIT Press, 1998, pp. 255-258.

[39] ujjwalkarn, "An Intuitive Explanation to Convolutional Neural Networks - the data science blog," The Data Science Blog, 11 8 2016. [Online]. Available: https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/. [Accessed 5 1 2017].

[40] e. a. Gabriel G´omez, "An adaptive learning mechanism for teaching a robotic hand to grasp," 2006.

[41] "Cloud Powered 3D CAD/CAM Software for Product Design | Fusion 360," Autodesk, 2016. [Online]. Available: http://www.autodesk.com/products/fusion-360/overview. [Accessed 2 1 2017].

[42] "Arduindo - ArduinoBoardUno," Arduino, 2017. [Online]. Available: https://www.arduino.cc/en/main/arduinoBoardUno. [Accessed 2 1 2017].

[43] "Arduino - ArduinoBoardMega," Arduino, 2017. [Online]. Available: https://www.arduino.cc/en/Main/arduinoBoardMega. [Accessed 2 1 2017].

[44] M. Guelker, "machinedesign.com: The difference between robotic grippers with parallel, three-finger, and angled designs," Penton, 1 September 2011. [Online]. Available: http://machinedesign.com/robotics/difference-between-robotic-grippers-parallel-three-finger-and-angled-designs. [Accessed 3 December 2016].

Appendix A

This section demonstrates the current state of the hand prototype.

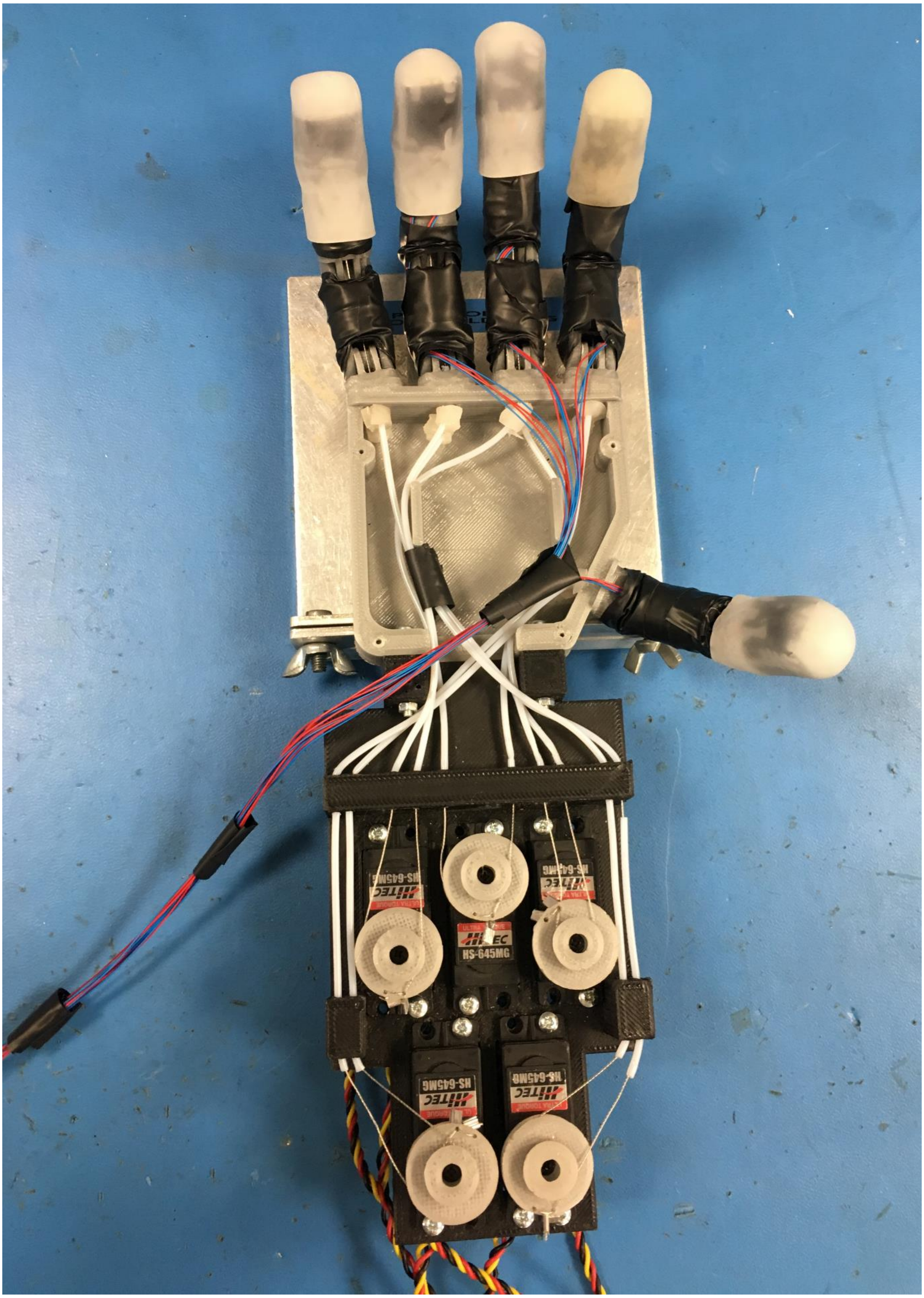

Figure 0-1: Current Assembly of Hand Prototype. The covering plate is removed to show Bowden system and space for camera.

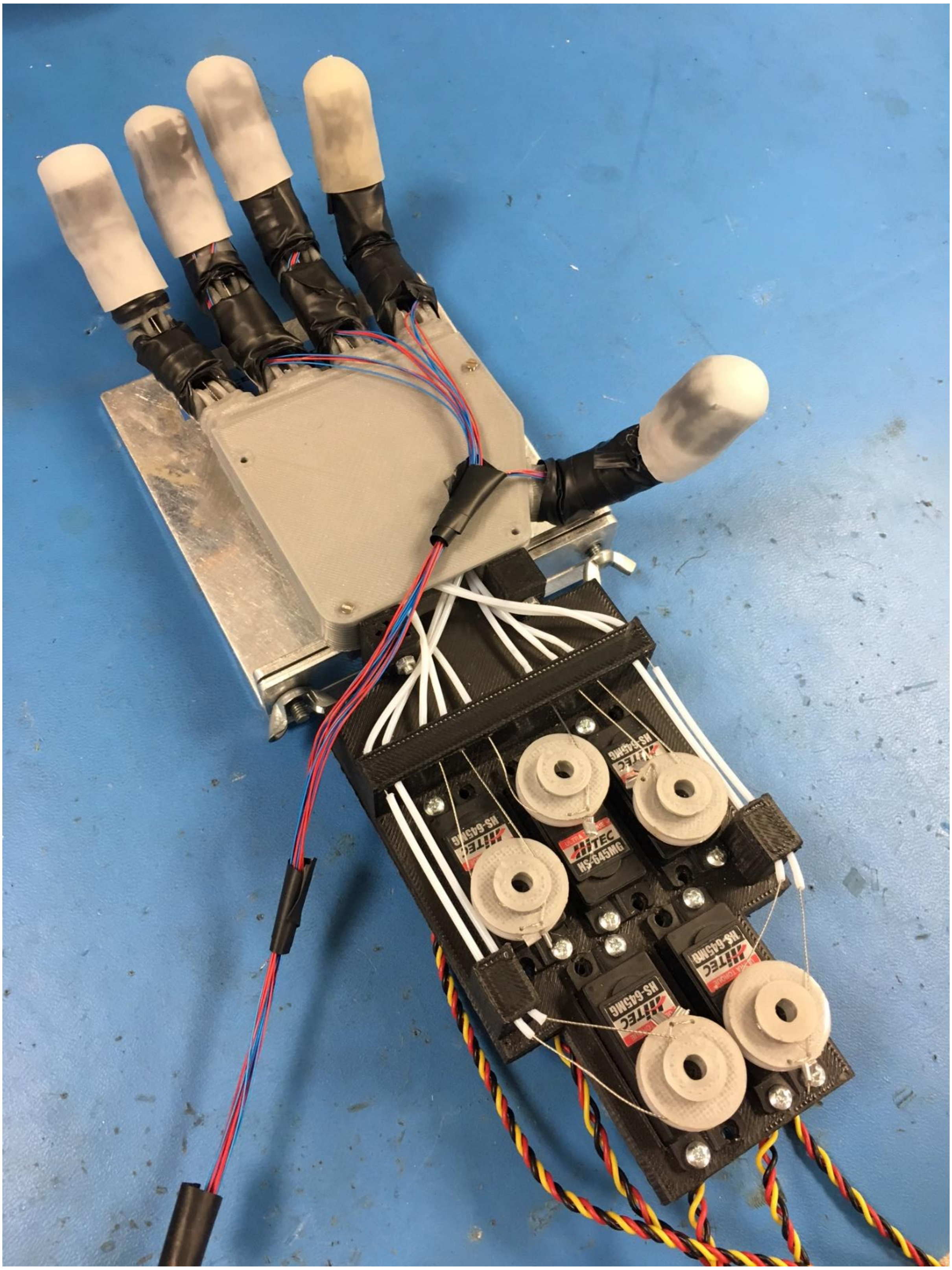

Figure 0-2: The Prototype Hand in its current assembly (without camera).

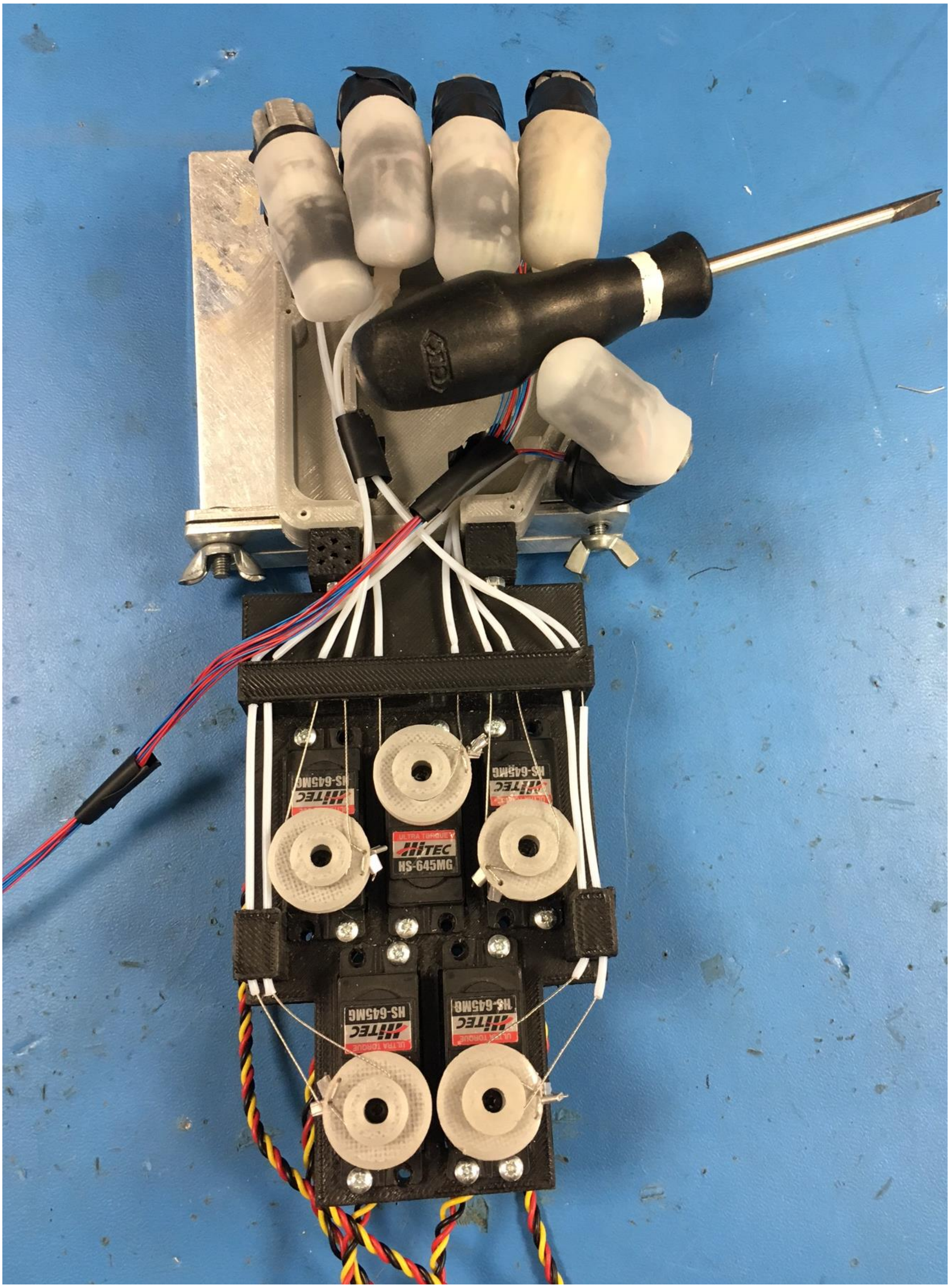

Figure 0-3: The Hand Grasping a Screwdriver.

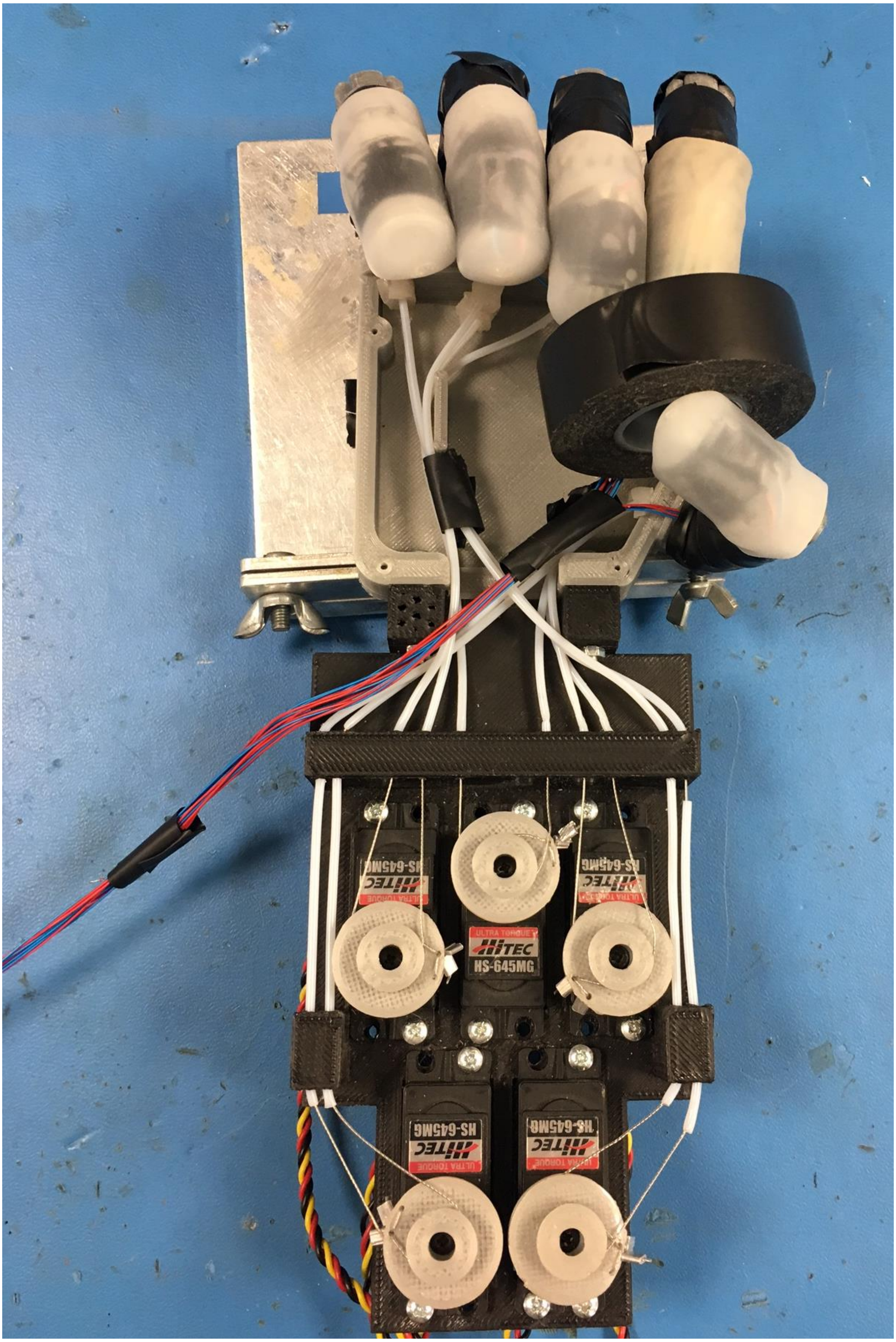

Figure 0-4: The Hand Grasping PVC Tape.

Appendix B

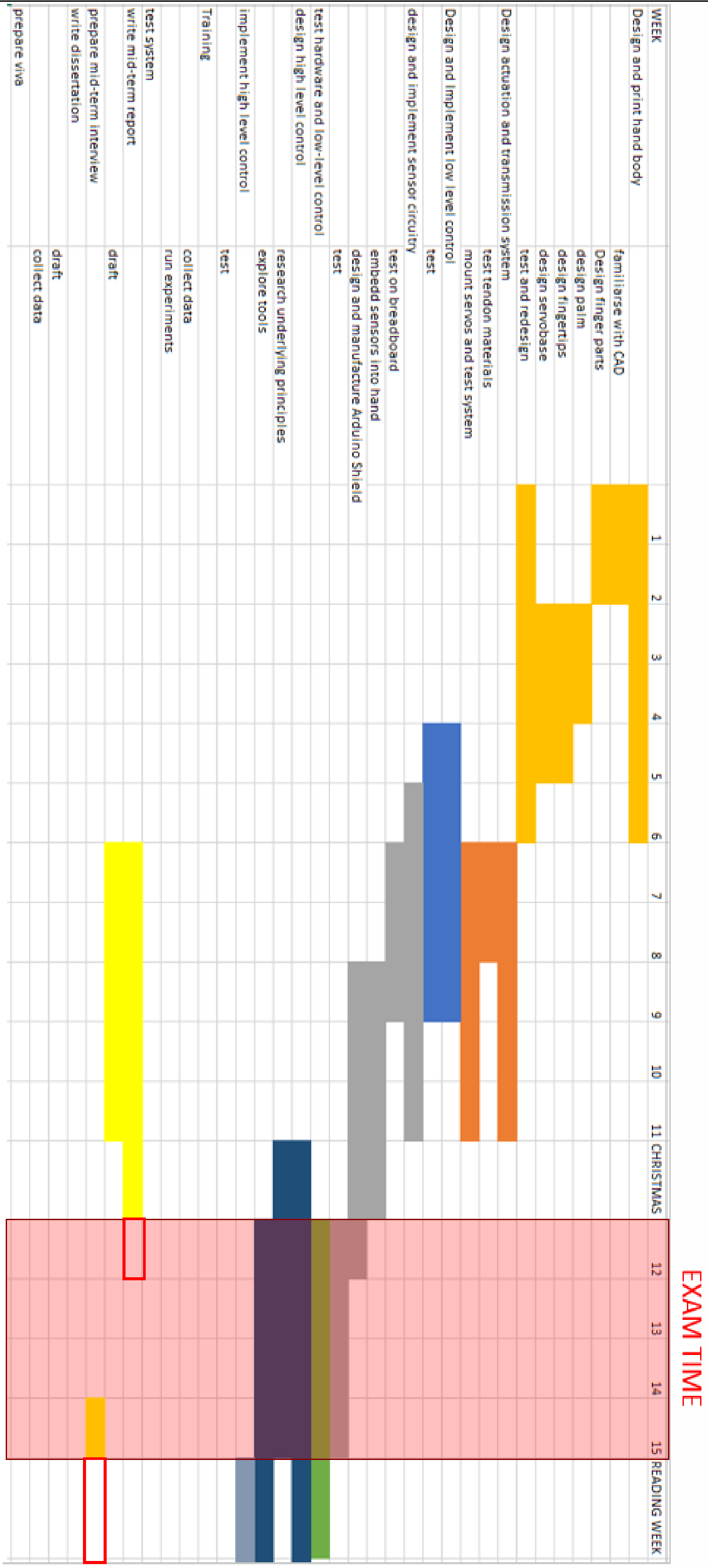

Figure 0-1: Semester 1 Gantt Chart.

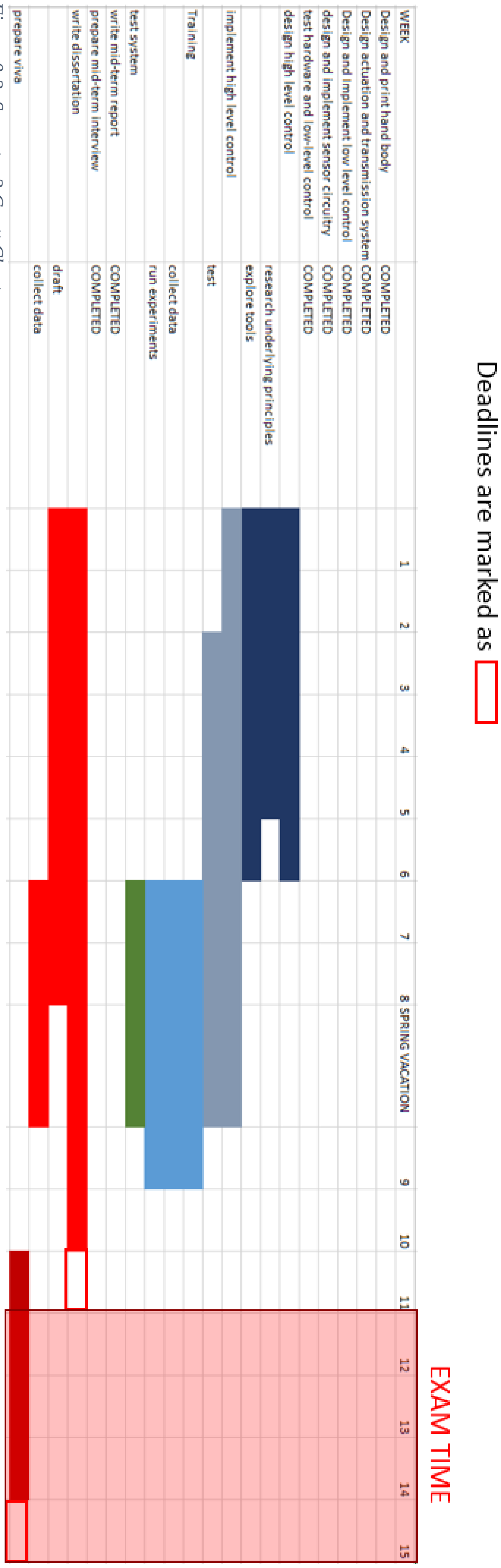

Figure 0-2: Semester 2 Gantt Chart.