Acknowledgements

First, we would like to thank Allah for providing us with the strength and determination to achieve this research work.
We want to acknowledge the immense support provided to us by our supervisor, Dr. Umar Shahbaz Khan. He offered continuous and unending technical and moral support. His feedback and inspirational ideas helped us majorly in achieving this work. Dr. Umar kept us motivated and disciplined throughout the research through his structured approach, and he provided guidance, especially regarding the computer vision and edge-computing deployment part of the project. It can be said undoubtedly that this research project could not have been able to achieve as much without the constant support of Dr. Umar Shahbaz Khan.
Furthermore, the support provided by Dr. Umsan Akram was extremely valuable and he provided us with every resource and help that he could. He continuously assisted us during our work and provided us with new avenues and platforms to complete this research project.
Lastly, we would like to thank our families for being our support system and giving moral support to this endeavor and others. We would especially like to acknowledge the efforts of our parents and their sacrifices and their constant prayers to help us reach our best potential. We would extend our sincerest appreciation to our friends for giving us unflinching moral support. Hence, every aforementioned person and those not mentioned (but in our hearts), helped us complete this research project to the best of our potential.

Imaan Shahid
Ali John Naqvi
Hassan Ashraf
Bilal Yousaf

Abstract

The rise in upper limb amputations has led to a major growth in the use of prosthetic hands throughout the world. Amputations can result from a variety of disorders, including cancer, diabetes, congenital deformities, infections, and vascular diseases, as well as severe injuries from accidents at work or in cars. Amputations can also be the result of disagreements and accidents at work in some areas. Prosthetic hands are recommended to address these issues, however many of the current models have frustratingly complex control schemes, scant sensory feedback, and unnatural movement patterns that drive users away.

In this capstone project, we offer a thorough solution that makes use of various control systems to operate prosthetic hands. In the first control approach, hand movements are interpreted by a camera and computer vision algorithms to identify the appropriate grip pattern. This enables gesture-based control. The second control method makes use of electromyography (EMG) signals produced by remaining muscles in the amputated limb. Surface electrodes are used to record and analyze these signals, which allows to precisely control grip patterns. As a third control option, a user-friendly mobile application is also created that enables people to manually select and modify grip patterns in accordance with their preferences. This project offers a user-centric approach to prosthetic hand control, lowering the learning curve and improving usability for people with upper limb amputations, enabling them to carry out daily tasks with greater ease. It does this by integrating camera-based control, EMG signal analysis, and a mobile app interface.

The mechanical design of the prosthetic hand, which has six degrees of flexibility, was greatly improved in this research. A significant improvement was made to the thumb, which had previously presented a problem because of its size and made it difficult for it to fit inside a glove. A new mechanism for the thumb was created with the goal of minimizing its size while preserving functionality to overcome this limitation. The user's comfort and usefulness are improved because to this redesign, which enables seamless integration of the 6 degrees of freedom prosthetic hand with a glove.

A haptic feedback system was a crucial addition to the prosthetic hand system. By giving users a feeling of touch, this device hopes to improve their entire experience and the prosthetic hand's functionality. Users can experience tactile sensations and feedback from the hand using the haptic feedback technology, simulating the sense of touch. The user's capacity to interact with items is significantly improved by this breakthrough because they can now feel texture, pressure, and other tactile information. The prosthetic hand offers a more intuitive and realistic experience because to the addition of haptic feedback, enabling users to carry out a larger variety of activities and better navigate their surroundings.

Contents

List of Figures

Chapter 1

INTRODUCTION

The prevalence of amputations worldwide was 1.6 million in 2005, with projections that the prevalence may double by the year 2050 [1]. This exponential increase in amputations has spurred technological advancements and developments in the field of prosthetic development. The advancements in this field are directed towards providing a solution for regaining functionality and a closer-to-reality experience for amputees.

Primarily, a great number of amputations occur because of traumatic injuries on the upper limbs at the workplace, for instance by electric shocks, burns, and industrial mishaps. Conflict and war are also very prevalent in the Asian region causing the participants to concede multiple injuries through gunshots or active combat. The diagnoses of medical conditions like cancer, vascular diseases, birth defects, and diabetes can cause the upper extremity to be amputated because of the danger associated with the limb. Additionally, congenital/birth defects are also a major cause resulting in the amputation of the upper limb.

An amalgamation of all these reasons has resulted in an increased number of amputees and an increased use of prostheses to allow amputees to reclaim their independence. The use of a prosthesis is highly crucial for an amputee for the improvement of the health and well-being of the patient along with allowing them to resume their daily activities, be it house chores or their employment responsibilities. However, it is important to note that rate of prosthesis rejection is high among upper-limb amputees [2]. The existing prosthetic hands have limited freedom of movement which creates a more affected experience for the user, making them disappointed. In addition, the patient is required to put in a significant amount of effort to learn the correct usage by attending rehabilitation sessions also requiring long training periods. Usually, the frustration of the patient and a lack of natural movement causes the abandonment of the prosthesis.

This project brings forward a more automatic, and intelligent alternate solution to lower the prosthesis rejection rate and create a natural experience for the amputees. The alternative solution introduces machine learning to automate the grip determination of the prosthesis which would enhance the usability and lower training periods for amputees. A camera is embedded in the palm of the prosthesis which captures the scene as approached by the patient. Machine Learning models are applied to this capture to determine the class of the object, the size of the object, and its distance from the embedded camera. This processing would allow the prosthesis to decide about the grip based on the information gained through the models.

Instead of cloud computing, edge computing is employed to deploy machine learning models to enable decision-making in real-time and provide a determination for the grip faster with minimal latency. This would make the prosthesis more responsive and intuitive, making it more natural to use.

Along with technical developments, the focus will be on enhancing the prosthesis' mechanical design. To improve the prosthesis's comfort, dependability, and utility there will be utilization of lightweight materials, cutting-edge joint mechanics, and ergonomic considerations. The enhanced mechanical design will make the prosthesis easier to integrate into the user's regular routines. Along with this, this solution will also provide a sense of touch by using haptic feedback; a motor band would send vibrations to the user depending on different frequencies. Lastly, a mobile application will be developed to enable users to control the grips of the prosthesis manually and gain more customization according to their needs and requirements. The mobile application will provide in-built grips and options for reducing power consumption by establishing a simple Bluetooth connection.

The central aim of this project is to overcome the limitations of the existing prosthesis by creating a user-centric design and solution by reducing the effort required by the patient. The modifications are intended to create a realistic experience for amputees and take a novel step in the advancements in the field of upper limb prostheses.

1.1 Scope

The scope of the project is to create a solution that includes an intelligent prosthesis upper limb prosthesis that would be deployed on an edge computing device. This scope would be achieved by employing machine learning models and creating an enhanced mechanical design. This would allow the development of a prosthetic device that would be able to automatically determine the grip position of the prosthetic hand along with creating natural and smooth movement and control. A mobile application will also be a part of the project which would enable the user to set manual grips and have a more intuitive experience.

The scope also encompasses the assessment of various computer vision algorithms inclusive of object detection algorithms and depth estimation for grip determination. This includes data set creation, collection, training, and testing. The deployment of these algorithms on an edge computing device would allow for real-time processing, reducing latency and enhancing responsiveness. Moreover, the project also entails the enhancement of the mechanical design of the prosthetic hand considering the weight of the hand, comfort, and ergonomic considerations.

Additionally, the range of the project includes the careful consideration of Human Factor Engineering (HFE), taking into account human characteristics, limitations, and capabilities, for the efficient design of the prosthetic hand system. The first step towards these considerations would be to create a sense of touch for the user by using haptic feedback techniques. Secondly, the development of a mobile application, an important aspect of the project, would allow the user to interact with the prosthetic hand to customize different grips based on specific requirements.

1.2 Motivation

Limb amputation has become a prevalent issue because of its exponentially increasing number of cases. People around the world suffer from limb absence because of various reasons. Out of these upper extremity affects approximately 41,000 people which is 3% of the "Limb Absence" population [3]. Although surgical removal is one of the main reasons for limb absence, the possibility of being born without the upper extremity is also high. Researchers estimate that about 1 in every 1,900 babies is born with a limb reduction defect in the United States. Some of these babies will have both upper and lower limb reduction defects [4]
Limb amputation is caused by several different reasons, the main reason being trauma followed by cancer. Upper limb amputations from trauma occur at a rate of 3.8 individuals per 100,000 [3]. Figure 1.1 shows the causes of amputations caused by traumatic injuries and experiences. The ever-rising statistics of amputations were one of the main motivations for this project. People who experience limb loss are excluded from the basic activities of society, be it common house chores or their job tasks. They are treated differently because of this they start developing mental illnesses. Hence, the motivation behind this project was to create a solution for the affected to make them valuable and equal citizens of society.

Additionally, the amputees were advised to use a prosthesis, an artificial limb for

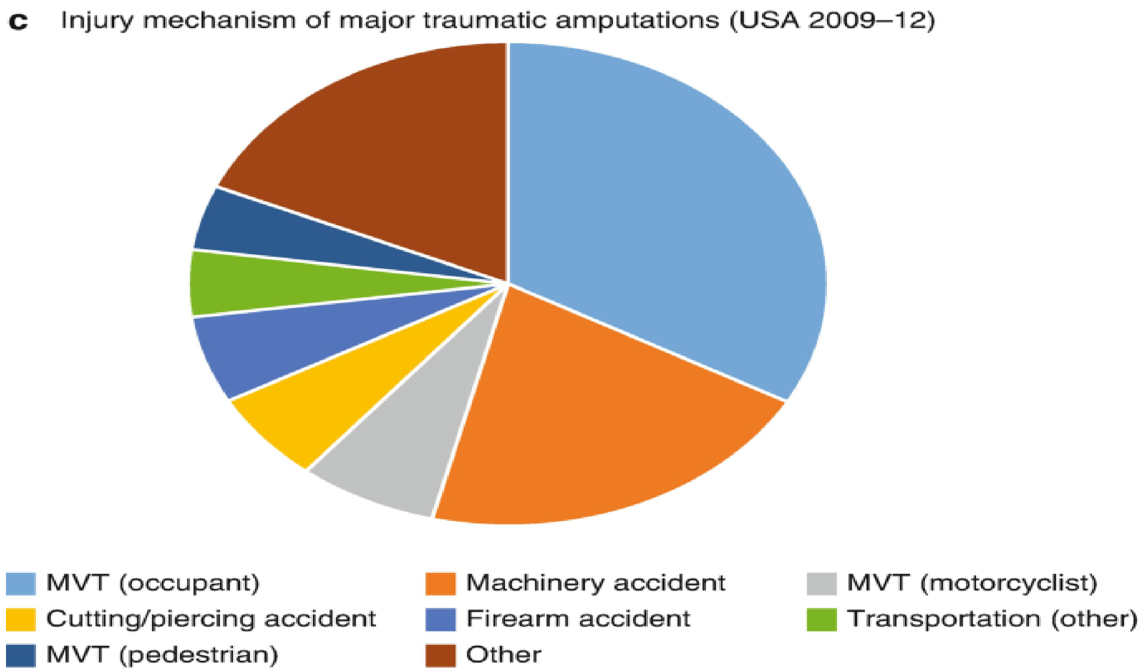


Figure 1.1: Reasons of major traumatic amputations (USA 2009-12) [5]

the replacement of the lost limb, to resume their daily routine. However, the users quickly became frustrated because of the long rehabilitation/ training periods associated with it and the discomfort. This frustration has led to an extremely high rate of prosthesis abandonment going up to as much as 44% (which is nearly half of its users) [6]. Figure 1.2 visualizes the main reasons given by the users for the abandonment of their prostheses. The alarming rate of abandonment was the primary reason for perusing this project. Our solution strives towards automating the training periods for the amputees and creating a lightweight design that would be a comfortable fit for the patient. Hence, the focus of the project is to be more user-focused and create a product that will be user-friendly.

In summary, the motivation behind this project was to strive towards better health and well-being of the amputees and reduce inequalities faced by the affected persons. Moreover, making the prosthesis "less of a fuss" for the users and making a product that could actually be called a "replacement" for the lost limb was the reason behind this project.

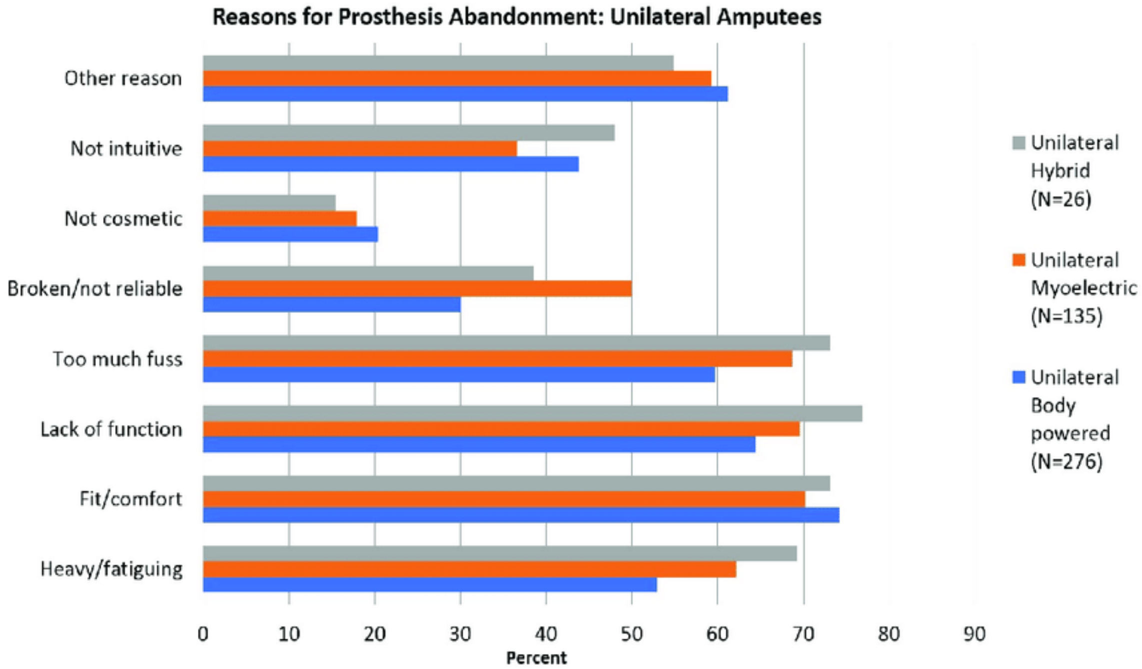


Figure 1.2: Reasons for prosthetic abandonment [7]

1.3 Objectives

The objective of this research study is to create grip patterns of a prosthetic hand in real time based on the object intended to be picked up by the user. The aim of the project extends to the improvement in mechanical design and creating a sense of touch for the user. These enhancements strive to decrease training periods and reduce the abandonment rate. These The objectives to achieve this goal are as follows:

1. To improve the mechanical design of the prosthetic hand to make a comfortable and fit product for its user.

2. To develop a haptic feedback solution for controlled force exertion to create a natural and realistic experience.

3. To implement a computer vision solution for the accurate grasp of objects implemented on a prosthetic hand.

4. To develop an edge computing-based solution for real-time analysis and increased responsiveness.

5. To develop a Flutter-based mobile application for monitoring power consumption and better user experience.

1.4 Problem Statement

In the Asian region, upper extremity amputation is also very common mostly because of poor and dangerous working conditions and constantly rising conflicts and wars. However, to account for this exponential increase, prosthesis development is still deficient. The prosthesis that already exists is sparsely used because of their shortcomings. Witteveen et al. (2012) suggest that 30% to 50% of prostheses wearers do not use their devices regularly, Additionally, 74% of non-users state they would consider using prostheses if they present significant technology improvement at an affordable price. [8]

The problems addressed by this project are the shortcomings of the existing prosthesis. These shortcomings have caused users to abandon their upper extremity prostheses. Following are the shortcomings which this project strives to create a solution:

1. Limited number of grips.
2. Limited degrees of freedom.
3. Long training periods for the user, to learn prosthesis control and coordination.
4. High prosthesis abandonment rate due to frustration from long training periods.

In summary, the research and development in this area are crucial for Asia and Pakistan to create a healthy community and society where amputees are given an equal chance of achieving an adequate lifestyle.

1.5 Contributions

In order to achieve the aforementioned objectives the following contributions were made during the course of the project:

1. Prototyping and Testing: various prototypes and iterations were performed to create a functional and improved mechanical design for user satisfaction.
2. Research and Development: machine learning models were tested for effective grip pattern determination. Data collection, training, and testing were the primary procedures.
3. Edge-Computing Integration: The models were integrated with an edge-computing device to increase response time and allow for real-time grip pattern determination.
4. Mobile Application Development: A mobile application was created to establish a connection with the prosthesis and allow users to manually control grip patterns for specific requirements.

1.6 Structure of Thesis

The structure of the thesis is as follows:

Chapter 2 discusses the different types and locations of upper limb amputation along with the various causes of amputation. Moreover, the most common cause and type of amputation are identified.

Chapter 3 presents a comprehensive literature review of the existing prosthetic hands in the market and the pros and cons of each of them.

Chapter 4 is a detailed explanation of the previous design of the prosthetic hand, the new and improved design requirements, and each of the design iterations performed to achieve the requirements.

Chapter 5 is the step-by-step explanation of the workflow of the entire circuit. Additionally, the electronic components used and their reason for choice are also discussed.

Chapter 6 elaborates on the software side of the project, including the discussion of object recognition models, depth estimation, and deployment of the models on the edge-computing device.

Chapter 7 concludes this thesis and presents the main contributions. Finally, future projections of the proposed work are discussed.

Chapter 2

LITERATURE REVIEW

A prosthesis is an artificial limb that is used for the replacement of an absent or amputated limb. In the case of an upper limb prosthesis, the prosthesis is designed to replace the upper limb to restore functionality and allow the patient to perform tasks seamlessly. Along with serving as an artificial appearance of the missing body part, it greatly improves the quality of life of the amputee. Due to its importance, the prosthetic industry, research, and development have always increased. Additionally, with the introduction of industrialization and an increased rate of upper limb injuries the requirement for prosthetic devices has soared up.

Chapter 3 is an overview of the research and development done in the field of prosthetics and how they have evolved from being only aesthetically pleasing to advanced functionality.

2.1 Upper Limb Amputation

Upper limb amputation refers to the removal of any part of the upper extremity due to surgery, trauma, or pathology. Standard levels of amputation include partial or complete removal of the arm, forearm, hand, or digits. Congenital hand deficiency, caused by exposure to toxins during pregnancy or genetic abnormalities, results in babies being born without hands or with underdeveloped hands. Traumatic injuries are the leading cause of upper limb amputations. The absence of an upper extremity can significantly impact a patient's daily life and limit their participation in activities.

After amputation, various approaches can restore functionality to the missing hand. The most effective solution recommended by doctors is a prosthesis, which requires patients to undergo extensive training and rehabilitation to learn how to use the entire system.

The discussed types include transmetacarpal amputation [9], wrist disarticulation , transradial amputation, elbow disarticulation [10], transhumeral amputation, shoulder disarticulation, and forequarter amputation. Transmetacarpal amputation involves the removal of the metacarpal bones while preserving a longer part of the forearm. Wrist disarticulation involves the amputation of the hand at the wrist joint. Transradial amputation is the most common type and involves the removal of the forearm below the elbow. Elbow disarticulation separates the forearm and the longer portion of the upper limb.

The primary reasons for upper limb amputations are trauma, congenital and genetic disorders, dysvascular diseases, cancer, and infections. Trauma, including workplace injuries and accidents, is the most common cause. Congenital deformities [11]2 at birth can also lead to upper limb absence, while dysvascular diseases affect blood flow and can result in surgical removal. Cancerous tumors in the upper limb may necessitate amputation, and severe infections that cannot be controlled may lead to the permanent removal of the limb.

In conclusion, upper limb amputations can have a significant impact on individuals' lives. Prosthetic devices are crucial in bridging the gap between normal life and amputation. However, many patients refuse to use or abandon prosthetic devices due to frustration. The scope of this project is to enhance and improve upper limb prosthetic devices to reduce patient frustration and abandonment rates.

2.2 Historical overview and previous work

The earliest prosthesis date back to 77 AD which was created for a Roman general to return back to battle. The prosthesis was made of iron and had very limited functionality. One of the earliest and the most famous prosthesis of early times was of a German Knight and was also made of iron [12].

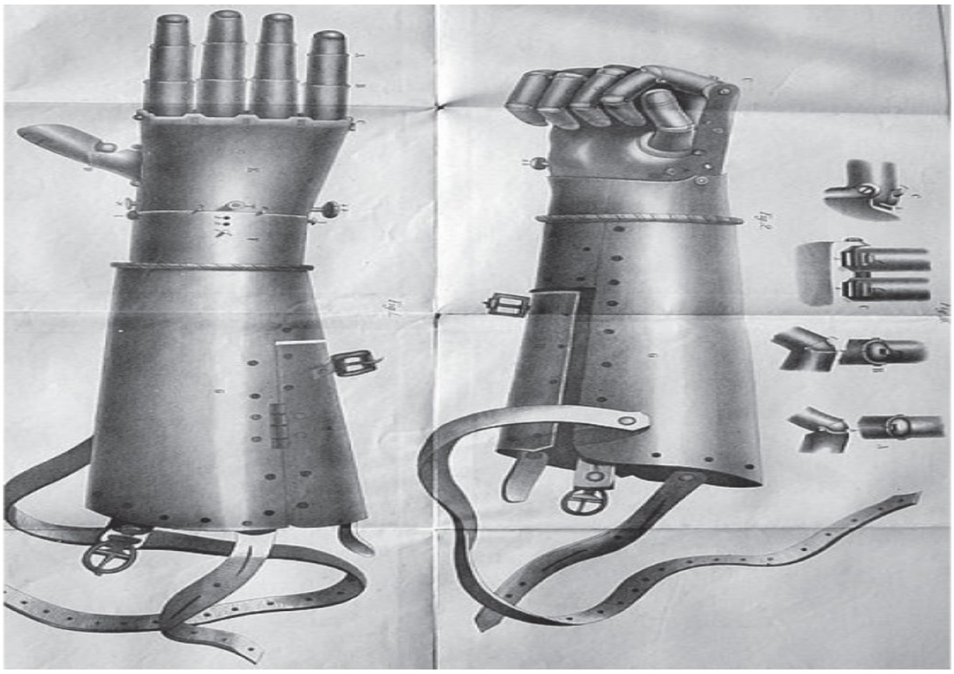


Figure 2.1: The earliest prosthetic hands [13]

Figure 2.1 shows the iron prosthetic hand owned by the German knight. This prosthesis could only move at certain points.

2.2.1 Classification of existing upper limb prosthesis

As the years passed by, developments in the appearance and functionality of the prosthesis started increasing. The different types of prostheses that were made are classified as follows:

- **Passive Prosthesis:** A passive prosthesis is a prosthesis in which the force required to adjust the gripping of the prosthesis is applied externally, for instance, by the sound hand [14]. Passive prostheses are not electric, but mechanical and provide the basic functionality of the prosthesis by giving stability to the user for grabbing objects. Moreover, they resemble the natural human arm and are sometimes used just to protect the remaining limb of the amputee. Only a fraction of the amputees use a passive prosthesis.

- **Active Prosthesis:** On the other hand, active prostheses are battery-powered and do not require an external force to produce grip patterns. They use different sensors, placed on the remaining arm, to receive signals from the flexion and extension of muscles to identify user intention and create grips. There was extensive research and development in this area, resulting in four further types of active prosthesis:

  1. Body-powered prosthesis: A body-powered prosthesis uses the body's movements and mechanical control systems. In fact, these prostheses use cables to transmit the user's intention and movement to the prosthesis control system. A cable uses the motion of a person's shoulder blade and upper arm to operate the hand, hook, or elbow joint [15]. Although body-powered prostheses provide helpful support for the amputee but heavily depend on the amputee's strength and range of motion after the surgical removal.

  2. Myoelectric prosthesis: Myoelectric prosthesis uses electrical signals produced by the amputee's remaining muscles to control the prosthesis and create grips. A myoelectric prosthesis uses a sensor that gets the surface electromyography (sEMG) signal from the users for controlling the prosthesis [16]. Myoelectric prostheses have an edge over body-powered prostheses because of their natural movements and a closer-to-reality experience for the users.

3. Hybrid: The hybrid model is an amalgamation of the body-powered and the myoelectric prosthesis. It provides better functionality along with a lesser weight and a compact shape.
4. Targeted Muscle Re-innervation(TMR): TMR is a surgical procedure whereby the transferring nerves of the residual arm to the muscle sites that might still be functional. This would allow the amputee to use these functional muscles to provide signals to the prosthesis to control it. TMR requires a lot of post-surgery rehabilitation and recovery, because of which it is less likely to be chosen as an option.

2.2.2 Previous work

As mentioned above, the prosthesis industry flourished very rapidly and continuous research and developments were being carried out. Apart from advancements in the types of prostheses, a lot of research went into improving the experience of the users. In addition to enhancing the control of the prosthesis, a lot of work was also being done to create a more realistic and human-like arm. Since amputation is a very demotivating process, so amputees prefer prosthetic devices that look and feel natural. To address this problem, various 3D printing techniques were used to create and customize the prosthetic hands according to the requirements of the user.
In addition to this, haptic feedback [17] was incorporated into prosthetic hands. Users solely relied upon the difficult visual feedback of the prosthetic hand which caused users to drop the objects they were holding. To solve this problem, a sense of touch was introduced to the domain of prosthesis for which different technologies were used: force-sensitive resistors, piezoelectric sensors, and shape memory alloys. This provided sufficient feedback about the object's properties.
Another important advancement in this domain was the incorporation of computer vision instead of EMG signal to make the prosthesis more intelligent [18]. Large

data sets of hand movements and EMG signals were fed to complex machine-learning algorithms to allow the prosthetic hands to adapt to new environments. Some studies also implemented different AI strategies to predict the user's intention and aid the user in prosthesis control.

In summary, there have been continuous research and development in the field of prosthetic. The incorporation of haptic feedback and machine learning has made the experience easier and more natural. Moreover, advancements in printing have brought about significant changes in the aesthetics of the arm and boosted the self-confidence of the amputee. However, it must be noted that there are various research gaps that need to be filled and research must be done continuously to make the prosthesis more efficient, comfortable, and natural to use.

2.3 Existing upper limb prosthesis technologies

2.3.1 Vincent (2009)

Vincent prosthetic hand [19] was designed using a high-strength aluminum alloy. The muscle ligaments in the fingers are represented using springs between the proximal and the distal joints. Moreover, the thumb of the hand moves independently and moves separately. The skin of the prosthetic hand is similar to the natural skin because of the use of a soft shell. The prostheses combine 10 axes which are bi-directionally motor driven. The VINCENTev2 has 12 different grips along with 2 EMG signals, and its successor offers 14 different grip types. The Vincent hand also comes with a training app.

2.3.2 iLimb

iLimb [20] was the first multi-articulating prosthetic limb, meaning that every finger has its individual motor. This prosthetic limb was developed by Touch Bionics [21].

However, the thumb must be rotated manually to create grips of the prosthetic hand. iLimb gives complete control to the users regarding the speed of the opening and closing of different grips. The hand is developed using aluminum for strength and for it to be long-lasting. One important feature that differentiates iLimb from other devices is the customization it provides to the users. iLimb has a dedicated application for the users to customize different types of grips and provide a more user-centric product.

2.3.3 Bebionic Hand

The bebionic prosthetic hand is a type of myoelectric prosthesis and is very similar to iLimb with minimal differences. Every finger is independent of the other because of which the hand provides a lot of different grips. It can sustain a weight of 45kgs with a power grip of 140 newtons, which is why it is deemed as one of the most durable prosthetic devices. The thumb positions are customizable and have a total of 14 grip patterns along with proportional speed control.

2.3.4 Michealangelo Hand

The thumb, index and middle finger of the Michelangelo hand [22] are active while the ring finger and little finger are passive The thumb has an independent motion from the other fingers and has a flexible wrist joint. Adding to this, this prosthetic device is the most human-like and natural-looking prosthetic hand, however, the main drawback is its weight, which is about 746g.

2.4 Mechanical Design

Prosthetic hands are artificial hands that help people who have lost their natural hands. There are different kinds of prosthetic hands that work in different ways. Some use cables and pulleys to move like a real hand, while others use the person's

Figure 2.2: Existing prosthetic hands Michelangelo, iLimb, Vincent, Bebionic (left to right) [23]

own body movements. Another important thing to consider is how many ways the hand can move. The more ways it can move, the more things it can do, but it also makes it more complicated and expensive. Researchers are working on making prosthetic hands that can do different types of grasping, so they can do more things like a real hand. A very important concept in the mechanical design of the prosthesis is "Degrees of freedom", which is an indicator of the allowed movement of the prosthetic hand along the x,y, and z axes.

2.4.1 Weight of the prosthesis

How a prosthetic hand feels to the user in terms of comfort and happiness is highly influenced by its weight. Because of how it is linked to their body, some individuals still feel it overly heavy even if it weighs the same as a genuine hand. By developing new techniques for attaching the artificial hand, scientists are attempting to address this. However, a significant issue that has an impact on how individuals feel about it is weight. Many prosthetic users feel that their devices are overly weighty. According to a poll, the prosthesis' weight is one of the key factors in determining how effectively it will function.

2.4.2 Actuators and driving mechanism

A direct current (DC) motor is the most prevalent actuator used in prosthetics today, barring a harness driven by the user's body. These motors may be carried in the hand and are compact and lightweight. Because they are simpler to regulate, brushed DC motors are more frequently employed in prosthetic hands. Higher torque-to-weight ratios are possible with brushless DC motors, but more complicated motor control strategies are needed. Normally, sensors that can offer extra position feedback are included with brushless motors. Brushless DC motors are also anticipated to overtake other motor options as control electronics get smaller. For usage in prosthetic devices, all DC motors produce excessive speed and inadequate torque by nature. Drive reductions are therefore required to decrease the speed and increase the torque that the actuator provides. These motors may be utilised with gearing, lead screws, or even harmonic drives to decrease speed and enhance the restricted torque. The proximal phalange of each finger houses a single motor and gear train in the iLimb and Vincent hands. A tiny DC motor is employed in the FluidHand III (Forschungszentrum Karlsruhe GmbH; Eggenstein-Leopoldshafen, Germany) to power a tiny hydraulic pump that is contained inside the hand's palm. The pressure is then transferred to bellows at each joint via five independent valves. Utilizing a pressure-based system has the benefit of each finger joint's compliance, which enables the device to withstand abrupt hits. Numerous hands use nonbackdriveable mechanisms (NBDMs) to control the flexing of the fingers in conjunction with the motor. With the help of the mechanism's compliance, NBDMs enable the finger to retain strong gripping pressures without continuously draining the battery. The most popular NBDMs are roller clutches, worm drives, and lead screws.

2.4.3 Grip Patterns

The variety of grip patterns that are currently accessible is a crucial part of how far modern prostheses have progressed in terms of design and functioning. The unique method a prosthetic hand or arm may hold or move an object is referred to as a grip pattern. Prosthetics have advanced in recent years to feature a variety of grasp patterns, giving users more flexibility and control in their everyday activities. The pinch grip, enables users to pick up small objects between their index finger and thumb, the power grip, enables users to hold larger, heavier objects with more force, and the lateral grip, enables users to grasp objects between their fingers, these are a few of the most popular grip patterns. Other grasp styles include the tripod grip, which is similar to the pinch grip but uses three fingers instead of two, and the key grip, which enables users to turn keys and other tiny items. These grip styles allow users to pick up things by hooking the prosthesis over them. Overall, the functionality and utility of current prostheses for amputees have been substantially enhanced by the availability of different grasp patterns. Users may complete more chores more easily and live better lives with a larger variety of alternatives. It's crucial to remember that not all prosthetic limbs offer the same grasp patterns and that each one is unique to the demands of the user. To ascertain which grip patterns would best fit a user's demands, it is crucial to speak with a trained prosthetist.

2.5 Computer vision and prosthetics

Computer vision has been greatly used in the field of prostheses and robotics. the use of surface and texture information was another leap in the use of computer vision for determining the grasp. Kootstra et al [24] created an early cognitive vision system that only used edge and texture information to determine the grasp of an unknown object. Using a stereo camera, the information on the contours was used to generate a surface-based and contour-based grasp of a 2/3 finger gripper.

In addition to this, Lenz et al [25] used a two-tier cascaded system with two different models. The second model re-evaluated the detections from the first one. When the image was acquired of the objects that had to be picked up, the first network made a decision about the grip depending on the characteristics of the object. After this, the second network picked the most suitable option for the gripping spots that were classified by the first model. This computer vision system was implemented on a two-finger gripper and used a technique known as group regularization to balance the information acquired regarding the object.

Saxena et al [26] used a stereo camera for the purpose of grasping unknown objects. The location of the objects was determined without the generation of a 3D model. Triangulation was used to determine the grip of the prosthesis.

In another study, the object detection method was used along with information about the depth of the objects using ultrasound sensors. This approach estimated the size of the object and applied a rule-based algorithm to determine the grasp appropriately [27]

Additionally, Markovic et al [28] incorporated augmented reality (AR) to provide proprioceptive feedback, through visual feedback, to the user. The stereo camera provided the depth of the objects and AR provided feedback regarding the grip aperture size. Complex machine-earning models were trained regarding segmentation, 3-dimensional point cloud generation, and geometrical model fitting. The system showed an appropriate accuracy and a decrease in response time when run on a computer.

Further advancements included Kopicki et al [29] to introduce a one-shot learning mechanism. Thousands of grips were generated for images acquired by a depth camera. The combination of a contact model and a hand-configured model was optimized.

Moreover, Convolutional neural networks (CNN) became increasingly popular in this application. Ghazaei et al [18] trained a CNN model on a total of 500 images and defined four different grips for the class of the objects. The model achieved an accuracy of 88% and got an overall score of 84% when run in real-time on a standard laptop.

Field	Success (%)	Time(s)	Hand
P	93	~4	CyberHand
P	94	~1	SmartHand
P	~99	0.75	Michelangelo hand
R	87	1.2	2-finger gripper
R	20–60	N/A	2/3-finger gripper
R	93.7	13.5	2-finger gripper
R	77.8	13–24	Boris hand

Figure 2.3: A comparison of current prosthetic hands using computer vision [18]

2.6 Sensory feedback techniques

Various methods of sensory feedback have been studied in this section, including temperature [30] [31], vibration [32] [33], mechanical pressure and skin stretching [34] [35] [36] [37] [38] [39], electro tactile stimulation [40] [41] [42] [43] [44] [45] [46], audio feedback

2.6.1 Vibration Feedback

Vibrational feedback is a technique that utilizes small vibrators placed on the skin to activate mechanoreceptors in the skin to communicate information to prosthetic users. Although the feedback reduces the cognitive load required to pick up objects compared to using visual feedback alone, the benefits are not consistent among all users. Vibrotactile stimulation is helpful for improving a user's sense of embodiment in their prosthetic, but users must undergo training to develop the full benefit. Desensitization may occur after prolonged use, so intermittent pulses are recommended. Feedback systems with more than three stages may become unreliable. Vibrotactile feedback may also be used to control grasping force and identify different surface textures. However, it may not be as effective outside the laboratory [47]

2.6.2 Mechanotactile Pressure Feedback

Currently, temperature feedback is mainly used to communicate the force of grip and finger position to prosthetic device users [48] However, temperature feedback has the potential to provide additional information about the environment and potential dangers related to heat. Current research on temperature feedback focuses on using a Peltier element to produce heat on the upper arm in response to the temperature detected by the prosthetic hand. Studies have shown that temperature feedback can be accurately communicated to users, but it draws upon a lot of power and may not be a priority until advances are made with other feedback methods. Further research could investigate combining temperature feedback with other feedback methods to occur simultaneously.

2.6.3 Temperature Feedback

Currently, temperature feedback is mainly used to communicate the force of grip and finger position to prosthetic device users [48]. However, temperature feedback has the potential to provide additional information about the environment and potential dangers related to heat. Current research on temperature feedback focuses on using a Peltier element to produce heat on the upper arm in response to the temperature detected by the prosthetic hand. Studies have shown that temperature feedback can be accurately communicated to users, but it draws upon a lot of power and may not be a priority until advances are made with other feedback methods. Further research could investigate combining temperature feedback with other feedback methods to occur simultaneously.

2.6.4 Audio feedback

A study on the potential of using audio as a feedback mechanism for prosthetic devices was conducted. The study found that while test subjects were able to interpret two channels of audio, there was a significant delay and high cognitive load. Another study by Gibson and Artemiadis showed that subjects could use auditory feedback alone to pick up objects with a robotic hand, with the variance in volume representing the level of grasping force and the varying frequency corresponding to the location of two different regions of the hand. In a different approach, Gonzalez utilized different musical instruments to represent the movement and force of the prosthetic hand's fingers. However, further research is required to determine the effectiveness of these feedback methods in real-world scenarios with background noise.

2.6.5 Augment reality feedback

In one study, researchers used Google glasses to provide visual feedback to prosthetic hand users about their grip strength, contact time, EMG strength, and aperture angle. This information helped the subjects improve their performance in tasks that required different levels of strength, but they didn't rely on the EMG strength signals very much. Another study explored the use of augmented reality to provide feedback on grasping force and the angle of grasp closure. Although subjects relied more on force feedback than closure feedback, the authors found that incorporating feedback increased cognitive load and prolonged the time required to pick up objects. The angle feedback may be more valuable in tasks that don't allow for as much visual attention on the hand.

2.6.6 Hybrid feedback

The studies reviewed so far have only focused on communicating one sensation at a time, which limits the ability to convey multiple sensations at once. However, some studies have investigated the possibility of using multiple feedback methods simultaneously, either to improve the recognition of one type of stimulus or to allow for the communication of two different stimuli simultaneously. In a study [49], the author found that hybrid feedback of electrotactile or vibrotactile stimulation allowed subjects to identify nine levels of stimulation, whereas using either mode alone only allowed for the identification of a single sensation. The same authors also demonstrated that subjects could identify patterns from four stimulation devices that used a combination of electrotactile and vibrotactile stimulation with higher accuracy than similarly sized vibrotactile devices [41]. However, the testing was limited to able-bodied subjects, and the authors suggested that further improvements could be made by reducing the size of electrodes. While combining mechanical pressure and vibration has also been explored, an experimental prototype was built, and no testing was performed on subjects. The device was also very bulky.

2.7 Gap analysis

As the previous work is discussed regarding the existing prosthesis, there is a lot of room for improvement and research in the field of prosthesis. The number of amputees is growing every day and requires attention and more advancements. The gap noticed during the research work included various factors. Firstly, regarding the mechanical design, the degrees of freedom of the existing prosthetic hands were very limited and caused limited and artificial movement. This artificial movement is one of the reasons for the abandonment of the prosthesis. Adding to this, the computer vision algorithms integrated with the prosthetic hand control system achieved only a decent accuracy leaving a huge room for improvement in accuracy. Accuracy in grip determination is a crucial part of prosthesis success and providing a realistic experience for the user. The datasets used for training of computer vision algorithms are mostly non-relevant to daily life and the product becomes obsolete for the users because of which the abandonment rate has increased. Moreover, the computer vision algorithms that were incorporated were deployed and run on laptops and not on an edge computing device. This caused more latency and decision-making was not done in real-time, which was one of the drawbacks of the computer vision prosthetics.

Moving on, very few prostheses were able to provide haptic feedback and determine the force required for the grasping of certain objects, which could be a way forward in further improving the user experience.

Chapter 3

MECHANICAL DESIGN

This project includes the enhancement of the mechanical design of an existing prosthetic hand to ensure an accurate formation of grip patterns for the grabbing of different objects. A highly functional design of the prosthesis is very important for ensuring dexterity and user comfort which would allow the amputees to use the prosthesis without fatigue. The mechanical design is a crucial part for reducing prosthesis abandonment: the scope of this project includes the improvement of design and creating a functional prosthesis from an existing non-functional one. Additionally, carrying out tests regarding the design and making sure that it is accurate and safe for the amputees to use is also one of the aims of this project.

Chapter 3 discusses the characteristics and specifications of the existing prosthesis, its limitations and how this project tackles these limitations and creates a highly functional, and user-centric mechanical design.

3.1 Previous Design

The prosthetic hand that was previously available has six degrees of freedom (6DOF). All four fingers and the thumb of the actuation system were equipped with linear

Figure 3.1: Previous Design from DE-37 Dept of Mechatronics Engineering, NUST

actuators. For the rotation of the thumb, worm gears that can drive other gears but not themselves (unidirectional) were utilized. In order to 3D print the hand, premium SLA printing material was used for assembly. Additionally, stress tests were carried out on the hand utilizing a force of around 20 Newtons, or about 2 kg, and the ANSYS Workbench's Static Structural Analysis was used to carry out the stress and strain analysis. The prosthetic hand still featured a number of drawbacks and restrictions even though it was an enhanced version of the original design. One important concern was that the hand couldn't fit inside a glove due to the thumb's huge size.

3.2 Modified Design

The improved prosthetic hand solved a number of issues with the older design's limitations. The main alteration was a smaller thumb, which was implemented as a key. The prosthetic hand's compliance with the proportions of a typical glove was made possible by reducing the thumb's size, which had previously been a challenge since it was too big.

The project focused on fixing any faulty structural linkages in the prosthetic hand. These previously broken components were meticulously repaired and precisely machined to return them to their best operational state. The hand's general functioning and robustness were guaranteed during the restoration procedure, enhancing its dependability as a prosthetic device.

The placement of the thumb also underwent a major change in order to accommodate the reduced thumb size and produce a seamless fit within the glove. The prosthetic hand's compatibility issue with conventional gloves was successfully handled by carefully relocating the thumb. This modification made it possible for the glove to snugly and pleasantly enclose the hand, facilitating usage and improving the wearer's entire experience.

After the thumb was successfully included in the glove, the team moved on to investigate and use a variety of unique grip patterns. During this phase, careful study and testing were conducted to identify the ideal grip arrangements that would allow the prosthetic hand to imitate real hand movements. The hand acquired increased versatility and adaptability to fulfill the wearer's unique demands in various occupations and activities by including a variety of grip patterns.

The prosthetic hand design was refined overall thanks to the repeated increases in a thumb size, fixes for broken linkages, relocation of the thumb, and the subsequent introduction of various grip patterns. These developments were made in order to overcome the initial flaws and restrictions, which ultimately led to a more useful and practical prosthetic solution.

The upgraded prosthetic hand overcame the prior difficulty of fitting under a glove owing to the size of the thumb through an iterative design process and the persistent work of the design team. Furthermore, the team's dedication to reaching a high level of performance and user happiness is demonstrated by the repairs and machining of damaged links along with the relocation of the thumb. By utilizing several grip styles, the hand was able to regain its dexterity as well as its adaptability to a variety of tasks.

This iterative process illustrates the commitment to ongoing innovation and development within the prosthetic industry, as designers work to develop more practical and flawlessly integrated solutions for those who have lost limbs.

Figure 3.2: Prosthetic hand (inside the glove)

3.3 Motion Mechanism

3.3.1 Fingers

A linear actuator-based actuation system was used in the design to facilitate finger movements. A unique linear actuator was installed in each proximal phalanx of each finger to make movement easier. However, a 4-bar linking mechanism was used in combination with a single linear actuator to provide an underactuated system for the middle and distal phalanges. The middle and distal phalanges were considered as a single joint in this arrangement.

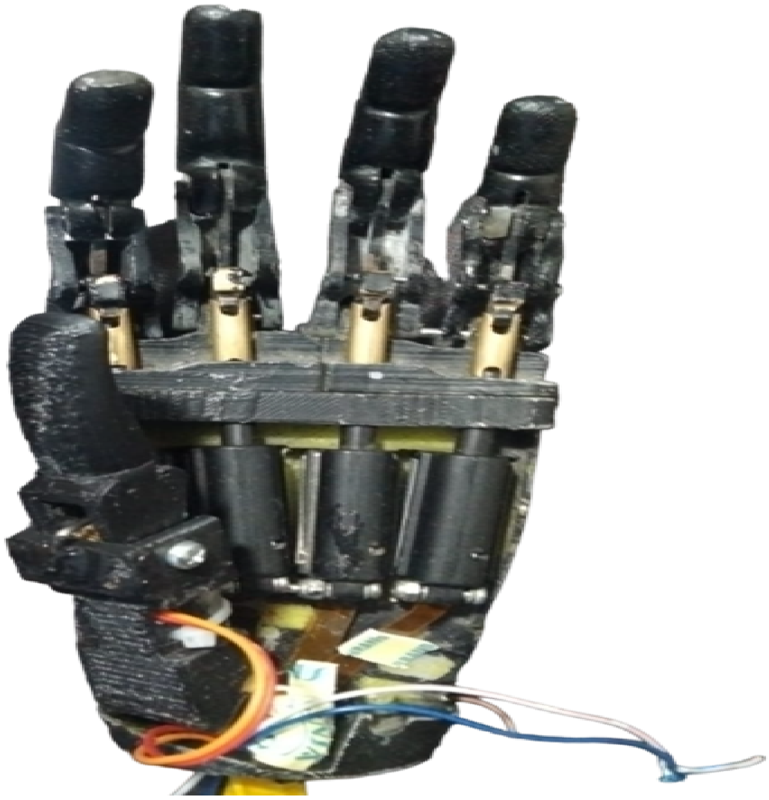


Figure 3.3: Prosthetic hand (outside of the glove)

The proximal phalanx may be rotated across a range of 90 degrees by the linear actuator used in the design. The finger's flexion and extension were significantly increased because to this rotation. The distal and middle phalanges may both rotate by an angle of 60 degrees. With the aid of this carefully regulated rotation, the hand was able to move the fingers naturally, simulating the flexion and extension of a human hand's joints.

The prosthetic hand was able to move its fingers effectively and functionally because to a combination of separate linear actuators and a 4-bar connection mechanism. The fingers' movements may be precisely controlled through the use of linear actuators, enabling a fluid and precise articulation. The design's general simplicity and dependability were enhanced by the underactuated mechanisms for the middle and distal phalanges, which guaranteed a perfect fit with the wearer's residual limb.

Overall, the prosthetic hand was able to mimic a variety of natural finger movements

Figure 3.4: Anatomy of hand [50]

thanks to the use of linear actuators and a 4-bar connecting mechanism. This design strategy not only guaranteed excellent dexterity but also gave users of the prosthetic hand an intuitive and responsive experience.

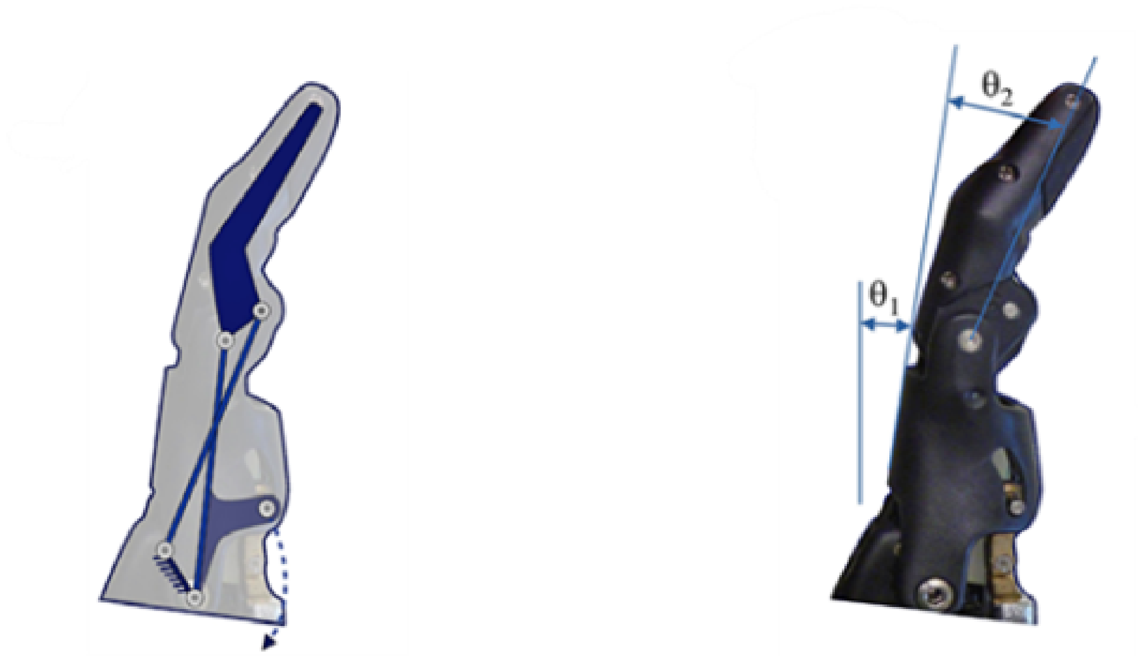


Figure 3.5: Under-Actuated mechanism of fingers

A circular worm gear pair mechanism was used to move the thumb in the prosthetic hand design. The rotation of the thumb along the palm was made possible by this mechanism. A round worm gear pair was also used to accomplish two important goals.

First off, the size of the motor needed to move the thumb was reduced thanks to the use of a round worm gear pair mechanism. The effectiveness of the worm gear pair, which resulted in a more compact overall thumb design, allowed for this reduction in motor size. The prosthetic hand's functionality and cosmetics were much enhanced by this size decrease since a lower thumb size allowed for better integration inside the hand structure.

The usage of a worm gear pair also offered two other advantages. The ability to self-lock guaranteed that the thumb remained firmly in position without any unintentional movement, which was the initial benefit. The thumb was more stable and under control overall because to this self-locking function, which also allowed for accurate alignment and prevented any unwanted rotation.

The worm gear pair's high-end torque production was its second benefit, made possible by the mechanism's built-in gear ratio. Due to the high torque production, the thumb was able to apply enough force to be useful and strong like a normal thumb. The prosthetic hand design gained an improved gripping capacity by including the worm gear pair, making it appropriate for a variety of occupations and activities requiring a firm and solid grasp.

A specific thumb mount was created in order to enhance the thumb's functionality even further and guarantee that it is compatible with the glove. This mount changed the thumb's position, making it possible for it to be placed and fastened to the glove in a secure manner. This change in the thumb's position was vital in overcoming earlier difficulties with the prosthetic hand fitting within a regular glove, improving user comfort and overall utility.

The angle of the thumb's metacarpal bone link was extended to 45 degrees in the final design of the prosthetic hand to match the angle of the glove. This modification guaranteed improved compatibility and a more organic appearance while the hand is at rest. The thumb's proximal and distal phalanges were also regarded as a single joint, permitting only a 60-degree rotation. This limitation kept the thumb inside the correct bounds and kept it from extending beyond its intended.

In conclusion, the thumb design benefited in a number of ways from the inclusion of a round worm gear pair mechanism. These featured a smaller motor, the capacity to lock itself, and high-end torque output. These benefits helped to increase the thumb's functioning and grip strength. Additionally, the thumb mount allowed for simple placement and seamless integration within the glove, overcoming past restrictions and assuring a secure fit for the user.

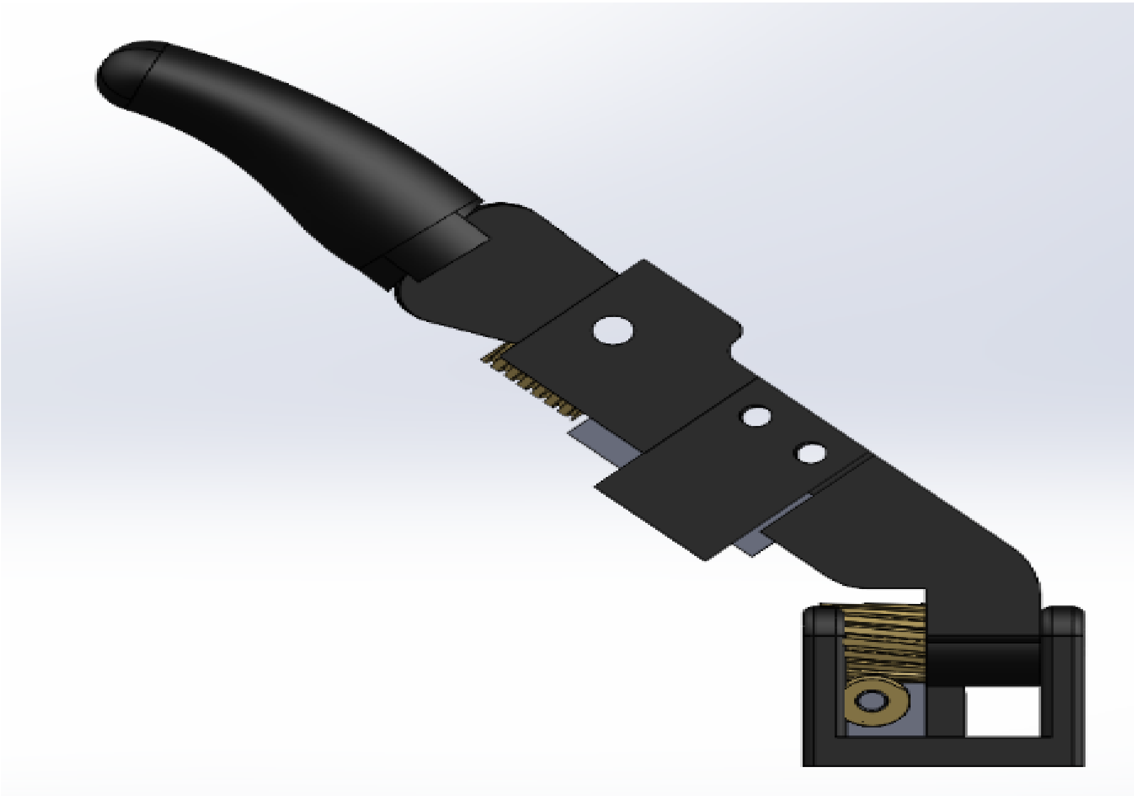


Figure 3.6: Improved thumb design - A

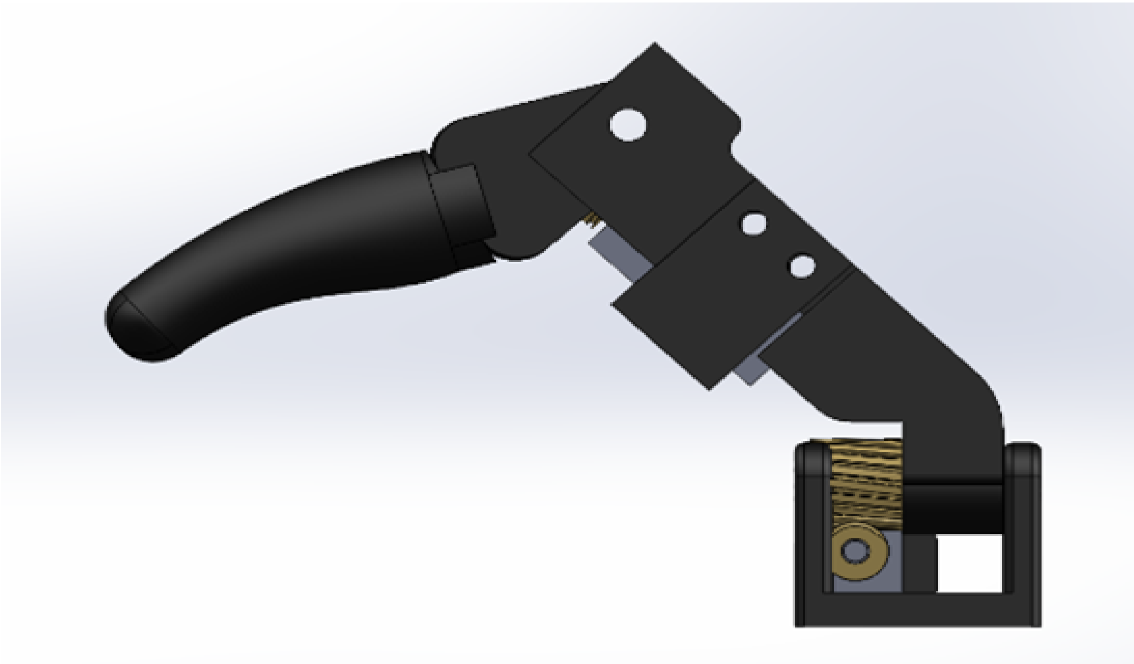


Figure 3.7: Improved thumb design - B

3.4 3D modeling and prototyping

In the next step of mechanical design improvement and to implement the design concept, 3D models were created and analyzed. After the creation of 3D models, the different prototypes were printed. A total of four iterations were carried out to complete the design requirements and enhance the design. The printing of the first three prototypes were completed using PLA plastic material. Following is the method followed in each iteration of prototype development:

1. **First iteration:** This iteration of the prototype development was for the proof of concept and to make sure that the design concept was practical. Hence, in this iteration a 3D model was created in which the space required by the N20 gear motor was specified and possible positions of the gear motor were explored. Moreover, the exploration of these positions were tested by checking if the thumb would not compromise on its range of motion and was rotating completely. The prototype was printed using PLA plastic material and was

approved and tested. However, the rotation of the thumb was not limited.

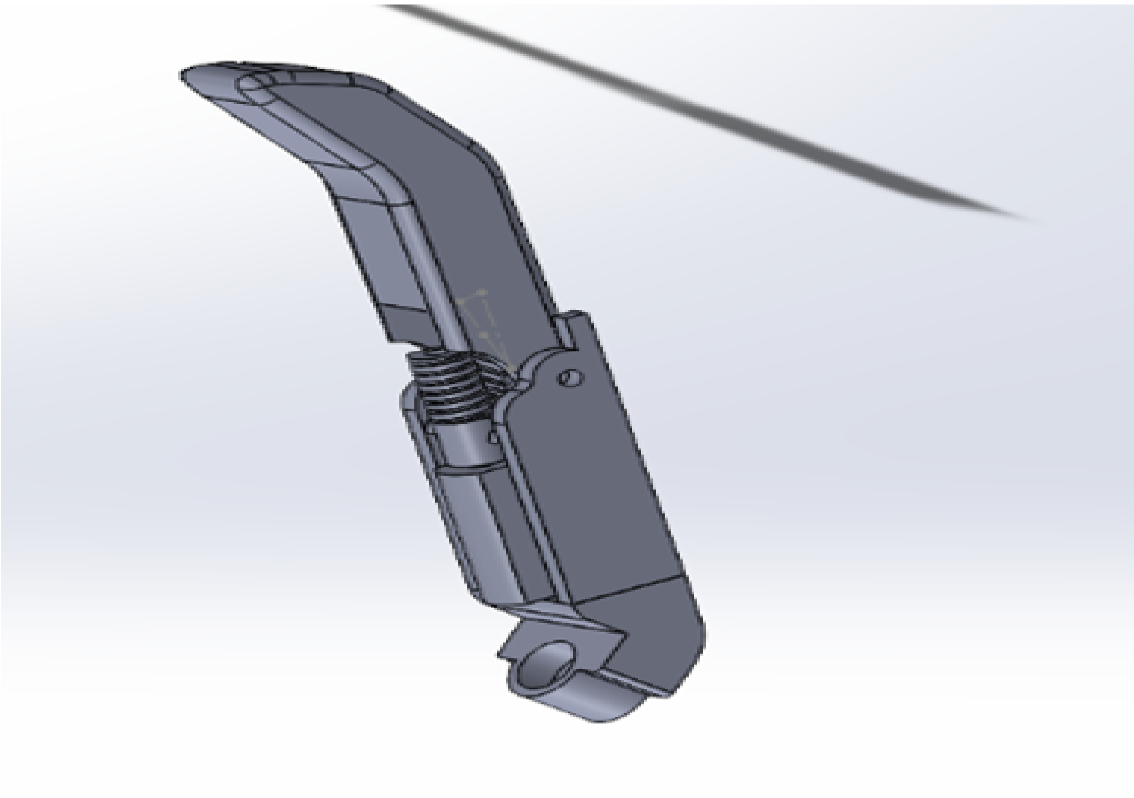


Figure 3.8: First iteration improved design

2. **Second iteration:** In the second iteration, the movement and the rotation of the thumb was restricted from zero degrees to sixty degrees in order to prevent the thumb to be moving limitless that would create a hindrance in the formation of many important grip patterns. After limiting the rotation of the thumb and making sure that the functionality of the prosthesis hand would not be compromised, a second prototype was PLA printed and tested. It was found that this design was thinner than expected and had some minor faults.

3. **Third iteration:** The third iteration was the proper design of the three iterations and the aim of this design iteration was to correct the faults and weaknesses of the second design. As stated, the previous design was thinner than expected, so the design was made according to the thickness requirements

and the minor weakness of the previous design were corrected. As a whole, this iteration was a process to enhance the design created in the second design iteration.

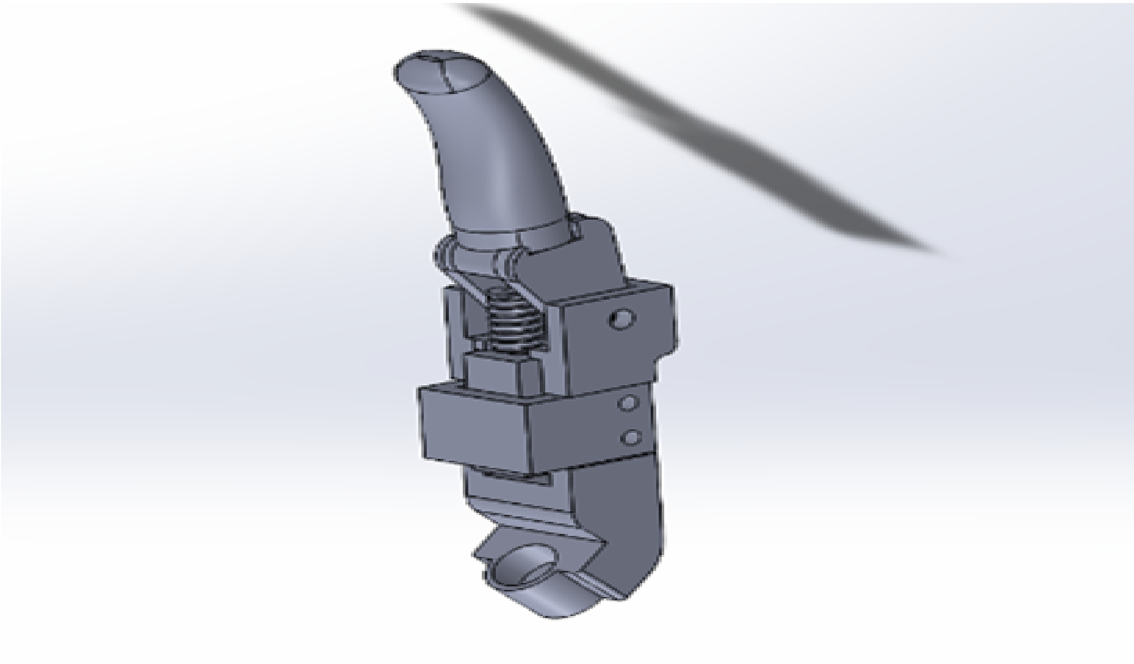


Figure 3.9: Third iteration improved design

After the rigorous testing of the mentioned designs, a fourth and final design iteration was created to make a flawless design of the the thumb that would allow the perfect fitting of the hand into the glove.

3.4.1 Final Design

The fourth and the final design iteration was printed using the SLA printing and fulfilled the design requirements. Using the N-20 gear motor the size of the thumb was decreased by more than half. The height of the thumb was decreased from 100mm to 60mm, which is a total of 60mm. Moreover, the width and breadth of the previous design was 30 x 20, which was decreased to 15 x 20 in the enhanced design. Apart from the design the unnecessary movements of the thumb were removed and realistic movements were incorporated.

Figure 3.10: Final modified design

The results of the simulations are as follows:

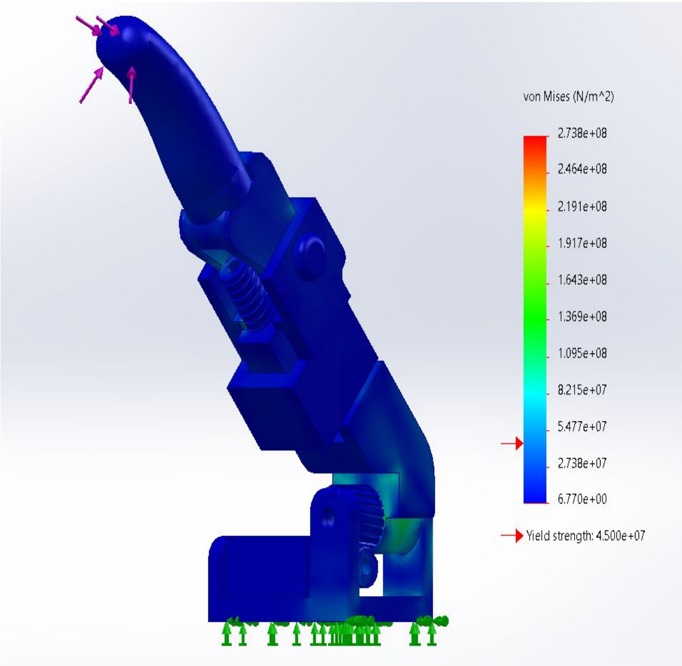
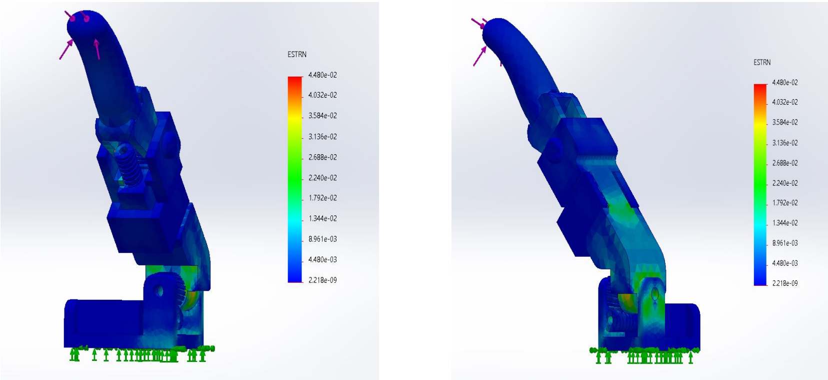


Figure 3.11: Stress Analysis of Modified design



(a) Front View (b) Back View

Figure 3.12: Strain Analysis of Modified design.

3.5 Actuators

The PQ-12 actuator is made particularly to apply force for pushing or pulling a load over the length of its stroke. According to the Load Curves, variables like the load being moved and the mount angle affect how quickly it moves. By lowering the driving voltage, the actuator speed may be slowed down.

Unless the applied load exceeds the back drive force, the actuator will remain its position when power is cut off. It is crucial to remember that continuously stalling the actuator or exposing it to extended stalling times will drastically shorten its lifespan. Therefore, it is advised to test the actuators within each individual application to determine how long they will really operate .

The prosthetic hand's thumb was designed to use the N20 motor, a tiny gear motor. This motor was chosen primarily because it produces a lot of torque, which is important to provide the thumb movement the necessary strength and usefulness. When used in conjunction with a worm gear mechanism, the device becomes self-locking, making it exceedingly challenging or impossible to accidentally slide the thumb back from the correct position. This self-locking mechanism guarantees that the thumb stays firmly in position, giving the person using a prosthetic hand stability and control.

Table 3.1: Comparison of PQ12 and N20 actuator

Model	PQ-12	N20AGB100-08230
Gear Ratio	30:1	150:1
Voltage	6V	6V
Mass	21g	24g
Max Force/Torque	18 N	0.1176 Nm
Speed	28 mm/s	120 rpm
Stall Current	210 mA	120 mA

Table 3.2: Specifications of worm gear pair used with N20

Model	Step Gear	Fit Worm Shaft
Hole	4mm	3mm
Length	12mm	12mm
Teeth Outer Diameter	11mm	7mm
Teeth Thickness	5mm	-
Step Diameter	9mm	-
Teeth	20	9T

Figure 3.13: Left: PQ12 actuator, Right: N20 actuator [51] [52]

The worm gear pair used with N20 motor has the gear ratio of 1:20 and the rest of the specs are as follows:

The calculation done for the selection of the motor and gear ratio to match the speed and force of linear actuator was as follows:

Motor speed: 120rpm

Gear ratio: 1:20

Output speed(W): 6rpm

Thumb length(r): 60mm

$$V = rW \tag{3.1}$$

$$V = 60 \times \frac{6 \times 2 \times \pi}{60} = 37.68mm/s \tag{3.2}$$

This speed is the nearest one we could find with the linear actuator. Similarly for force:

$$r = 60mm \tag{3.3}$$

Since gear ratio is 1:20, torque would be multiplied by 20

$$T = 0.1176 \times 20Nm \tag{3.4}$$

$$F = \frac{2.352}{60 \times 10^{-}3} = 39.2N \tag{3.5}$$

So, thumb can provide approximately 40N of force while each finger can provide.

3.6 Conclusion

In conclusion, a crucial component of the study was to enhance the mechanical design of the prosthetic hand. By improving both usability and functionality, this project attempted to keep patients from giving up their prosthetic hands out of frustration. The procedure included a number of important elements that were fundamental to its overall success.

First, the transformation of the non-functional hand into a completely functioning one required extensive testing and subsequent pin repair. The pin mechanism's problems or shortcomings were found and fixed, making the prosthetic hand more dependable and useful. This advancement made it possible for patients to easily do a variety of jobs and successfully regulate the hand's movements, enhancing their independence and quality of life.

The redesign of the thumb and its purposeful location within the prosthetic hand also represented a significant advance. This invention was particularly significant since it made it possible for the hand to fit comfortably inside the available aesthetic glove. The prosthetic hand was given a more realistic look and movement by paying close attention to the proportions, articulation, and alignment of the thumb. Patients' trust and acceptance of the prosthesis were increased by the aesthetically beautiful product that was made available to them and easily integrated into their overall look.

The enhanced design also served to shield the prosthetic hand from any harm brought on by outside forces. The glove protected the hand from accidental collisions, harsh environments, or potential wear and tear because the thumb and other parts were safely enclosed inside the glove. This design improvement, which protected the prosthetic hand, not only increased its longevity but also decreased the frequency of repairs or replacements, minimising inconvenience for the patients.

Overall, a functioning, user-friendly, and visually beautiful result was only possible because to the diligent attention to mechanical design advancements in the prosthetic hand project. The hand became dependable and responsive after extensive testing and pin corrections. The prosthetic thumb's novel redesign and thoughtful positioning within the prosthetic hand allowed for seamless integration with a cosmetic glove, providing patients with a natural-looking and aesthetically pleasing option. This design improvement also offered defence against external harm, assuring the prosthetic hand's durability. The enhanced mechanical design effectively met the demands and preferences of the patients by integrating utility, aesthetics, and durability, lowering the risk of abandonment and improving their entire experience with the prosthetic hand.

Chapter 4

ELECTRONICS

4.1 Electrical Hardware

This project is an amalgamation and integration of the software and the hardware components. The scope of this project included the deployment of machine learning models on an edge-computing device for a faster response. This chapter specifically discusses the electronic components used in the circuit and how these components communicate with each other to complete the grip pattern determination iteration. Moreover, it is a complete and detailed overview of how the entire circuit works together in creating the desired results.

4.1.1 Raspberry pi 4b

In the initial phases of the project, a great amount of research went into the selection of an appropriate edge computing device for the deployment of machine learning models. The requirements of the edge computing device were as follows:

1. Appropriate size of the edge computing device since it had to be embedded on the prosthetic hand.

2. Sufficient computing power to run and deploy the machine learning models.

Table 4.1: Specifications of Raspberry pi 4

Name	Specification
Processing power	64 bit @ 1.8 GHz
Bluetooth	Bluetooth 5.0 BLE
Size	85.6mm × 56.5mm
Availability	Readily available in Pakistan

3. Capability of in-built Bluetooth system for communication between different modules.

4. Availability of the edge computing device along with its cost being within the appropriate budget.

After careful consideration and research of all the above-mentioned criteria, we were able to shortlist some of the edge-computing devices to fulfill this purpose. Hence, the final decision about the selection was Raspberry pi 4.

Specifications Raspberry pi 4 were sufficient for our purpose Figure 4.1 is an image of the Raspberry pi 4 used in the project.

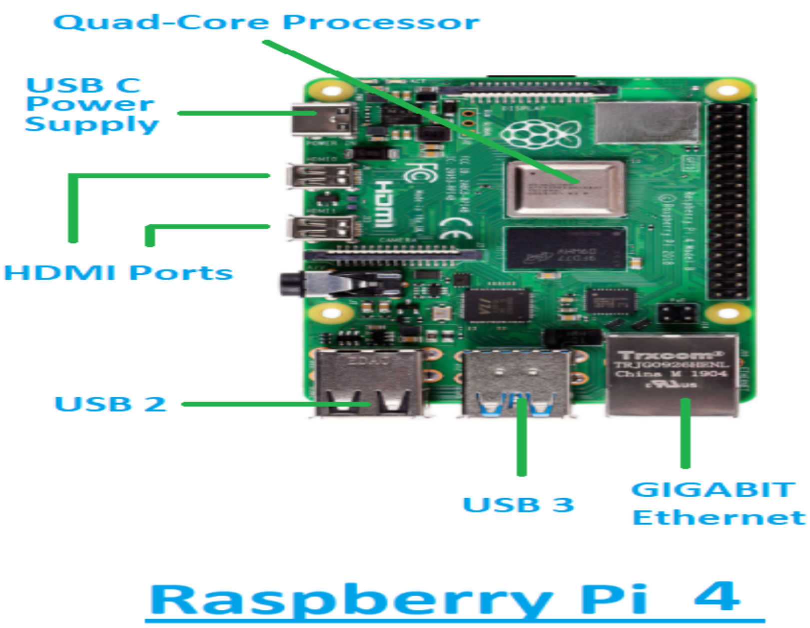


Figure 4.1: Raspberry pi 4 model b [53]

4.1.2 Arduino

The Arduino Uno was used as the control system of the prosthetic hand. Firstly, the Arduino is connected to the L298N mini motor driver. The Arduino is responsible for generating and providing a signal to the motor driver. In the case where the Arduino sends a high signal (1) to the motor driver, the motor driver causes the finger to open. While, when the Arduino sends a low signal (0), the motor driver causes the fingers of the prosthetic hand to close.

The determination of the type of signal being sent to the motor driver by the Arduino is dependent upon the signal received from the EMG sensor. Arduino is also connected to the EMG sensor which decides the type of signal that needs to be sent. After applying signal processing techniques ref on the EMG sensor, if the signal acquired is greater than the specified threshold a high signal is sent to the motor driver. Alternatively, a low signal is sent when the acquired signal is below the threshold.

Figure 4.2: Arduino Uno [54]

Arduino also communicates with the Raspberry Pi module through serial communication. It receives real-time data from the raspberry pi to determine the type of grip that is required for the object that has to be picked up.

The main objective of using an Arduino is for it to act as an intermediary between all the other modules and create a control system for the prosthetic hand.

4.1.3 Motor driver

L298N mini motor driver

L298N is a dual H-bridge motor driver, it allows for the control of speed and direction simultaneously. The mini motor driver L298N provides a continuous current of 1.5A and a peak current of about 2A. The small size of this motor driver, 24.7 * 21 * 5mm, along with its light weight makes it a perfect choice for use in this project. It is able to drive DC motors that have voltages between 2V to 10V. The motor driver can operate two DC motors at the same time or a four-wire two-phase step motor. The product parameters are listed below:

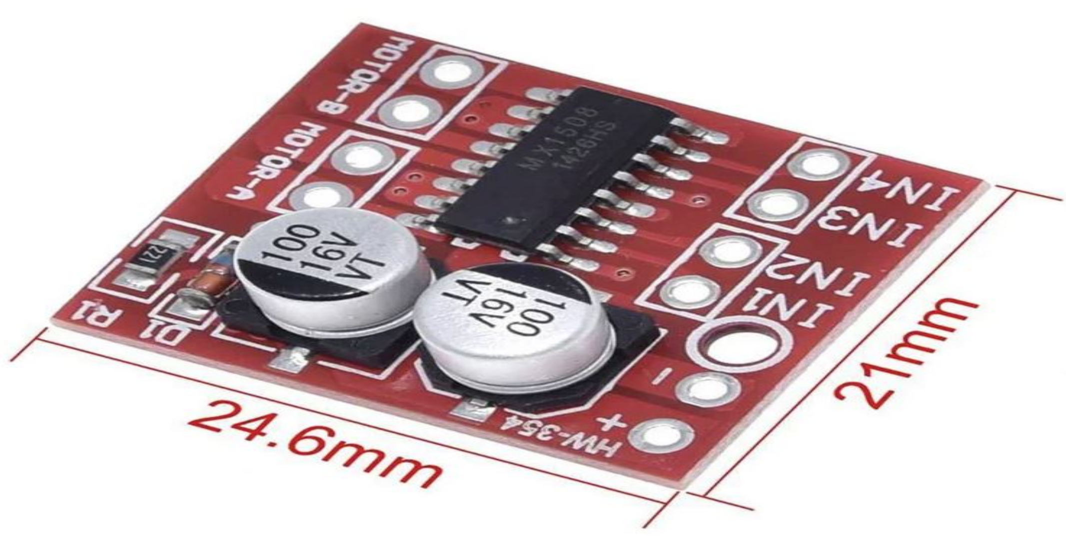


Figure 4.3: The mini motor L298N [55]

1. Supply voltage 2V-10V

2. Signal input voltage: 1.8V-7V
3. Operation current = 1.5A, Peak current ¡=2.5A, standby current ¡= 0.1 .
4. Internal thermal protection circuit
5. Size: 24.7mm x 21mm x 5 mm

4.2 Buck converter

The battery being used in the circuit provides 12V, however, the circuit operates at 5V. In order to achieve the operational voltage, a DC-to-DC buck converter is used. Buck converters are used to step down the input voltage and provide a lower voltage but step up the current and provide a larger current. The primary purpose of the Buck converter is to convert a higher input voltage to a lower output voltage.

The components of a buck converter include an input voltage source, an inductor, a switch, a diode, and an output capacitor. One of the main principles used by the buck converter is that of Pulse Width Modulation (PWM). The duration of the switch on and off states are varied because of which the average output voltage is regulated. Another important use of the buck converter is its characteristic of reducing power losses. Although it reduces the voltage but makes sure that minimum power is lost in this process, hence it has the advantage of excellent efficiency.

The use of the buck converter was essential in this project for the purpose of reducing the input current to the appropriate value without considerable loss in power.

4.3 Sensors

4.3.1 ACS712 current sensor

For the purpose of measuring the current being consumed by the motors, a Hall effect-based current sensor was used: ACS712. The sensor uses its conductor to measure and sense the current that is being applied across it.

As mentioned above, ACS712 uses the hall effect to determine the current across it, which enables it to measure the current without being in direct electrical contact. To accomplish this, a hall-effect transducer is used which measures the magnetic field produced by the current flow, and by determining the power of this magnetic field, ACS712 can accurately measure the current flowing.

The advantages of the ACS712 include its ability to measure positive and negative currents. This characteristic makes it appropriate for current measurement in bidirectional current measurement applications.

In addition to this, this component is available in different sensitivity values and allows flexibility to the user to choose according to their requirements. Moving forward, ACS712 has a mechanism to minimize power loss by using an internal conductor with low resistance. It also incorporates an amplifier for the purpose of integration with analog-to-digital converters.

Further, it includes inbuilt protection for high current values that could damage the integrated circuit (IC).

In summary, ACS712 was an efficient and suitable device for this project because of minimum power loss, and its inbuilt safety precautions.

4.3.2 EMG Sensor

This project uses a surface electromyography sensor (sEMG) to measure the electrical signal produced by the muscles of the user when the user makes an intention to move their muscles. The **Otto Bock EMG sensor** is used in this project since it is one of the most widely used sensors for the application of prosthetics. The electrodes of the Otto Bock sensor are four times more powerful than the old ones. It has a lower sensitivity to high and low-frequency interferences and has effective noise filtration. The Otto Bock EMG sensor is made out of titanium, a relatively allergy-free material, which makes the sensor suitable for patients with allergies. Moreover, it has enhanced sensitivity in low muscle signal range and increased differential in high muscle signal range to broaden the spectrum of potential patients [56].

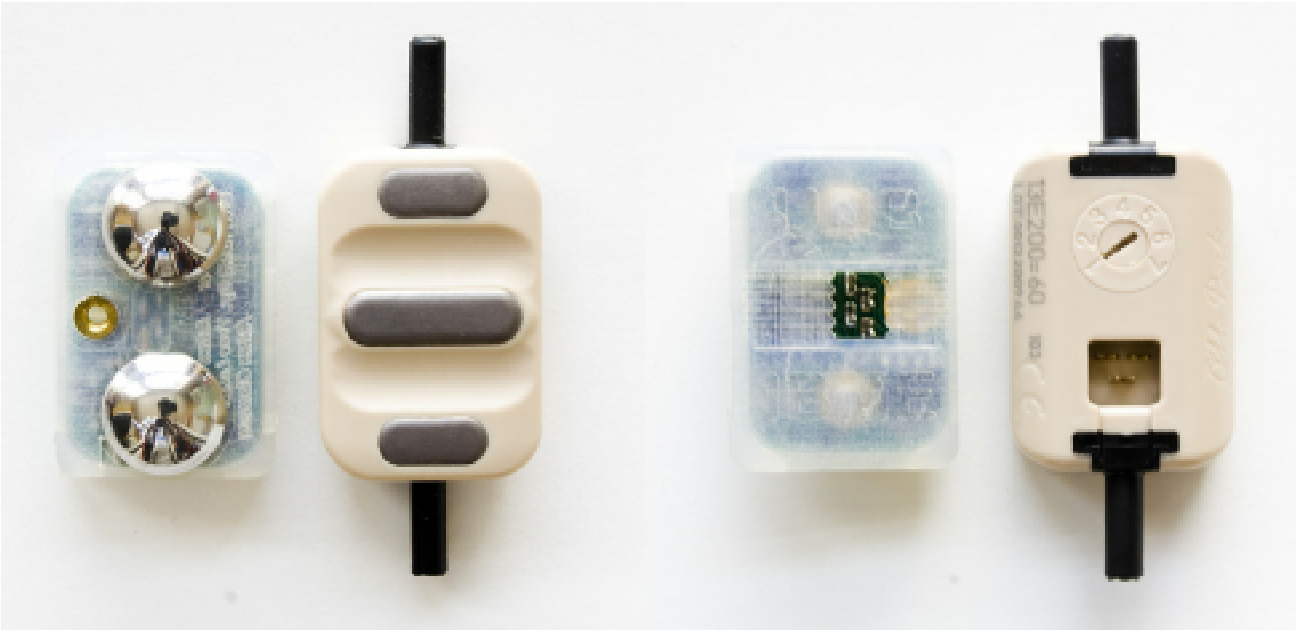


Figure 4.4: The Otto bock EMG sensor [57]

EMG signal acquisition and processing

The remaining muscles of the patient after the removal of the upper limb are used to acquire an EMG signal. When the patient intends to move a muscle, small electrical signals are produced which are recorded by the electrodes. The EMG sensor is placed on these remaining muscles and the electrodes present on the EMG sensor capture these electrical signals.

After the successful acquisition of the EMG signal, the signal was smoothed and the noise was removed using pre-processing techniques. In the next step, a thresholding algorithm was implemented to perform processing on the acquired signal. After rigorous testing, it was found that the active signal range was at about 650. Hence, the threshold value for an active signal was kept at 650. Any signal above the value of 650 would indicate an active signal and would produce a high output, while any value below 650 would be considered as an inactive signal and would produce an inactive signal.

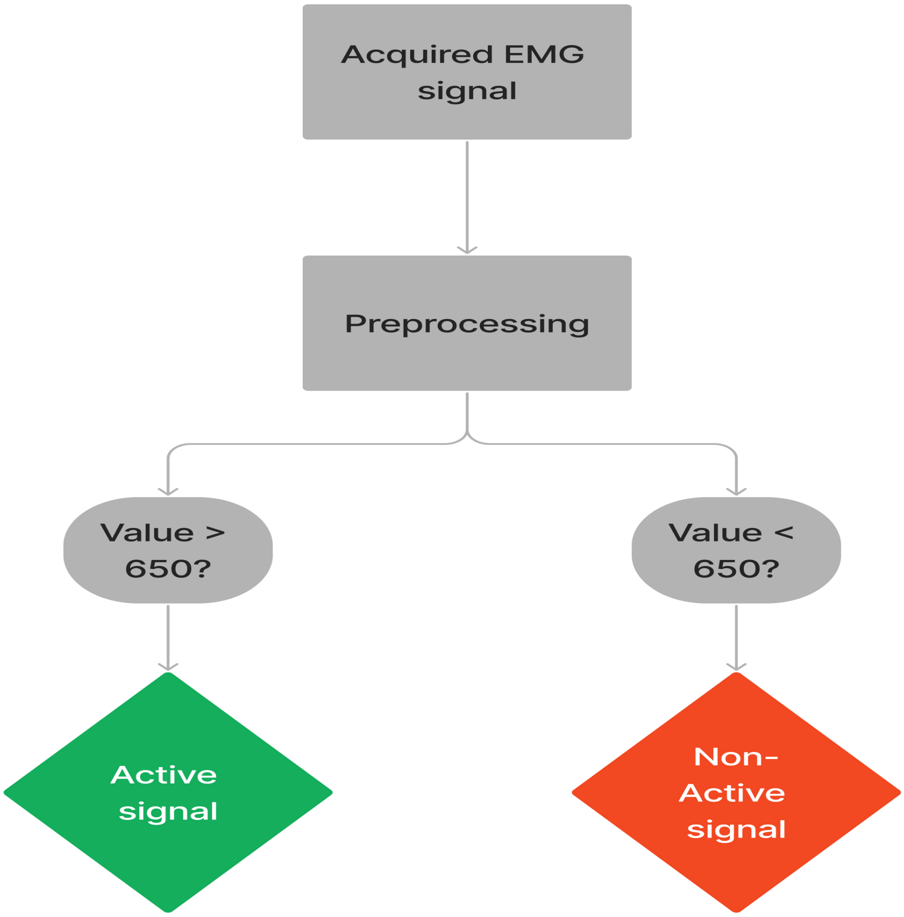


Figure 4.5: Flow diagram for thresholding algorithm

4.4 Communication between components

4.4.1 Communication between Arduino and Raspberry Pi

The Arduino (control system of the prosthetic hand) and Raspberry pi have serial communication between them. The serial connection was established using the receive (RX) and transmit (TX) pins. It is important to note, however, that the Raspberry Pi 4B operates at 3.3 V while the Arduino operates at 5V because of which there was a challenge in creating a connection between them.

To solve this problem, a voltage shifter was used that was used to change the voltage to an appropriate value for both devices. When the signal was being sent from the Raspberry Pi, the 3.3 V was stepped up to 5V for being received at the Arduino. However, when it was the other way around, 5V was stepped down to 3.3V for it to be received by the Raspberry Pi. The main objective of the voltage shifter was to regulate the voltage as required.

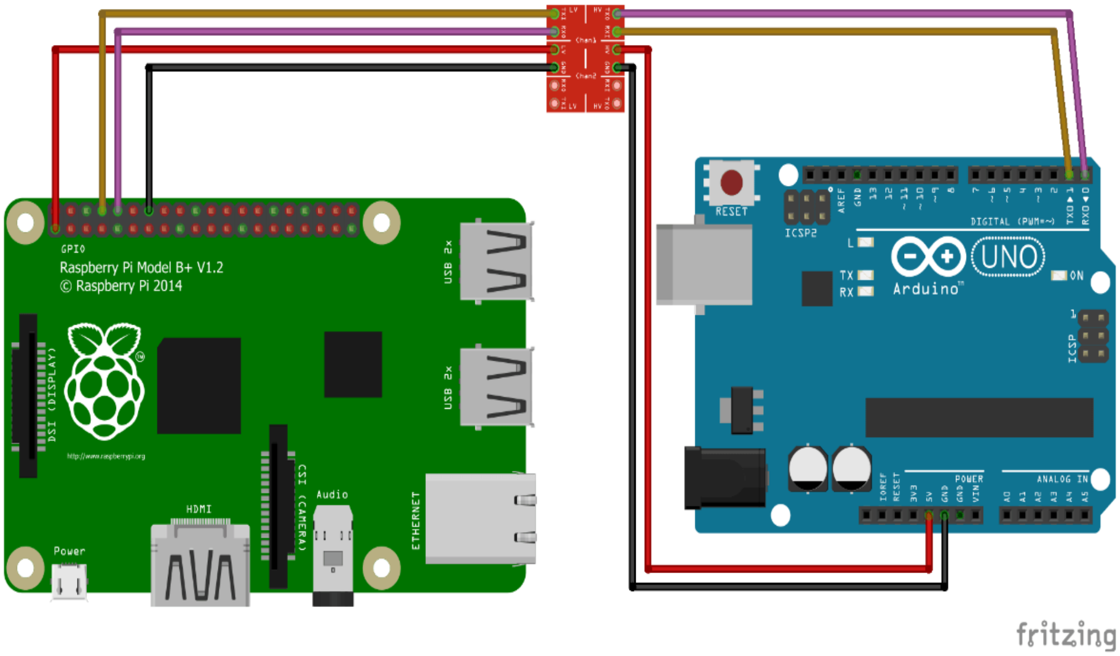


Figure 4.6: The connection of Raspberry Pi and Arduino through RX/TX pins [58]

4.4.2 Communication between Mobile Application and Raspberry pi

This project includes the development of a mobile application to provide customization to its user. The user will be able to change grip patterns depending on their requirement. Hence, for this to work it was imperative for the raspberry pi to be connected to the mobile application. Since the raspberry pi has a 5.0 in-built Bluetooth, a Bluetooth connection was established between the mobile application and the raspberry pi. The library used in the flutter framework was **Flutter Blue-**

tooth serial. A serial communication between the two was established, using the RFCOMM protocol. The RFCOMM protocol is a type of Bluetooth protocol that has a set of transport protocols. RFCOMM provides a reliable data stream, multiple concurrent connections, flow control, and serial cable line settings [59]. The following steps were taken to successfully create an RFCOMM connection:

1. Await a Bluetooth connection using the Flutter Bluetooth serial library in Flutter
2. turn on the Bluetooth on the raspberry pi (using command line interface(CLI) or GUI) - the commands for CLI are as follows
   - discoverable on
   - pairable on
   - agent on
   - default-agent
   - scan on
3. Pair your phone to the raspberry pi by using **pair (your device mac address)**
4. Open the RFCOMM port at the raspberry pi using **rfcomm watch hci0**
5. press the connect button in the mobile application to establish the RFCOMM connection successfully

4.5 The circuit workflow

The entire circuit of the project is a complex one, this section discusses the step-by-step connections and workflow of the entire circuit. Firstly, a 12V battery is used for providing voltage, this voltage is stepped down to 5V using a buck converter.

This 5V voltage provides power to the Arduino and the motor driver. Further, the Arduino is connected to the Otto Bock EMG sensor. Moreover, the current sensor is placed between the motors for the fingers which determines the amount of current being drawn from the motors. At the user intention, a signal is received by Arduino, where pre-processing and a thresholding algorithm are used to determine if the signal is an active signal or not.

The raspberry pi is connected to an external power source and to Arduino to which it is communicating. The Raspberry Pi sends the type of grip that is to be made by the prosthetic hand. The Arduino receives a signal from the EMG sensor and the raspberry pi, after which it sends a signal to the motor driver to move the intended motor to make the determined grip.

Further, the Raspberry Pi is also connected to the mobile application via Bluetooth, and a signal sent by the mobile application for a certain grip would be sent to the prosthetic hand, and a live feed of the camera would be available on the mobile application.

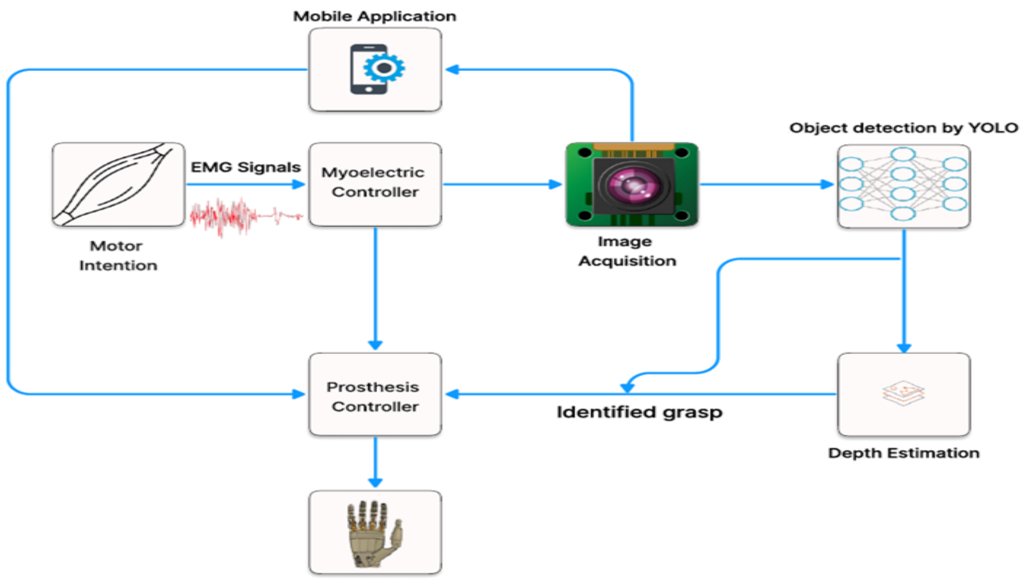


Figure 4.7: The workflow

Chapter 5

SOFTWARE

The software section of our project consists of two main subsections: computer vision and a mobile application. The computer vision section focuses on the development of a computer vision system that enables our smart prosthetic hand to detect objects in real-time and send a signal to the main controller to make a grip according to the detected object. The mobile applications section focuses on the development of a mobile application, that allows users to view the remaining battery of the prosthesis and send grip commands directly to the prosthesis.

5.1 Computer Vision

5.1.1 Image Acquistion and Pre-processing

The acquisition of high-quality images is a critical component of any computer vision system. In our project, we use the Raspberry Pi Camera v2 to capture images of objects The Raspberry Pi Camera v2 is a custom-designed add-on board for the Raspberry Pi, featuring a high-quality 8-megapixel Sony IMX219 image sensor and a fixed focus lens. The camera is capable of capturing static images with a resolution of 3280 x 2464 pixels and supports video recording at 1080p30, 720p60, and 640x480p60/90. The camera attaches to the Raspberry Pi via the dedicated CSi interface, which is designed specifically for interfacing with cameras. The camera board itself is tiny, measuring around 25mm x 23mm x 9mm and weighing just over 3g, making it ideal for mobile or other embedded applications where size and weight are important.

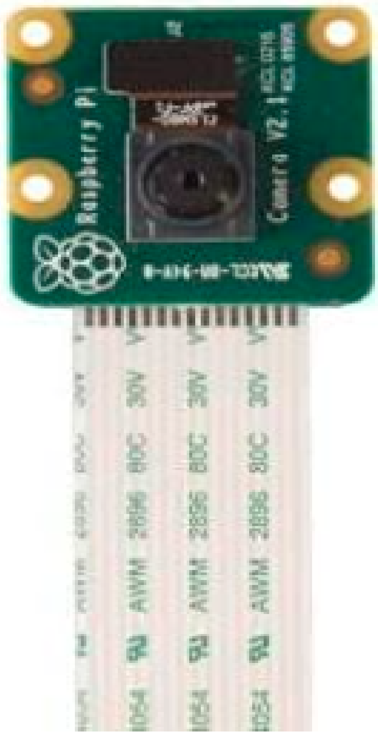


Figure 5.1: Raspberry pi camera [60]

To read images from the Raspberry Pi Camera v2, we use the OpenCV library, which provides a simple and efficient way to work with images and video streams. OpenCV is a popular open-source computer vision library that provides a wide range of functions for image and video processing, including image acquisition, filtering, feature detection, and object recognition. OpenCV is written in C++ and has bindings for Python, Java, and other programming languages, making it accessible to a wide range of developers. For our application, we are using OpenCV 4.7.0 version.

The Raspberry Pi Camera v2 communicates with the Raspberry Pi through serial communication, and OpenCV is used to read the image data from the camera and convert it into a format that can be processed by our computer vision algorithms. The OpenCV library provides a VideoCapture class that can be used to read frames from a video stream. We create an instance of the VideoCapture class and specify the camera index to initialize the camera. We then use the read() method of the VideoCapture class to read frames from the camera and store them in a variable for further processing.

Before sending the image to our deep learning model for object recognition and manipulation, we pre-process the input video stream by resizing it to 640 x 640pixels. This is done to reduce the computational complexity of the deep learning model and improve its performance. The resizing operation is performed using OpenCV, which provides a variety of image processing functions for manipulating images and video streams. We create a new image with the desired size using the cv2.resize() function and pass the original image and the new size as arguments. The cv2.resize() function uses interpolation to resize the image and maintain its aspect ratio.

In summary, in image acquisition and pre-processing steps, We acquire images from the Raspberry Pi Camera using OpenCV and pre-process the input video stream by resizing it to 640 x 640 pixels. These steps are essential for preparing the input data for our deep-learning model, which will be used for object recognition and manipulation.

5.1.2 Data Set

The need for a large volume of data is crucial for the effective training of deep neural networks. Existing datasets for robotic grasp detection, such as the RoboCup dataset [61], do not provide images that are suitable for our specific environment. Additionally, certain classes, such as "scrubby," contain an insufficient number of images for accurate detection results. Another dataset, COIL-100 [62], focuses on visual object recognition and includes object categories as labels. However, for our grasp classification problem, the labels should represent appropriate grasp types. Ghazaei et al. [63] utilized the COIL-100 dataset and manually defined a grip for each object category. This approach, though, is not suitable for real-world scenarios where a wide variety of objects are encountered. Consequently, we were motivated to develop our own dataset, GRASP, in which the label for each object corresponds to the appropriate grasp type. This new dataset aims to address the limitations of existing datasets and provide a more comprehensive solution for robotic grasp detection in everyday environments.

Data Collection

The acquisition of a comprehensive database is a crucial aspect of many research projects, and the experimental setup used to obtain this data is of utmost importance. In Figure 5.2, we present the experimental setup used to acquire our database, which involved the use of a Galaxy A21s camera to capture images. This camera is equipped with a 48MP Samsung ISOCELL GM2 (S5KGM2) 1/2.0" sensor, featuring 0.8µm pixels, an f/2.0 lens, and a Quad Bayer color filter. To ensure diversity in our dataset, images were taken in various environments and with variable backgrounds. Normal room lighting conditions were used to avoid any shadows that could affect the quality of the images.

To capture images of the objects, each was placed at a different distance from the camera, specifically at distances of 24cm, 16cm, and 30cm. The camera was then rotated around the object to capture images from at least 15 different views, providing a comprehensive dataset for our research. The images were saved in PNG format with a dimension of 4000*3000, which was necessary to ensure that the images were of high quality and could be used effectively in our research.
However, due to disk size limitations, we further processed each image and resized it to 1000*1000. This was done to ensure that the images could be stored efficiently without compromising their quality. The images were then numbered sequentially

Figure 5.2: Data acquisition setup

as 1.png, 2.png, and so on, to use in our research.

In our dataset, we have included a total of four grip patterns, which were selected based on their frequency of use in day-to-day activities [64]. These grip patterns are the power grip, pinch grip, tripod grip, and hool grip. To ensure that our dataset was comprehensive, we divided each object into these grip classes. For instance, a ball is typically gripped using a power grip, while a pen is gripped using a pinch grip. We have created 500 samples of different objects per class, and a selection of these samples can be seen in Figure 5.3.

The inclusion of these grip patterns in our dataset is a critical aspect of our research, as it allows us to accurately classify objects based on their grip type. By dividing each object into these grip classes, we can ensure that our dataset is representative

Figure 5.3: Object and their respective grip type

of real-world scenarios and can be used effectively in our research. The creation of 500 samples per class also ensures that our dataset is comprehensive and provides a sufficient number of samples for our research.

Data Annotation

Creating a dataset is a crucial step in machine learning, and annotating it is equally important. In our case, we created a dataset and annotated it using the bounding box technique. This technique involves drawing a rectangular box around the object of interest in an image. The bounding box technique is widely used in object detection tasks, where the goal is to identify and locate objects within an image.

To annotate our dataset, we used the Roboflow platform, which is a powerful tool for data annotation and management. The platform allowed us to upload our images and annotate them using the bounding box technique. We were able to draw bounding boxes around the objects of interest in our images, and the platform automatically saved the annotations as JSON files and txt files.

The JSON file format is a lightweight data-interchange format that is easy for humans to read and write, and easy for machines to parse and generate. It is a popular format for storing and exchanging data between web services and applications. The txt file format, on the other hand, is a simple text file format that is widely used

for storing and exchanging data. It is a versatile format that can be easily read and edited by humans and machines alike.
Having our dataset annotations available in both JSON and txt file formats was important for us, as it allowed us to easily integrate our dataset into different machine learning frameworks and tools. We were able to use the annotations to train and test our machine-learning models, and to evaluate their performance.

5.1.3 Object Recognition

A fundamental problem in computer vision called object detection involves finding and identifying things in digital photos or movies. It is essential for many applications, from self-driving automobiles to robotics and surveillance. Computer systems can better comprehend and interact with their environments by precisely recognizing and localizing objects [5]. Two essential questions are addressed by object detection: "What is the object?" and "Where is it?" While the second question seeks to pinpoint the exact placement of the object within the image, the first question concentrates on recognizing the particular object within the frame.
Over the years, object detection algorithms have evolved significantly, driven by advancements in deep learning and convolutional neural networks (CNNs). These algorithms can be broadly categorized into two main types: single-shot detectors and two-stage detectors [65]. Figure 5.4 shows the classification of these algorithms.

As the name implies, single-shot detectors seek to identify objects within a single pass of the input image. These methods are computationally efficient and suited for real-time applications because they process the full image at once. YOLO (You Only Look Once) is a single-shot detector that has gained a lot of popularity. By creating a fully convolutional neural network design that can process an entire image, YOLO revolutionized the area of object detection. It creates a grid out of the image and uses this grid to forecast bounding boxes and class probabilities. Because of its remarkable accuracy and speed, YOLO is the best option for activities requir-

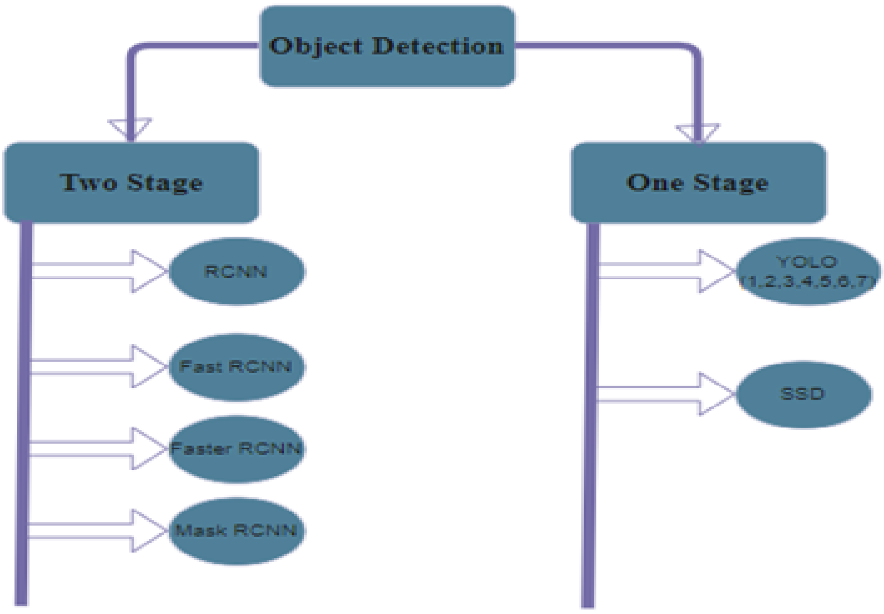


Figure 5.4: One and Two-stage detectors [66]

ing real-time object detection.

Two-stage detectors, on the other hand, adopt a different strategy. To find objects, they use a two-step method. These methods produce a set of candidate regions or possible object proposals in the initial step that is likely to contain objects. In the second stage, these suggestions are then improved and categorized to arrive at final projections. The Fast R-CNN, Faster R-CNN, and Mask R-CNN are three common two-stage detectors. The benefit of two-stage detectors is that they are more accurate. These algorithms can more accurately localize objects and obtain greater detection rates by using a proposal generation stage. But because of the higher computing complexity caused by this increased accuracy, they are less suited for real-time applications with constrained resources.

YOLOv7 Tiny for real-time grasp detection

We have chosen YOLOv7 tiny for our application. Our decision for this model is based on the following reasons:

First off, the YOLOv7 Tiny algorithm's outstanding performance was a significant reason for choosing it. In the context of prosthetic hand control, real-time performance is essential since it allows for rapid and smooth contact with the environment. YOLOv7 Tiny ensures that the grab detection system can keep up with real-time motions and changes in the surroundings by processing photos quickly and effectively. Because of its speed advantage, it is the best option for applications that need immediate responsiveness, improving the user experience as a whole.

Second, while accuracy and speed are equally critical in grasp detection for prosthetic hands, speed is still very significant. YOLOv7 Tiny achieves an acceptable balance between precision and quickness. It makes use of a simplified design that enables quick processing while keeping a respectable level of detection accuracy. This is especially helpful in situations when accurate object detection is necessary but real-time performance is also crucial. YOLOv7 Tiny is a good option for my research because it can produce reliable outcomes without sacrificing performance. The graph of YOLOv7, Figure 5.5, shows the trade-off between accuracy and speed.

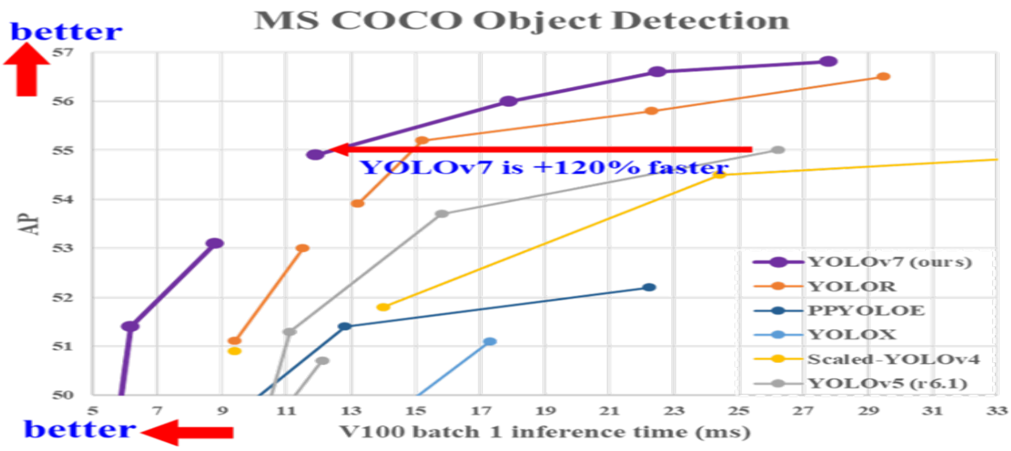


Figure 5.5: Comparison of YOLOv7 with other versions of YOLO

The x-axis represents the inference time taken by the model to process a batch of 100 images, while the y-axis represents the mean average precision (mAP) score. The mAP score is a measure of the accuracy of the algorithm, with higher scores indicating better accuracy. The graph shows that YOLOv7 achieves a high mAP score while maintaining a low inference time, indicating that it strikes a good balance between accuracy and speed.

Compared to previous versions of YOLO, YOLOv7 is better in terms of both accuracy and speed. YOLOv7 achieves a higher mAP score than YOLOv5 and YOLOv4 while maintaining a similar FPS. This means that YOLOv7 is more accurate than previous versions of YOLO, while still being fast enough for real-time applications. In addition, YOLOv7 also introduces several new features that improve its performance. For example, YOLOv7 uses a new backbone network called CSPDarknet53, which is more efficient than the backbone networks used in previous versions of YOLO. YOLOv7 also uses a new training strategy called self-adversarial training, which helps to improve the robustness of the algorithm.

Results for grasp detection

A significant part of our research involved training the YOLOv7 model for grip detection on our own dataset, GRASP. To speed up the training process, we choose to make use of Google Colab's capabilities, particularly their GPU service. Our training arrangement had a learning rate of 0.01 across 60 epochs. It took over 7 hours to complete the training, which was time-consuming. Due to the YOLOv7 model's complexity and the magnitude of our dataset, this duration can be explained. Deep learning model training frequently needs a lot of computer power and time, especially for huge datasets. But by utilizing the GPU offered by Google Colab, we were able to speed up the training procedure and gain from fewer training iterations. We evaluated the model's performance using the mean average precision (mAP) metric at a confidence level of 0.5 throughout the training procedure. The result of our trained model is shown in Figure 5.7. The obtained mAP of 0.79 illustrates how well our trained model performs in precisely identifying grasps in the provided dataset. A greater mAP shows that the model correctly locates grasps with a high degree of recall and precision. Additionally, we looked at the training loss to gauge the progress of the training. The difference between the expected and actual bounding

```
Running evaluation...
100% 78/78 [00:24<00:00, 3.14it/s]
Accumulating evaluation results...
DONE (t=0.17s).
IoU metric: bbox
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.620
 Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.795
 Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.720
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = -1.000
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.620
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.669
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.734
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.734
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = -1.000
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.734
```

Figure 5.6: Represents trained model average precision result

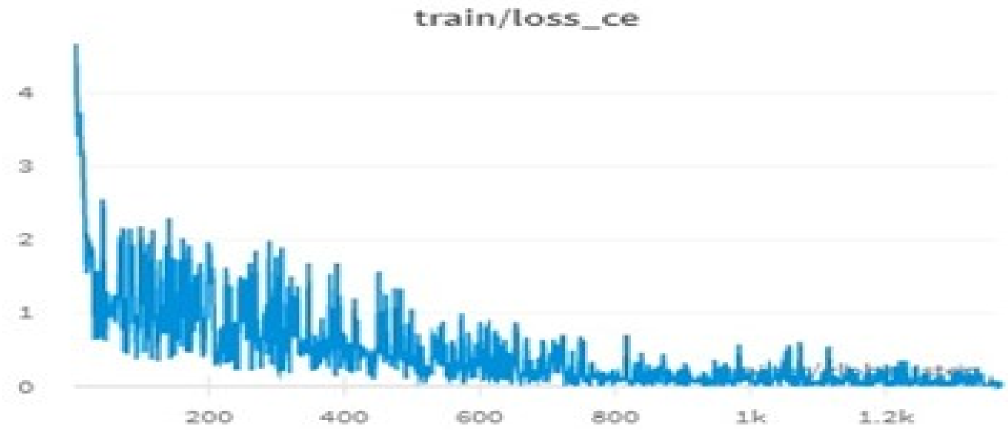


Figure 5.7: Represents the training loss curve

boxes during training is represented by the training loss. In our situation, the model was trained to minimize the error and precisely predict the grasp locations because the training loss was minimal, close to 0.01.

The dataset's quality and diversity, the complexity of the objects being detected, and the optimization of hyperparameters like the learning rate are a few of the elements that can affect whether a mAP and training loss is sufficient. As researchers, our goal is to optimize the performance of the model while balancing model complexity, computational resources, and training time. Our research on grip detection is well-founded thanks to the successful YOLOv7 model training on our GRASP dataset. The model has mastered the ability to precisely recognize and localize grasps in our particular domain, as evidenced by the acquired mAP of 0.79 and negligible training loss. These outcomes support the YOLOv7 model's efficacy and boost confidence in its prospective use in real-world scenarios.

Depth Estimation

Depth estimation is an important task in machine vision that involves figuring out the distance of objects from the camera or sensors. It has several uses in many different industries, including robotics, driverless cars, and augmented reality [67]. Accurate depth estimation can help us in object recognition, navigation, and scene comprehension. Stereo vision, structured light, and time-of-flight are a few methods for estimating depth [68]. Utilizing two cameras to take pictures of the same thing from slightly different angles is known as stereo vision. To determine the depth of items in the scene, one can estimate the difference between the two photographs. In structured lighting, a pattern of light is projected onto the scene, and the pattern's distortion is used to determine depth. To determine the distance, time-of-flight entails timing how long it takes a light pulse to reach an object and return to the sensor.

Monocular depth estimation, which includes determining depth from a single camera, has gained popularity in recent years. Due to the inherent ambiguity of converting an intensity or color measurement into a depth value, monocular depth assessment is a difficult operation. In situations where direct depth sensing is unavailable or impractical, it is an essential task. Using a convolutional neural network (CNN) trained on a sizable dataset of pictures with accompanying depth maps is a typical method for monocular depth estimation [69]. A depth map created by CNN from an input image can be used to estimate the distances between various objects in a scene. Large datasets, like the NYU Depth dataset and the KITTI dataset, which give a substantial number of images with matching depth maps, are major factors in the effectiveness of this method. Structure from motion (SfM), which includes estimating the 3D structure of a scene from numerous 2D images obtained from various angles, is another method for monocular depth assessment [70]. By assuming a given camera orientation and using the estimated 3D structure to infer depth, SfM can be used to estimate depth from a single image. In situations when the camera is moving, like in robots or autonomous vehicles, this method is especially helpful. Other methods for estimating monocular depth exists in addition to CNNs and SfM, such as depth from defocus, depth from motion, and depth from shading. Depth from defocus involves calculating depth using the blur that the camera's aperture produces. Depth estimation by object motion is a technique known as "depth from motion." Using the difference in brightness across an object's surface to estimate depth is known as "depth from shading." Although monocular depth estimation has advanced significantly in recent years, there are still a number of issues that must be resolved. The absence of ground truth depth maps for CNN training is one of the main obstacles. Although there exist big datasets and related depth maps, their size is frequently constrained, and they might not be an accurate representation of all cases. The trade-off between precision and speed is another difficulty. Despite its ability to produce precise depth maps, CNNs can be computationally expensive and aren't always appropriate for real-time applications.

Monocular Depth Estimation

Monocular depth estimate is the method of determining an object's depth or distance in a scene from one camera only. Given that 2D images produced by monocular cameras are fundamentally devoid of depth information, it is a difficult task. However, it is possible to infer depth information and rebuild a depth map from a single image or a series of photos by using a variety of approaches and algorithms. Figure [?] shows the monocular depth estimation pipeline where a single image is fed to the deep neural network, and it outputs a depth map that corresponds to the given input image.

For our project, we experiment with an open-source monocular depth estimation

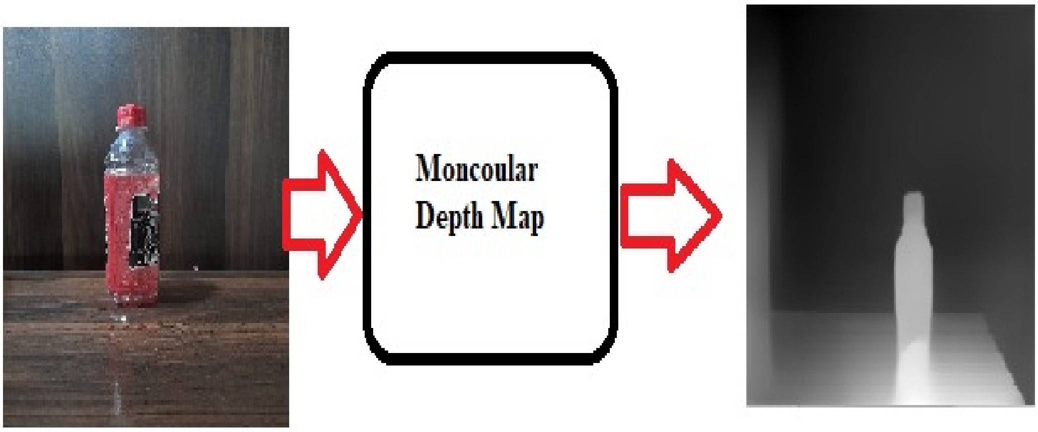


Figure 5.8: Monocular Depth estimation pipeline. Single image input and depth map output

model, Midas. Midas (Multi-Dataset depth estimation), a cutting-edge monocular depth estimation approach, solves the difficulties in obtaining dense ground-truth depth at scale across various contexts. Large and varied training sets are necessary for monocular depth estimation to be successful, and Midas is built to accommodate numerous datasets with different traits and biases, even if their annotations are incompatible [71]. To accomplish reliable depth estimation, the Midas model contains a number of significant improvements. It first suggests a reliable training target that is unaffected by variations in depth range and scale. As a result, the model can handle datasets with different depth properties. Second, it combines data from several sources using principled multi-objective learning, allowing the model to benefit from the advantages of each dataset. Finally, it emphasizes the significance of pretraining encoders on support tasks, which can enhance the generalization abilities of the model. The tests reported in the Midas paper demonstrate that monocular depth estimation is significantly enhanced by combining data from complementary sources. The model sets a new standard for monocular depth estimation by outperforming rival methods on a variety of datasets. Users can select the most appropriate Midas model for their unique application and hardware limitations thanks to the model's availability in a variety of configurations, with variable numbers of parameters and inference resolutions.

The Midas small model runs on a Raspberry Pi 4B at about 2 FPS (frames per second) after being translated to ONNX format. Although this performance might be enough for some applications, it is regarded as being extremely poor for real-time depth estimate tasks, particularly in situations where quick scene comprehension and object recognition are essential. Since the Raspberry Pi 4B is made for low-power and economical applications, its restricted performance can be attributed to its limited computing capabilities. Even though it is a more compact version of the original Midas model, the Midas small model still needs a lot of computing resources to complete the depth estimate. Figure 5.9 shows the result of the Midas model on raspberry pi in real-time. One of the drawbacks of the Midas small model is that

Figure 5.9: Midas model results on raspberry pi 4

it is less sensitive to subtle depth fluctuations, especially when objects are closer together than 18 centimeters. This flaw can be traced to the compact architecture of the model, which favors computing effectiveness above the capacity to record fine-grained depth information. The Midas small model might not be appropriate in situations where fine depth differences are essential for proper picture comprehension and object recognition, such as those that call for exact depth estimation for closely spaced objects. In many applications, like robotic manipulation, where precise depth information is necessary for actions like grasping and item placement, the diminished sensitivity to small depth changes might be troublesome. Similar to virtual reality, precise depth estimation in augmented reality is essential for integrating virtual things into the real world seamlessly, especially when such virtual objects are close to actual objects. In these situations, the application's effectiveness may be constrained by the Midas small model's inability to distinguish between minute depth changes.

Depth estimation using the triangulation principle

Depth estimation using the simple triangle rule is a straightforward method that relies on the geometric properties of similar triangles to determine the distance between an object and the camera. This approach can be particularly useful when the height of the object is known beforehand, and the camera's focal length has been pre-calculated. In this explanation, we will discuss the process of calculating the focal length of a Raspberry Pi camera and then using it to estimate the distance between the object and the camera.

The first step in this method is to calculate the camera's focal length. To do this, we placed an object of known height at a known distance from the camera. By capturing an image of the object and measuring its height in pixels, we used the following formula to calculate the focal length:

$$F = \frac{dx}{dZ} \times dZ \tag{5.1}$$

Where F is the focal length of the camera in pixels, dx is the height of the object in the image, dX is the real height of the object and dZ is the distance between the camera and object. In this case, the Raspberry Pi camera's focal length was determined to be 700 pixels. With the focal length known, we now used it to estimate the distance between the camera and any object with a known height.

To estimate the distance between the object and the camera, you can use the same formula as before, but this time, rearrange it to solve for the distance. By capturing an image of the object with a known height and measuring its height in pixels, you can plug the values into the formula to calculate the distance between the object and the camera.

The simple triangle rule for depth estimation relies on the principle of similar triangles, which states that if two triangles have the same angles, their corresponding sides are proportional. In this case, the triangles are formed by the camera's optical center, the top and bottom of the object, and the top and bottom of the object's image on the camera sensor. Since the angles between the sides of these triangles are the same, the ratio of their corresponding sides is constant. This constant ratio is the focal length of the camera, which allows you to estimate the distance between the object and the camera using the known height of the object and its height in pixels. It is important to note that this method assumes that the camera is level with the object and that the object is perpendicular to the camera's line of sight. If these conditions are not met, the depth estimation may be less accurate. Additionally, this method may not be suitable for objects with varying heights or complex shapes, as the height measurement in pixels may not accurately represent the object's true height.

5.1.4 Deployment on Raspberry Pi

Due to its potential for real-time, low-latency inference and improved privacy, the use of deep learning models on edge devices has drawn a lot of attention recently. The Raspberry Pi, a little and reasonably priced single-board computer with many features, is one such well-liked edge gadget. In this section, we look at how to use a Raspberry Pi to deploy our YOLOv7 deep learning model for our smart prosthetic hand project. The benefits it provides led us to decide to run our model on a Raspberry Pi. We can achieve real-time inference without relying on cloud connectivity by bringing the power of artificial intelligence right to the edge. This increases the autonomy and reactivity of our smart prosthetic hand by enabling it to make decisions on the spot. Deploying deep learning models on limited-resource devices like the Raspberry Pi, however, presents particular difficulties. Due to Raspberry Pi's restricted computing capabilities, such as its CPU speed, memory capacity, and energy economy, model deployment must be carefully optimized. Successful integration of our model with the Raspberry Pi environment also depends on variables like compatibility, model conversion, and inference speed. We'll go into detail about how to deploy our YOLOv7 model on the Raspberry Pi in this section, covering crucial topics like hardware and software setup, model optimization techniques, installing and configuring dependencies, model conversion and compatibility, assessing inference performance, designing the user interface, and deployment considerations. We want to enable our smart prosthetic hand to perform real-time object detection and grip detection directly on the edge device, lowering latency, boosting privacy, and enhancing its overall usefulness by successfully implementing our deep learning model on the Raspberry Pi. The development of a self-contained, intelligent prosthetic hand that can adjust and react to its environment in real-time, ultimately enhancing the user's quality of life, has advanced significantly with this deployment.

Choosing the Right Operating System

For the Raspberry Pi model to be deployed, selecting the appropriate operating system is essential for assuring optimal performance and compatibility with the intended applications. With the introduction of the Raspberry Pi 4, the computer is now fully capable of running 64-bit operating systems, which presents both consumers and developers with a wide range of new opportunities.

Running 64-bit programs like Pytorch, which were previously unreachable on the Raspberry Pi 3 due to its constrained RAM capacity, is one of the most important benefits of adopting a 64-bit operating system on the Raspberry Pi 4. Users can now fully benefit from the performance advantages that 64-bit applications have to offer thanks to the Raspberry Pi 4's enhanced RAM capacity. It is important to keep in mind that using a 32-bit operating system like Raspbian can cause problems with image files and updates, which can affect the Raspberry Pi's overall performance and stability. As a result, using a 64-bit operating system is strongly advised for the best performance and compatibility with the most recent software and upgrades.

Enabling Camera

Enabling the camera on a Raspberry Pi can be a challenging task, but by following the steps below, we successfully enabled the camera and start capturing images and videos.

1. Install rpi-update using the command: sudo apt-get install rpi-update
2. Update the firmware using the command: sudo rpi-update
3. Install cmake and build-essential using the command: sudo apt-get install cmake build-essential
4. Install userland code and reboot using the command: sudo apt-get install libraspberrypi-dev sudo reboot
5. Run the command: raspistill -o test.jpg. If you encounter an issue with "failed to open vchiq instance," proceed to the next step.
6. Run the command: sudo chmod 777 /dev/vchiq
7. Test the camera using the command: raspivid -p 0,0,640,480 -t 0. If you still encounter an error, proceed to the next step.
8. Run the following commands:
   - vcgencmd get_camera
   - sudo ls -l /dev/video
   - sudo vcgencmd get_camera

- sudo lsmod — grep bcm2835
- sudo modprobe bcm2835-v4l2
- sudo rpi-update
- sudo modprobe bcm2835-v4l2
- v4l2-ctl –stream-mmap=3 –stream-count=1 –stream-to=somefile.jpg

These steps install the necessary drivers and dependencies needed to run the Raspberry Pi camera successfully. By following these steps, we enabled the camera and start capturing images and videos on your Raspberry Pi.

Pytorch and TorchVision Installation

It's crucial to keep in mind how embedded systems and desktop computers differ in terms of CPU architecture when working with devices like the Raspberry Pi. While most embedded systems employ the Reduced Instruction Set Computing (RISC)-based Arm CPU architecture, most desktop computers use Intel's x86 CPU architecture, which uses Complex Instruction Set Computing (CISC). Because of this fundamental distinction, x86-compatible software requires significant modifications to operate on Arm, and vice versa. It's crucial to check that machine learning libraries, like PyTorch and TorchVision, are compatible with the embedded system's CPU architecture before using them. In this instance, the stable distribution of PyTorch and TorchVision only contains installations for x86 computers. However, the developers also provide nightly builds that operate on aarch64 (ARM) devices, however these may be incomplete due to ongoing development.
To install PyTorch and TorchVision on a Raspberry Pi, we followed the following steps:

- Update your pip3 file installer using the command: sudo apt-get install python3-

pip

- Install NumPy, a large mathematical library for Python, using the command: sudo pip3 install numpy
- Run sudo pip3 install –pre torch torchvision

Move to the "yolov7-on-rpi4-2020" directory in the terminal. Copy the Python libraries in the "dist-packages" folder into your Python 3.8 library directory using the command: sudo cp -r dist-packages/* /usr/local/lib/python3.8/dist-packages/. Copy the files in the "local-bin" folder into your Raspberry Pi's local bin using the command: sudo cp -r local-bin/* /usr/local/bin/. Ensure that the Python programs "convert-caffe2-to-onnx" and "convert-onnx-to-caffe2" are included. Download other necessary Python libraries using the command: sudo pip3 install typing-extensions. By following these steps, you can install PyTorch and TorchVision on your Raspberry Pi and ensure that they are compatible with the Arm CPU architecture.

5.2 Mobile Application

Our project's Flutter-based mobile app seeks to provide smooth communication between a smartphone and a Raspberry Pi utilizing a Bluetooth module. Users may easily communicate with the GPIO pins on the Raspberry Pi with this program, which gives them the ability to transmit signals and instructions to the attached hardware. Users of the mobile application can manage several features of their Raspberry Pi-based system through its user-friendly and straightforward interface. Users can connect to the Raspberry Pi through Bluetooth after running the software, ensuring a steady and dependable communication channel. The application offers a variety of functions and functionalities once connected. The main advantages of developing a flutter-based mobile application are:

- Cross-Platform Compatibility: Flutter enables the development of mobile applications that work flawlessly on both Android and iOS devices. Regardless of the user's preferred mobile platform, the app can reach a larger user base thanks to its cross-platform compatibility.
- Intuitive User Interface: The user experience is improved by Flutter's ability to create aesthetically appealing and understandable user interfaces. Because of the application's user-friendly design, even those with little technical expertise can easily communicate with the Raspberry Pi.
- Bluetooth Connectivity: The app does not require an internet connection because Bluetooth is used for communication. This makes it especially helpful in circumstances where internet connectivity may be constrained or nonexistent. A direct and secure connection between the mobile device and the Raspberry Pi is also made possible by Bluetooth communication.
- Portability: Users can remotely manage their Raspberry Pi-based system using the mobile application. Users can monitor and manage their projects from any location that is within Bluetooth connection's range thanks to portability and simplicity.

Our project's Flutter-based mobile application offers a simple and effective way to communicate with a Raspberry Pi through Bluetooth. The application enables users to easily operate and interact with their Raspberry Pi-based devices by providing GPIO control, signal transmission, and data monitoring. The application is a useful tool for remote control and monitoring in many projects and applications because of its benefits of cross-platform compatibility, intuitive user interface, Bluetooth connectivity, and portability.

5.2.1 UI Design and Functionality

With its well-designed user interface (UI) and simple functionality, the Flutter application we created for our project enables seamless communication between the Raspberry Pi-based system and the prosthetic hand. Figure 5.10 shows the UI design of the mobile application we created for our project. The application has two

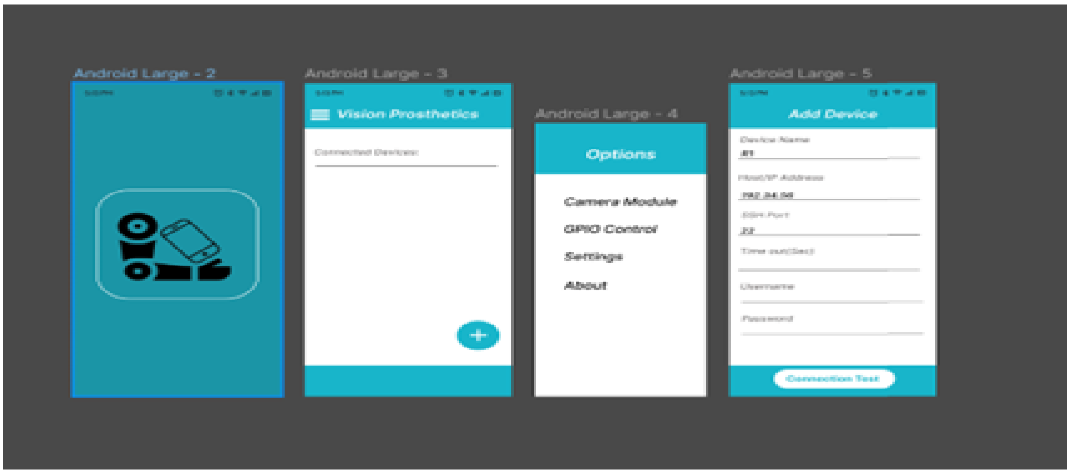


Figure 5.10: UI design for our mobile application (initial)

major menus: Battery Camera Module and GPIO Control. Here is a detailed examination of each of these modules:

- GPIO Control: The main interface for managing the grip patterns of the prosthetic hand is the GPIO Control module. This module's user interface was designed with simplicity and usability in mind. A variety of preconfigured grip patterns are displayed on the main screen and are represented by visually unique buttons or icons. The software instructs the prosthetic hand to perform a specific grip when the user taps on a particular grip pattern, sending a corresponding signal to the Raspberry Pi in the process. The user interface (UI) gives them immediate feedback, verifying their choice of grip pattern or updating them on the grip's progress. The GPIO Control module may also include customization choices that let users create and preserve their own grip patterns. Figure 11 shows this module.

- Battery Camera module: The focus of the Battery Camera module is on giving consumers access to live camera feed from the Raspberry Pi while also providing crucial information about the prosthetic hand, such as the percentage of battery life left. Users of this module may easily keep track of the prosthetic hand's power level thanks to the UI's prominent display of the current battery state. To accurately reflect the percentage of battery life left, it could incorporate a visual representation like a battery icon with a dynamic fill or a numerical display. The module incorporates a live camera feed from the Raspberry Pi in addition to battery data. Users can directly access the application's real-time video stream from the camera attached to the Raspberry Pi. Users can conveniently watch their surroundings thanks to the responsive design and seamless viewing optimization of the camera feed display. The Battery Camera module might offer choices for changing camera settings, such as brightness, contrast, or resolution, to improve the user experience. With the use of these options, users can tailor the camera feed to their unique needs or the surrounding circumstances.

5.2.2 Bluetooth Communication

The RFCOMM (Radio Frequency contact) protocol and the Bluetooth API are used to establish contact between the Raspberry Pi and our Flutter application. With this setup, wireless communication between the two devices is dependable and easy. A virtual serial port emulator is offered via the Bluetooth connection using the straightforward and dependable RFCOMM Bluetooth protocol. Similar to an established wired serial communication link, it enables the creation of a dependable data channel between devices. Due to RFCOMM's ability to transmit data both ways, the Raspberry Pi and the Flutter application can effortlessly share data. A

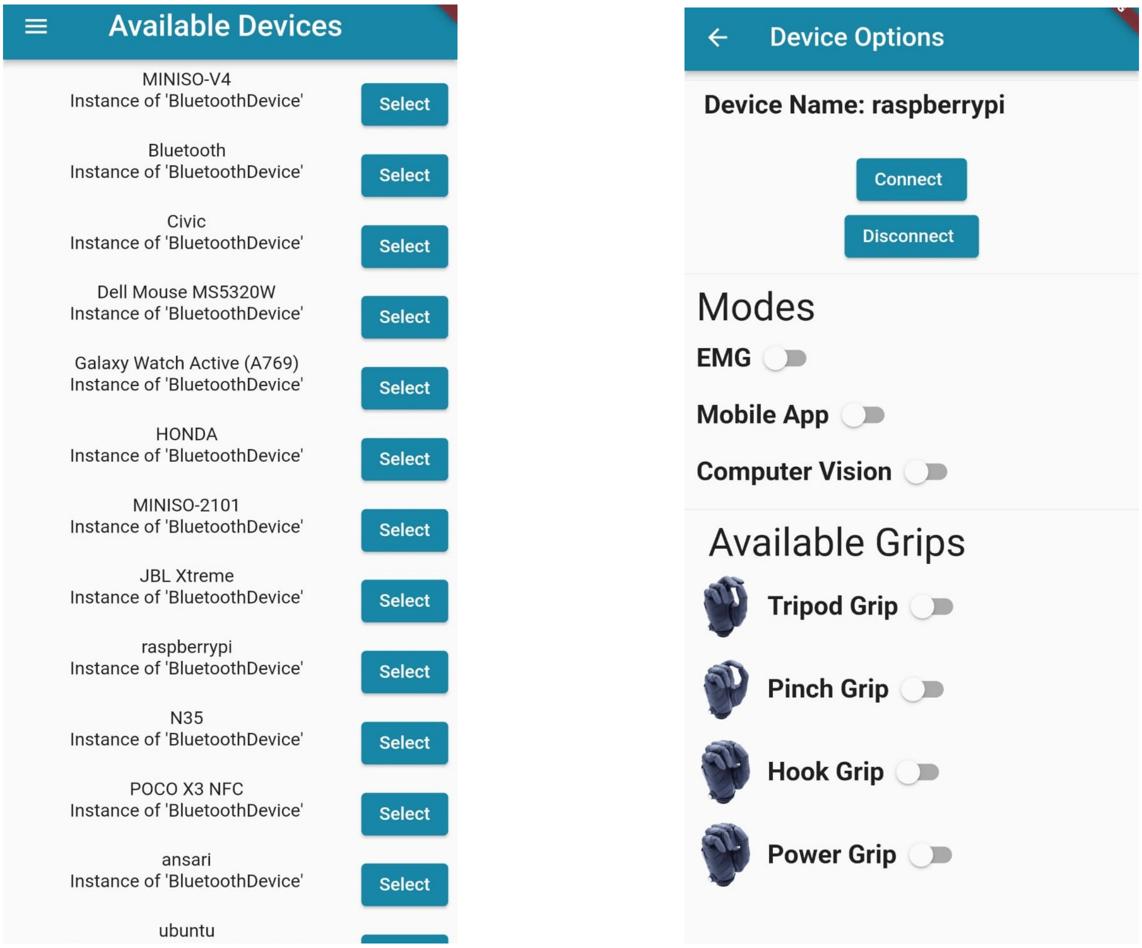


Figure 5.11: Initial working of the mobile application. Left: Available and paired devices, Right: Device options

collection of programming interfaces and protocols called the Bluetooth API makes it easier to integrate Bluetooth functionality into software programs. The Bluetooth API in the context of our project makes it possible for the Raspberry Pi and the Flutter application to find and connect to one another. The Bluetooth API on the Raspberry Pi supports setting up a server socket and configuring the Bluetooth module. The Flutter application's server socket keeps an eye out for incoming Bluetooth connections. The Bluetooth API facilitates the construction of a client socket connection to the Raspberry Pi and the discovery of available Bluetooth devices on the Flutter application side. The client socket enables two-way communication between the Raspberry Pi and the Flutter application.

5.2.3 Conclusion

As a result, the application of computer vision and depth estimation methods on a Raspberry Pi for prosthetic hand grasp categorization marks a significant development in the field of assistive technology. We have successfully created a smart prosthetic hand that can perceive its surroundings and makes wise grip judgments in real time by integrating these technologies. We have achieved efficient and precise object identification using deep learning models, in particular the YOLOv7 model, deployed on the Raspberry Pi, allowing the prosthetic hand to identify and localize items in its environment. We have given the prosthetic hand the ability to carry out these functions locally without relying on cloud connectivity by utilizing the computational capacity of the Raspberry Pi, protecting privacy and lowering latency. Furthermore, the prosthetic hand's grasp classification abilities have been improved by the incorporation of depth estimation methods employing a monocular camera. We have been able to analyze the spatial relationships and geometry of things and determine the best grab technique for various objects by inferring depth information from the camera's visual input. The prosthetic hand now has a more thorough understanding of the environment thanks to depth estimation algorithms, which have enhanced grasp precision and effectiveness. There have been difficulties in successfully implementing this depth estimation and computer vision methods on the Raspberry Pi. It took significant thought and optimization work to adjust the deep learning model to the Raspberry Pi's constrained resources, speed up inference, and ensure compatibility. To achieve precise depth estimates, the monocular camera needed to be calibrated, and issues like occlusion and changing lighting had to be addressed. In conclusion, the effective use of computer vision and depth estimation methods on the Raspberry Pi for prosthetic hand grasp classification shows the promise for intelligent and context-aware assistive technology. We have made tremendous progress towards developing more effective and natural prosthetic solutions by utilizing edge computing and incorporating depth awareness, which brings us closer to the objective of improving the lives of people with limb differences.

As a result of the creation of a Flutter-based mobile application for our project, an effective and simple tool for interfacing with the Raspberry Pi-based prosthetic hand system has been made available. By acting as a link between the user and the hardware, the mobile application enables seamless control, monitoring, and customization of numerous system components. The mobile application's user interface puts simplicity, usability, and accessibility first. Users may easily regulate the grip patterns of the prosthetic hand thanks to the GPIO regulation module. The program adapts to each user's preferences and needs by providing predefined grip patterns as well as the option to customize and save new patterns. The application's real-time input improves the user experience and makes it easier to have exact control over the movements of the prosthetic hand. The mobile application's Battery Camera module is critical for giving pertinent information and improving situational awareness. Users may simply check the prosthetic hand's battery life to ensure ongoing operation and prevent unplanned power interruptions. The live camera stream feature adds another level of convenience and safety by enabling customers to observe their surroundings in real-time. This mobile application serves as a foundation for future improvements in the creation of prosthetic hand systems and is an important tool in the field of assistive technology.

Chapter 6

HAPTIC FEEDBACK

For those who are missing an upper limb, the lack of tactile sensations is a significant barrier that prevents them from multitasking efficiently and from making full use of the dexterity of their prosthetic hands. Currently, despite the mechanical ability of contemporary artificial hands to separately operate all five digits, myoelectric prosthetic hands can only regulate one grip function at a time. We undertook an inquiry to see if people might gain fine control over the grip pressures delivered to two separate items simultaneously held by a dexterous robotic hand in order to overcome this constraint. To help with this, we created a wearable soft robotic wristband that simulates human touch for those who use robotic hands. We wanted to enable people to effectively grab and transfer things using the dexterous artificial hand, without running the danger of breaking or dropping them, even while their visual perception of both objects was obscured, by adding numerous channels of haptic input

The usage of the wearable soft robotic armband proven to be helpful in providing consumers with fake touch sensations. The haptic feedback system used a variety of channels to improve the individuals' use of the dexterous robotic hand to grab and operate items. It's noteworthy that this was done even when the subjects' eyesight of the manipulative objects was compromised. Individuals were able to move many things at once as a consequence, making deliveries successfully without the requirement for a one-at-a-time strategy.

Comparing the simultaneous movement of goods to the traditional one-at-a-time technique resulted in substantial time savings. Individuals with upper limb absence displayed increased efficiency in their delivery activities by taking use of the additional control and sensory feedback offered by the wearable soft robotic armband. In addition to increasing their productivity, being able to handle and move many things at once also lessened the restrictions put forward by the sequential technique. This study overcomes a significant barrier that prevents people without an upper limb from multitasking and using the full dexterity of their prosthetic hands. We were able to successfully enable users to control the grip pressures produced by the dexterous artificial hand on two separate items at the same time by creating and utilizing a wearable soft robotic armband capable of communicating synthetic touch sensations. Individuals were able to precisely grab and move things even when their visual perception was hindered because to the integration of several haptic feedback channels. Additionally, the contemporaneous execution of deliveries resulted in significant time savings, demonstrating the potential influence of this invention on the effectiveness and productivity of people utilizing prosthetic hands.

These findings hold promising implications for the field of prosthetics, as they offer a path towards overcoming the limitations of current myoelectric prosthetic hands and unlocking the full dexterity and multitasking capabilities for individuals with upper limb absence. By incorporating wearable soft robotic technology and harnessing the power of haptic feedback, we have taken a significant step forward in enhancing the functionality and usability of artificial limbs. Future advancements building upon these findings could contribute to a transformative shift in the quality of life for individuals with limb absence, enabling them to engage in a wider range of activities and tasks, ultimately promoting their independence and inclusion within society.

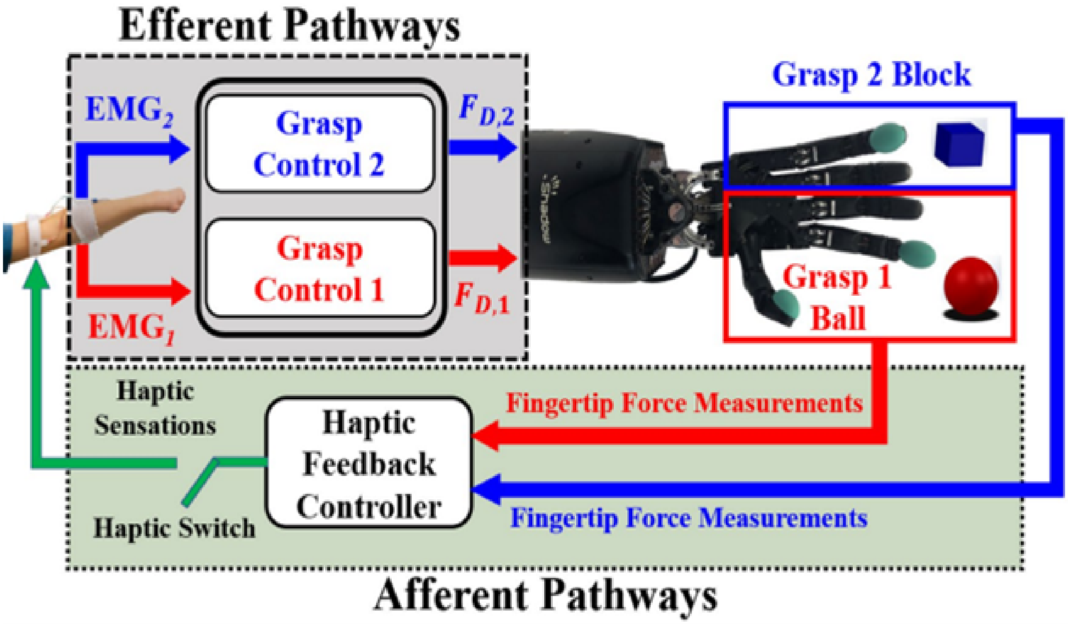


Figure 6.1: Flowchart of the haptic feedback system

6.0.1 FSR Feedback

Force-sensitive resistive (FSR) sensors were used in the early design to give feedback on the force applied by the prosthetic hand's fingers. The prosthetic hand's fingers were 3D printed using PLA (Polylactic Acid) material, and the FSR sensors were embedded into the fingers themselves. This method made it possible to create finger structures that were specifically suited to the consumers' needs.

At the distal phalanx of the thumb and index finger, a unique mount was designed in order to integrate the FSR sensors into the finger design. For the FSR sensors, these mounts acted as housing modules. The silicon substance was used to cover the FSR sensors, assuring their stability and protection, and they were then firmly installed inside the mounts.

This method of using FSR sensors made it possible to quantify the force used by the prosthetic fingers when gripping and manipulating objects. The FSR sensors' resistance characteristics altered in response to pressure, enabling the detection and measurement of the applied force. The user was then given force feedback and a feeling of touch, simulating the sensations felt with natural hands, using the information as feedback.

In order to enable precise and accurate mounting of the FSR sensors and the subsequent integration of the mounts, the CAD (Computer-Aided Design) models of the fingers were developed. The positioning of the FSR sensors and optimisation of the finger design were both made easier by the CAD models' virtual portrayal of the finger anatomy.

The real finger constructions based on the CAD models might be created using the 3D printing technique with PLA material. The 3D-printed prosthetic fingers were strong and long-lasting thanks to PLA, a popular thermoplastic polymer. The fingers' low weight, adaptability, and ability to successfully house the FSR sensors and their mounts are all results of the 3D printing process.

The development of the unique mount at the distal phalanx of the thumb and index finger was essential for housing and stabilising the FSR sensors. The finger design was made easier to include the FSR sensors onto because of the mount's role as a structural element. The silicon material covering was used to encapsulate and preserve the FSR sensors once they had been put inside the specifically built mount.

There were various advantages to using silicon as the FSR sensors' covering material. Due to silicon's exceptional flexibility, the FSR sensors may be seamlessly integrated with the finger surfaces while still following the curves of the finger structures. Additionally, silicon had strong shock-absorbing qualities that protected the FSR sensors from outside shocks and vibrations that may have harmed their accuracy or operation.

Overall, using 3D printing technology, a specifically created mount, and silicon coating, this initial design approach was effective in integrating FSR sensors into the fingers of the prosthetic hand. The user's capacity to sense and regulate the gripping force while doing various activities was improved by the incorporation of FSR sensors, which allowed for the measurement and feedback of the forces applied by the prosthetic fingers. This design marks a significant development in prosthetic technology, enhancing the usefulness and usability of prosthetic hands for people who are missing their upper limbs and bringing tactile input closer to natural feelings.

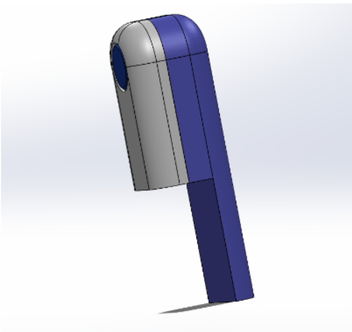


Figure 6.2: 3D modelled FSR incorporated

Figure 6.3: 3D printed FSR incorporated distant phalanx

In the first design of the prosthetic hand, force-sensitive resistors (FSR) had a number of drawbacks. To begin with, FSRs did not offer a linear fluctuation in resistance in accordance with the applied force. It was difficult to properly measure the force generated by the prosthetic hand's fingers due to their non-linear behavior. The lack of linearity made it difficult to gauge forces accurately, which is essential for people using prosthetic hands to move items successfully. In the first design of the prosthetic hand, force-sensitive resistors (FSR) had a number of drawbacks. To begin with, FSRs did not offer a linear fluctuation in resistance in accordance with the applied force. It was difficult to properly measure the force generated by the prosthetic hand's fingers due to their non-linear behavior. Lack of linearity made it difficult to gauge forces accurately, which is essential for people using prosthetic hands to move items successfully.

Additionally, even before the fingers of the prosthetic hand were engaged, the pressure was applied to the FSRs since they were housed inside the proximal phalanx of the finger and covered by the glove. The value of the FSRs was impacted by the pre-existing pressure brought on by the glove, which resulted in incorrect force measurements. The pressure applied by the glove changed when the prosthetic hand's fingers opened and closed, causing further changes in the FSR readings. Due to these discrepancies in FSR measurements, the user's force feedback was no longer reliable or stable.

To get beyond these constraints—the non-linear response of FSRs and the interference brought on by the pressure the glove applied during finger movements—alternative sensing technologies have to be investigated. The non-linearity problem may be resolved, and applied force measurements might be more accurate, with the development of more sophisticated and accurate force sensors. It would be feasible to create a precise connection between the force applied by the fingers and the sensor data by using sensors that have linear behavior.

To lessen the interference brought on by the pressure applied by the glove, the installation and integration of the force sensors should also be given further thought. The mounting method might be redesigned to reduce the effect of outside forces on the force sensors, perhaps by adding cushioning or isolating materials. More precise and dependable force feedback might be produced by shielding the sensors from outside effects. This would provide users with trustworthy information about the strength of their grasp and improve their control over the prosthetic hand.

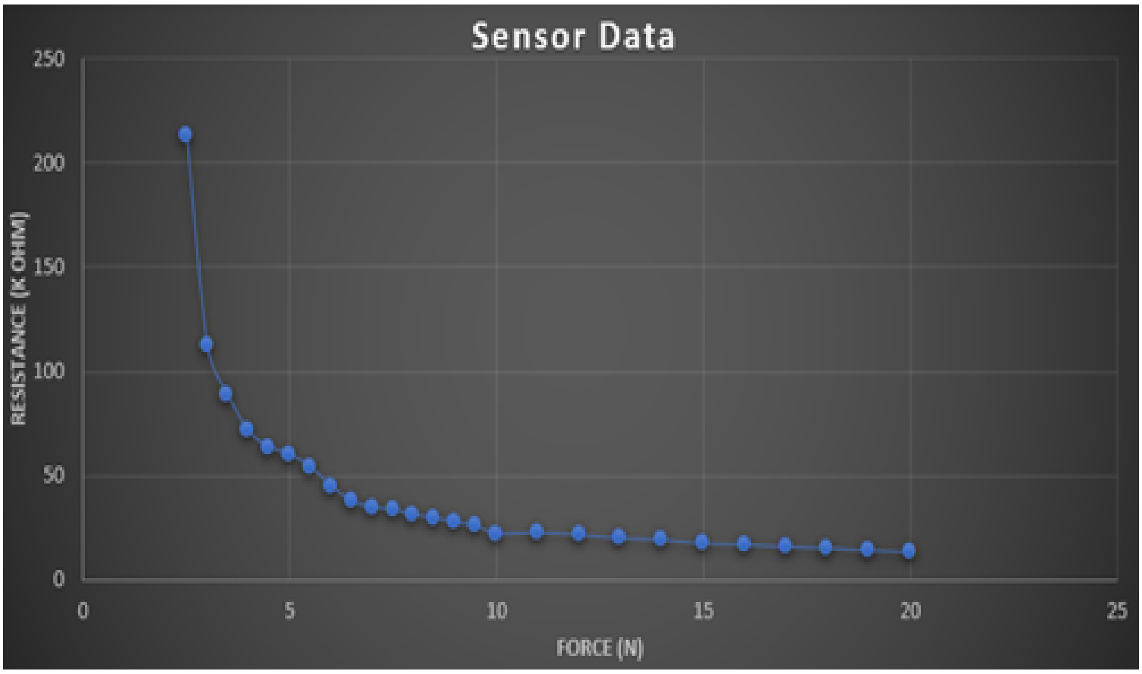


Figure 6.4: Force(N) vs Resistance (kOhm) curve of FSR

6.1 ACS712 current sensor

We switched to a current sensor as an alternate sensing technique to address the issues with force-sensitive resistors (FSR) that we had. In this method, we measured the current flowing through the prosthetic hand's thumb. We were able to determine the current's maximum, which corresponded to the motor's stall current and the point of greatest force application. The motor was then stopped at the proper value using the current sensor data, allowing us to regulate the prosthetic hand's grip strength.

This switch to a current sensor resolved the issues with FSRs and offered a more accurate and linear way to measure force. We were able to get a more precise picture of the force applied by directly seeing the current flowing through the thumb. The maximal force applied by the hand was accurately determined using the stall current as a threshold.

By leveraging the current sensor and its correlation to motor performance, we could effectively control the grip strength of the prosthetic hand. This improved control allowed users to achieve a more natural and precise manipulation of objects, enhancing their overall dexterity and functionality.

This shift to a current sensor represents a significant advancement in the development of prosthetic hands, as it overcame the limitations associated with FSRs. By accurately measuring current and utilizing it as an indicator of force exertion, we achieved a more reliable and linear feedback mechanism, ultimately improving the user's ability to interact with the environment and perform tasks with their prosthetic hand. The calculations carried out were as follows:

$$\text{Amax} = 5\text{A}$$
$$10 \text{ bits} = 1023$$
$$\text{At current} = 0, \text{ ADC value} = 1023/2 = 512$$
$$\text{For current} \geq 0:$$

$$\frac{5}{512} = 9.76mA/bit \tag{6.1}$$

This shows that a minimum 9.76mA of current can be detected using Arduino ADC and ACR 712 for 200 mA of current:

$$\frac{200}{9.76} + 512 = 532.5 \tag{6.2}$$

This shows the value of the ADC value of the Arduino required for the stall position.

6.2 Haptic Band

Furthermore, the vibration persisted for around 1 second to provide excellent timing and usage. Users might get enough input about their grip within this time frame before the vibration stopped. The haptic band offers a dependable and natural way to communicate the grip state to the user by delivering a brief and regulated vibration.

An inventive method of improving the tactile feedback of prosthetic hands is the incorporation of vibration motors inside the haptic band. Users who used this technology during item handling had a better sensation of touch and awareness. The timed vibration was an effective tool for determining if a grasp was successful and for boosting user confidence in their ability to hold items firmly in their prosthetic hand.

This haptic band design not only improved the prosthetic hand's functionality, but it also vastly improved user experience. Vibration motors that incorporate tactile feedback might help users feel more connected to their prosthetic hands and enable more intuitive and organic interactions with their surroundings.

Figure 6.5: Left: Haptic Band, Right: Vibration motor

Chapter 7

Conclusion and Future Work

7.1 Conclusion

This research study emphasizes the prevalence of upper limb amputations and how they are increasing day by day. Moreover, the causes and the types of upper limb amputations are discussed, and how they affect the patients. The amputees are stripped away of their independence and are unable to perform simple daily tasks. One of the treatments for amputees is the use of a prosthetic hand which allows them to return back to their daily routine. However, patients around the world who use prosthetic hands become frustrated because of the long and tiring process of training and rehabilitation. This causes a large number of people to abandon the prosthesis. The shortcomings of the existing prosthesis are high latency, low accuracy, fewer degrees of freedom, artificial feel, and long training periods. This research study/project aims to address the pain points of the patients which force them to abandon the prosthesis.

In this project, we aimed towards making an existing prosthesis functional and increasing its degrees of freedom in this process. Moreover, we embedded a camera on the palm of the prosthesis which would take real-time images of the scene. These images would then be processed under the computer vision algorithms and determine a suitable grip for the object depending on its shape and class. Haptic feedback has also been incorporated into the prosthetic hand. All these improvements are for the purpose of creating a realistic product and encouraging the user to continue their journey with the prosthetic hand to make their life easier. Moreover, the project aims to provide control to the user, for which we have developed a mobile application to provide customization.

The dataset for this project is a custom one, created in the process of this project. It includes the relevant objects that are used in daily tasks. The major contributions of this research study are listed as follows:

1. A custom dataset for the purpose of use in prosthetic hands including objects patients might encounter on a daily basis.
2. Improving the mechanical design of the prosthetic hand and making it functional for use.
3. Training custom object detection model and using depth estimation to intelligently determine grip patterns.
4. A mobile application that connects to the Raspberry Pi and is able to send signals to enable different grip patterns.
5. Integration of machine learning models on an edge computing device for better responsiveness.

7.2 Future Work

As mentioned earlier, there is a dire need for research in this field especially in Asian countries. The following could be the future direction and further improvements for the field of prosthesis:

- Attempt to increase the degrees of freedom of the fingers and the thumb without a massive increase in price.
- Formation of a customized dataset solely for the purpose of training computer vision algorithms for prosthetic hands.
- Deployment of computer vision algorithms on edge-computing device for a real-time and fast response.
- Incorporation of object detection, depth estimation, and EMG-based computer vision models for grip determination.
- Development of mobile applications for giving maximum control of the prosthetic hand to the user.
- Providing useful feedback to the user using different techniques to allow for a realistic experience.
- Incorporating a training program inside the mobile application to reduce training frustration.
- Integration of different models to make a fully autonomous prosthetic hand control system.

Bibliography

[1] K. Ziegler-Graham, E. J. MacKenzie, P. L. Ephraim, T. G. Travison, and R. Brookmeyer, "Estimating the prevalence of limb loss in the united states: 2005 to 2050," *Archives of physical medicine and rehabilitation*, vol. 89, no. 3, pp. 422–429, 2008.

[2] L. Resnik, M. R. Meucci, S. Lieberman-Klinger, C. Fantini, D. L. Kelty, R. Disla, and N. Sasson, "Advanced upper limb prosthetic devices: implications for upper limb prosthetic rehabilitation," *Archives of physical medicine and rehabilitation*, vol. 93, no. 4, pp. 710–717, 2012.

[3] D. Latour, "Advances in upper extremity prosthetic technology: rehabilitation and the interprofessional team," *Cursrent Physical Medicine and Rehabilitation Reports*, vol. 10, no. 2, pp. 71–76, 2022.

[4] C. T. Mai, J. L. Isenburg, M. A. Canfield, R. E. Meyer, A. Correa, C. J. Alverson, P. J. Lupo, T. Riehle?Colarusso, S. J. Cho, D. Aggarwal, and R. S. Kirby, "National population?based estimates for major birth defects, 20102014," *Birth Defects Research*, vol. 111, p. 14201435, Oct 2019.

[5] C. Gstoettner, S. Salminger, and O. C. Aszmann, "General considerations on upper limb amputation and its levels," p. 3–11, Jan 2021.

[6] S. Salminger, H. Stino, L. H. Pichler, C. Gstoettner, A. Sturma, J. A. Mayer, M. Szivak, and O. C. Aszmann, "Current rates of prosthetic usage in upper-limb

amputees–have innovations had an impact on device acceptance?," *Disability and Rehabilitation*, vol. 44, no. 14, pp. 3708–3713, 2022.

[7] "A national study of veterans with major upper limb aputation," *ResearchGate*, 2019.

[8] M. Espinosa and D. Nathan-Roberts, "Understanding prosthetic abandonment," in *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 63, pp. 1644–1648, SAGE Publications Sage CA: Los Angeles, CA, 2019.

[9] J. Murray, W. Carman, and J. MacKenzie, "Transmetacarpal amputation of the index finger: a clinical assessment of hand strength and complications," *The Journal of hand surgery*, vol. 2, no. 6, pp. 471–481, 1977.

[10] C. Lake and R. Dodson, "Progressive upper limb prosthetics," *Physical Medicine and Rehabilitation Clinics*, vol. 17, no. 1, pp. 49–72, 2006.

[11] Aug 2021.

[12] "The evolution of functional hand replacement: From iron prostheses to hand transplantation," vol. 22.

[13] K. J. Zuo and J. L. Olson, "The evolution of functional hand replacement: From iron prostheses to hand transplantation," *Plastic surgery (Oakville, Ont.)*, vol. 22, no. 1, p. 44–51, 2014.

[14] "Passive prosthetic hands and tools," vol. 42.

[15] J. J. Stokosa, "Limb prostheses options," Jan 2021.

[16] "A compact-sized surface emg sensor for myoelectric hand prosthesis," vol. 9.

[17] M. A. Abd, J. Ingicco, D. T. Hutchinson, E. Tognoli, and E. D. Engeberg, "Multichannel haptic feedback unlocks prosthetic hand dexterity," vol. 12, Feb 2022.

[18] G. Ghazaei, A. Alameer, P. Degenaar, G. Morgan, and K. Nazarpour, “Deep learning-based artificial vision for grasp classification in myoelectric hands,” *Journal of neural engineering*, vol. 14, no. 3, p. 036025, 2017.

[19] “Vincent prosthetic hand,” 2015.

[20] “Ilimb prosthetic hand,” 2023.

[21] “Touch bionics,” Oct 2008.

[22] “Michelangelo prosthetic hand,” 2016.

[23] F. Cordella, A. Ciancio, R. Sacchetti, A. Davalli, A. Cutti, E. Guglielmelli, and L. Zollo, “Literature review on needs of upper limb prosthesis users,” 2016.

[24] G. Kootstra, M. Popović, J. A. Jørgensen, K. Kuklinski, K. Miatliuk, D. Kragic, and N. Krüger, “Enabling grasping of unknown objects through a synergistic use of edge and surface information,” *The international journal of robotics research*, vol. 31, no. 10, pp. 1190–1213, 2012.

[25] I. Lenz, H. Lee, and A. Saxena, “Deep learning for detecting robotic grasps,” *The International Journal of Robotics Research*, vol. 34, no. 4-5, pp. 705–724, 2015.

[26] A. Saxena, J. Driemeyer, and A. Y. Ng, “Robotic grasping of novel objects using vision,” *The International Journal of Robotics Research*, vol. 27, no. 2, pp. 157–173, 2008.

[27] S. Došen and D. B. Popović, “Transradial prosthesis: artificial vision for control of prehension,” *Artificial organs*, vol. 35, no. 1, pp. 37–48, 2011.

[28] M. Markovic, S. Dosen, D. Popovic, B. Graimann, and D. Farina, “Sensor fusion and computer vision for context-aware control of a multi degree-of-freedom prosthesis,” *Journal of neural engineering*, vol. 12, no. 6, p. 066022, 2015.

[29] M. Kopicki, R. Detry, M. Adjigble, R. Stolkin, A. Leonardis, and J. L. Wyatt, "One-shot learning and generation of dexterous grasps for novel objects," *The International Journal of Robotics Research*, vol. 35, no. 8, pp. 959–976, 2016.

[30] Y. Cho, K. Liang, F. Folowosele, B. Miller, and N. V. Thakor, "Wireless temperature sensing cosmesis for prosthesis," in *2007 IEEE 10th International Conference on Rehabilitation Robotics*, pp. 672–677, IEEE, 2007.

[31] Y. Ueda and C. Ishii, "Development of a feedback device of temperature sensation for a myoelectric prosthetic hand by using peltier element," in *2016 International Conference on Advanced Mechatronic Systems (ICAMechS)*, pp. 488–493, IEEE, 2016.

[32] H. Yamada, Y. Yamanoi, K. Wakita, and R. Kato, "Investigation of a cognitive strain on hand grasping induced by sensory feedback for myoelectric hand," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3549–3554, IEEE, 2016.

[33] J. M. Walker, A. A. Blank, P. A. Shewokis, and M. K. OMalley, "Tactile feedback of object slip facilitates virtual object manipulation," *IEEE transactions on haptics*, vol. 8, no. 4, pp. 454–466, 2015.

[34] M. Aziziaghdam and E. Samur, "Providing contact sensory feedback for upper limb robotic prosthesis," in *2014 IEEE Haptics Symposium (HAPTICS)*, pp. 575–579, IEEE, 2014.

[35] J. Wheeler, K. Bark, J. Savall, and M. Cutkosky, "Investigation of rotational skin stretch for proprioceptive feedback with application to myoelectric systems," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 18, no. 1, pp. 58–66, 2010.

[36] T. Morita, T. Kikuchi, and C. Ishii, "Development of sensory feedback device for myoelectric prosthetic hand to provide hardness of objects to users," *Journal*

of robotics and mechatronics, vol. 28, no. 3, pp. 361–370, 2016.

[37] S. Casini, M. Morvidoni, M. Bianchi, M. Catalano, G. Grioli, and A. Bicchi, "Design and realization of the cuff-clenching upper-limb force feedback wearable device for distributed mechano-tactile stimulation of normal and tangential skin forces," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1186–1193, IEEE, 2015.

[38] C. Antfolk, A. Björkman, S.-O. Frank, F. Sebelius, G. Lundborg, and B. Rosen, "Sensory feedback from a prosthetic hand based on air-mediated pressure from the hand to the forearm skin," *Journal of rehabilitation medicine*, vol. 44, no. 8, p. 702, 2012.

[39] A. Akhtar, M. Nguyen, L. Wan, B. Boyce, P. Slade, and T. Bretl, "Passive mechanical skin stretch for multiple degree-of-freedom proprioception in a hand prosthesis," in *Haptics: Neuroscience, Devices, Modeling, and Applications: 9th International Conference, EuroHaptics 2014, Versailles, France, June 24-26, 2014, Proceedings, Part II 9*, pp. 120–128, Springer, 2014.

[40] N. Jorgovanovic, S. Dosen, D. J. Djozic, G. Krajoski, and D. Farina, "Virtual grasping: closed-loop force control using electrotactile feedback," *Computational and mathematical methods in medicine*, vol. 2014, 2014.

[41] M. Isaković, M. Belić, M. Štrbac, I. Popović, S. Došen, D. Farina, and T. Keller, "Electrotactile feedback improves performance and facilitates learning in the routine grasping task," *European journal of translational myology*, vol. 26, no. 3, 2016.

[42] H. Xu, D. Zhang, J. C. Huegel, W. Xu, and X. Zhu, "Effects of different tactile feedback on myoelectric closed-loop control for grasping based on electrotactile stimulation," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 24, no. 8, pp. 827–836, 2015.

[43] G. K. Patel, S. Dosen, C. Castellini, and D. Farina, "Multichannel electrotactile feedback for simultaneous and proportional myoelectric control," *Journal of neural engineering*, vol. 13, no. 5, p. 056015, 2016.

[44] M. Štrbac, M. Belić, M. Isaković, V. Kojić, G. Bijelić, I. Popović, M. Radotić, S. Došen, M. Marković, D. Farina, *et al.*, "Integrated and flexible multichannel interface for electrotactile stimulation," *Journal of neural engineering*, vol. 13, no. 4, p. 046014, 2016.

[45] B. Xu, A. Akhtar, Y. Liu, H. Chen, W.-H. Yeo, S. I. Park, B. Boyce, H. Kim, J. Yu, H.-Y. Lai, *et al.*, "An epidermal stimulation and sensing platform for sensorimotor prosthetic control, management of lower back exertion, and electrical muscle activation," *Advanced Materials*, vol. 28, no. 22, pp. 4462–4471, 2016.

[46] L. Jiang, Q. Huang, J. Zhao, D. Yang, S. Fan, and H. Liu, "Noise cancellation for electrotactile sensory feedback of myoelectric forearm prostheses," in *2014 IEEE International Conference on Information and Automation (ICIA)*, pp. 1066–1071, IEEE, 2014.

[47] M. A. F. Ismail and S. Shimada, "robothand illusion under delayed visual feedback: Relationship between the senses of ownership and agency," *PloS one*, vol. 11, no. 7, p. e0159619, 2016.

[48] M. Hans Rietman *et al.*, "Myoelectric forearm prostheses: State of the art from a user-centered perspective," *Journal of Rehabilitation Research and Development*, vol. 48, no. 6, p. 719, 2011.

[49] M. D'Alonzo, S. Dosen, C. Cipriani, and D. Farina, "Hyve: hybrid vibro-electrotactile stimulation for sensory feedback and substitution in rehabilitation," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 22, no. 2, pp. 290–301, 2013.

[50] "anatomy, pathology and treatment of the wrist hand - articles advice — white house clinic$_2$023,$''$ 2023.

[51] *PQ12 Actuator.*

[52] *handson technology datasheet g12-n20 geared mini dc motor.*

[53] andrewshaw, "What is raspberry pi 4? pinout, specs, projects datasheet - the engineering projects," Mar 2021.

[54] *Arduino® Uno R3.*

[55] *l298n dual h-bridge motor driver.*

[56] "Otto bock emg sensor," 2021.

[57] "An overview of the unb hand system$_2$023,$''$ 2023.

[58] ed, "Raspberry pi arduino serial communication - everything you need to know - the robotics back-end," Nov 2019.

[59] S. Rathi, *Bluetooth Architecture.*

[60] R. P. Ltd, "Raspberry pi camera module 2 – raspberry pi," 2016.

[61] D. D. R. Meneghetti, P. H. S. Domingues, B. d. F. V. Perez, T. S. B. Meyer, K. K. G. Cardoso, A. M. de Lima, M. Y. Gonbata, F. d. A. M. Pimentel, and P. T. Aquino Junior, "Annotated image dataset of household objects from the robofei@home team," 2020.

[62] S. A. Nene, S. K. Nayar, H. Murase, *et al.*, "Columbia object image library (coil-20)," 1996.

[63] G. Ghazaei, A. Alameer, P. Degenaar, G. Morgan, and K. Nazarpour, "An exploratory study on the use of convolutional neural networks for object grasp classification," 2015.

[64] J. Z. Zheng, S. De La Rosa, and A. M. Dollar, "An investigation of grasp type and frequency in daily household and machine shop tasks," in *2011 IEEE international conference on robotics and automation*, pp. 4169–4175, IEEE, 2011.

[65] J. Deng, X. Xuan, W. Wang, Z. Li, H. Yao, and Z. Wang, "A review of research on object detection based on deep learning," in *Journal of Physics: Conference Series*, vol. 1684, p. 012028, IOP Publishing, 2020.

[66] "Yolo algorithm for object detection explained," 2022.

[67] W. Lee, N. Park, and W. Woo, "Depth-assisted real-time 3d object detection for augmented reality," in *ICAT*, vol. 11, pp. 126–132, 2011.

[68] R. Zhang, P.-S. Tsai, J. E. Cryer, and M. Shah, "Shape-from-shading: a survey," *IEEE transactions on pattern analysis and machine intelligence*, vol. 21, no. 8, pp. 690–706, 1999.

[69] D. Eigen and R. Fergus, "Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture," in *Proceedings of the IEEE international conference on computer vision*, pp. 2650–2658, 2015.

[70] R. Szeliski, "Computer vision. texts in computer science," *Texts in Computer Science*, 2011.

[71] R. Ranftl, K. Lasinger, D. Hafner, K. Schindler, and V. Koltun, "Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 3, 2022.