Sensor Controlled Robotic Hand with Haptic Feedback

Law Chor Hon

The University of Hong Kong

April 15, 2017

COMP 4801 Final Year Project

Supervisor: Dr. Y.K. Choi

FYP Account: fyp16036

FYP Website: http://i.cs.hku.hk/fyp/2016/fyp16036/

Abstract

Many different types of Robots are invented in order to perform dangerous tasks in place of humans. However, in many cases, human workers are still being sent to hazardous environments, seeing as how current robots are not capable of performing complicated tasks. This paper describes the design, implementation and evaluation of the "Haphand", a sensor controlled robotic hand that provide haptic feedback. Control over a robotic hand would obviously be most intuitive to the user when actual hand gestures and movements are closely mirrored by the robotic appendage, and this could be achieved with the use of sensor gloves. Haptic feedback also provides more information to the user other than mere visual output, and these features in tandem will allow the robotic hand to handle far more complicated tasks and replace humans in more dangerous scenarios.

Acknowledgement

I would like to express my deep gratitude to Professor Loretta Choi, my research supervisor for her patient guidance, contribution in stimulating suggestion, encouragement and granting me access to the HKU maker lab. Her support assisted me a lot in completing this report. I would also like to thank Mr. David Lee for his advice and assistance in helping my progress on schedule and teaching me 3D modeling techniques.

Table of Contents

List of Figures

List of Abbreviations

Abbreviation	Explanation
The Haphand	The sensor controlled robotic hand with Haptic Feedback
PLA	Polylactic Acid
ABS	Acrylonitrile Butadiene Styrene
FSR	Force sensitive resistor
PWM	Pulse width modulation
ADC	Analog to Digital Conversion

1 Introduction

Robotics is a rapidly growing field in the past few decades, and the potential uses and functionality of robots have been researched substantially in variety of fields. In recent years, robots are used to perform tasks in dangerous environment in place of humans. For example, many robots have been designed and built to clean up radioactive contaminated places like nuclear facilities, defuse bombs, search for survivors in unstable ruins, etc. However, these robots have a lot of limitations, as their functionality harshly limits flexibility and efficiency when compared to humans. With such restrictions in the application of robots, human workers remain essential for work in hazardous situations, in particular tasks that robots fail to perform satisfactorily. With such shortcomings in mind, the focus of this paper will be a sensor-controlled robotic hand with haptic feedback, the Haphand, which will mimic the gesture of the user wearing the sensor glove. This robotic hand is capable of performing complicated tasks with nearly the same dexterity as a human hand and thus can eliminate the need for human presence in hazardous locations to the greatest possible extent.

In the remainder of this paper, we shall first offer background information on the current robotics and their limitation, and then elaborate more on how this project can solve the challenges and improve upon current practices. Subsequently, we shall discuss the design and approaches for the implementation of this project. We will then mention the model design and 3D printing and finally the results and future development.

1.1 Background

Contemporary robots usually consist of moving components and one or more robotic arms. These robotic arms are designed for specific purpose. Pliers are attached to the robotic arms for defusing bombs, while a screwdriver might be substituted for maintenance of a machine in radioactive environment. The functionality of these robotic arms is limited to the very specific tasks they are designed for, and the design of controller places another shackle on the flexibility of robots. In most cases, the operator obtains information about the surrounding environment solely through attached cameras, sorely restricting the channel of receiving information.

Several pitfalls emerge from current practice, for instance, unknown challenges are often faced when performing missions in hazardous places. Having robotic arms that are designed for too specific a purpose limit the functionality and flexibility of robots. The robots cannot perform complicated operations and tackle every eventuality. In Fukushima Daiichi nuclear disaster, human workers were sent to stabilize the reactors instead of sending robots because current robots are not qualified for these jobs in terms of functionality and efficiency. Second, the reception of surrounding environment is restricted to sight. The operator cannot feel what the robots touch. Some tasks should be effortless with tactile cues like cutting a wire However, without the pressure feedback, the operator need to apply constant visual confirmation. This increases the cognitive load of the operator and lowers the efficiency and functionality of the robots. Finally, using traditional controllers to control a robotic hand is counter-intuitive and hence necessitates lengthy training on the part of the operator. The effectiveness of robots is also largely dependent on the operator’s technique in handling the controller and ultimately the design of the controller.

1.2 Objective

In this project, the robotic hand will mimic the gesture of operator’s hand through a pair of sensor gloves. The robotic hand will have adequate strength to hold light instruments and objects but strength will not be the focus of this project. This project will focus on the flexibility and functionality of the robotic hand. The ability to handling different instrument will ensure the multi-functionality of the robots.

The robotic hand will have measure on pressure bearing and provide haptic feedback to the sensor glove so operator can feel the same. Operator can control the robotic hand to perform tasks even with no visual information. The glove will measure the force that operator applies and the robotic hand will apply a similar force. This is important in performing precise tasks such as holding fragile objects like eggs.

Wearable sensor glove instead of traditional controller will be used in this project. This will provide an intuitive way to control the robots. The robotic hand will move in the way that the operator move and consistently smoothen the learning curve of using the robotic hand.

2 Related Works

2.1 InMoov Robotic Hand

Figure 1: InMoov 3D life-size robot

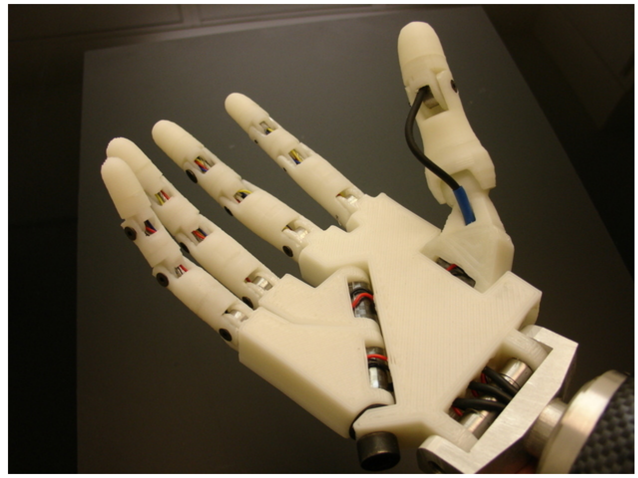

Figure 2: InMoov 3D robotic hand

InMoov is an open source humanoid robot that can be 3D printed and controlled by Arduino microcontroller. The robotic hand of InMoov is controlled by five servomotors and cables. This is the skeleton used in this project but it is not equipped with haptic feedback.

2.2 Dextrus Hand

Dextrus Hand is a 3D printed robotic hand that is designed to be control by two EMG sensors by placing them on muscles. In Dextrus Hand, different fingers are controlled with a separated string and servomotors which makes the hand can fit to the object that it griped. In contrast, some robotic hands move the fingers in synchronous way which all fingers are controlled by only one motor and string. The gesture that these hands can make are limited. It is not equipped with haptic feedback. There is no movable part in the palm which

Figure 3: Dextrus Robotic Hand

limits its flexibility. Figure 4 have shown the difference between Dextrus Hand and synchronous Hand.

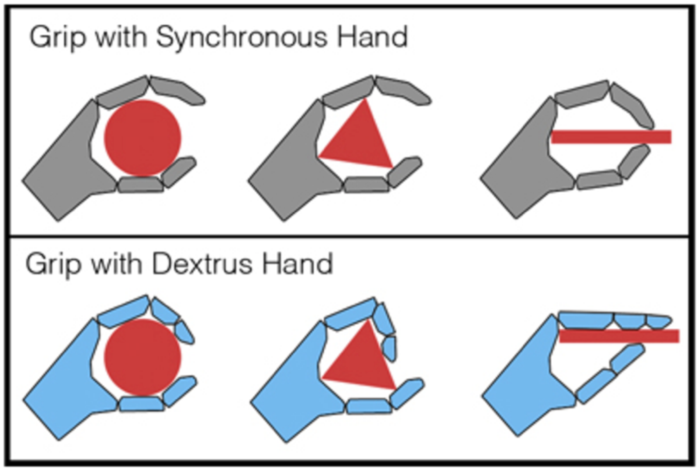

Figure 4: Grip with Synchronous Hand and Dextrus Hand

3 Designs

3.1 Abstract

In this section, the detail design of the Haphand will be explained. First, we will go through the components of the two main parts in the Haphand, namely robotic hand and sensor glove. Then the working principle of the Haphand including how to control the robotic hand and the way of providing haptic feedback will be described. Finally, the circuit design and program design will be mentioned.

3.2 Components

The Haphand is composed of robotic hand and sensor glove. Their components will be described in this section.

3.2.1 Robotic hand

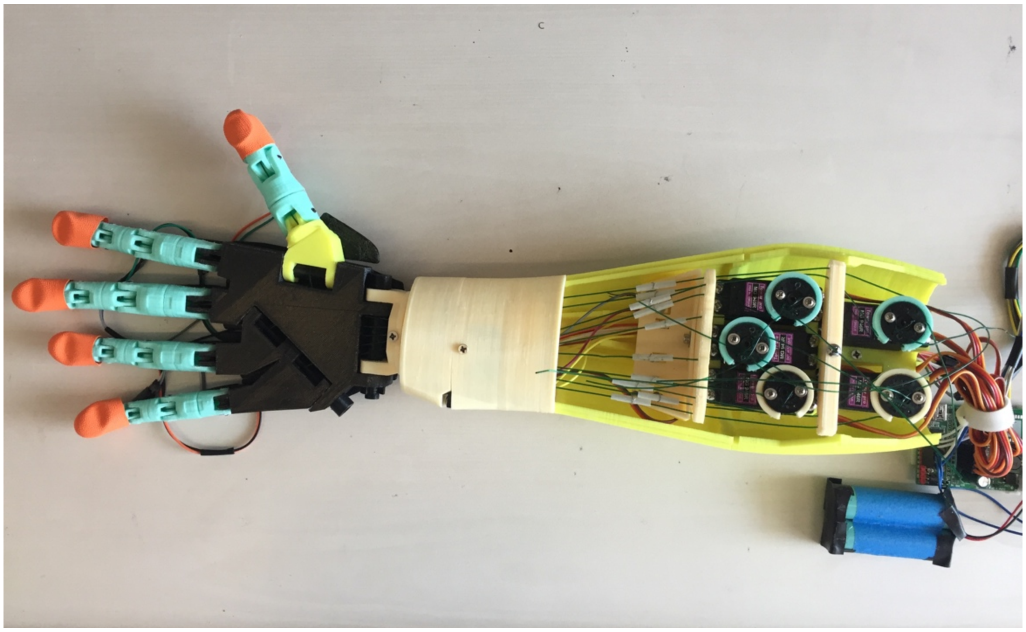

Figure 5: The Haphand Robotic Hand

3.2.1.1 Skeleton

All the skeletons of robotic hand are 3D printed with the design from InMoov since the design of robotic hand is beyond the scope of this project.

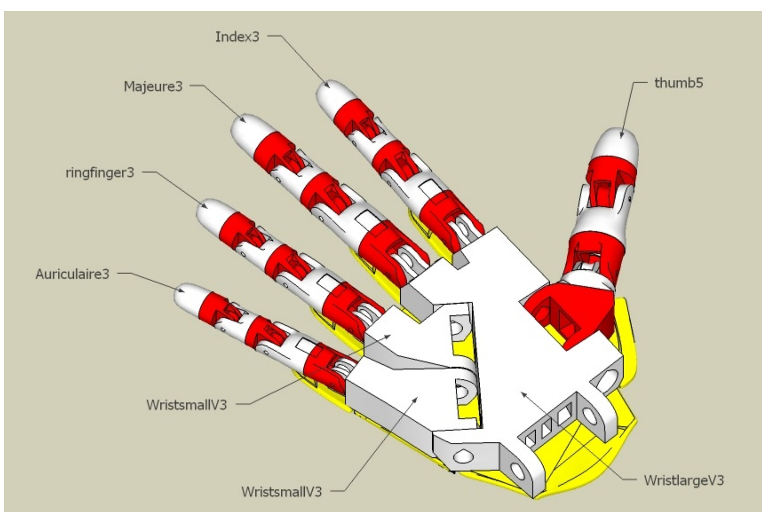

Figure 6: InMoov Robotic Hand (hand)

Figure 6 shows the structure of the robotic hand. It consists of finger parts and palm parts. Palm consists of 2 parts which gives more flexibility than palm that is made of one whole parts. This also provide a better grip to the object and better simulate the movement of human hand. This is one of the reason InMoov hand is chosen as parts of the Haphand. The fingers excluding the thumb consist of 6 parts including 3 knuckles which is the same as the human hand.

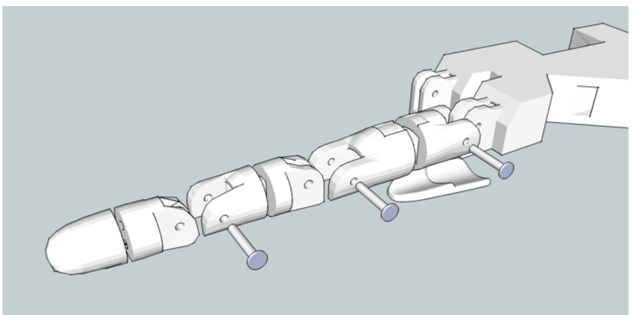

Figure 7: InMoov Robotic hand (finger)

Figure 8: Fingers with rubber cover

However, in this design the movement of the finger can only be controlled in the fingertip, the knuckles are connected using bolt and cannot be controlled individually. This is a compromise between flexibility and complexity.

As shown in figure 7, rubber cover is installed on the fingers to provide a better griping force and protect the pressure sensor.

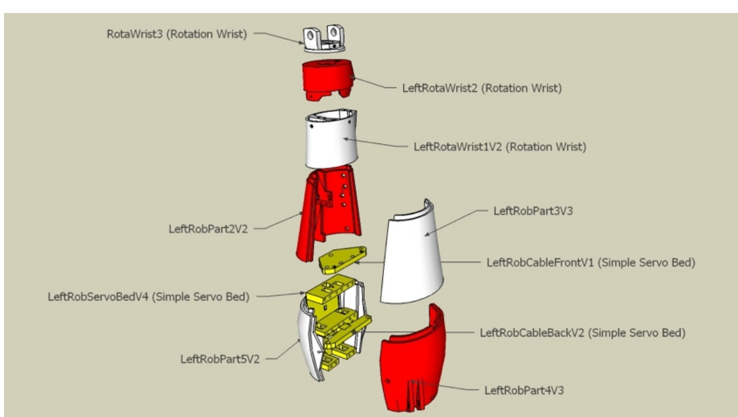

Figure 9: InMoov Robotic Hand (wrist and forearm)

Figure 9 shows the structure of the wrist and forearm of the robotic hand. It helps supporting the robotic hand and keeping the servo motors and servo bed. It also directs the wire from the robotic hand to the servo motors.

All parts of the the InMoov hand can be 3D printed including the bolts. This provide a convenient and cheap solution to Haphand but it still requires assembling of the parts. This is the second reasons InMoov hand design is chosen. More details will be mentioned in Part 4 Model design and 3D printing.

3.2.1.2 Servo Motors

Tower Pro MG946R Servo Motor is used in the robotic hand to control the movement of fingers. It can rotate from 0 to 180 degree and provide torque up to 13 kg. The operating voltage is 4.8V to 6.0V. It has fast rotating speed with 0.17sec/60^{o}. There are 3 wires, red and brown one is for power supply while orange one is for communication. MG946R is used as it has newer PCB and IC control system which makes it more accurate compare to Tower Pro MG945.

Figure 10: Tower Pro MG946R Servo

3.2.1.3 Microcontroller

GSTduino is used as the microcontroller in the robotic hand. It is modified based on Arduino Nano by The University of Hong Kong. It provides 14 digital I/O pins, 8 analog input pins. It can be powered via micro USB connection or battery connector. Most importantly,

Figure 11: GSTduino

It integrates connectors for RC servos, providing multiple servo connectors so extra bread board is not required for using multiple servo motors. It also integrates a battery step up power regulator which can step up the voltage from 3.7V to 5V. It greatly reduces the chances on making mistakes when wiring the servos and wiring different modules together. These are the major reason of choosing GSTduino instead of Arduino Nano.

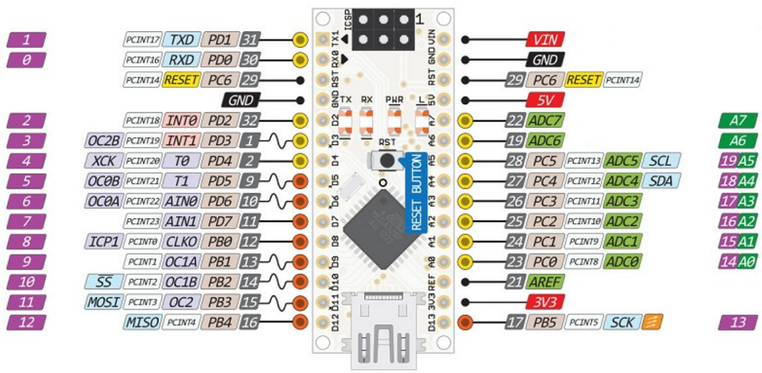

Figure 12: Arduino Nano pin out diagram

GSTduino shares similar I/O pin assignments and functionality with Arduino Nano. It also supports I2C, SPI and serial communication. It can be programmed by Arduino Software or Scratch.

3.2.1.4 Pressure Sensor

Interlink Electronics Force Sensitive Resistor 402 is used as the pressure sensor of the robotic hand. It has a round, 0.5” diameter sensing area. It can sense applied force in the range of 100g to 1kg by providing an inverse change in resistance in response the applied force.

Figure 13: Pressure sensor

The resistance changes of pressure sensor with respect to the force applied can be observed from figure 14.

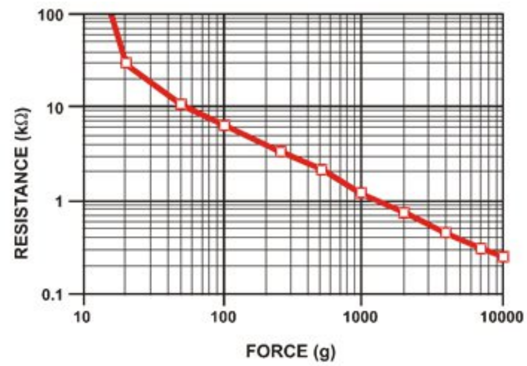

Figure 14: Resistance change of pressure sensor

3.2.1.5 Voltage Divider

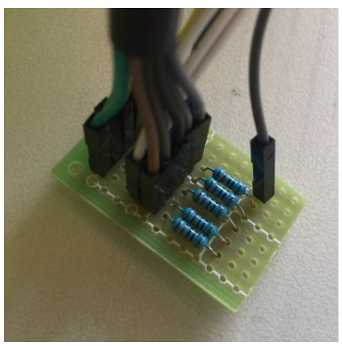

Figure 15: Voltage Divider

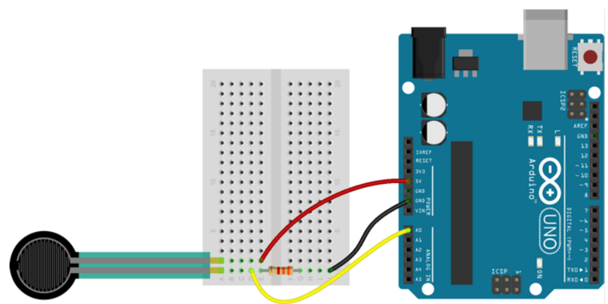

Figure 16: FSR voltage divider diagram

The resistance change of pressure sensor cannot be directly read by the microcontroller therefore voltage divider is required. It can convert the resistance change to voltage change which can be read by the microcontroller. The fix resistor used is 10K resister. Details will be described in Part 3.3.2 Capturing Users Gesture.

3.2.1.6 Suspension System

Tubes and springs are used to create a suspension system. This help to maintain a constant tension on the connecting wire between fingers and the servo motors. This help the robotic hand to move normally and lower its response time.

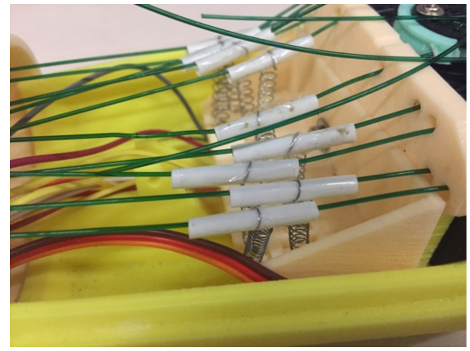

Figure 17: Suspension system

3.2.2 Sensor Glove

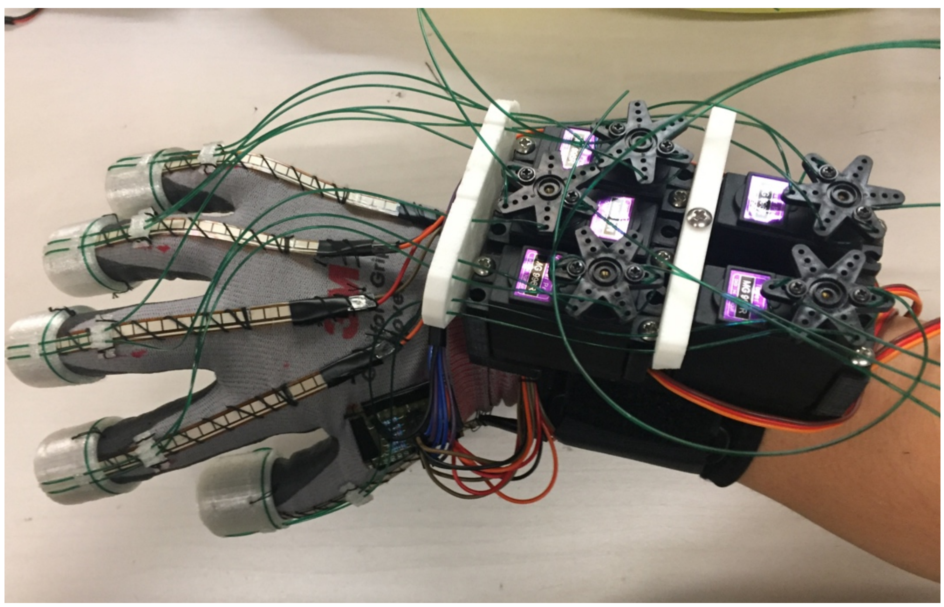

Figure 18: Sensor glove

3.2.2.1 Skeleton

The sensor glove is composed of glove, finger cap, wire director, servo bed and wrist band. For the glove, 3M Comfort Grip Glove is used instead of normal glove as a lot of components are sewed into the

Figure 19: 3M Comfort Grip Glove

glove. 3M Comfort Grip Glove provides higher durability to withstand the high tension provided by other components including the servo motors.

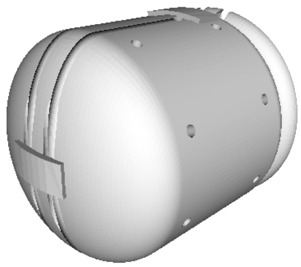

Figure 21: Finger cap

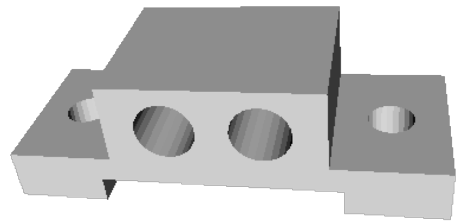

Figure 20: Wire director

Finger cap and wire director is specifically designed and modelled for this FYP. They are components involved in providing haptic feedback. They ensure that the wires that link user's finger and servo motors will be parallel to user's finger to provide a correct and normal haptic feedback. More details will be discussed in Part 4 Model Design and 3D Printing.

The servo bed is a modify version of the InMoov design. A surface is attached to the base of the original servo bed to in order to attach the Mackerel Wristband to the servo bed.

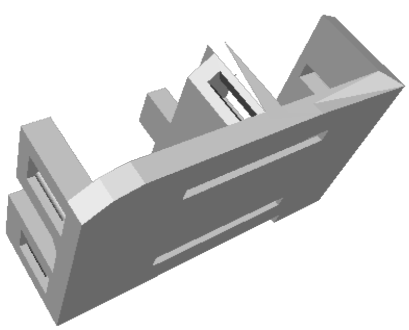

Figure 22: Modified servo bed

Figure 23: MaxKare Wristband

3.2.2.2 Servo Motors

Please refer to Part 3.2.1.2 Servo Motors under robotic hand section.

3.2.2.3 Microcontroller

Please refer to Part 3.2.1.3 Microcontroller under robotic hand section

3.2.2.4 Flex Sensor

Figure 25: Spectra Symbol Flex Sensor

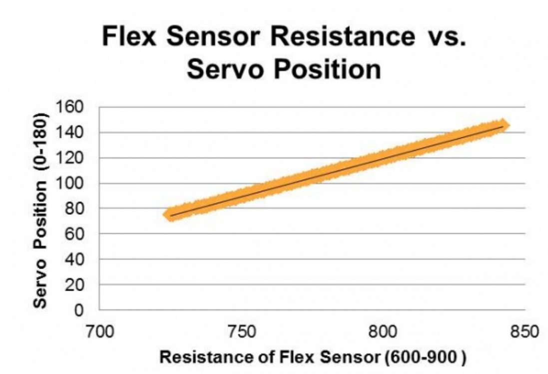

Figure 24: Resistance change of flex sensor

Spectra Symbol flex sensor is used in this project. It is 4.5” in length, has a flat resistance of 25K ohms and bend resistance from 45K to 125K ohms depending on the bend radius. It can operate from -35^{o}C to 80^{o}C.

Flex sensor is used to measures the amount of deflection or bending. Its resistance is directly proportional to amount of bend as shown in figure 24.

3.2.2.5 Voltage Divider

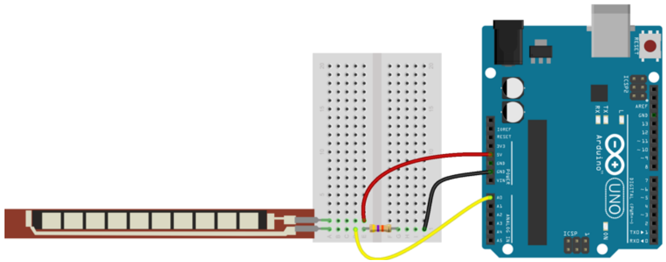

Figure 26: Flex sensor voltage divider diagram

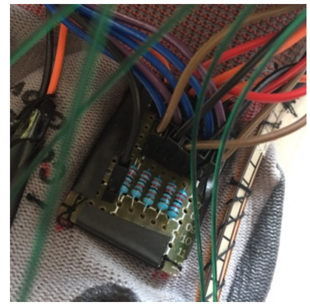

Figure 27: Voltage divider on sensor glove

The voltage divider used on sensor glove is very similar to that on robotic hand. The only difference is the fixed resister used is 22K resister. It is used to transfer the change in resistance to change in voltage so microcontroller can get the value. Details please refer to Part 3.2.2 Capture User's Gesture

3.2.3 Accessories

3.2.3.1 Battery

18650 Lithium Ion Rechargeable Batteries are used in this project. 18650 battery has large capacity, long lifespan, life cycle can be up to more than 500 times. It is also non-toxic, non polluting, light weight and no memory effect. These are the reasons it is chosen. Two 3.7V 18650 Batteries is connected in parallel to provide a longer available time for the robotic hand and sensor glove.

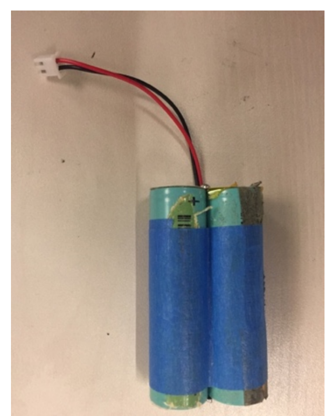

Figure 28: 18650 batteries

3.2.3.2 Wire

0.8mm Fuchuan Fishing Line is used to link the fingertips and servo motors in both robotic hand and sensor glove. It has a much larger diameter than normal fishing line to ensure that it can withstand the strong tension created by the servo motors without breaking. It is also inelastic so the robotic hand control and haptic feedback can be more responsive and accurate.

Figure 29: Fuchuan Fishing Line

3.3 Working Principle

This section will explain how the Haphand work. We will first have an overview to the Haphand system, explaining how different parts are linked together. Then more details will be discussed including the way of capturing user's gesture, controlling the robotic hand, providing haptic feedback and communicating between microcontroller.

3.3.1 Overview

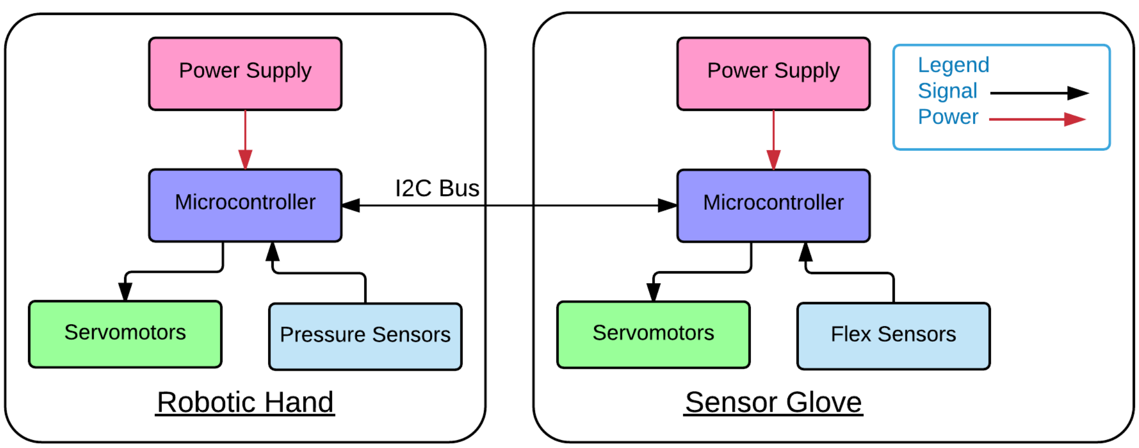

Figure 30: Block Diagram of the Haphand

From figure 30, we can see that the Haphand system is divided into 2 main parts, the robotic hand and the sensor glove. They have similar structure, consist of a power supply, servo motors, sensors and a microcontroller that connects all the components. Both microcontrollers get signals from the sensors and provide signals to the servomotors. The robotic hand and sensor glove communicates using I2C Bus by the microcontrollers. This allows the microcontrollers to get signals and send signals to components that are not directly linked to them. The details will be explained later in Part 3.3.5 Communication between microcontrollers.

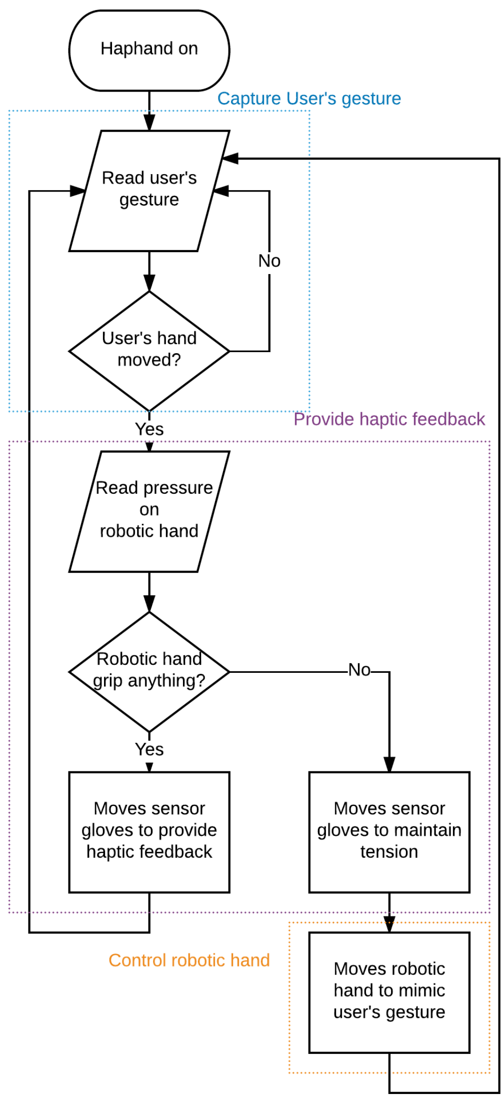

Figure 31: Flow chart of Haphand system

Figure 31 demonstrates the entire work flow of the Haphand system. After the Haphand is turned on, user's gesture will be continuously monitored using flex sensors. If user moves his/her hand, system will check if user has touched anything with his/her finger using pressure sensors. If no, sensor gloves will be moved to maintain appropriate tension of the wire on user's hand. Robotic hand will also be moved to mimic the user's gesture. If user has touched something which means haptic feedback is required, sensor gloves will be moved to constraint user's action and therefore provide a touching sensation to user.

3.3.2 Capturing User's Gesture

5 flex sensors are attached on each user's finger. As mentioned before, flex sensors can measure its degree of bend by the vary in resistance across it. By measuring the degree of bend of each finger, the user's gesture can be captured. If user's gesture is continuously monitored, the user's hand movement can be estimated.

However, the microcontroller cannot read the resistance of the flex sensors directly. Therefore, a voltage divider is required. Voltage divider is a passive linear circuit that produces an output voltage that is a fraction of input voltage. It is used to create reference voltages using the formula on figure 32. Therefore, the change in resistance will be converted to change in voltage and can be read by the microcontroller as analog signals.

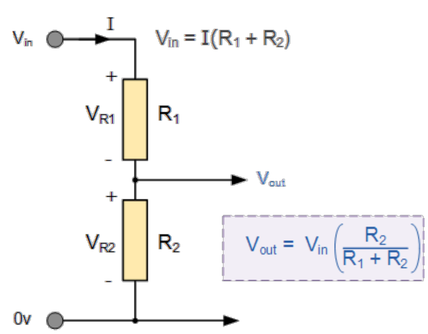

Figure 32: Voltage divider circuit

3.3.3. Robotic Hand Control

After capturing the user's gesture using the flex sensor, the microcontroller of the sensor glove is programmed to analyze the analog data received then covert them to digital signals using Analog to Digital Conversion (ADC) on Arduino. This microcontroller will then transmit the digital signals to the microcontroller of robotic hand using I2C communication. The latter microcontroller will then transfer the received digital signals to analog signals and send to the servo motors to control the fingers. The signal path is shown as a red line in figure 33.

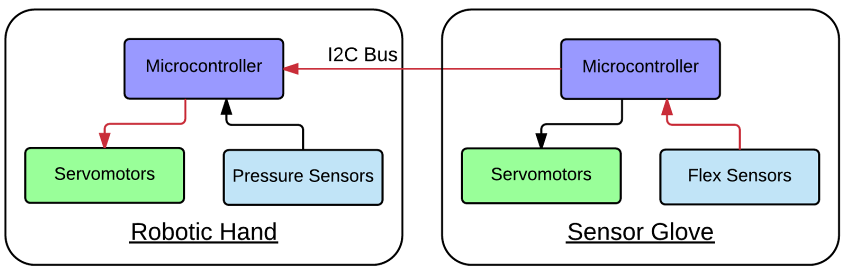

Figure 33: Signal path of robotic hand control

The aforementioned analog signal is created by pulse width modulation (PWM). It is a technique used to encode a message into a pulsing signal. Different frequency of pulse represents different rotating angle. The servo will rotate to corresponding angle by receiving 500,000 to 2,500,000 ns duration of pulse.

There are 5 servo motors attached on the robotic hand. Each servo motor is link to the fingertips of the robotic hand with two wires. One is attached on the palm side and the other one is attached on the back side. Pulling the wire on the palm side will crook the finger while pulling the other wire which is on the back side will straighten the finger.

The rotation direction of the servo motors will decide which wire is being pulled and therefore able to control the movement of each fingers and hence the gesture.

3.3.4 Haptic Feedback

5 pressure sensors are each attached on a fingertips of the robotic hand to detect the pressure applied on the robotic hand and send the signal back to the microcontroller of the robotic hand. Similar to flex sensors, the pressure is detected with the change of resistance across the pressure sensors which cannot be directly read by the controller. Therefore, a voltage divider is installed to transfer the change in resistance to change in voltage.

Detecting pressure applied on the pressure sensors represents the sensors are being pressed. The robotic hand is likely griping something and haptic feedback is required. After the microcontroller received the analog signal, it will analyze the signal and transfer to digital signal the transmit to the microcontroller of the sensor glove through I2C communication. The latter microcontroller will then transfer the digital signal back to analog signal using PWM and move the servomotors attached on the sensor glove. The signal path is shown as a red line in figure 34.

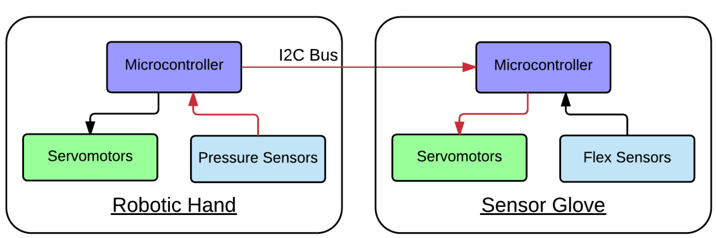

Figure 34: Signal path in providing haptic feedback

5 servo motors are attached on the sensor glove. Each of them is linked to the user's fingertips using 2 wire. When haptic feedback is required, both of the wire will be pulled to constraint the movement of the user and therefore create the sensation of feeling the object. The higher the pressure detected by the pressure sensor, the tighter the wire being pulled. User will experience a higher pulling force and stronger haptic feedback.

It is noteworthy that the servomotors attached on the sensor glove are not only affected by the pressure sensor on robotic hand. It will also move according to the signal received from flex sensor. This means when user moves his/her hand, the servos on both robotic hand and sensor glove side will be moved even when no pressure detected. Figure 35 shows both the signal paths when flex sensors are bent.

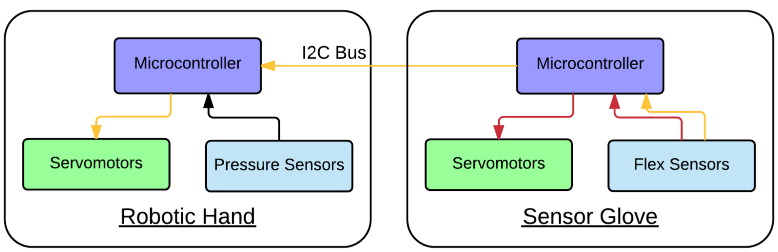

Figure 35: Signal paths of flex sensor

The reason is to maintain an appropriate tension of the wire linking the servos and user's fingertips. If it is too tight, user's movement will be constraint even when no pressure is detected. On the opposite side, if it is too loose, the servos will require a long time to tighten the wire again to provide the haptic feedback hence create an unresponsive user experience. As a result, the servos on the sensor glove side need to adopt with the user's gesture, maintaining an appropriate tension without hindering the user's fingers.

3.3.5 Communication between microcontrollers

In this FYP, 10 sensors and 10 servo motors are involved therefore 10 analog input pin and 10 digital I/O pin is required. However, the GSTduino only has 8 analog input pin which is not enough. As a result, 2 GSTduino boards have to be used and communication between microcontrollers is involved. The Inter-Integrated Circuit (I^2C) Protocol is used for this purpose. It is a serial computer bus that allows multiple "slave" digital integrated circuits to communicate with usually one "master" chips. Only two signal wires are required, the SCL and SDA line. SCL is the clock line that used to synchronize all data transfers over the I2C bus while SDA is the data line. Master chips will be the device that drives the SCL clock line. When the master devices want to initiate the communication to the slave devices, it will begin by issuing a start sequence on the I2C bus then follow by the message and end with an ending sequence.

In this project, microcontroller of the sensor glove is the master device while microcontroller of the robotic hand is the slave service. Since the slave device cannot initiate a transfer over I2C bus, only master device can do so. Therefore, the microcontroller of sensor glove will constantly require the reading of the pressure sensor on robotic hand side.

3.4 Arduino Program Design

As there are two microcontrollers with one on robotic hand side and one on sensor glove side, there are two Arduino programs with one program acting as master program and one acting as slave program.

For the microcontroller on sensor glove, it acts as the master in I2C communication protocol. It is responsible for receiving signal from flex sensors and then send signals to servos on robotic hand and sensor gloves. In the program, pins are first allocated to servos, then initiate the wire library and join the I2C bus as a master. After the setup, the following program will be looped: Read the signal from flex sensor; Map the flex sensor reading to the angle that the servos are required to move; Constraint the maximum value that the servos can move; Begin a transmission to I2C slave device and send the value of each servos on robotic hand is required to move; Request the value received by pressure sensor on robotic hand through I2C communication.; Map the value received to the value that required servos on sensor glove to turn in order to provide haptic feedback; Move the servos on sensor glove according to the difference of value got by flex sensor and pressure sensor. If the pressure sensor does not being pressed, the servos will adopt to the user's gesture. If the pressure sensor is pressed, the servos will move to restraint user's movement and hence provide haptic feedback.

For the microcontroller on robotic hand, it acts as the slave in I2C communication. It is responsible for receiving signal from the flex sensor on sensor glove through I2C and turn the servos on robotic hand accordingly. It also received signal from pressure sensor and send to the master device, the robotic hand using I2C. In this program, pins are first being allocated to the servos on robotic hand, then initiate the wire library and join the I2C bus as a slave. It will then register a function to be called when it receives a transmission from the master. The transmission received will be the value of servos that is required to turn. Therefore, in this function, the servo values will be read than write to the servos on robotic hand to mimic user's gesture. Another function will be registered to be called when the master requests data from it. The data requested will be the value of pressure sensor. Therefore, in this function, the microcontroller will get the value from pressure sensor; map it to degree that requires the servos to turn to restraint user's movement to provide haptic feedback. Then send the values to the master device using I2C bus.

It is noteworthy that moves the servo in the range of 0^o to 90^o is usually enough to make it moves from crooking entirely to fully straighten. To the servos at robotic hand, 0^o represents fully straighten but to the servos at sensor glove, 0^o represents fully crooked. The reason behind this is the servos cannot move at negative value but can move beyond 90^o to a maximum of 180^o. For the robotic hand, the servo can move beyond 90^o to provide extra griping force to the robotic hand and for the sensor glove, the servo can also move beyond 90^o to provide extra haptic feedback to the user so the user will have a stronger feedback.

4 Model Design and 3D Printing

This project has involved a lot about 3D printing as most parts of the Haphand can be 3D printed. Also, as the sensor glove is designed by me, it requires component that are highly customized. Being able to 3D modeling and printing them has greatly saved my time and make me able to do multiple trial to explore the best shape of the components. In the following section, some basic about 3D printing will be mentioned. The 3D modeling of the customized parts will also be discussed.

3D printing is a processes that used to create 3D object in which layers of material are formed under computer control to create the object. 3D printing involves of 3D modeling, fixing process, the printing process.

4.1 3D modeling

3D modeling is the process of developing a mathematical representation of any 3D object via specialized software including Sketchup, Blender, Adobe Rhino, etc.
In this Project, Sketchup is chosen. It is a 3D modeling software for wide range of drawing applications such as architectural, interior design, civil and mechanical engineering, etc. Sketchup is chosen because it is much more easy to use and the measurement of objects is accurate compare to other alternatives.

In this project, I have designed multiple components using Sketchup. The first one is finger cap.

Figure 37: Finger cap version 1

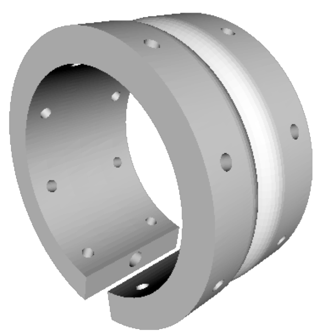

Figure 36: finger cap version 2

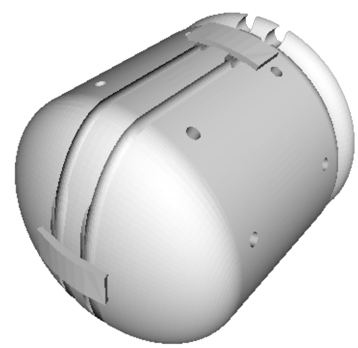

Figure 38: Finger cap version 3 (final)

The above 3D components are served for the same purpose, attaching on user's finger and link to the servo using wires to provide haptic feedback. 3 version of finger cap have been made. In the first version, user will wear the finger cap like wearing a ring. When it is being pulled, it tends to rotate in the same direction of the wire being pulled. The side of the ring will press the user's finger and may hurt the user. More importantly, although the haptic feedback can be provided, it is not realistic. Therefore, a second version is developed with larger surface but it faces similar problem too. Finally, the third version is made and change from ring shape to cap shape. This applied pressure to user's finger uniformly and provide a realistic haptic feedback.

The second component is the wire director which use to lead the wire connecting the finger cap and servos to move along the user's finger without deflection so the finger is pulled upwards instead of sideward and hence provide a realistic haptic feedback.

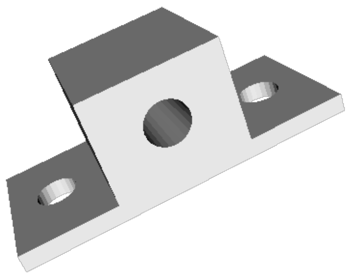

Figure 40: Wire director version 1

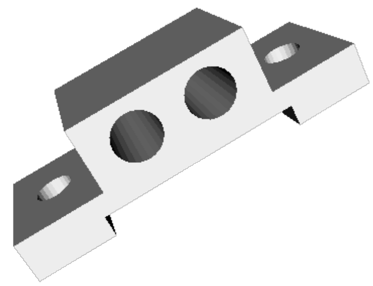

Figure 39: Wire director version 2 (Final)

In wire director version 1, there is only one hole but 2 wires need to be directed. This make the connecting wire unorganized and easier to break. Furthermore, the bottom of wire director version 1 is flat but it will be sewed to the sensor glove with the flex sensor underneath, this may damage the flex sensor. Therefore, the wire director version 2 is created to solve the abovementioned problems. It has 2 holes and has space underneath to cover the flex sensor without pressing it.

The servo bed of InMoov design is also modified so a wrist band can be attached on it and be worn by the user. An extra layer is added to the original design which contain space inside that allow the wristband to pass through. However, since the design of the original servo Box is download as STL file. It can not be directly edited by Sketchup. A plugin, SketchUp STL have to be installed so it can be processed in Sketchup.

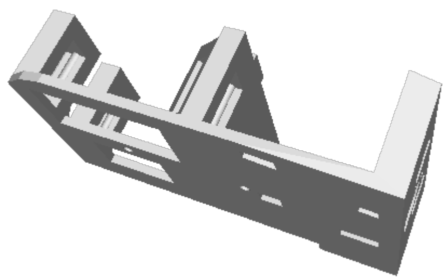

Figure 41: Original Servo Box

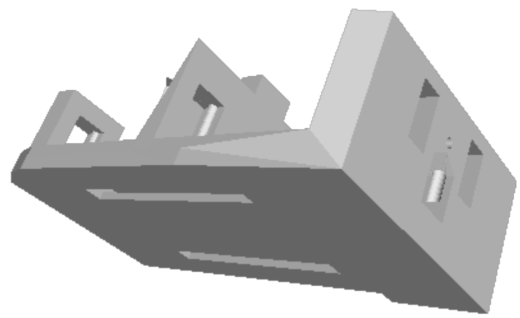

Figure 42: Modified Servo Box

4.2 Fixing process

Sketchup is convenient to use but this convenience may cause error sometimes. In Sketchup, connecting lines can form surface automatically and can pull up a surface to form a solid. It also allows you to erase a line from a surface. If this is carried out carelessly, an open object may form where there are holes in the object as a face is missing. A 3D printer is damaged in this project because there are abnormal surface and hole on the object which causes the 3D printer to operate under abnormal condition. This problem can be solved by creating a new surface to close the hole. Sometimes, this problem can be auto repaired by the printer software.

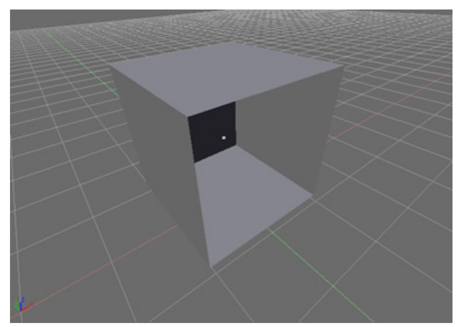

Figure 43: Open object

4.3 Printing

After the 3D modeling, the 3D model will be converted to g-code file which uses numerical control programming language. It tells the 3D printer how to print the 3D model including where to move, how fast to move and what path to follow. The 3D printer will then print the 3D object from bottom to up, layer by layer.

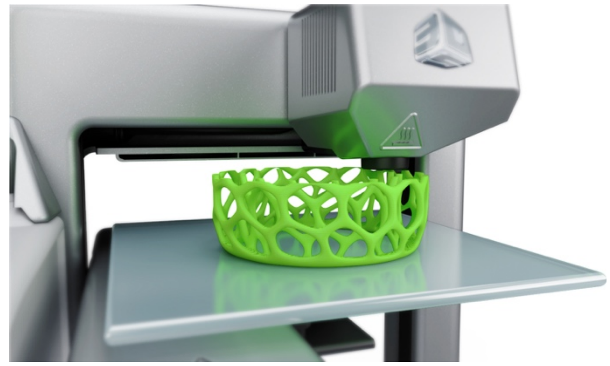

Figure 44: 3D printing

Multiple issue may occur during 3D printing due to the model, the position of the printing bed, etc. The first common issue is overhanging. Since the 3D objects is built layer by layer, each layer must be supported. If it is ignored, the parts that without any supports will fail to print correctly as shown in figure 45. This issue can be solved by adjusting the orientation of the printing object or adding support to hanging parts.

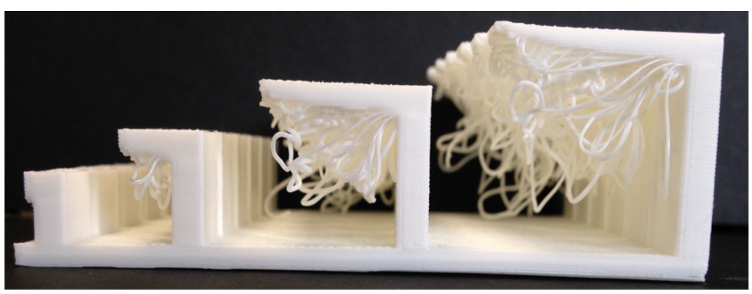

Figure 45: 3D model with overhanging problem

The second common issue is warping. Warping occurs due to the material shrinkage in 3D printing. Corners are especially vulnerable to this issue. Since when plastics are printed, they

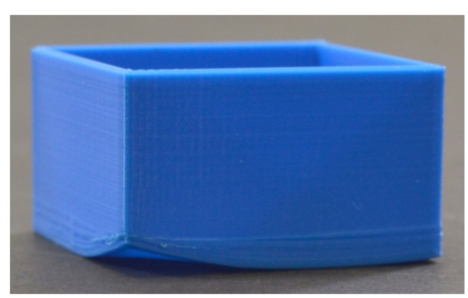

Figure 46: Object with warping issue

will contract as they cool down, if it contracts too much, the print may bend up from the build plate. To avoid this issue, a heated build plate can be used to keeps the printed material at temperature just below the point that the material starts to solidifies. This problem may also occur if the distance between nozzle and build plate is too wide.

5. Results and Limitations

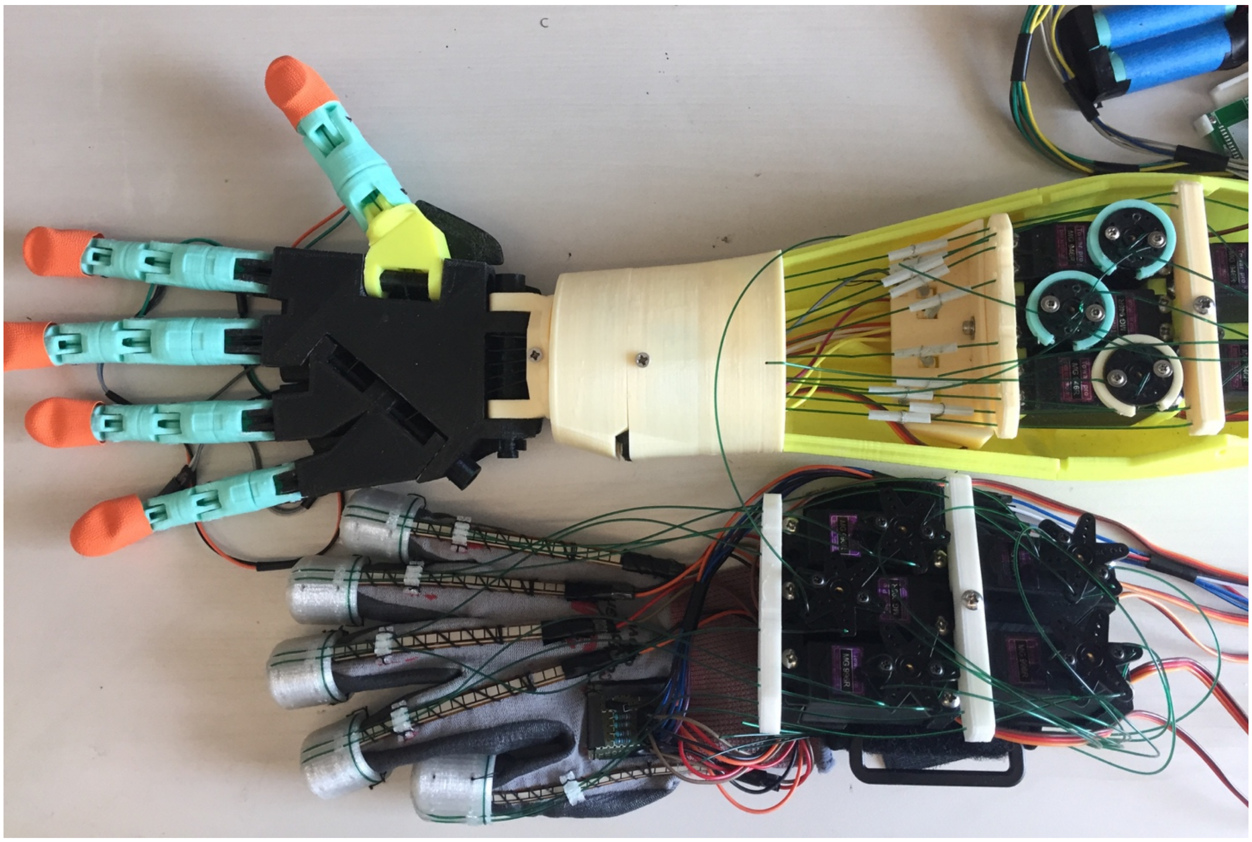

Figure 47: The Haphand

At the end of this FYP, a robotic hand that can be controlled by sensor glove and able to provide haptic feedback has been made successfully. It has an around 0.2s delay between user moves his/her hand and the robotic hand response by mimic the user's gesture. The delay can be noticed but it is acceptable and does not affect the whole user experience a lot. The delay is mainly due to the rotating speed of the servos. Because of the strong torque that is provided by the wire and the good strength of the wire, this robotic hand can lift heavy object up to 2kg. However, since only the fingertips of the robotic hand are linked to the servo. There are no individual servos for controlling the knuckles. The fingers are crooked just relying on the tension of the wire. There are limitations on the gestures that the robotic hand can mimicked. Furthermore, the way of crooking the fingers depends on the friction of bolts in the knuckles. If the bolts in top knuckles have the least friction, it will bend first when crooking the fingers. Therefore, sometimes the way of the robotic hand bending fingers may not similar to human hands.

For the haptic feedback, user's can experience a significant pulling force when pressure sensor of the sensor glove is being pressed. This make the user can roughly feel the object that the robotic hand is gripping. If the user tries to resist the pulling force and keep moving, the pressure detected by the pressure sensor will be higher and apply a larger pulling force to the user's finger. However, the tension of the wire is affected by many factors including the position of the servo bed and the angle of the wrist. This make calibration difficult and user's have to maintain steady to achieve realistic haptic feedback.

6. Future Development

The user experience can be further enhanced by using motors with higher rotating speed to lower the delay between the user moves his/her finger and the robotic hand moves to mimic user's gesture. Motors that can provide higher torque can also be used so the robotic hand can be able to grab heavier object. More motors can also be used to control different part of the robotic hands such as the knuckles, even allowing the fingers to move sideward like human do to mimic more gesture of the user.

For the haptic feedback, as mentioned before, the tension of the wire linking the finger cap and servomotors is affected by a lot of factors which gives a lot of uncertainties. This can be solved by fixing the angle of the user's wrist which will make it difficult to use. Otherwise, muscle sensor can be used to detect the entire movement of the user's hand and arm and adjust the tension of the wire based on the position of user's hand and arm.

By applying the above solution, the robotic hand will be more intuitive to use with the lower delay and the haptic feedback provided will be more realistic. The Haphand can be applied to the following area.

6.1 Rescue robots

Figure 48: Rescue robot

For performing a specific task, a robotic arm that is specifically designed for that purpose is more suitable than robotic hand. However, the difference between normal robots and rescue robots is the latter will face much more uncertainties because of the greatly varied environment. In this case, robotic hand is the best solution to handle many different scenarios. They can be used to carry victims out of the hazardous environment or carry out delicate task like tying a knot. In this case, the haptic feedback provided by the Hap hand can be useful.

6.2 Gaming Industry

Figure 49: PS4 VR

Nowadays, the gaming industry emphasizes more and more on immersive gaming experience. From providing better graphics, better sound effects to the development of joystick and the developing virtual reality technology is all served for the same purpose. The haptic feedback technologies used in the Haphand may be used in gaming to provide more feedback to the user and hence providing a more immersive gaming experience.

7. Conclusion

The Haphand has immense potential in enhancing the versatility and overall usefulness of robots. This sensor glove-controlled robotic hand with haptic feedback can let user control it intuitively and provides more information about the environment, thanks to the haptic feedback supplementing visual information. Being able to fully utilized the 3D printing technology, this is an economic way to ensure some emergency cases can be handled quickly without on-site human operation. The Haphand has great developing potential in many areas including the rescue robots and gaming industry.

8 References

1. Finn A, Scheding S. Developments and Challenges for Autonomous Unmanned Vehicles: A Compendium. Berlin: Springer Science & Business Media; 2010
2. France AK. Make: 3D Printing: The Essential Guide to 3D Printers. Canada: Maker Media, Inc.; 2013
3. Thanked S, Azhar SMH, Miskon A. Prosthetic Hand Controlled by Wireless Flex Sensor on EOD Robot. Singapore: Springer Singapore; 2016
4. Moll J, Sallnas EL. Communicative Functions of Haptic Feedback. Heidelberg: Springer Berlin Heidelberg; 2009
5. History of Industrial Robots from the first installation until today. Germany: International Federation of Robotics; 2012