DEVELOPMENT OF A VISION AIDED REACH-TO-GRASP PATH PLANNING AND CONTROLLING METHOD FOR TRANS-HUMERAL ROBOTIC PROSTHESES

Dannangoda Gamage Kanishka Madusanka

148039D

Thesis Submitted in Partial Fulfillment of the Requirements for the Degree Doctor of Philosophy in Biomedical Engineering

Department of Mechanical Engineering

University of Moratuwa

Sri Lanka

May 2018

DECLARATION

I declare that this is my own work and this thesis does not incorporate without acknowledgment any material previously submitted for a Degree or Diploma in any other University or institute of higher learning and to the best of my knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgment is made in the text.

Also, I hereby grant to University of Moratuwa the non-exclusive right to reproduce and distribute my thesis, in whole or in part in print, electronic or other medium. I retain the right to use this content in whole or part in future works (such as articles or books).

Signature: Date:

The above candidate has carried out research for the PhD thesis under our supervision.

Prof. R.A.R.C Gopura
Head/Professor,
Department of Mechanical Engineering,
University of Moratuwa, Sri Lanka.

Dr. Y.W.R. Amarasinghe
Senior Lecturer,
Department of Mechanical Engineering,
University of Moratuwa, Sri Lanka.

Prof. G.K.I. Mann
Professor,
Faculty of Engineering and Applied Science,
Memorial University of Newfoundland, Canada.

Abstract

This study proposes a reach-to-grasp path planning and controlling method for trans-humeral prostheses. Trans-humeral prostheses are used to replace the missing body part after the loss of upper limb (UL) above elbow. Reach-to-grasp paths refers to the paths taken by the human UL to reach towards an object with the intention of grasping.

A trans-humeral prosthesis has been designed and fabricated with 5DOF. A simulation environment has been proposed using the design. Simulation environment consists of a virtual shoulder joint which can be actuated according to a natural human shoulder using an Inertial Measurement Unit (IMU). Prosthesis and the simulation environment has been used to experimentally evaluate the proposed path planning method.

A reach-to-grasp path planning method combining Electromyography (EMG) signals and vision signals has been proposed. EMG Based Module (EBM) is capable of controlling prosthesis elbow motion effectively with an accuracy of 92%. Visual Servoing Module (VSM) consists of a 2-1/2D visual servoing system to center the object of interest to the hand of the prosthesis and to correct the orientation. An object reaching algorithm has been proposed to reach towards the object. Later, the EBM and the VSM has been fused using an fusion filter.

An improvement to the above method has been proposed to make the paths straight. It consists of a path generation module and a path tracking module. Path generation module is capable of generating a path towards the object. The object position is located and a path is generated from the current position of the prosthetic hand to the object position with the aid of vision. Path tracking module takes the prosthetic hand on the generated path considering shoulder motions. Two path tracking methods has been proposed: spatial path following method and Model Predictive Controller (MPC) based path tracking method. Proposed path planning method has been experimentally evaluated.

Keywords- **Trans-humeral prosthesis, electromyography, visual servoing, reach-to-grasp path planning, path following**

DEDICATION

To my parents,

D.G. Sunil Santha and *Indralatha Udayakanthi Wijesinghe*

Without whom none of my success would be possible

ACKNOWLEDGMENTS

After an intensive period of three years, today I am writing this note of thanks in the finishing touch on my thesis. It has been a period of intense learning for me, not only in the scientific arena, but also on a personal level. Working on this thesis has had a big impact on me. I would like to reflect on the people who have supported and helped me throughout this period.

I would first like to thank my thesis supervisor Prof. R.A.R.C. Gopura for insightful guidance, valuable suggestions, and timely encouragement. The door to his office was always open whenever I ran into a trouble spot or had a question about my research or writing. He consistently allowed this thesis to be my own work, but steered me in the right direction whenever he thought I needed it. Furthermore my co-supervisors Dr. Y.W.R. Amarasinghe and Prof. G.K.I. Mann shred their tremendous support towards making this thesis a success. Their comments and suggestions on improving research work and writings was outstanding. I am extremely grateful for them for giving me the opportunity and support towards reading for my doctoral degree under their supervision.

I take this opportunity to express my gratitude towards the Senate Research Council of University of Moratuwa for the financial support granted to me under the grant no. SRC/LT/2013/07 to conduct my studies throughout three years.

Moreover, enormous thanks to the progress review panel members: Prof. Rohan Munasinghe, Dr. Palitha Dassanayake, and Dr. Sanath Jayawardena who guided me throughout the study. Thank you very much for your comments and suggestions which helped me to drive my study in the right direction.

I wish to express my sincere thanks towards Dr. Thilina Lalitharatne, Mr. Pubudu Ranaweera, Mr. Achintha Mihiran and all colleagues of Bionics Laboratory for their tremendous support. Besides that special thanks goes to Mr. Viraj Muthugala, a PhD candidate of Department of Electrical Engineering for the support and the comments provided and helping me at hard times.

In recognition of all the help and support I would like to remind Dr. W.K. Wimalsiri (previous Head of the Department), Dr. Damith Chathuranga (research coordinator), and all the staff of the Department of Mechanical Engineering, University of Moratuwa.

Special thank goes to all of my friends who helped me in all sorts of forms to make this thesis a success. Moreover, I would like to express my gratitude towards Ms. Nathasha Peiris for the tremendous support provided.

Without any doubt, I must thank my parents, for the endless support and commitment. Their love and encouragements helped me a lot to lift me up in every downfall.

Kanishka Madusanka
kanishkam@ieee.org

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

AAD	Abduction/Adduction
ADL	Activities of Daily Living
ANN	Artificial Neural Network
API	Application Package Interface
BLDC	Brush Less DC
CCD	Charge Coupled Device
CMOS	Complementary Metal Oxide Semiconductor
DAQ	Data Acquisition
DC	Direct Current
DH	Denavit Hartenberg
DLS	Damped Least Squares
DOF	Degrees of Freedom
EBM	EMG Based Module
ECoG	Electrocorticography
EEG	Electroencephalography
EFPMB	EMG Force Proportional Moment Balance Model
EMG	Electromyography

EPE	End Point Error
FE	Flexion/Extension
FIS	Fuzzy Inference System
FK	Forward Kinematics
IBVS	Image Based Visual Servoing
IER	Internal External Rotation
IK	Inverse Kinematics
IMU	Inertial Measurement Unit
IP	Inertia Point
MAV	Mean Absolute Value
MDA	Multiple Discriminant Analysis
MMG	Mechanomyography
MPC	Model Predictive Controller
MSE	Mean Square Error
MU	Motor Unit
MUAP	Motor Unit Action potential
MUAPT	Motor Unit Action potential Train
PBVS	Position Based Visual Servoing
PC	personal Computer
PCA	Principal Component Analysis
PID	Proportional Integral Derivative

RMS	Root Mean Square
RMSE	RMS Error
ROM	Ranges of Motions
SD	Standard Deviation
SP	Supination/Pronation
SSC	Slope Sign Change
TDANN	Time delayed ANN
TMR	Targeted Muscle Reinnervation
UI	User Interface
UL	Upper limb
URD	Ulnar Radial Deviation
US	Ultrasonic
USA	United states
V-REP	Virtual Robotic Experimentation Platform
VSM	Visual Servoing Module
WL	Waveform Length
ZC	Zero Crossings

Chapter 1

INTRODUCTION

Prosthesis (plural: Prostheses) is an artificial device which replaces a missing body part. "Prostheses" is a word developed from ancient Greek words "*prós*" means for "in addition" and "*thésis*" means for "a placing" [1]. Among prostheses limb prostheses holds importance since loss of limbs make amputees unable to carry out his/her Activities of Daily Living (ADL). Limbs can be lost due to traumas, accidents, diseases, or congenital conditions [2].

Development of prosthetic limbs has been started way back in the history. First confirmed use of a prosthetic device dates back to 950-710 BC [3] which was a toe prosthesis found attached to the right foot of a female. This prosthesis was built using wood and leather. Prostheses were developed using wood, iron, steel, and copper in old days. However, their use is more cosmetic rather than functional. Development of prostheses were highly influenced by world war I and II, which resulted in developing more functional prostheses. National Academy of Sciences (NSA) of United States (USA) began to support research and development of prostheses after world war II [4]. Hence, more scientific efforts have been flooding towards the development and control of prostheses.

The concept of cybernetics which means communications and control systems in machines and living things influenced the development of functional prostheses [5]. First, electrically powered prosthesis have been developed with the support of USA government and IBM in 1949 [6]. Prostheses should be capable of being controlled according to human motion intention in order to carryout ADL

properly. Numerous efforts have been taken by the researchers to achieve functional prostheses which can be controlled according to the motion intention of the wearer.

However, the prostheses stand way back from the natural limb compared to the functionality, controllability, and appearance.

1.1 Prosthesis as a Robot

A robot can be treated as a machine which can be programmed by a computer. Moreover, robots are capable of carrying out complex series of actions automatically. Robots are used to perform human activities safely and productively. These machines are used in many areas such as manufacturing, space exploration, military applications, medical applications, etc. In most applications humans and robots are not directly connected. However, in prosthetic applications the human robot interaction is significant.

Most of externally powered Upper Limb (UL) prostheses can be considered as serial link manipulator robots. A serial link manipulator robot is made up of mechanical links connected serially by joints. These robots consist of the mechanical manipulator (links), end effector, controllers, actuators, and sensors. Generally, the number of joints can be treated as the number of Degrees of Freedom (DOF). The devices attached to the output links are the end effectors. The end effector is used to manipulate objects. In an UL prosthesis, prosthetic hand acts as the end effector. Robot controllers ranging from low level joint controllers to high level intelligent controllers are used to drive the end effector as desired. Potentiometers and/or encoders are used to measure robot joint motions which will also be used for the purpose of feedback control. Apart from that external sensors such as vision, Inertial Measurement Unit (IMU), etc. can be used to control a robotic manipulator depending on the application.

1.2 Motivation

Throughout the day a wide variety of tasks are performed using the human UL. Hence, loss of part of the UL leaves the amputees with difficulties in carrying out even the basic ADL which makes them helpless. Prostheses are developed in order to uplift their living standards and to add lost functions. However, the human UL is a complicated biological structure with bones and muscles acting together to provide many functions. Therefore, providing the same functionality with a mechanical device is a difficult task. Moreover, these devices should be developed to work with human motion intentions and to achieve human-like motions. Researchers are consistently working on developing prostheses and their controllers which can mimic human UL motions and functionality. However, current developments need improvements to match the functionalities of a human UL.

Developments in task level prosthetic controllers are very rare compared to the vastly available joint level controllers. Hence, task level prosthetic controllers should be developed so that several tasks can be performed by the prosthesis. Since most of the tasks performed by an UL involves reach-to-grasp motions it is important to develop methods to achieve such capabilities in prostheses to progress research in task level prosthetic controllers.

Moreover, the prosthesis is a separate device which operates using the signals extracted from the amputee. However, the extracted biological signals cannot exactly identify the motion intentions of the amputee. Hence, external sensors are used in order to assist bio-signal based controllers.

Following are the drawbacks of available prosthetic controllers which motivated the work presented in this thesis.

- Most of the available prostheses are not capable of generating human-like motions due to few control inputs available after an amputation.
- In joint level prosthetic controllers each and every joint should be controlled separately in order to complete a task.
- The prostheses did not have required sensory inputs to judge weather the prosthesis moves towards the object of interest.
- Reach-to-grasp path planning methods for trans-humeral prostheses are not yet been developed, however it is an essential part in developing task level prosthetic controllers.

Therefore, the work presented in this thesis is towards developing vision aided reach-to-grasp path planning methods for trans-humeral prostheses considering stump arm motions. The prosthetic motions should work according to the human motion intentions. Hence, an Electromyography (EMG) based elbow motion prediction method has been developed. Moreover, a 2-1/2D visual servoing based reach-to-grasp path planning method has been developed integrating EMG based elbow prediction method. This method is capable of taking the prosthetic hand towards an object. However, the paths taken by the prosthetic hand has some curvatures. Hence, improvement to the previous reach-to-grasp path planning method has been proposed. This method consists of a path generation module and a path tracking module. Path generation module is used to generate a path towards the destination. Two path tracking methods have been proposed: spatial path following method and the Model Predictive Controller (MPC) based path tracking method. These two methods are capable of taking the prosthetic arm towards the object of interest in a stringent line path. These path tracking methods are used to plan reach-to-grasp motions of the trans-humeral prosthesis.

1.3 Contributions of the Thesis

The research work presented in this thesis is towards the development of vision based reach-to-grasp path planning methods for a trans-humeral prosthesis. The major contributions are outlined below.

- Propose an EMG-Force Proportional and Moment Balance (EFPMB) model for elbow motion prediction.
- Propose a vision aided path planning method for reach-to-grasp motions of the trans-humeral prosthesis based on a 2-1/2D method of visual servoing.
- Propose a spatial path following method for the trans-humeral prosthesis, compensating shoulder motions.
- Propose an improved dynamic path tracking method based on a Model Predictive Controller (MPC) for the trans-humeral prosthesis.
- Develop an improved vision based path planning method for reach-to-grasp motions of the trans-humeral prosthesis using proposed path following methods.

1.4 Thesis Overview

The thesis consists of six chapters. Organization of the thesis is as follows. Chapter 2 covers the literature review. Chapter 3 explains the prosthesis and the simulation environment used for the experimental evaluation of path planning methods. Chapter 4 proposes the 2-1/2D visual servoing based reach-to-grasp path planning method. Chapter 5 proposes the reach-to-grasp path planning method based on path tracking methods. Chapter 6 concludes the thesis with future directions. The contents of the following chapters are elaborated below.

Chapter 2:- Literature Review

This chapter covers the bio-mechanics of human UL and currently available prosthetic controllers. Bio-mechanics of UL plays a vital role in designing prostheses and their controllers. Moreover, evolution of prosthesis controllers has been extensively reviewed in this chapter. besides, reach-to-grasp paths of a UL and path planning methods has been discussed.

Chapter 3:- The Trans-humeral Prosthesis Used to Evaluate the Path Planing Method

The simulation environment and the 5 DOF trans-humeral prosthesis which has been developed is elaborated in this chapter. The simulation environment has been developed using virtual robot experimentation platform (V-REP) to simulate prosthetic control algorithms. Main motion units of the prosthesis: elbow, forearm, wrist, and hand have been explained with their designs. The low level controllers used to actuate these motion units have been explained and evaluated experimentally. The prosthesis and the simulation environment have been used in the following chapters to evaluate the proposed path planning Method.

Chapter 4:- Reach-to-Grasp Path Planning Based on a 2-1/2D Method of Visual Servoing

This chapter proposes a reach-to-grasp path planning method based on a 2-1/2D Visual Servoing Module (VSM) and a EMG Based Module (EBM). The EBM is used to controls elbow flexion/extension (FE). A camera and an Ultrasonic (US) sensor are fitted on the palm of the prosthesis. The VSM plans the path towards the detected object by the camera if a reach request is received. Since both VSM and EBM produce elbow FE angles, these two modules are fused to get a resultant elbow FE angle.

Chapter 5:- Reach-to-Grasp Path Planning Based on Path Tracking Methods

This chapter proposes an improvement to the previous method based on dynamic path tracking algorithms. This method makes the paths straight line, which was missing in the method discussed in chapter 4. The proposed method consists of two modules: path generation module and path tracking module. The path generation module generates a path towards the object of interest with the sensory feedback received from the camera and the US sensor fitted on the palm. The path tracking module uses two novel algorithms proposed in this study, in order to track the generated path by the prosthetic hand.

Chapter 6:- Conclusion

This chapter concludes the thesis with future directions.

Chapter 2

LITERATURE REVIEW

Loss of part of the UL leaves an amputee with difficulties in performing even the most basic ADL. Full or partial loss of the UL can be caused by diseases, accidents, trauma, etc. [7]. With the aim of uplifting the living standards of amputees, continuous efforts on developing functional and reliable prostheses have been conducted by various researchers. [8–11].

Currently, different combinations of biological and sensory signals are used as inputs in achieving smoothly functioning and reliable controllers for UL prostheses. However, still these prosthetic devices and their controllers have not been able to match the performance of a natural UL.

This chapter covers the bio-mechanics & kinematics of human UL, UL prostheses, available prosthetic control methods, and reach-to-grasp path planning methods.

2.1 Bio-mechanics and Kinematics of Upper Limb (UL)

Bio-mechanics and kinematics of the UL plays an important role when designing and controlling UL prostheses. Since the joints and their motions of a UL prosthesis should be developed in accordance with the bio-mechanics and the kinematics of human UL.

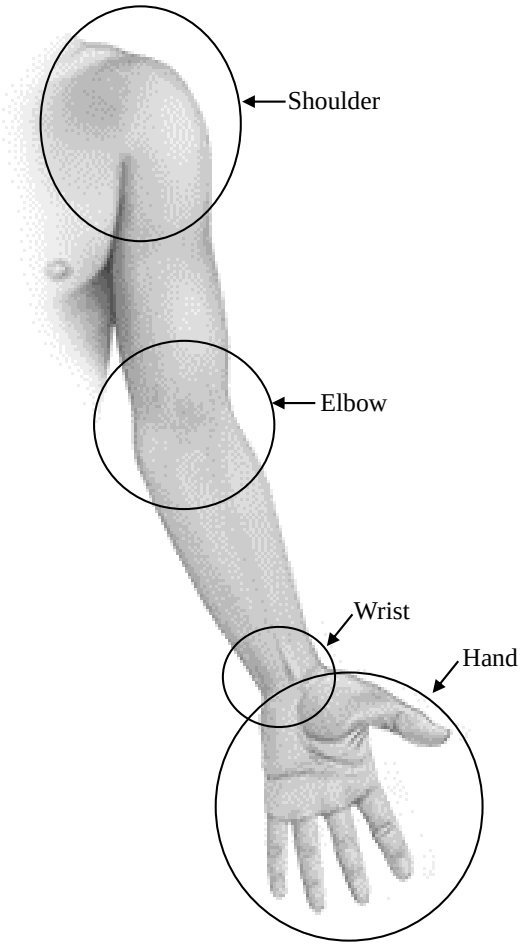


Figure 2.1: Four major sections of UL. UL is divided into four major sections in order to ease analyzing bio-mechanics. These four sections are shoulder, elbow, wrist, and hand.

2.1.1 Bio-mechanics of Upper Limb (UL)

Bio-mechanics is the study of mechanical laws relating to the movement or structure of living beings. In order to analyze the bio-mechanics, the human UL can be divided into four major sections: shoulder, elbow, wrist, and hand (see Fig. 2.1). The movements of the UL are three dimensional (3D). These movements are described using 3 planes defined with respect to the human body: sagittal plane, transverse plane. and frontal plane (See Fig. 2.2).

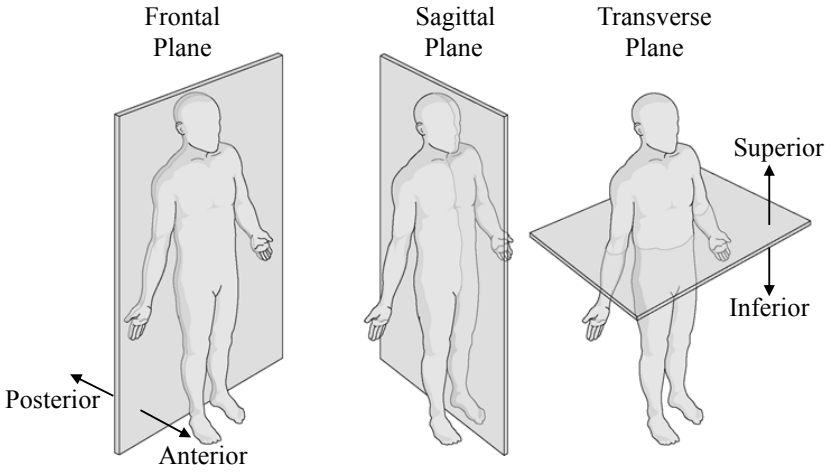


Figure 2.2: Planes of human body. Human body movements can be explained using three planes: frontal, sagittal, and transverse. Frontal plane is also refereed to as coronal plane. Transverse plane is also refereed to as horizontal plane [12].

Shoulder

The shoulder consists of three major bones viz. clavicle, scapula and humerus. The clavicle and the scapula supports suspension of the UL and also serves as a location for muscle attachment. the humeral head (ball) of the humerus and the glenoid cavity (socket) of the scapula is combined to form the gleno-humeral joint which is known as the shoulder joint (refer Fig. 2.3) [13]. The humerus bone connects the shoulder and elbow. Shoulder joint can be modelled as a ball and socket joint with 3 DOF: shoulder FE, abduction/adduction (AAD) and internal/external rotation (IER). Movement of the gleno-humeral joint in sagittal plane results in shoulder FE motion. Shoulder AAD is the movement of the UL in frontal plane around the shoulder joint. IER is the rotation of the UL towards and away from the trunk around the shoulder joint with the elbow flexed 90 degrees (Fig. 2.4). The range of motions (ROM) of shoulder joints are given in Table 2.1 [14].

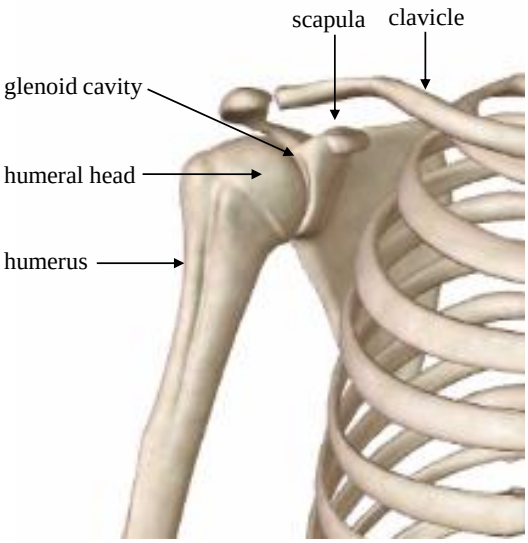


Figure 2.3: Bones of shoulder joint. Humeral head and glenoid cavity creates the shoulder joint. Shoulder joint can be treated as a ball and socket joint with 3 DOF.

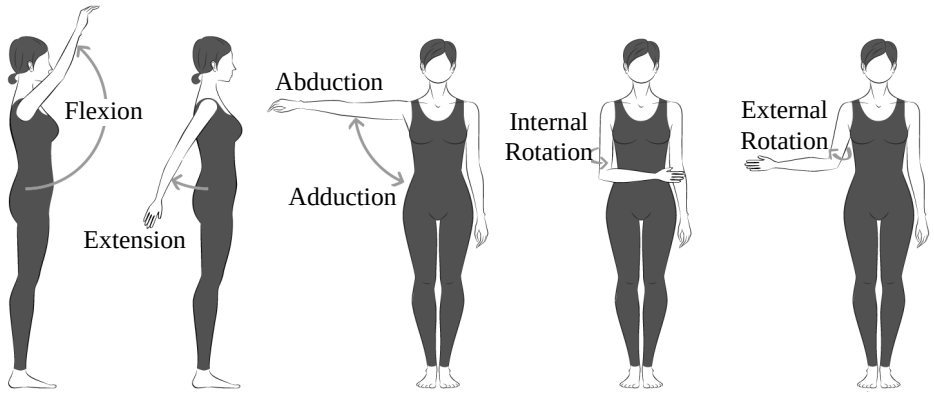


Figure 2.4: Motions of shoulder joint

Elbow

The elbow joint consists of two articulating surfaces: between the trochlea & trochlear notch and between the capitulum & the head of the radius. Elbow flexion is caused by contraction of the anterior muscles of the upper arm and elbow extension is caused by contraction of the posterior muscles of the upper arm. The elbow joint can be modelled as a hinge joint. The proximal radioulnar joint is located immediately after the elbow joint. Radial notch moves around the head of the radius which permits axial rotation of the forearm, known as

Table 2.1: Range of motions of shoulder [14]

Motion	**Range**
Shoulder Flexion/Extension	158° / 53°
Shoulder Abduction/Adduction	170° / 0°
Shoulder Internal/External Rotation	70° / 90°

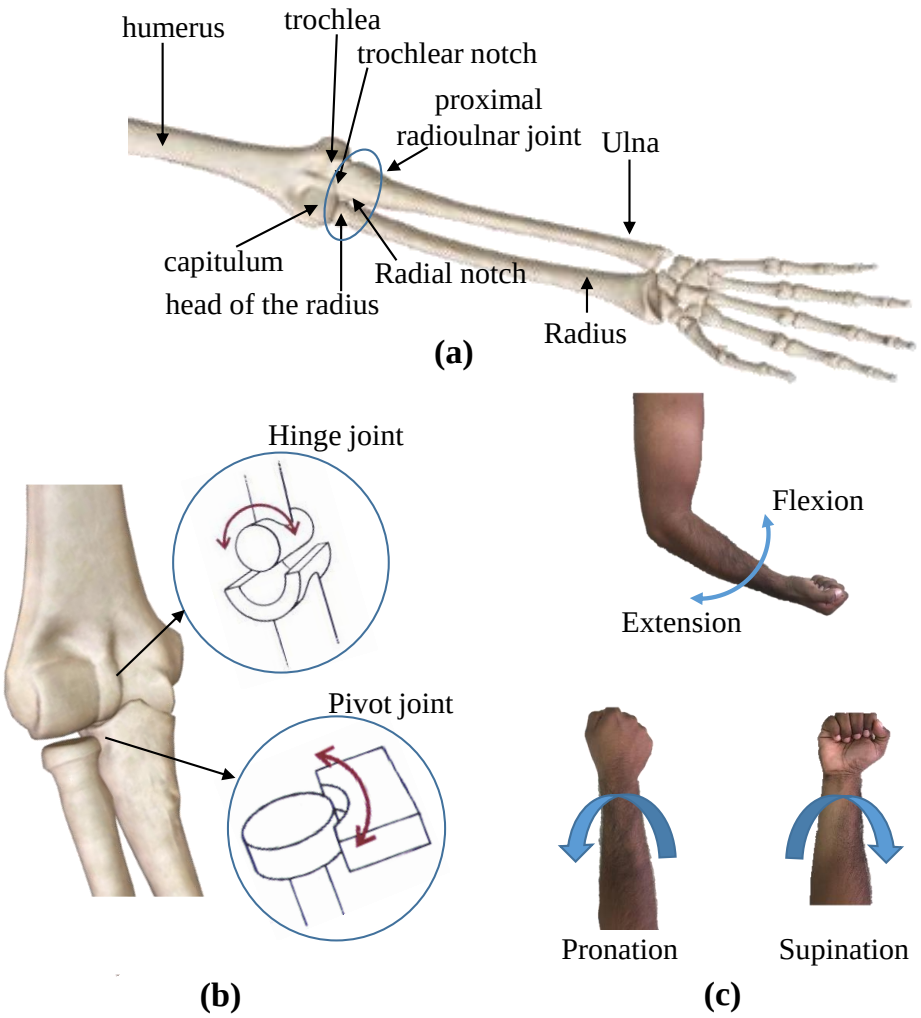


Figure 2.5: Motions of elbow joint. (a) Bones of forearm and elbow joint, (b) Modeling elbow joint as 2 DOF, (c) 2 DOF at elbow joint.

supination/pronation (SP). This motion is visible at the distal end of the forearm which can be modelled as a pivot joint (Refer Fig. 2.5). Since both the proximal radioulnar joint and the elbow joint is closely located, these two joints form the elbow complex. Hence, the elbow joint is considered to have 2 DOF.

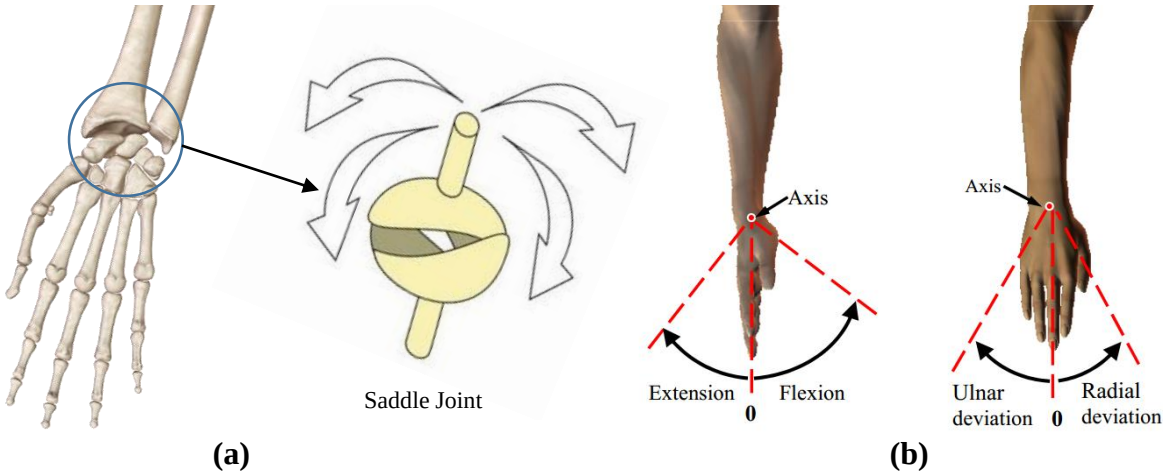


Figure 2.6: Motions of wrist joint [15]. (a) Modeling wrist as 2 DOF, (b) 2 DOF at wrist joint.

The muscles along humerus contracts and extend to provide elbow FE. Three muscles are located anterior to the humerus: Biceps brachii, brachialis and coracobracialis and the triceps muscle is located posterior to the humerus.

The average elbow joint is capable of 0° in extension and 146° in flexion. Pronation and supination of the forearm have ranges of 71° and 84° respectively [14].

Wrist

The wrist is connected to the elbow with the radius and ulna bones. A number of muscles which extend along the radius and ulna, cause two wrist motions: wrist FE and ulnar/radial deviation (URD) (Fig. 2.6). FE and URD are commonly assumed to occur around two axes or rotation slightly offsetting one another. Hence, wrist is considered to have 2 DOF. The wrist can be modelled as a saddle joint [refer Fig. 2.6(a)]. Average ROM in the wrist are 73° in flexion, 71° in extension, 19° in radial deviation, and 33° in ulna deviation [14].

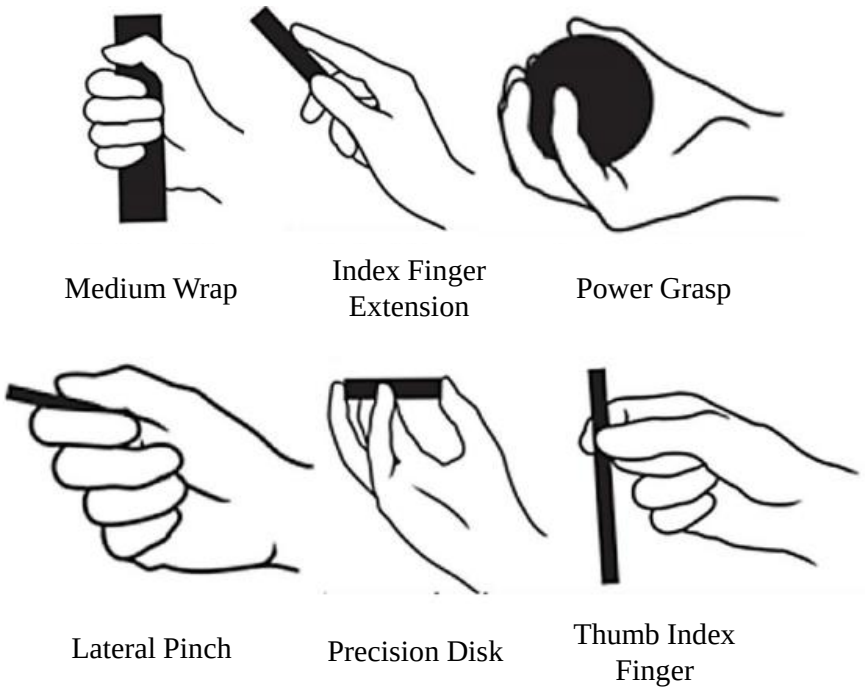


Figure 2.7: Grasping patterns frequently used in ADL [17]

Hand

Most complicated component of the UL is the hand which consists of carpal, metacarpal and phalangeal bones. Five metacarpal and 14 phalangeal bones form the palm and fingers of the hand. Eight carpal bones form the wrist [16]. The five digits (or fingers) of the hand are the thumb, index finger, middle finger, ring finger and little finger. The fingers, cumulatively have 21 DOF. The hand being a highly functional and important component of the human body, it is essential in carrying out most of the ADL.

Hand is responsible for grasping and manipulating objects. In order to grasp objects, the hand is capable of adapting into different forms which are referred to as grasping patterns. In ADL there are few grasping patterns which are frequently used. They are, medium wrap, index finger extension, power grasp, lateral pinch, precision disk, and thumb-index finger [17]. Fig. 2.7 depicts these grasping patterns.

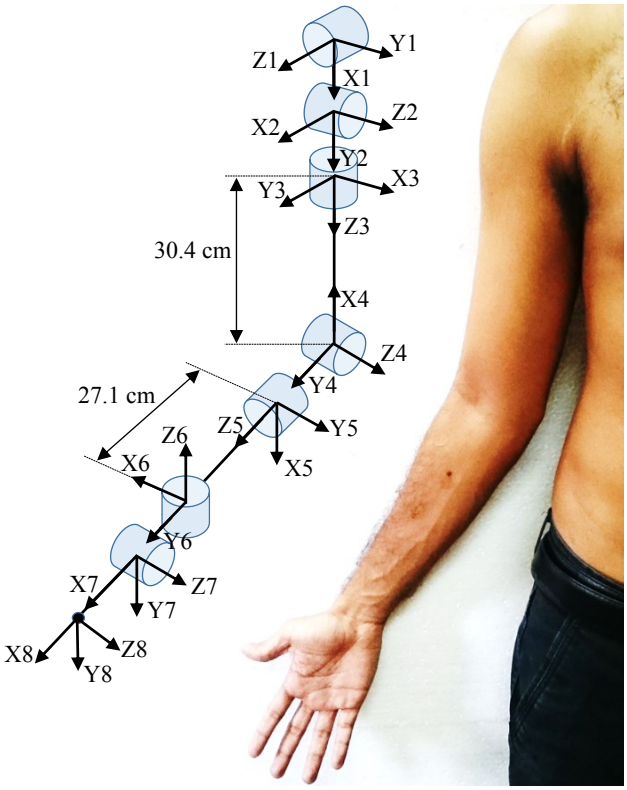


Figure 2.8: Kinematic structure of upper limb. Distance from shoulder to elbow and elbow to wrist are extracted from [18], where 50^{th} percentile data of men were taken.

2.1.2 Kinematics of Upper Limb (UL)

Kinematics is the study of motion without considering the forces which cause the motion. Kinematics plays an important role in analyzing bio-mechanics of the UL. The UL can be modelled as a 7 DOF structure: 3 DOF at shoulder, 2 DOF at elbow, and 2 DOF at wrist. The hand acts as the end effector of the kinematic chain. The kinematic structure of the UL is shown in Fig. 2.8. All 7 DOF of the UL can be represented using revolute joints. According to the shoulder model, 3 DOF at shoulder can be represented as 3 revolute joints with mutually perpendicular joint axes. Similarly, the motion of elbow and wrist joints can be represented as 2 revolute joints each with mutually perpendicular axes.

The kinematcs of UL are analysed using the conventional Denavit-Hartenberg (DH) convention [19]. The motion of end effector (hand) is represented with re-

spect to (w.r.t.) the shoulder. Hence, shoulder acts as the base of the kinematic chain. The coordinate frame attached to the base is frame 1 ($[X1, Y1, Z1]$). Coordinate frame 8 ($[X8, Y8, Z8]$) is attached to the hand of the UL. Axes of rotation of shoulder AAD, FE, and IER are $Z1$, $Z2$, and $Z3$ respectively. Similarly the axes of rotation of elbow FE, SP, wrist URD, and wrist FE are $Z4$, $Z5$, $Z6$, and $Z7$ respectively.

In DH convention, 4 link parameters are defined for a particular link between two joints: θ, d, a, and α [20]. These parameters are substituted to the DH matrix to derive the transformation from one joint to the other. The DH matrix used to represent pose of j w.r.t. to $j-1$, T_j^{j-1} is given in (2.1). In (2.1), $c\theta$, $s\theta$, $s\alpha$, and $c\alpha$ represents $sin(\theta)$, $cos(\theta)$, $sin(\alpha)$, and $cos(\alpha)$ respectively.

$$T_j^{j-1} = \begin{pmatrix} c\theta & -s\theta c\alpha & s\theta s\alpha & ac\theta \\ s\theta & c\theta c\alpha & -c\theta s\alpha & \alpha s\theta \\ 0 & s\alpha & c\alpha & d \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{2.1}$$

The DH parameters: θ, d, a, and α are defined as follows (refer Fig. 2.9). θ is the angle between x_{j-1} and x_j axes about z_{j-1} axis. d is the distance from origin of frame $j-1$ to x_j axis along z_{j-1} axis. a is the distance between z_{j-1} and z_j axes along x_j axis. α is the angle from z_{j-1} to z_j about x_j axis.

The DH parameters for the human UL is given in Table 2.2. DH parameters $\theta(\theta_1$ *to* $\theta_7)$ are the joint angles from shoulder to wrist in the same order as coordinate frames (frame 1 to frame 7 in Fig. 2.8) are defined. Substituting these parameters in the DH matrix [refer (2.1)], transformation matrices ($T_2^1, T_3^2, T_4^3, T_5^4, T_6^5, T_7^6$, and T_8^7) can be obtained.

The transformation from shoulder (base, frame 1) to the hand (end effector, frame 8) can be stated as (2.2). Transformation matrix, T_8^1 gives the pose (position and orientation) of the hand w.r.t. the shoulder.

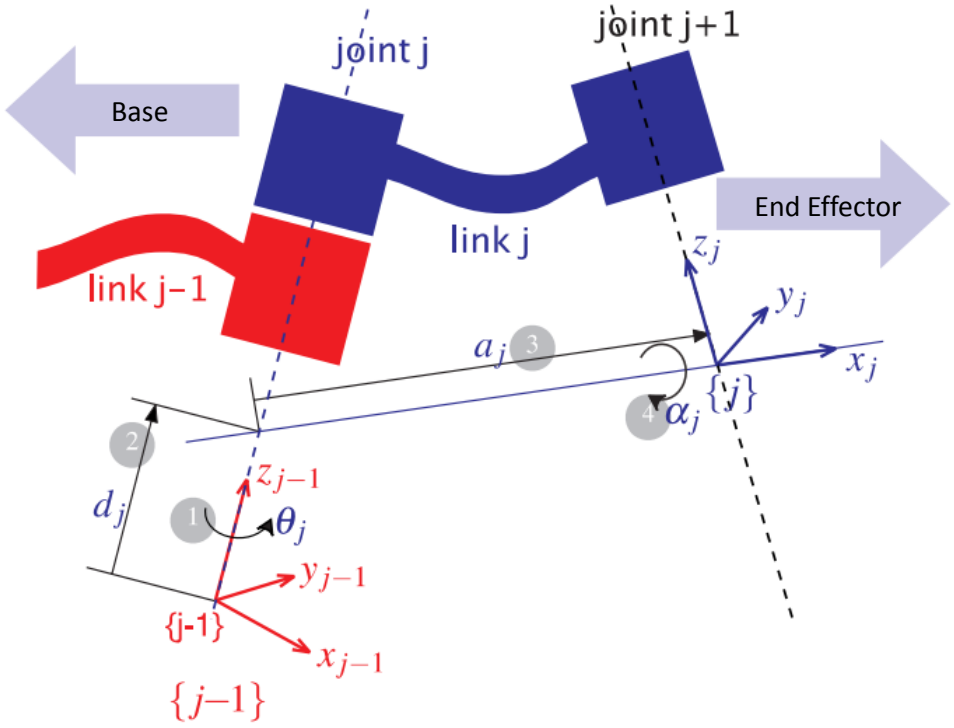


Figure 2.9: Definition of DH parameters. Parameters θ_j, d_j, a_j, and α_j are marked on the figure.

$$T_8^1 = T_2^1 \times T_3^2 \times T_4^3 \times T_5^4 \times T_6^5 \times T_7^6 \times T_8^7 \tag{2.2}$$

2.2 Upper Limb (UL) Prostheses

UL prostheses are used to replace the missing part of the UL after an amputation. The UL performs a wide variety of tasks, from the delicate and complex to the strong and forceful. An ideal UL prosthesis should be capable of doing all of these. Unfortunately, current UL prostheses can not perform the tremendous array of functions routinely done by our natural hands.

UL prostheses are designed and used for different levels of amputations such as shoulder disarticulation, trans-humeral amputation, elbow disarticulation, trans-radial amputation, wrist disarticulation, loss of partial hand, and loss of fingers. Fig. 2.10 depicts the levels of amputations.

Table 2.2: DH Parameters of human UL

Link	θ	$d(mm)$	$a(mm)$	α
1	θ_1	0	0	$\pi/2$
2	θ_2	0	0	$\pi/2$
3	θ_3	304	0	$\pi/2$
4	θ_4	0	0	$-\pi/2$
5	θ_5	271	0	$-\pi/2$
6	θ_6	0	5	$-\pi/2$
7	θ_7	0	0	0

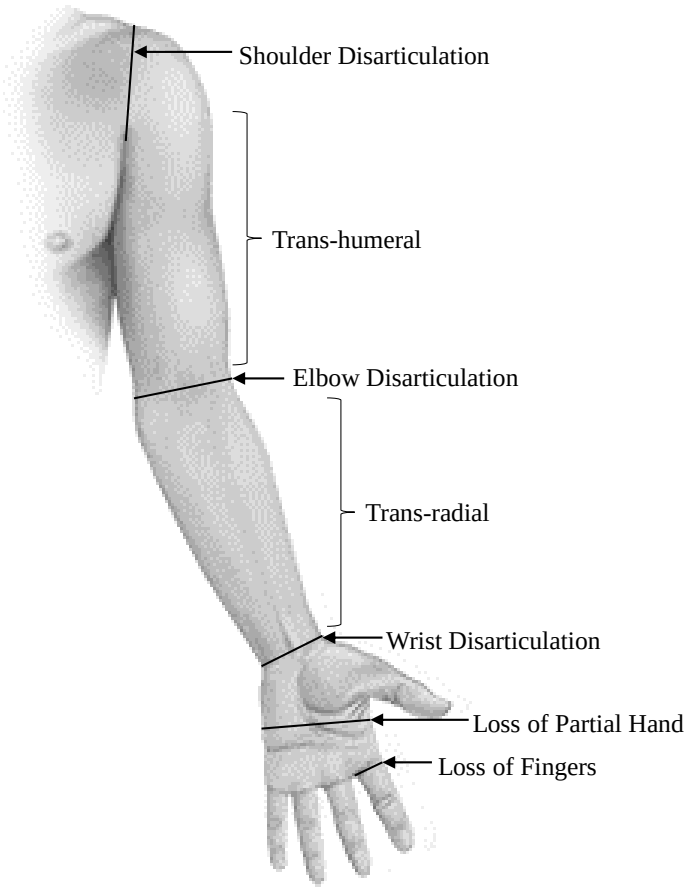


Figure 2.10: Levels of amputation. Amputation between shoulder and elbow is referred to as trans-humeral amputation and amputation between elbow and wrist is referred to as trans-radial amputation.

Moreover, prostheses can be categorized based on the way they are powered as cosmetic prostheses, body powered prostheses, and externally powered prostheses. The cosmetic prostheses are worn just for the appearance and body powered

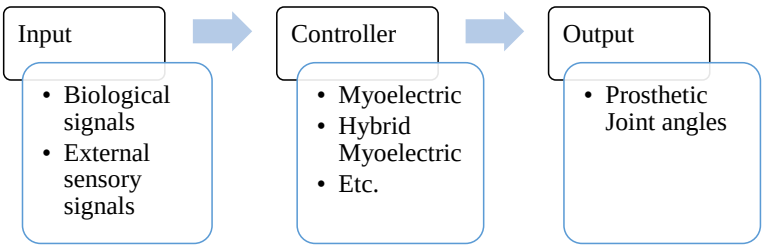


Figure 2.11: Overview of a prosthetic control system. Inputs to the system can be biological signals from the amputee or any external sensors used. The inputs are processed using the prosthetic controller to derive the prosthetic joint angles.

prostheses use wearer's body power to operate. For example, a body powered UL prosthesis can use the shoulder power of the amputee's healthy limb to operate the elbow, hand, and wrist using cables and mechanical locks [21]. Externally powered prosthetic devices are powered by an external power source such as batteries. Externally powered prostheses are the most functional among three categories. different types of controllers are developed using different input signals to control externally powered prostheses. These control systems are discussed in the following section.

2.3 Upper Limb Prosthetic Control Systems

Controlling a device that has been developed to replace a missing body part is a challenging task. Moreover, prostheses should be able to control according to the wearer's motion intention. Since biological signals carry information related to the motion intentions of the wearer, biological signals such as Electromyography (EMG) [10, 22], Electroencephalography (EEG) [23], and Electrocorticography (ECoG) [24] are used to control prostheses. Theses inputs are processed using a prosthetic controller to derive the prosthetic joint angles. Overview of a UL prosthetic control system is shown in Fig. 2.11.

EEG signals can be used to extract motion intentions of the human from the surface of the head [23]. ECoG uses electrodes placed on the surface of the brain

(under the skull) to capture signals related to motion intentions [24]. Among these biological signals, EMG is preferred by researchers due to its advantages such as higher signal to noise ratio and ease of extraction [21, 22]. EMG signals can be easily extracted from the muscles. However, amputations make muscles unavailable for EMG signal extraction. For example, after a trans-humeral amputation, only parts of the biceps brachii and triceps brachii are available. All other muscles below the point of amputation are lost.

Hence, prosthetic developers are experimenting with different combinations of sensory signals and biological signals to achieve smooth and reliable control of the prostheses.

2.3.1 Myoelectric (EMG) Signals

Human motion intentions are generated in the brain and transferred to the muscles through the nervous system. EMG (Myoelectric) are the signals that represent the current generated by the ionic flow across membranes of the muscle fibers. Muscle fibers are in groups called Motor Units (MU) where the activation of MU creates a Motor Unit Action Potential (MUAP). Continuous firing of muscle fibers creates a Motor Unit Action Potential Train (MUAPT). EMG signal is the summation of these MUAPTs (see Fig. 2.12).

EMG signals contain rich information regarding limb motions and can be used to control prostheses. EMG signals can be extracted in two ways. In the first method, surface electrodes are used to extract EMG signals non-invasively from the surface of the muscles. These signals are referred to as surface EMG. Second method is to extract EMG signals by inserting needle electrodes into the muscle which is an invasive process. These EMG signals are called as intramuscular EMG [25, 26].

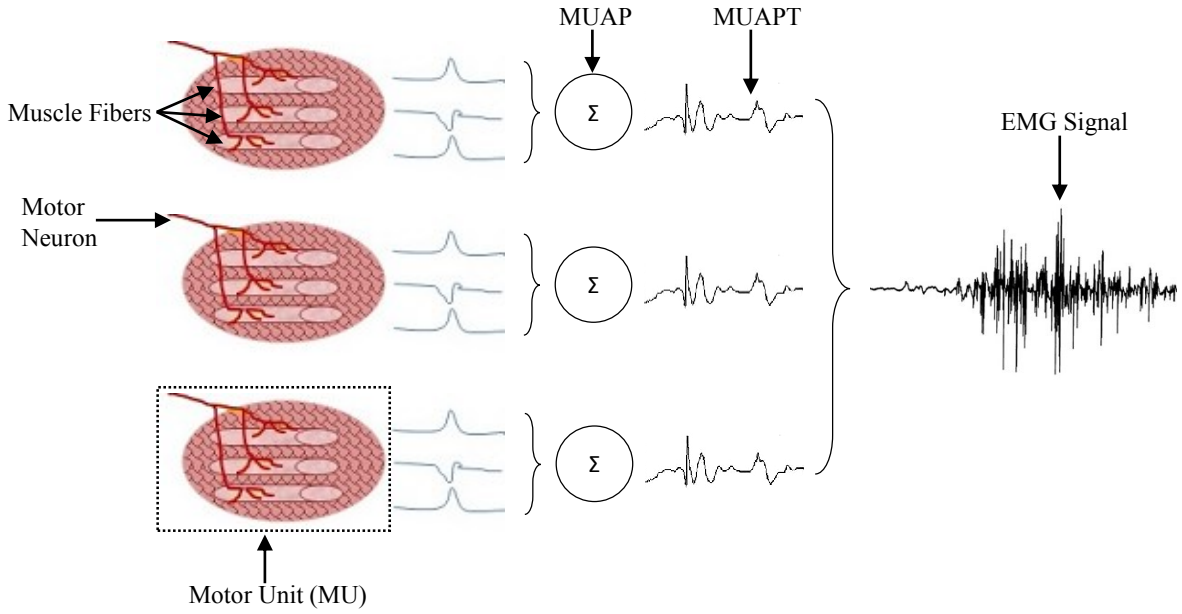


Figure 2.12: EMG signal. Muscle fibers are in groups called Motor Units (MU) where the activation of MU creates a Motor Unit Action Potential (MUAP). Continuous firing of muscle fibers creates a Motor Unit Action Potential Train (MUAPT). EMG signal is the summation of these MUAPTs.

2.3.2 Myoelectric Control Systems

Myoelectric control systems are based on EMG signals and first developed in late 1950s and early 1960s [23]. Surface EMG is mostly used in myoelectric control systems due to the fact that it can be easily extracted non invasively from the surface of skin above muscles.

There are two types of myoelectric control systems: pattern recognition based and non-pattern recognition based. Fig. 2.13. shows a pattern recognition based myoelectric control system in which, the input signals are converted into output commands using features. Features are extracted from a signal of small duration which is called a segment. Moreover, features can be categorized into the domain that they are being extracted from as time domain features, frequency domain features, and time-frequency domain features [27]. Time domain features are the most commonly used features in controlling prostheses. Some of the commonly used features according to [28–30] are: root mean square (RMS), mean absolute

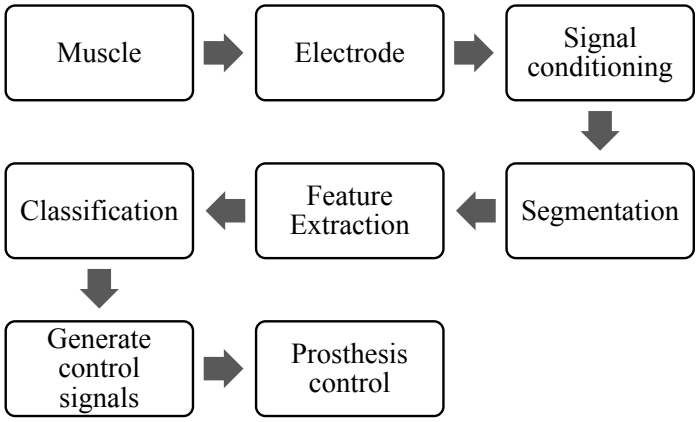


Figure 2.13: Pattern recognition based myoelectric control system

value (MAV), MAV slope, zero crossings (ZC), slope sign changes (SSC), and waveform length (WL). Refer Appendix A for details about these features. Depending on these features the segments are classified into different tasks using a classifier and those tasks are used to control prostheses [27, 31].

The classification can be done in 3 stages for better accuracy (see Fig. 2.14). These stages are pre-processing, classification, and post-processing. Pre-processing is performed in most cases to get rid of the curse of dimensionality where the classification error will increase with many features in few training samples. Principle component analysis (PCA), linear discriminant analysis (LDA), and multiple discriminant analysis (MDA) are the mostly used methods in preprocessing [32, 33]. In classification, these inputs (feature vector) are classified into classes using artificial neural networks (ANN), Bayesian networks, LDA, fuzzy inference systems (FIS) or fuzzy-neuro classifier [29]. Post-processing on these classified outputs are performed to reduce misclassifications and increase classification accuracy. Post processing techniques such as majority voting, moving average and fuzzy logic have been used after classification for improved accuracy [34, 35].

Non-pattern recognition based control systems do not use classification. Some of the non-pattern recognition based control methods are proportional control, onset analysis, and threshold control [27, 36]. As an example, in proportional control, speed or torque of a prosthetic joint is determined to be proportional to

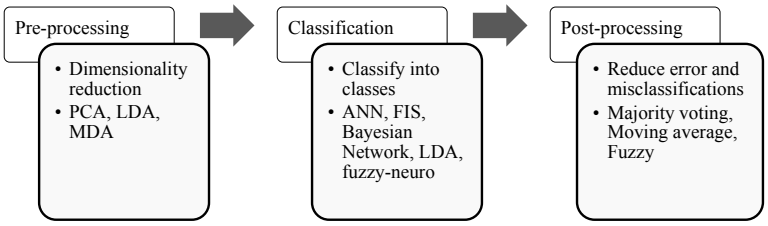


Figure 2.14: Stages of classification. Classification of myoelectric signals can be done in 3 stages for better results. These stages are pre-processing, classification, and post-processing.

the amplitude of EMG signals [37]. In threshold control, output command is generated if amplitude of EMG signal is greater than a threshold [36]. Non-pattern recognition based control algorithms are simple to implement compared to pattern recognition based control algorithms, but the number of functions that can be controlled are limited [27]. Behavior of non-pattern recognition based control systems depends on characteristics of the data acquisition system, anatomy and physiology of muscles, position of sensors on the skin and muscle fatigue [36]. It is most effective to use non-pattern recognition based methods alongside pattern recognition based control systems [27].

Both methods discussed above require the existence of muscles to extract EMG signals, however as a result of amputation there are only few muscles available to extract signals. In the case of trans-humeral amputation, all the forearm muscles and part of the biceps brachii and triceps brachii are not available. In order to control the prosthesis, the available muscle segments of biceps brachii, triceps brachii and shoulder muscles are used. In most cases the prosthesis is controlled using an agonist and antagonist muscle pair, such as biceps brachii and triceps brachii. The user of prosthesis is capable of activating these muscles with different intensities and patterns so that the prosthesis developers can take this into account when developing prostheses. The different patterns of these two muscle signals are used to achieve the required DOF [38]. However, this method of controlling prosthesis requires higher amount of training to adapt to the system and the user need to perform several contractions to get a single task done [38].

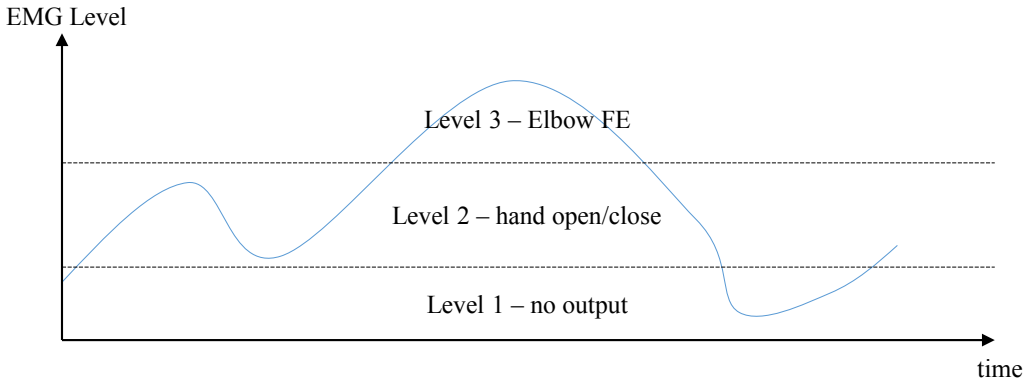


Figure 2.15: EMG contraction levels used in [39].

A non-pattern recognition based myoelectric control system has been used to control a prosthesis in [39] for a 9 year old girl with a trans-humeral amputation. Biceps brachii and Triceps brachii of the amputee has been used to control two motions of the prosthesis: elbow FE and hand open/close. Two methods of controlling were discussed: three-level method and contraction rate detection. In three-level method, contraction level is used in controlling 2 motions where slight contractions of muscle activate open/close of the hand and higher contractions activate elbow FE (refer Fig. 2.15). In the second method, higher contraction for a period of time (T) which goes into level 3 activates elbow FE until EMG level drops into level 1. In the same way if the contraction for the time period T ended up in level 2, hand open/close is activated. [39]. In [40], elbow FE and forearm SP of healthy humans are classified using a pattern recognition based myoelectric control system. Biceps brachii, medial head of triceps brachii, posterior deltoid, anterior deltoid, middle deltoid, and clavicular pectoralis major were the muscles used to extract surface EMG. Intramuscular electrodes were used to extract signals from brachialis (see Fig. 2.16). Four EMG features are used for the classification: MAV, WL, ZC, and SSC. Time delayed artificial neural networks (TDANN) have been used for the classification. A motion capture system has been used to extract motions of the human UL. The algorithm is capable of predicting elbow FE and forearm SP motions simultaneous. This method resulted in an average root mean square error (RMSE) of 11.6° for single joint motions and 13.4° for simultaneous joint motions. Antfolk *et al.* proposed a method which

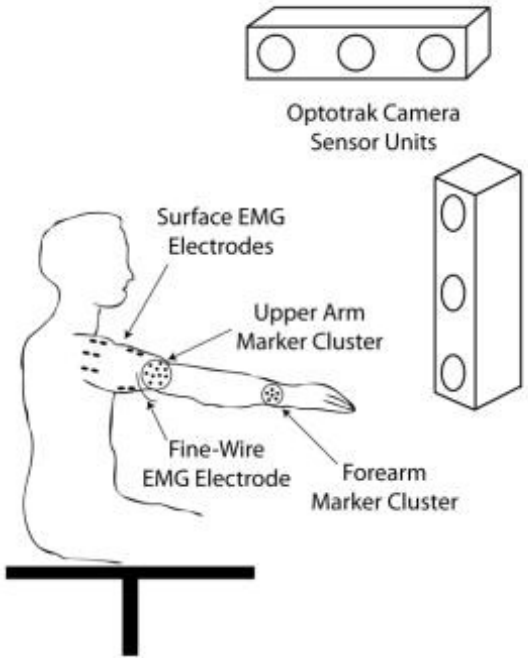


Figure 2.16: EMG electrode placement in [40]. Surface EMG signals are extracted from Biceps brachii, medial head of triceps brachii, posterior deltoid, anterior deltoid, middle deltoid, and clavicular pectoralis major. Intramuscular EMG signals are extracted from brachialis. These EMG signals are used to classify elbow FE and forearm SP

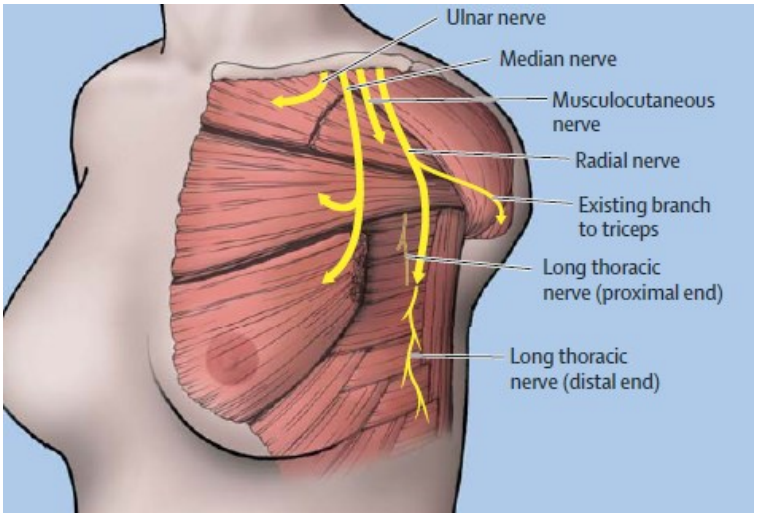
uses 16 myoelectric signals in a prosthesis designed for trans-radial amputees in which, EMG electrodes were placed on superficial extensor muscles on dorsal side and superficial flexor muscles on the volar side of the forearm [41] . The subject could learn in less than 2 hours to perform intended tasks with the prosthesis. The prosthesis is capable of performing power, lateral, and precision grips. Fingers of the prosthesis were equipped with force sensors to provide feedback to the user [41]. In [36], surface EMG signals from biceps brachii, triceps brachii, and pectoralis major were used to control a prosthesis designed for trans-humeral amputees. Pattern recognition based control strategy has been used which uses an autoregressive model and a neural network to extract features and classify. The arm is capable of achieving wrist FE, forearm SP, and hand open/close. According to [30], the speed of opening and closing of the hand can be changed based on EMG signals. Various motions of the arm and hand as well as combinations of the motions are classified in this research. The study has been conducted using

a classifier based on Bayesian theory and evaluated using amputees and non-amputees. For non-amputees, 6 pairs of electrodes were used which were placed 2 cm distal to the elbow around the circumference of the forearm in an equidistant manner. For amputees, eight pairs of electrodes were used which were placed on the biceps brachii and triceps brachii. A motion classification system was suggested in [42] for trans-radial amputees. It is capable of identifying 9 motions: wrist flexion, wrist extension, ulnar deviation, radial deviation, forearm supination, forearm pronation, opening of fingers, closing of fingers, and relaxation. Four EMG signals were used for the classification process. Furthermore, [8] has used a self-correcting pattern recognition based method using a neural networks to control an UL prosthesis. The self-correcting post-processing algorithm detects potential erroneous classifications which in turn increases the classification accuracy by 30 percent. [24] researched about the effects on a prosthetic hand when changing its impedance. Both stiffness and damping could be changed in a virtual environment. Adapting for different conditions of the hand was seen to be increasing after incorporating impedance control. Farina *et al.* reviewed simultaneous control of multiple DOF in UL prosthetic devices in [22]. [43] uses a method based on quantum information processing for pattern recognition. Simultaneous movements of multiple DOF were simulated in a virtual environment. Furthermore, authors discussed about the possibility of adapting the system or classifying surface EMG. Details of the work discussed are summarized in Table 2.3.

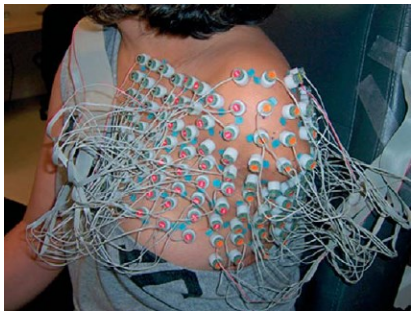
Apart from above discussed methods there is a surgical procedure known as targeted muscle re-innervation (TMR) for persons with low control input sites (muscles) [44–46]. In TMR the nerve endings of the amputated limb section is transferred onto a muscle where the activation level is low. For UL amputations, muscles at the chest are used in most cases (refer Fig. 2.17(a)) [44, 45]. The re-innervated muscles are used to extract EMG signals for the prosthesis control (refer Fig. 2.17(b)). Further, the persons who undergo TMR can get the sensations relevant to the missing limb from the surface of the re-innervated

Table 2.3: Comparison of pattern recognition based prosthetic control systems

Classifier	Prosthetic type	Ref.	Accuracy	Motions
ANN	trans-humeral	[40]	RMS angle errors 15.7 and 24.9 for elbow FE and SP respectively	Elbow FE and SP
Local approximation and lazy learning	trans-radial	[41]	86%	16 DOF including individual finger movements
AR model and ANN	trans-humeral	[36]	91-98 %	Wrist FE, SP, and hand open/close
Conditional parallel approach based on Bayesian theory	trans-humeral	[30]	96.4% for discrete motions and 89.1 % for combinations	Hand open/close, wrist FE, SP, and combinations of two of above motions.
SVM	trans-humeral	[27]	95.5 +/- 3.8 %	Five limb motions
LDA	trans-humeral	[27]	94.5 +/- 4.9 %	Five limb motions
LDA and MLP	trans-radial	[42]	97.4%	4 DOF of the hand



(a)



(b)

(c)

Figure 2.17: Targeted Muscle Reinnervation (TMR). (a) Rewired Nerves, (b) EMG Extraction, (c) Controlling a prosthesis with TMR [44]

muscles [47]. TMR has some risks associated due to the invasive surgery that the amputee needs to undergo. They are permanent paralysis of the target muscles, phantom limb pain, development of painful neuromas, and standard risks of elective surgery. As a result amputees are reluctant to undergo TMR surgery.

2.3.3 Hybrid Myoelectric Control Systems

Due to unavailability of required number of muscle sites to extract EMG signals and invasiveness of the TMR surgery hybrid control systems are developed. A hybrid control system is where two or more control inputs are used to control the prosthesis [48]. For example a hybrid myoelectric prosthesis can have EMG

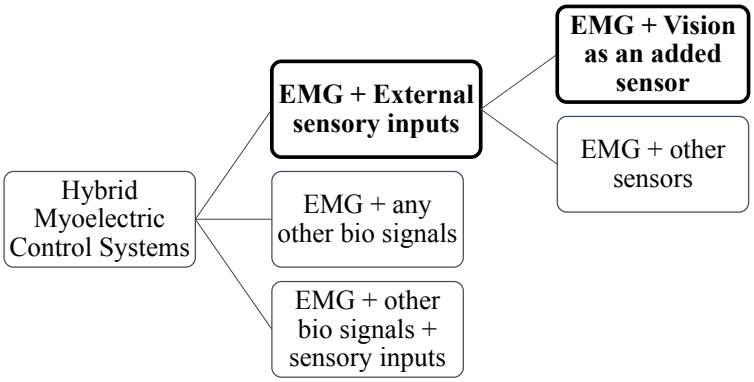


Figure 2.18: Categorization of hybrid myoelectric control systems

signal inputs and some other inputs such as foot pressure sensors, and switches. which are used collaboratively to achieve the desired motions of the prosthesis. Furthermore, some research activities are being carried out to fuse EMG signals with EEG signals to design prosthetic controllers [49]. Hybrid myoelectric control systems can be categorized into 3 (see Fig. 2.18). In the first category the EMG signals are fused with other sensory inputs such as foot pressure sensors, IMU, and switches to reinstate the lost functionality of the human limbs. Vision aided myoelectric control systems are the latest addition to this category and discussed in a separate section. In the second category the myoelectric signals can be fused with any other biological signals such as EEG or ECoG. This field is still under development and literature related to this category is limited. Under the third category the EMG signals can be fused with bio signals and external sensory inputs to build robust controllers and research related to this category needs more improvements.

EMG + External Sensory Inputs

Boston elbow and Utah arm are commercially available prosthetic devices which are equipped with hybrid myoelectric control systems falling under this category. These devices use mechanical switches to switch between motions [28]. In these devices, the user needs to use the other hand or any other means to press

Figure 2.19: ACHILLE interface used to control prostheses. The sensitive areas are marked in the figure [50]. This device contains 4 sensitive areas and can be worn inside a shoe.

the switches and activate different motions. ACHILLE [50] is a foot pressure sensory wearable interface for prosthesis control which is shown in Fig. 2.19. This device is equipped with 4 sensitive areas and can be worn inside a shoe. This is an instrumented insole and it can wirelessly transmit the signals to a prosthetic controller. Signals from these kind of devices combined with EMG signals can be used to control a prosthesis. ACHILLE can be used even by an amputee who has lost both hands to control a prosthetic device. Moreover, motion sensors such as accelerometers or IMU can be used as an added devices to control a prosthesis. The signals from the IMUs are combined with the EMG signals to achieve the prosthesis motions in [51]. Eight EMG sensory inputs from muscles around the forearm and two 3-axis accelerometer inputs fitted near brachioradialis and biceps brachii has been used in a pattern recognition based control algorithm to distinguish hand motions.

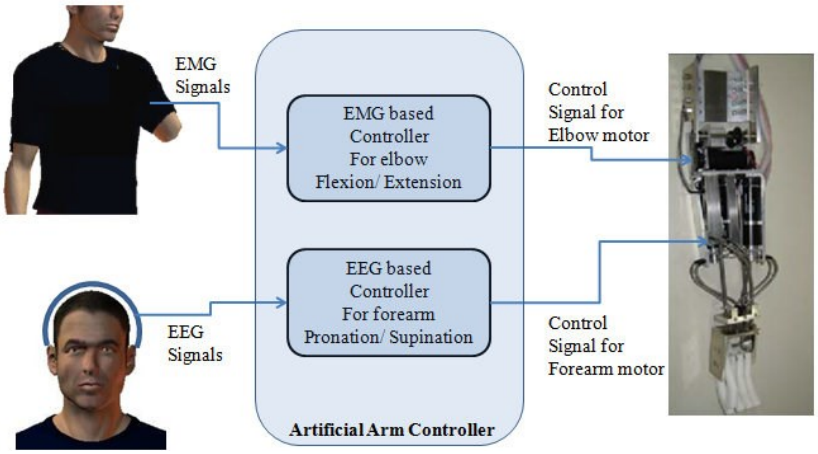


Figure 2.20: Use of both EMG and EEG to control a trans-humeral prosthesis [49]

EMG + Any Other Bio Signals

Possibility of using EEG and EMG based hybrid control approaches have being discussed in [49] for bio robotics applications which includes the control of prosthesis. They have drawn the conclusion that EEG and EMG combined control systems can perform well compared to the control systems which are only based on EEG or EMG alone. The combined system can be formed in two ways as simultaneous and sequential in which the simultaneous control strategy outperforms. Furthermore, appropriate fusion of EMG and EEG signals can improve the accuracy of the control system. An EMG and EEG combined prosthesis controller is shown in Fig. 2.20.

EMG + Vision as An Added Sensor

Vision sensors can be used as an external input in order to control prostheses. Need for vision sensors arise due to lack of capability in identifying the objects that the prosthesis needs to grasp. Moreover, integrated vision systems can estimate the distance to the object, shape of the object and size of the object. Natural limb is controlled through the nervous system which is connected with

the human vision system including eyes. External vision systems can fill the gap created by the loss of natural limb. Loss of natural limb breaks the link between the humans natural vision system with the limb. Vision systems such as stereo cameras, charge coupled devices (CCD), and complementary metal oxide semiconductor (CMOS) sensors can be integrated with prosthetic control systems.

One of the main functions required by a UL prosthesis is reaching towards an object and grasping it. When considering vision-based reaching and grasping, the first step is to identify and locate the target object. Secondly, the prosthesis should be moved to an appropriate position to perform the grasp. Subsequently, identification of how to grasp the object is necessary. Finally, the grasp is being executed. According to [52], grasping can be classified into two: blind grasps and visually guided grasps. In the former, the location of the object is determined. Then the prosthesis is moved without any visual feedback. In the latter method, a visual feedback loop is established. This method is named as visual servoing. Visual servoing is more accurate than the blind grasps [52].

Number of cameras used is an important factor in designing a vision-based control system. Use of multiple cameras improve accuracy, and depth sensing can be achieved by using the stereoscopic vision of the cameras [53] however, calibration is required. On the other hand, using a single camera reduces the need of calibration drastically [52]. However, an additional depth sensing method must be deployed such as laser or US depth sensor [53]. In [54], two cameras have been used with a laser pointer for object recognition. It proposes a hat that consists of two web cameras and a laser pointer which should be worn by the user. The user points to the object with the laser pointer. The vision system calculates the distance to the object from cameras by comparing the two images taken by two cameras. Then it detects the object after a colour segmentation process and comparing the segmented image with a set of images stored in a database. The algorithm is depicted in Fig. 2.21. It also uses tactile sensors which work alongside vision system to identify and recognize objects.

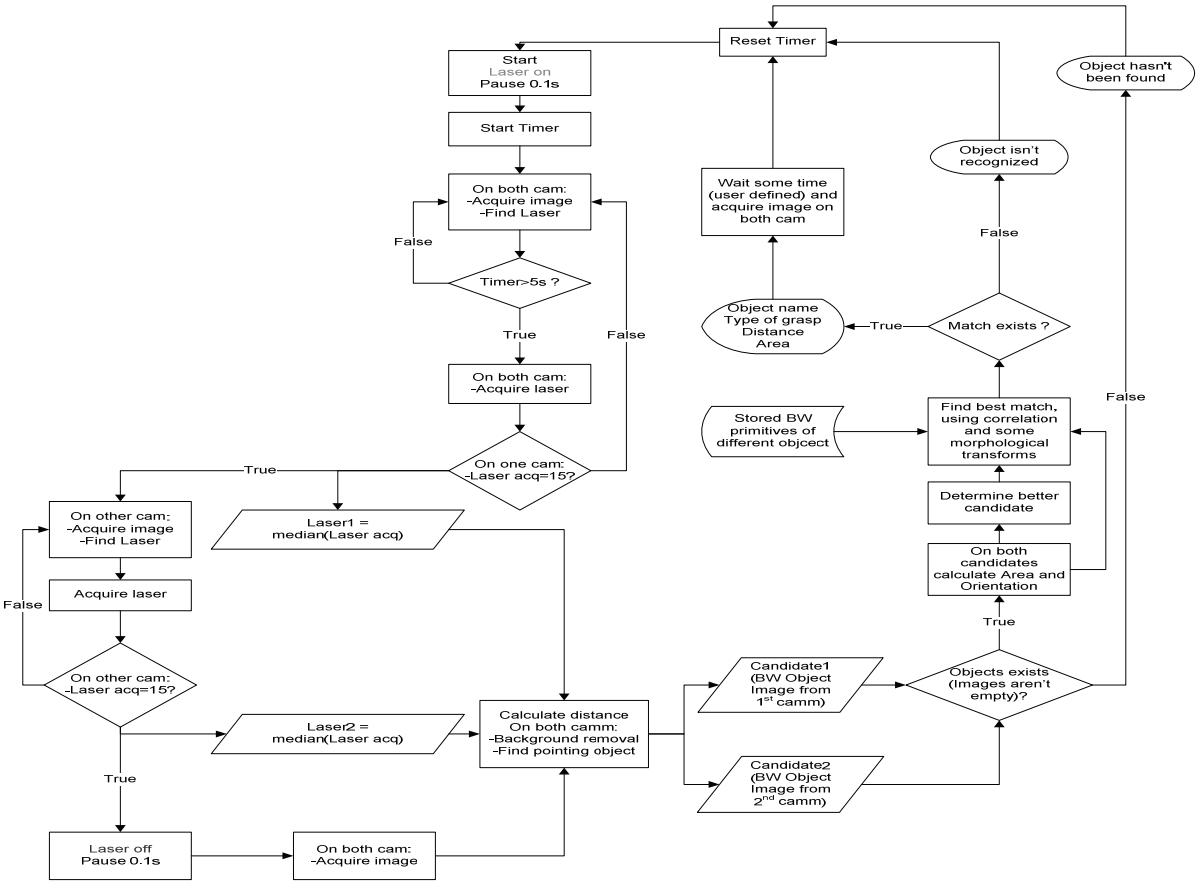


Figure 2.21: Object determination and grasp type identification algorithm used in [54]

Cameras can be placed statically on a structure which does not move or they can be mounted on a moving part such as the manipulator itself (eye-in-hand) [52, 53]. Similarly, combination of these two methods can be used [53]. If cameras are mounted statically, the manipulator might occlude the targeted object. Furthermore, camera to manipulator coordinate transformation has to be calibrated [52]. Mounting the camera in the manipulator eliminates these drawbacks, but introduces the problem of changing lighting conditions since the amount of light received by the camera changes due to its position being changed [52]. [53] employs LED illumination mounted on the hand to overcome this problem.

Few vision based prosthesis controllers are reported in literature which are listed in Table 2.4. Majority of research carried out by integrating vision sensors

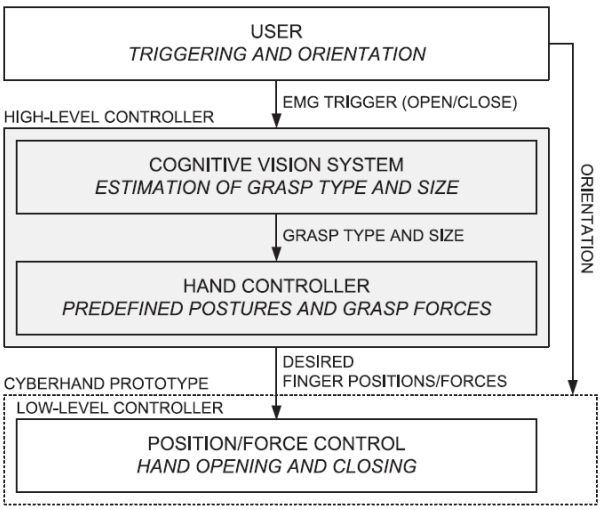


Figure 2.22: Control system architecture of [56]

to control upper limb prostheses are conducted to control trans-radial prostheses [53–57]. In these, wrist and hand motions are performed with the aid of vision [53, 54, 56] and in some of them EMG is used only as a trigger to open and close the hand [55, 56]. Only the orientation of hand is corrected in these prostheses to reach towards an object of interest [53, 57]. This has been achieved by directly calculating the required rotation angle from image data in [57], and using an accelerometer in [53].

Dosen *et al.* used a web camera, US distance sensor and a laser pointer to control a prosthetic hand along with an EMG interface [56]. The user should move the arm so that the laser pointer is pointed to the target object. It can be reliably identified by the camera. The prosthetic controller selects the grasp type and the size based on target object. The control system architecture is shown in Fig. 2.22 and the implementation of the control system is shown in Fig. 2.23. The control system of [56] is named as Cognitive Vision System (CVS) which decides the grasping pattern. It is a rule-based approach. After grasp types are decided based on the size of the object, CVS generates control commands for the hand control module, which in turn determines the appropriate finger positions and forces required. The user issues commands with the EMG interface for opening and closing the hand.

Table 2.4: Vision based prosthetic controllers

Ref.	Task	Level	DOF	Sensors	Cons	Accuracy
[57]	Estimate grasping pattern (4 types), object size, and orientation of the object to control prosthetic hand accordingly.	TR	SP and hand DOF.	Web camera, US distance sensor	Lack of sensory feedback.	Size estimation accuracy is higher than 36%.
[55]	Grasp identification (2 types) and control of a bebionic V2 hand prosthesis.	TR	Hand DOF to adapt two grasping patterns.	Two mechanomyogram (MMG) sensors, web camera.	Lack of grasping types.	Average success rate 84.4%.
[53]	Estimate grasping pattern (3 types), object size, and object orientation to control a hand prosthesis with wrist rotator.	TR	Wrist rotation (SP) and hand DOF.	Web camera, US distance sensor, Accelerometer, laser pointer, LED illumination.	Lightning is unnatural for a prosthesis.	Success rate, With lighting 90%, Without lighting 70%.
[56]	Select grasp type (4 types) and size (3 sizes). Triggers using an EMG signal.	TR	Hand DOF	2 EMG sensors, web camera, US distance sensor, laser pointer.	No closed loop (Look and move) vision control.	Correct type and size 84%. Wrong size, correct type 3%. Correct type, large size 3%.
[54]	Stationary fixed stereo cameras are used to estimate the grasping pattern (4 types) for Electrotherapy.	N/A	N/A	Two CCD cameras, laser pointer,	Stationary cameras are not suitable for prosthesis applications.	Higher than 90%.

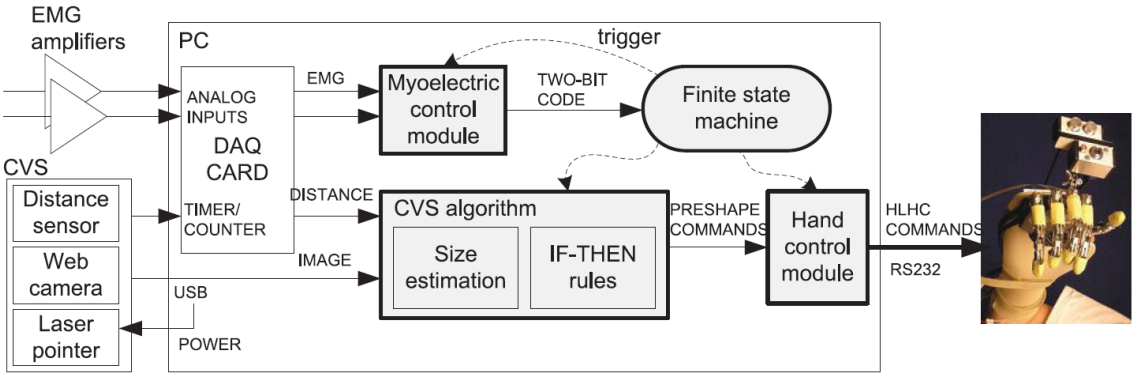


Figure 2.23: Implementation of control system in [56]

In [53], a camera, an US sensor and an accelerometer has been used to control a trans-radial prosthesis. The camera and the accelerometer is turned on only when the distance measured by the US depth sensor reaches a critical value (25 cm). The prosthetic controller estimates the shape and size of the object in real time. Furthermore, it estimates the hand orientation relative to the object. Based on this data, control signals are generated to control wrist rotator of the prosthesis and actuators of the hand. Prosthetic hand can form three grasping patterns: palmar, lateral, and precision. Signal processing algorithm used in the trans-radial prosthesis is depicted in Fig. 2.24.

According to [53] and [54], using the laser pointer will not benefit when the target object is of the same colour as that of laser. This will also happen when the target object is very bright, reflective or transparent. [54], also suggests that inconsistent backgrounds and shadows can disrupt proper object recognition. If the visual data to be part of the feedback loop of the prosthetic control system, it should track the target object as the limb moves. In order to track the object, [52] suggests to use features corresponding to points on the perimeter of the object. [53] extracts image contour to estimate its shape. [53] and [56] have estimated the size of the target object taking distance to the object (obtained with the distance sensor), length of the object's short & long axes and focal length of the camera as inputs. Types of grips are decided based on the size of the object [53, 56].

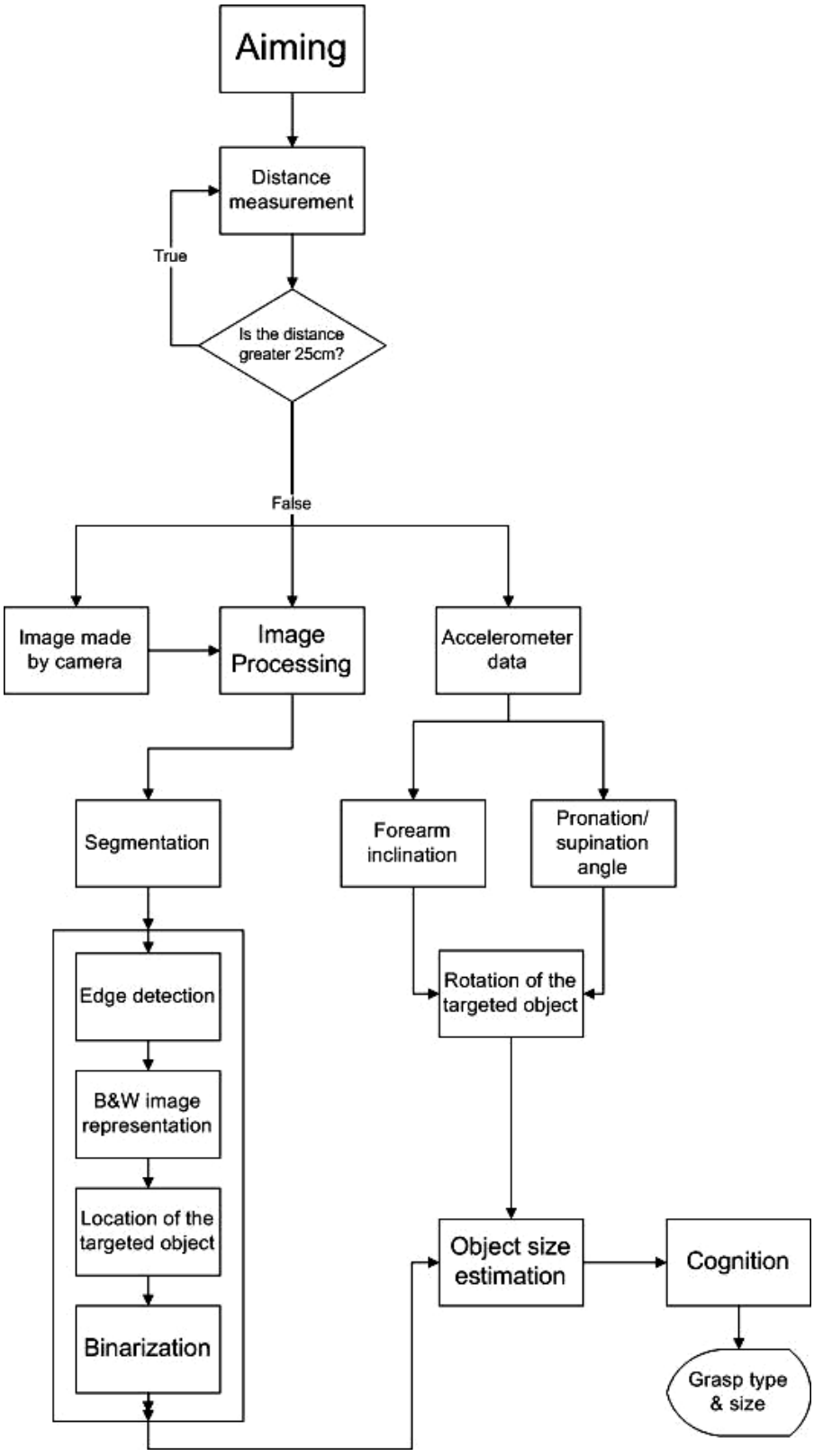


Figure 2.24: Signal processing algorithm used in [53]

Vision aided myoelectric control systems are considered as important since it has the ability to see things that needs to be grasped. Vision can be used in prosthesis for assisting grasp and taking feedback regarding the positions of prosthetic limb segments. Grasping assist was possible since the vision sensors can be equipped in the hand to see the object and path can be planned depending on the vision feedback.

2.4 Reach-to-grasp Path Planning

Natural ULs are capable of providing necessary articulations to reach towards an object when the human needs to grasp that object, which is known as reach-to-grasp. Typically, a healthy human hand takes a straight line path with a bell shaped velocity profile for reach-to-grasp motions [58, 59]. Moreover, the human hand follows a similar path when pointing at an object. In order to obtain human-like motions in prostheses, the prosthetic hand should be able to follow a human-like path.

Majority of trans-humeral prosthetic controllers tend to control the prosthesis at joint level rather than at task level. However, a few task level prosthetic controllers can also be found in literature. In [60], several tasks are associated with stump arm kinematics, where the tasks and prosthetic movements are predefined. However, upper limb motions required for the same task may vary from person to person. Furthermore, a person can perform the same task with different upper limb motions. For example, reaching towards the mobile phone on a table with the intention of grasping. In this, the phone placement and the current position of the person can change from time to time, which requires different movements of the UL to perform the task.

Furthermore, if a person wants to grasp an object, the human hand should be moved closer to the object from the current position. The human hand normally takes a straight line path from the current position to the destination position where the object is placed. In order to have this functionality in trans-humeral

prostheses, the hand of the prosthesis should be capable of taking on a reference path which is straight line if the points on the path are kinematically feasible. The reference path is created from the current position of the hand to the destination position (object position).

The methods that are found in literature for path planning has been developed for manipulators where the kinematics from base to the end effector are known [61–63]. Moreover, each and every DOF of manipulators is controllable. When considering trans-humeral prostheses, it can be treated as a manipulator fitted on a base. Stump arm is acting as the base where the base movements are not known-a-priori. Furthermore, the prosthetic controller is not capable of controlling the stump arm.

2.5 Summary

Loss of part of the UL makes people unable to perform most of the ADL. In order to regain the lost functionality externally powered UL prostheses are developed. These prostheses are controlled using biological signals such as EMG, EEG, and ECoG. EMG is preferred due to high signal to noise ration and ease of extracting.

Two methods of EMG based control systems can be found: pattern recognition based and non-pattern recognition based. Pattern recognition based algorithms for different limb motion classifications have led to promising results in prosthetic control. Among them time domain features are the mostly used features due to the fact that they are able to perform well compared to the complexity and the computing power that is being demanded. According to the literature the EMG only controllers are not able to classify the motions in full accuracy. As the level of amputation increases the classification becomes difficult because the EMG signals also vanish. This problem was addressed using TMR and were able to control prosthetic limbs with a reasonable level of accuracy. However, TMR includes an invasive surgical procedure and it has some risks associated with it.

In order to overcome the drawbacks of EMG only control systems and TMR, hybrid myoelectric control systems were developed. In hybrid myoelectric control systems, different sensory inputs are used in combination with EMG signals. Among them foot pressure sensors and mechanical switches are the easy to implement reliable methods. However, these types of control systems have drawback of complexity and the user of the prosthesis needs to press switches or foot against floor to perform the required task which is quite unnatural. Fusion of sensory inputs has a long way to go in order to reach the capabilities of natural limb.

Vision aided myoelectric control systems are considered as important since it has the ability to see things that needs to be grasped. Vision can be used in prostheses for assisting grasp and taking feedback regarding the positions of the prosthetic limb segments. Grasping assistance was possible since the vision sensors can be fixed in the hand to see the object and path can be planned depending on the vision feedback. Currently developed vision aided prosthetic control systems are for trans-radial prostheses.

Reach-to-grasp motions are performed to reach towards an object with the intention of grasping that object. Natural UL takes a straight line path towards the object with a bell shaped velocity profile if there are no any obstacles in the path. Giving path planning capability to perform reach-to-grasp motions is required to progress task level prosthetic controller development for trans-humeral prostheses.

Path planning of a trans-humeral prosthesis to perform reach-to-grasp motions requires hand to be moved closer to the object and to correct the orientation of hand according to the user intention. In order to achieve this vision signals and EMG signals should be integrated. Moreover, new algorithms should be developed in order to follow the path towards the object of interest compensation shoulder motions as shoulder motions are solely performed by the wearer.

Chapter 3

THE TRANS-HUMERAL PROSTHESIS USED TO EVALUATE THE PATH PLANING METHOD

This chapter explains the simulation environment and the prosthesis that has been used for experimental evaluation of path planning algorithms. 3 DOF at the shoulder joint prevails after a trans-humeral amputation. The latter 4 DOF of elbow and wrist along with DOF at the hand are lost and replaced by the prosthesis.

3.1 Design Criteria for the Trans-humeral Prosthesis

It is evident that existing trans-humeral prostheses are hardly able to achieve required DOF [64, 65]. Therefore, a prosthesis with simultaneously working major UL motions while having the relevant anthropometric aspects is required to achieve human-like motions. However, almost all the available trans-humeral prostheses do not include two wrist motions at the same time. Furthermore, the natural UL has a axis shift between two wrist motions which can be hardly seen in UL prostheses. If one of the motions are absent the shoulder or the remaining motions are required to perform the missing motion.

When designing a prosthesis, weight is a crucial factor. Overweighted prostheses limit the ROM and cause musculo-skeletal disorders. Thus, the weight of a prosthesis should be similar or lesser compared to the human UL. The UL

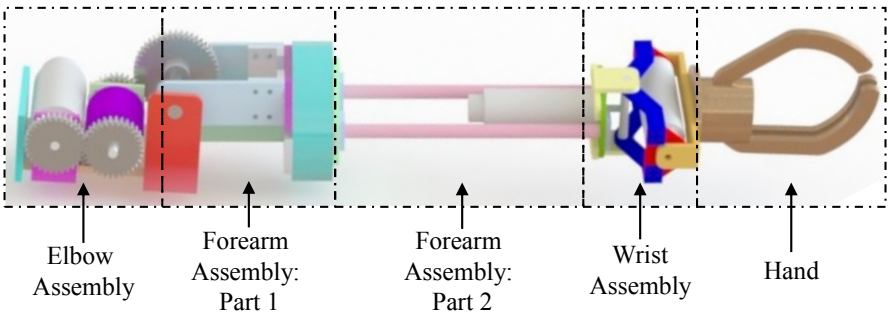


Figure 3.1: Major components of The prosthetic arm

of a human with a weight of 75kg weighs around 3kg [66]. Moreover, length of the forearm should be according to the anthropometry. Length of the forearm of a grown man is 27.1 cm according to [18] considering the 50^{th} percentile data. Though the fact remains as such, due to the components added to achieve the required motions, it is difficult to keep up according to the anthropomorphic data. Hence, keeping the co-relation between the functions of prosthesis and the anthropometry has become a challenging task.

3.2 Design of the Trans-humeral Prosthesis

As a solution, a 5 DOF prosthesis has been designed. It has a novel 2 DOF wrist which has a axes shift between two wrist motions. Moreover, it can achieve full ROM similar to a human UL. Furthermore, prosthesis is fabricated considering the weight. Major components of the design are shown in Fig. 3.1.

3.2.1 Elbow

The design of the elbow is shown in Fig. 3.2. A brushless DC (BLDC) motor (EC 4 Pole, Maxon motors) is used to initiate the rotary motion of the elbow. BLDC is fixed to the elbow housing. Elbow housing can be connected to the stump arm of the amputee. The BLDC motor is connected to a harmonic drive gear box (100:1) through a spur gear pair (ratio 1:1). The output of the harmonic

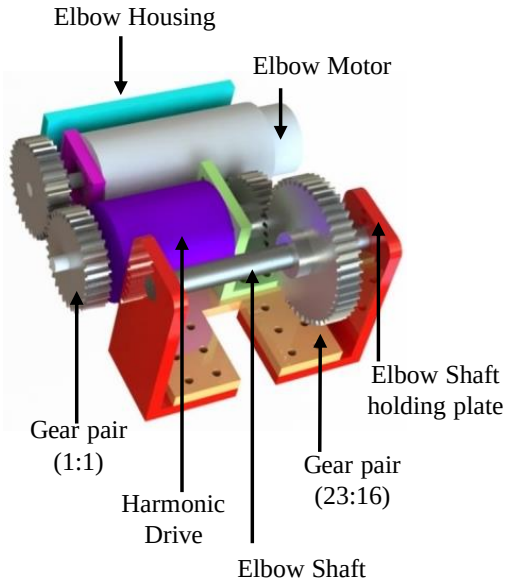


Figure 3.2: 3D Model of elbow assembly

drive gear box is coupled with the elbow shaft through a spur gear pair with a gear ratio of 23:16. The elbow shaft holding plate is rigidly attached to the elbow housing. The elbow shaft is supported by elbow shaft holding plates with the aid of two bearings. Elbow shaft can rotate relative to the Elbow shaft holding plates. The forearm connecting plates [see Fig. 3.3(a)] rigidly connects with elbow shaft so that the part 1 of forearm assembly can rotate relative to the elbow assembly. Thus the elbow FE motion is achieved.

3.2.2 Forearm

The forearm design of the prosthesis is shown in Fig. 3.3. Forearm motor (DCX22S, Maxon motors) is used to generate the forearm SP motion. The forearm motor is fixed to the forearm wheel and the forearm wheel is fixed to forearm connecting plates. Output shaft of the DC motor is connected to the internal gear using a spur gear attached to the motor shaft. The internal gear is rigidly fixed to the forearm base wheel to enable the forearm SP motion. Forearm base wheel is placed inside the forearm wheel. In order to support the relative motion between two wheels, a needle roller bearing is used. Needle roller bearing is press fitted

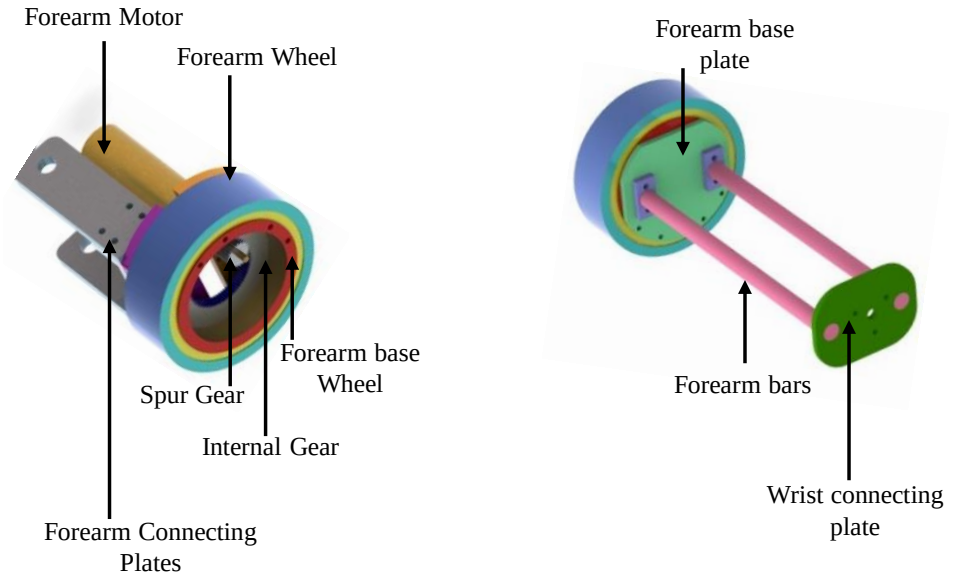


Figure 3.3: 3D Model of forearm assembly, (a) part 1, (b) part 2

to the forearm wheel and base wheel is press fitted to the needle roller bearing. Forearm base plate is connected to the forearm base wheel. The forearm base plate can rotate with the forearm base wheel and achieve the relative motion between two forearm parts [in Fig. 3.3, (a) and (b)]. Forearm bars have been used to connect the forearm base plate to the wrist connecting plate.

3.2.3 Wrist

UR motor (DCX22S, Maxon Motors) of the wrist (see Fig. 3.4(a)) is connected through the wrist connecting plate. A bevel gear pair rigidly attached to the output shaft of UR motor and URD shaft enables the URD motion. Bevel gear attached to the output shaft of the UR motor transmits rotary motion generated by the UR motor to the URD shaft through the bevel gear attached to the URD shaft. This mechanism gives the flexibility to the design adjusting the distance between two perpendicular axes of wrist motions. Wrist of the prosthesis has a 20 mm shift between these two axes [see Fig. 3.4(b)]. L brackets are rigidly attached to the wrist connecting plate. URD shaft is supported by L brackets with two

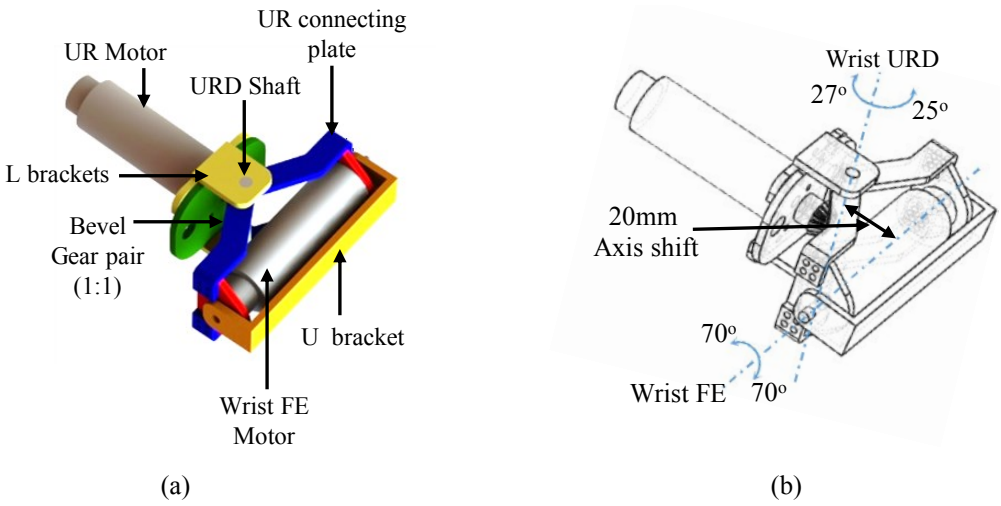


Figure 3.4: Wrist of the prosthesis. (a) 3D model of wrist assembly, (b) Wrist design. It can achieve 2 DOF wrist motions while maintaining a 20mm axis shift between wrist URD and wrist FE motions.

bearings. Hence, the URD shaft can rotate relative to the L brackets. Shape of the UR connecting plates are designed to achieve the full range of motion of URD. U bracket is connected to the wrist FE motor (DCX22S, Maxon Motors) to achieve the wrist FE motion. One end of the U bracket is connected to the wrist FE motor shaft. Other end of the U bracket is supported with a bearing. U bracket enables the connection to the prosthetic wrist.

3.2.4 Hand

The hand (see Fig. 3.5) comprises of a hand base which enables the connection with the U bracket of the wrist. A DC motor is located inside the hand. A screw is fixed coaxially with the DC motor shaft. A ball which moves along the screw is fit to the fingers (thumb and index finger). When the DC motor starts rotating, thumb and index fingers are pushed apart or towards each-other due to the ball and screw mechanism [67]. A force sensor is attached to the thumb to detect and limit the grasping force. A limit switch is used th detect the full open of the hand.

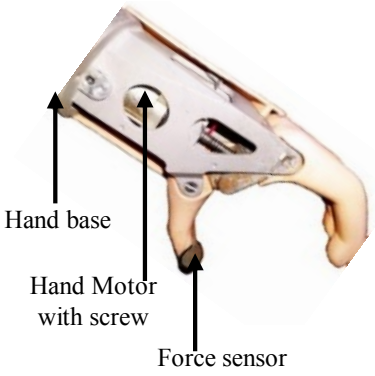


Figure 3.5: Prosthetic hand

3.3 Simulation of the Trans-humeral Prosthesis

In most circumstances, evaluation of control algorithms of prostheses are carried out in actual prostheses or prototype of prostheses. However, this limits the capability of developing a control algorithm if the required hardware is not available. Even if the hardware is available, a considerable amount of time will have to spent on setting up for the evaluation. Therefore, this simulation environment is proposed to simulate trans-humeral prosthetic control algorithms.

The simulation environment is implemented using V-REP [68]. V-REP is a robot simulator which can be used to build simulation environments. External inputs such as IMU sensory inputs, inputs from a camera, and EMG signals can be fed into V-REP. [21, 26, 69].

The simulation environment consist of a trans-humeral prosthesis and a virtual shoulder joint. The virtual shoulder joint can mimic human shoulder motions using an IMU attached to the stump arm of an amputee. IMU is a sensor consist of an accelerometer and a gyroscope. The accelerometer is capable of capturing the instantaneous translational accelerations applied on all 3 axes. The gyroscope is capable of capturing instantaneous angular velocities around 3 axes. These accelerations and angular velocities are used to calculate the rotation angles of the human shoulder joint.

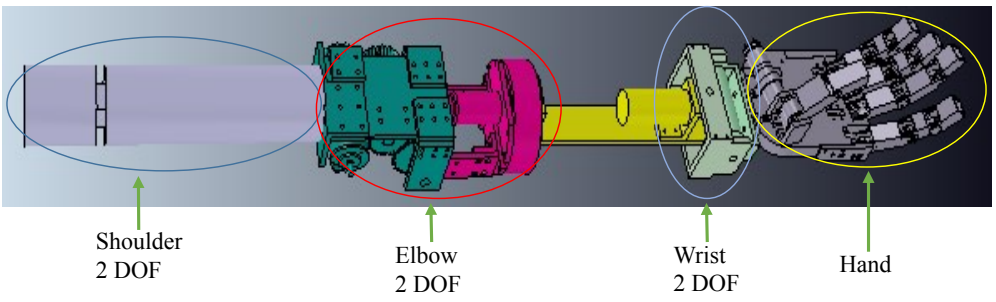


Figure 3.6: Prosthesis model used in simulation environment. The prosthesis has 5 DOF and it has been fitted to the end of a virtual shoulder.

The simulation environment is used to verify the ranges of motions and kinematics of the design before moving into the fabrication of the hardware. Moreover, this simulation environment is used to simulate trans-humeral prosthetic control algorithms without an amputee.

3.3.1 Development of the Simulation Environment

Simulation environment is developed using several components. First the prosthesis model is developed in the V-REP using the design of a prosthesis. The 3D model of the prosthesis is converted into STL files and imported into V-REP. Later, the model is built inside V-REP by adding revolute joints where necessary. Furthermore, a virtual shoulder joint is added to the prosthesis.

The virtual shoulder joint is modelled using 3 revolute joints as shown in Fig. 3.7(a), elbow FE as a revolute joint [Fig. 3.7(b)], forearm SP as a revolute joint [Fig. 3.7(c)], and wrist FE & wrist URD as two revolute joints [Fig. 3.7(d)]. The virtual shoulder is controlled according to the stump arm motions of an amputee through the IMU. IMU can be attached to the stump arm of an amputee. The developed prosthesis model is shown in Fig. 3.6.

Shoulder AAD is implemented as a revolute joint using a cylinder and a cube [see Fig. 3.7(a)] after the cube shoulder FE is implemented as a revolute joint using the cube and another cylinder. At the bottom of the second cylinder,

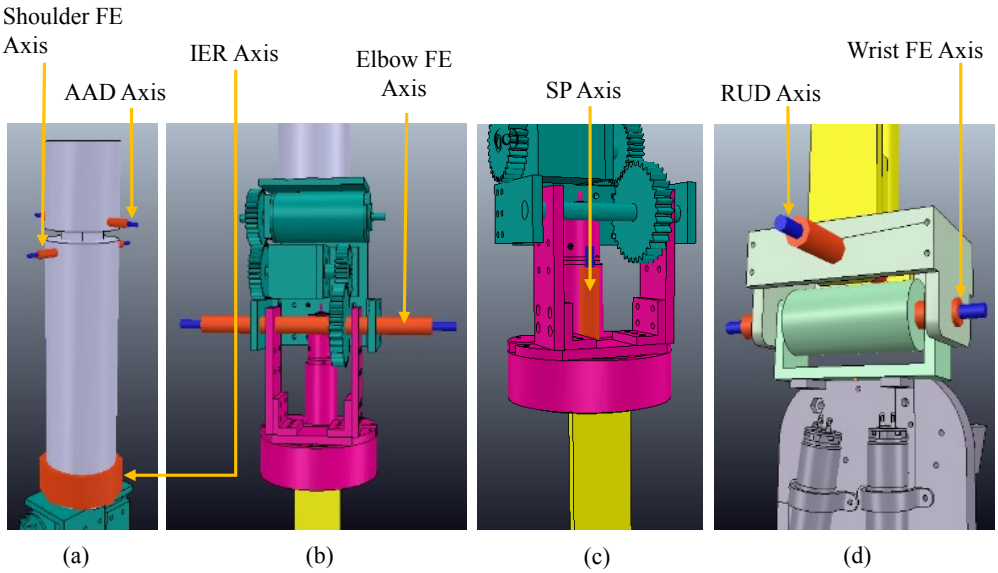


Figure 3.7: DOF at each joint of the simulator (joint axes are shown in orange colour) (a) Shoulder joint, (b) Elbow joint, (c) Forearm SP, (d) Wrist joint

shoulder IER is implemented as a revolute joint. The drive train of the elbow FE is included adjoining to the shoulder IER. A revolute joint is inserted to get the elbow FE as per Fig. 3.7(b). Furthermore, the forearm SP is implemented as a revolute joint after the elbow FE [see Fig. 3.7(c)]. The forearm is modelled as a link between elbow and wrist. At the bottom of the forearm, wrist joint is implemented. After a revolute joint for URD at the distal end of the forearm link, wrist FE is implemented as shown in Fig. 3.7(d).

After developing the simulation model in V-REP, control algorithms are implemented as embedded scripts in V-REP using LUA (a programming language used by V-REP). Since the shoulder joint is preserved in a trans-humeral amputee, the shoulder joint motions are mapped into the simulation environment through the IMU (MPU 6050). Control inputs of the shoulder are fed into the simulation environment using a micro-controller (ATmega2560, Atmel) through serial communication (RS232). The setup is shown in Fig. 3.8. The IMU can be placed on the stump arm of the amputee. The shoulder AAD and shoulder FE can be controlled through this setup. The angle data is calculated using the accelerometer and the gyroscope present in IMU. A complementary filter [refer

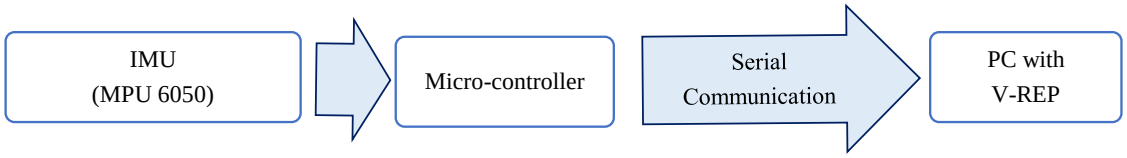


Figure 3.8: Shoulder controller. Natural human shoulder angles are extracted using the IMU and fed into the simulation environment through a micro-controller.

(3.1)] is implemented inside the micro-controller to calculate the resultant angle for both shoulder AAD and shoulder FE.

$$A_{cn} = K(A_{pn} + A_{gn} \times \delta t) + (1 - K)(A_{an}) \tag{3.1}$$

where A_{cn}, A_{pn}, A_{gn}, δt, A_{an}, and K are current angle of shoulder AAD ($n = x$) or shoulder FE ($n = y$), previous angle of shoulder AAD or shoulder FE, angular rate from gyroscope for shoulder AAD or shoulder FE, sample time of IMU, angle calculated from accelerometer for shoulder AAD or shoulder FE, and a constant between 0 and 1 respectively. K is selected as 0.98.

Angle calculated from accelerometer, $A_{an}(n = x, y)$ is given as follows.

$$A_{ax} = \tan^{-1}\left(\frac{a_x}{\sqrt{{a_y}^2 + {a_z}^2}}\right) \tag{3.2}$$

where A_{ax}, a_x , a_y , and a_z are angle calculated from accelerometer for x-axis, acceleration along x-axis, acceleration along y-axis, and acceleration along z-axis respectively. Similarly A_{ay} can be calculated.

The resultant angles calculated as per (3.1) is sent to the simulation environment (V-REP) through the micro-controller.

PID controllers are implemented inside the simulation environment to control shoulder AAD and shoulder FE joints. PID controllers are implemented in a threaded child script. The respective joints are set to motion mode. The

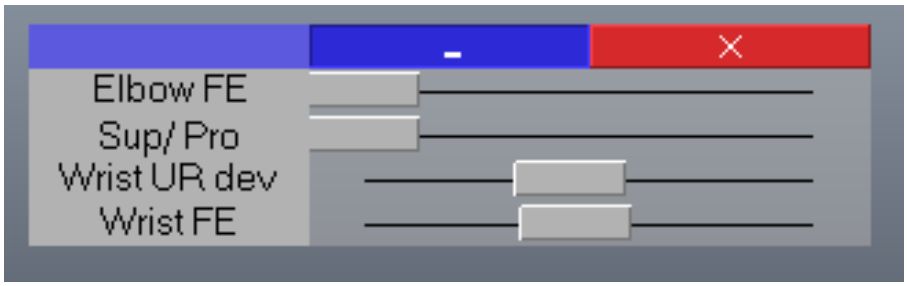


Figure 3.9: UI used to control joints from elbow to wrist

implemented PID controller calculates respective velocities for those two joints. V-REP can rotate these joints at a given set velocity. Hence, the shoulder joints of the simulation environment are rotated according to the stump arm motions extracted thorough IMU. Script limits the maximum and minimum angles that each joint can rotate.

Rest of the joints from elbow to wrist are also set to motion mode. The joints are fed with the respective joint velocity. A user interface (UI) is developed for those four joints. The UI can turn those joints up to the specified joint limits. The UI is shown in Fig. 3.9. In UI the 4 sliders are available for the 4 joints of the prosthesis. The joints are controlled using PID controllers as in (3.3) where the sliders provide the desired joint angles. The PID controllers for these joints are implemented inside a non-threaded child script.

Furthermore, calculated joint angles from other control algorithms can be used as inputs instead of the slider inputs from UI. These joint angles can be specified inside the child script. Moreover, the simulation environment can be used as a base for developing sensor based control algorithms by adding sensors and placing them on the prosthetic arm model using the V-REP's build in sensor database. EMG or any other biological signal can be fed into the simulation environment through serial communication or any other method preferred by the developer.

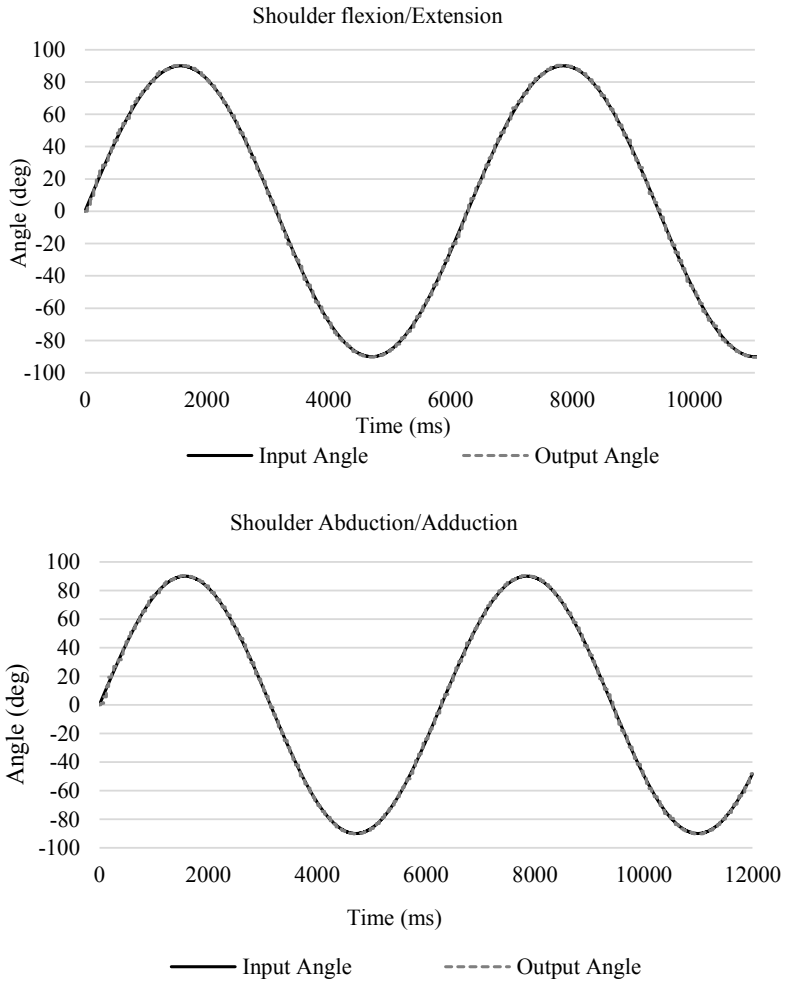


Figure 3.10: Simulation results of shoulder FE and shoulder AAD

3.3.2 Simulation Results of the Prosthesis

Several simulations were carried out on the developed simulation environment to evaluate its effectiveness. First, the modelled shoulder joint and its controller was evaluated. Input angles for the shoulder are given as a sine wave and the outputs were recorded. Results show (refer Fig. 3.10) that the implemented controller is capable of effectively following the input angles. Remaining joint controllers are evaluated for different angle combinations which are fed through the sliders in UI (see Fig. 3.9). The results are shown in Fig. 3.11. According to the results, the simulator can effectively generate required joint angles fed through the UI.

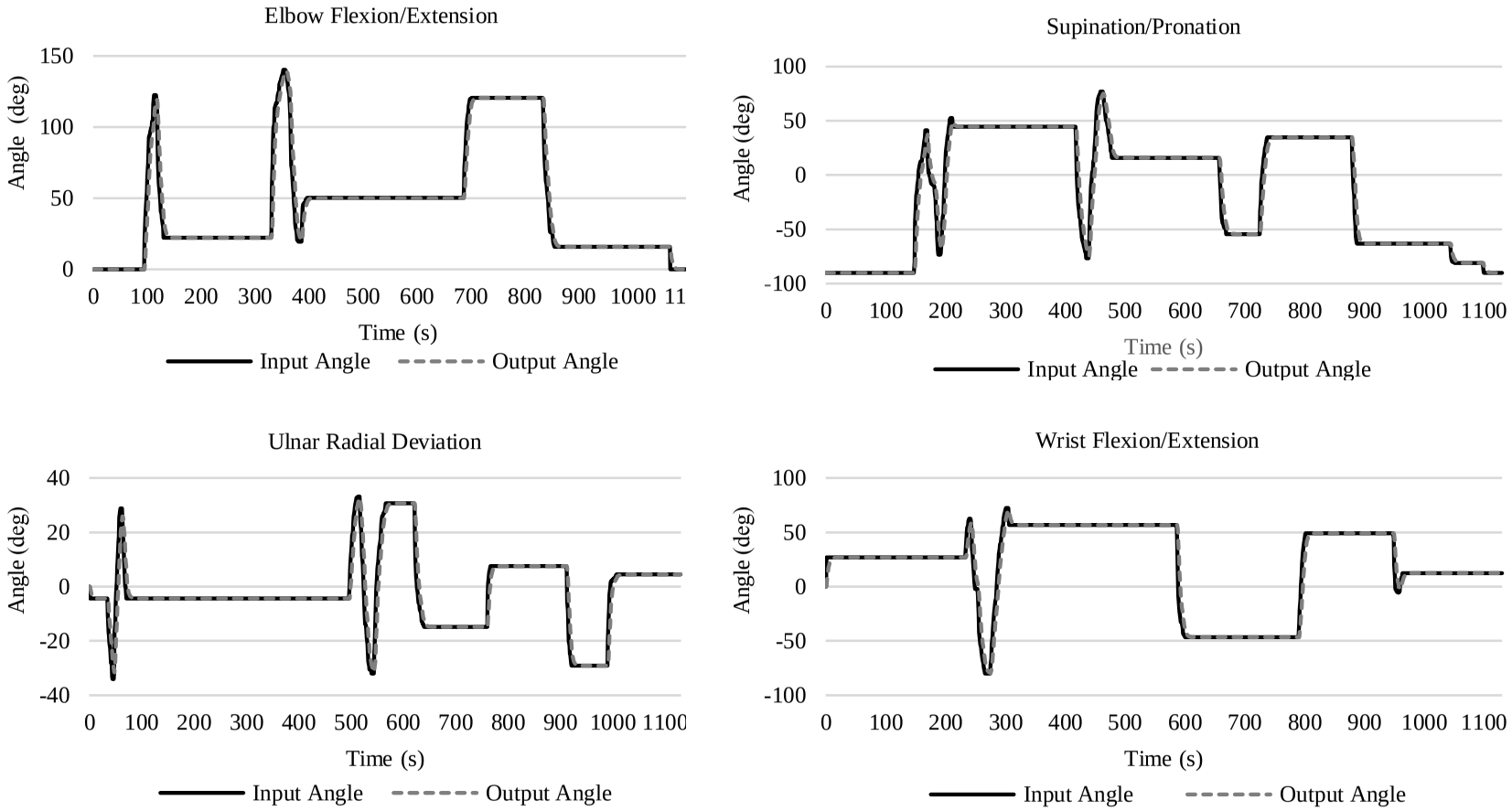


Figure 3.11: Simulation results of prosthetic joints: elbow FE, forearm SP, wrist URD, and wrist FE.

Figure 3.12: Trans-humeral prosthetic arm. The prosthetic arm has 5 DOF: elboe FE, forearm SP, wrist FE, wrist URD, and compound motion of hand. Fabricated prosthesis weighs 3.2kg.

It is evident that the simulation of prosthetic control algorithms can be done almost with the fed joint angles be reflected in the output, in contrast this is difficult to achieve in a hardware prosthesis. Moreover, simulation environment verified the effectiveness of the proposed design.

3.4 Fabrication of the 5 DOF Trans-humeral Prosthesis

The manufactured prosthesis is a right arm trans-humeral prosthesis. In order to perform human-like motions, it has 5 DOF: Elbow FE, forearm SP, wrist FE, URD and compound motion of the hand. The prosthesis is shown in Fig. 3.12. Most of the components of the prosthesis are built using Aluminum considering the high strength to weight ratio. Gear wheels of the elbow are fabricated using cast iron since elbow requires high torques and applies high dynamic tooth loads on the gear wheels. Fabricated prosthesis weighs 3.2kg. The prosthesis weight is almost similar to that of a human UL.

3.4.1 Prosthetic Controller

Connection diagram of the components (sensors, actuators, and controllers) of the prosthesis is shown in Fig. 3.13. The prosthesis consists of 5 motors: one BLDC, three DC motors with encoders, and one DC motor without an encoder. The BLDC is controlled through a BLDC controller (EPOS2, Maxon Motors).

It is used to actuate elbow FE. Three DC motors with encoders are used to actuate forearm SP, wrist FE, and wrist URD. DC motor without an encoder is used to actuate the hand. All 4 DC motors are controlled using H-bridge motor controllers (L298). BLDC controller can be directly connected to a personal computer (PC). BLDC controller communicates serially (RS232) with the PC. DC motors are controlled through micro-controllers (ATmega2560, Atmel). Three micro-controllers are used. DC motors with encoders are connected to these three micro-controllers. DC motor without an encoder is connected to one of the micro-controllers. Micro-controllers are communicating with the PC serially (RS232).

BLDC controller has its own position controllers build-in. Hence, it requires only the joint angles to be sent using the PC. DC motors with encoders are controlled using proportional-Integral-Derivative (PID) controllers implemented at micro-controllers. Joint angles are sent to the micro-controllers using the PC. PID controllers are implemented as (3.3).

$$S_j = K_p \times E_p + K_i \times E_i + K_d \times E_d \tag{3.3}$$

where, S_j, K_p, K_i, K_d, E_p, E_i, and E_d are joint command, proportional gain, integral gain, differential gain, proportional error, integral error and differential error respectively. Joint commands are given as pulse width modulation (PWM) outputs of micro-controller. These PWM signals are sent to the H-bridge motor controllers and to the motors subsequently.

$$E_p = A_D - A_A \tag{3.4}$$

where, A_D and A_A are desired angle and actual angle (current angle of the joint) respectively.

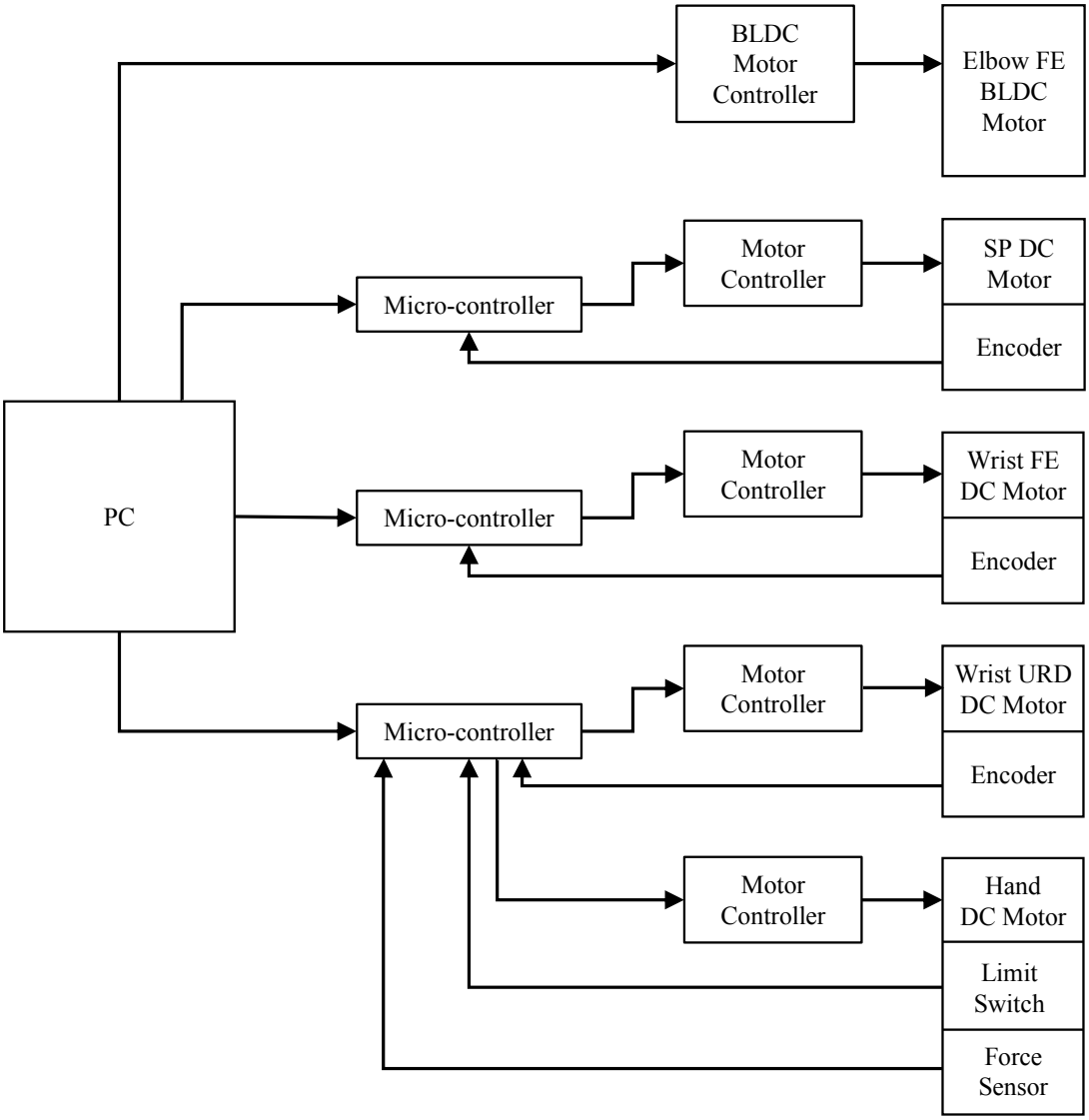


Figure 3.13: Interconnection of components of the prosthesis. Prosthesis consists of 5 actuators to achieve 5 DOF: BLDC and 4 DC motors. 3 DC motors are equipped with encoders. 2 sensors are equipped inside the hand: limit switch and a force sensor. 3 micro-controllers and a BLDC controller has been used to provide control commands to the actuators.

$$E_i = E_i(t-1) + E_p \tag{3.5}$$

where, $E_i(t-1)$ is previous integral error.

Algorithm 1 Hand Operation Algorithm

```
1: OC ← Place holder for hand open or close state
2: Limit ← Place holder for limit switch state
3: force ← variable for force sensor value
4: Thresh ← force threshold
5: if OC = Open then
6:     if limit = 1 then
7:         stop
8:     else
9:         hand open
10: else
11:    if force > Thresh then
12:        stop
13:    else
14:        hand close
```

$$E_d = E_p(t) + E_p(t-1) \tag{3.6}$$

where, $E_p(t)$ and $E_p(t-1)$ are current proportional error and previous proportional error respectively.

Hand is operated using the hand DC motor, a limit switch, and a force sensor attached to the thumb. The "hand open" and "hand close" commands are sent by the PC to the micro-controller. The control algorithm is implemented on the micro-controller as depicted in Algorithm 1. "Hand open" will open the hand until it hits the limit switch. The "hand close" will close the hand until a specified force threshold is detected by the force sensor.

3.4.2 Experimental Validation of The Prosthesis

The prosthetic limb was evaluated for its effectiveness and usability. Joint angle response of low level PID controllers were evaluated. A sinusoidal wave is generated through micro-controllers as the desired motion to the PID controllers of the DC motors. Sinusoidal input and encoder feedback (output motion) values are collected using the micro-controller.

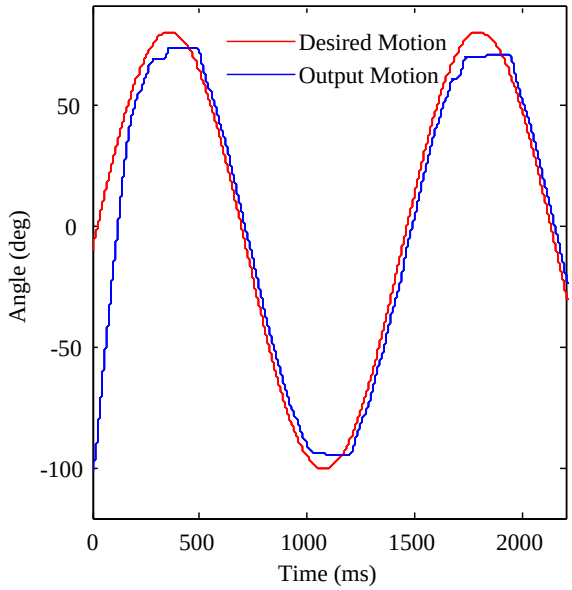


Figure 3.14: Motion output of the SP motor to a desired motion input

Table 3.1: Range of motions comparison of actual arm and prosthesis

Motion	**Range (deg) Human Limb**	**Range (deg) Prosthesis**
Elbow Flexion/Extension	0 - 145	0 - 150
Supination/Pronation	-85 - 70	-85 - 70
Wrist Flexion/Extension	-70 - 70	-60 - 60
Wrist Ulnar/Radial deviation	-35 - 20	-27 - 25

Fig. 3.14 and Fig. 3.15 depicts the output responses of the motors for forearm SP, wrist FE and wrist URD motions. In each graph it can be seen that output motions do not reach the peak of the desired motions since the ROM of prosthesis are limited to the values as shown in Table 3.1.

A comparison of experimentally obtained ROM of the prosthesis and ROM of human arm [66] is given in Table 3.1.

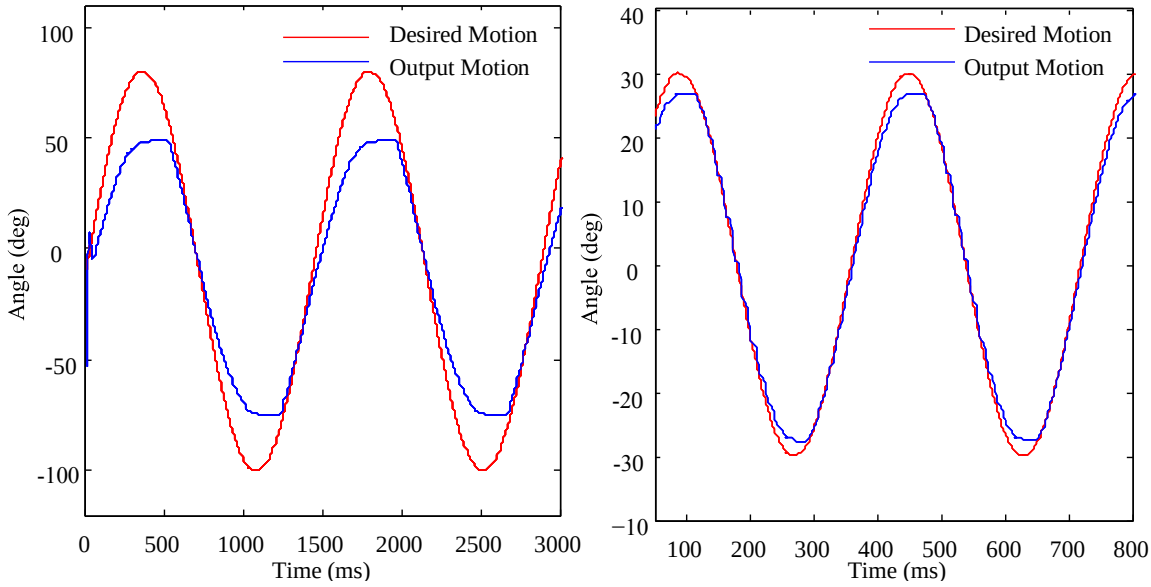


Figure 3.15: Motion output of the wrist. (a) Motion output of the wrist FE motor to a desired motion input (b) Motion output of the wrist UR motor to a desired motion input

3.5 Kinematics of the Stump Arm and the Prosthesis

The kinematic model of the prosthesis (excluding the hand) along with the stump arm is depicted in Fig. 3.16. The kinematics have been analyzed using conventional Denavit-Hartenberg (DH) parameters [19]. Table. 3.2 depicts the DH parameters related to the kinematic model. In Table. 3.2, q_1 to q_3 represent the joint angles of the stump arm and q_4 to q_7 represent the joint angles of the prosthesis.

Link 1 to Link 3 of Table. 3.2 is related to the stump arm and is used to derive transformation from shoulder (frame 1) to elbow (frame 4), T_e^{sh}. Transformation from elbow (frame 4) to hand (frame 8), T_p^e of the prosthesis is derived using Link 4 to Link 7. Furthermore, the transformation from shoulder (frame 1) to hand (frame 8) is stated in (3.7).

Table 3.2: DH parameters

Link	θ	d	a	α
1	q_1	0	0	$\pi/2$
2	$\pi/2 - q_2$	0	0	$\pi/2$
3	$\pi + q_3$	ls	0	$\pi/2$
4	$-q_4$	0	l_2	$-\pi/2$
5	$\pi/2 + q_5$	l_1	0	$-\pi/2$
6	$-\pi/2 - q_6$	l_3	l_4	$-\pi/2$
7	$-t_5 - q_7$	0	l_e	0

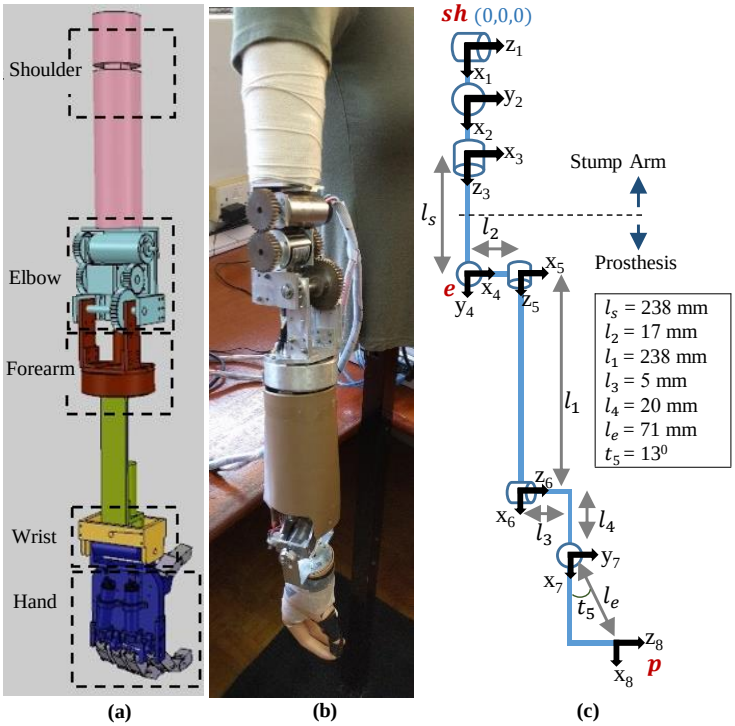


Figure 3.16: Kinematic model of stump arm and trans-humeral prosthesis, (a) Simulation environment, (b) Fabricated prosthesis, (c) Kinematic model

$$T_p^{sh} = T_e^{sh} \times T_p^e \tag{3.7}$$

where, T_p^{sh}, T_e^{sh}, and T_p^e are transformation matrix from shoulder to hand, transformation matrix from shoulder to elbow, and transformation matrix from elbow to hand respectively.

3.6 Summary

This chapter proposed a 5 DOF trans-humeral prosthesis which can mimic human motions and a simulation environment to simulate control algorithms of trans-humeral prosthesis. The prosthesis design consist of a novel 2 DOF wrist mechanism which has a axis offset as well.

simulation environment comprised of a virtual model of the trans-humeral prosthesis and a virtual shoulder joint. The virtual shoulder joint can mimic the human stump arm motions with the aid of an IMU. PID controllers are implemented to control joints of the prosthesis and the virtual shoulder. Simulation environment demonstrated effective results for the shoulder joint control through an IMU and remaining joints are controlled using the UI. Prosthetic control algorithms can easily be tested on the simulation environment. Use of this simulation environment can be extended for other types of prostheses.

The 5 DOF prosthesis has been fabricated and experimentally verified. The prosthesis can achieve slightly below ROM compared to a human UL in elbow FE, forearm SP, wrist URD and wrist FE. Prosthesis weighs about 3.2kg which is similar to a grown human's UL.

Moreover, the kinamatic analysis of the prosthesis and stump arm was carried out. The kinematics were analyzed using conventional DH parameters.

Chapter 4

REACH-TO-GRASP PATH PLANNING BASED ON A 2-1/2D METHOD OF VISUAL SERVOING

This chapter proposes a reach-to-grasp path planning method for trans-humeral prostheses which can reach towards objects of interest according to the wearers intention. The proposed path planning method uses a 2-1/2D visual servoing module (VSM) to plan reaching (reach-to-grasp) motions. Visual servoing is the use of feedback from a vision sensor (camera) to control the motion of a robot. Visual servoing can be performed in two ways: position based visual servoing (PBVS or 3D) and image based visual servoing (IBVS or 2D). These two methods can be combined to avoid drawbacks of each method and resulted in 2-1/2D visual servoing.

An object reaching algorithm is proposed along with the VSM to control the elbow FE to drive the hand towards the object of interest. The VSM is integrated with an EMG based module (EBM). The EBM is used to control elbow FE based on a novel EMG-force proportional and moment balance model (EFPMB). EMG is the signals extracted from the muscles which carry motion intentions of the subject. EMG can be extracted by inserting a needle into the musce (iEMG) or plcing a sensor on top of the surface of the muscle (sEMG). sEMG is preffered by researchers due to the non invasive extraction procedure.

Since elbow FE has two angular inputs from EBM and VSM, two modules are integrated to achieve a resultant angle for the elbow FE to reach towards the object of interest while keeping the user controllability.

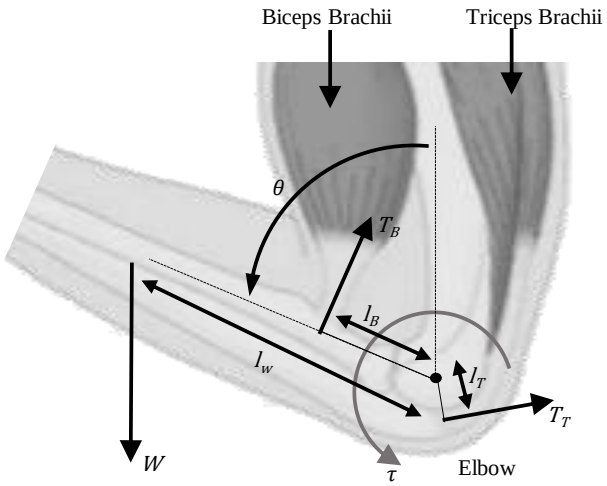


Figure 4.1: Forces acting on the forearm. Tendon extending from biceps brachii is connected to the radius bone. It is almost perpendicular to the radius bone. Tendon extending from triceps brachii is connected to the ulnar bone and slides around the elbow joint

4.1 EMG Based Module (EBM)

EFPMB model is proposed to generate elbow motion according to the motion intention of the prosthesis user.

It is assumed that significant portions of biceps brachii and triceps brachii are available after a trans-humeral amputation and the EMG signals generated from amputated muscles can be considered as equivalent to that of a healthy person [70, 71].

Since, the force generated during an isometric contraction is proportional to the EMG signal generated in the respective muscle [72, 73], root mean square (RMS) of EMG signals generated in each muscle is assumed to be directly proportional to the force (tension) generated by the muscle assuming that the user performs elbow motions at a slower phase. The forces acting on the forearm is modeled as shown in Fig. 4.1. Hence, the elbow angle can be stated as follows.

By considering the balance of moments around elbow axis as per the forearm model which is given in Fig. 4.1,

$$T_B l_B = \tau + T_T l_T + W l_W sin(\theta) \tag{4.1}$$

where, τ, T_B, T_T, W, l_B, l_T, l_W, and θ are torque applied on elbow joint, force generated by biceps brachii, force generated by triceps brachii, weight of the forearm, perpendicular distance to T_B from elbow axis, perpendicular distance to T_T from elbow axis, distance to W from elbow axis, and current elbow angle respectively. (4.1) can be rearranged as follows,

$$I\alpha = T_B l_B - T_T l_T - W l_W sin(\theta) \tag{4.2}$$

where,

$$\tau = I\alpha \tag{4.3}$$

where, α and I are angular acceleration and moment of inertia of the forearm respectively.

When discretizing (4.2), α can be stated as shown in (4.4),

$$\alpha = \delta\theta/(\delta T)^2 \tag{4.4}$$

Hence, by discretizing (4.2),

$$\delta\theta = \frac{T_B l_B - T_T l_T - W l_W sin(\theta)}{I} \times (\delta T)^2 \tag{4.5}$$

where, $\delta\theta$ and δT are elbow angle changes produced by the EFPMB and sample time respectively.

Assuming EMG RMS is proportional to the force generated by the muscles,

$$T_B = M_1 E_B \tag{4.6}$$

$$T_T = M_2 E_T \tag{4.7}$$

where, E_B , E_T, M_1 and M_2 are EMG RMS of biceps brachii, EMG RMS of triceps brachii, and proportional constant for biceps brachii, and proportional constant for triceps brachii respectively. Force generated by the biceps brachii is taken as $M_1 E_B$. The RMS is taken for a sample size of 100, sampled at 2000 Hz.

Since I, δT, W, and l_W are constant and l_B, l_T can be assumed to be constant around 90° of elbow angle, (4.5) can be stated as,

$$\delta\theta = K_1 E_B - K_2 E_T - K_3 sin(\theta) \tag{4.8}$$

where, K_1, K_2, and K_3 are proportional constants as given below,

$$K_1 = \frac{M_1 l_B (\delta T)^2}{I} \tag{4.9}$$

$$K_2 = \frac{M_2 l_T (\delta T)^2}{I} \tag{4.10}$$

$$K_3 = \frac{W l_W (\delta T)^2}{I} \tag{4.11}$$

The elbow angle, θ produced by the EFPMB is,

$$\theta_{(t)} = \theta_{(t-1)} + \delta\theta \tag{4.12}$$

where, $\theta_{(t-1)}$ and $\theta_{(t)}$ are previous elbow angle and elbow angle produced by EFPMB respectively.

4.2 Visual Servoing Module (VSM)

The proposed VSM uses the eye-in-hand camera configuration, where the camera is fitted on to the palm or the end effector of the robotic prosthesis. The reach-to-grasp path planning method is developed using the 2-1/2D VSM. The proposed method is capable of altering the orientation of the prosthesis to match with the target object orientation while reaching towards the object. The VSM consists of an image based visual servoing system (IBVS), a position based visual servoing system (PBVS), and a object reaching algorithm.

In order to map image features into the end effector velocity of prosthesis, a transformation matrix is used in IBVS. This matrix is known as Image Jacobian matrix. From the end effector of the prosthesis (Cartesian space) to its joint space the inverse kinematics (IK) can be derived. Fig. 4.2 depicts the IBVS process.

IK of the prosthesis is derived using an artificial neural network (ANN) [74,75]. ANN is used as it can be trained for the workspace that the robot can reach. Hence, ANN based IK will result in joint angles which are achievable and within joint limits. Moreover, conventional methods such as robot Jacobian may produce joint angles that the robot cannot achieve due to singularities present and the joint limits [76]. This is very critical for prosthesis applications since some human joints have limited angular ranges compared to a robot with same DOF.

The visual servoing process used in this study is as follows.

Velocity of the object ($\dot{P}$) with respect to camera frame can be expressed as given in (4.13) (Refer Fig. 4.2).

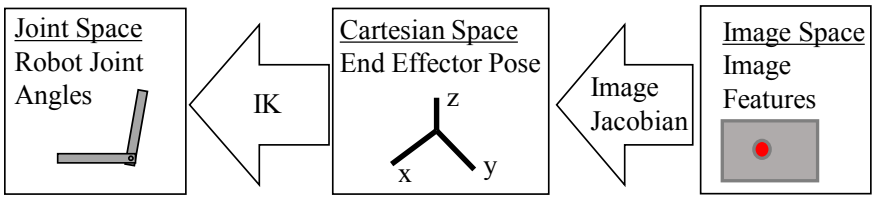


Figure 4.2: Image based visual servoing process. Image features captured using the camera are transformed into the cartesian space using image jacobian. Cartesian space to the robot joints are the inverse kinematics of the robot

$$\dot{P} = \omega \times P + V \tag{4.13}$$

where P, ω, and V are position of the object, angular velocity of point P, and translational velocity of point P respectively. (4.13) can be expanded as follows,

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix} = \begin{pmatrix} z.\omega_y - y.\omega_z \\ x.\omega_z - z.\omega_x \\ y.\omega_x - x.\omega_y \end{pmatrix} + \begin{pmatrix} V_x \\ V_y \\ V_z \end{pmatrix} \tag{4.14}$$

where $[x, y, z]^T$, $[\omega_x, \omega_y, \omega_z]^T$, and $[V_x, V_y, V_z]^T$ are point P, angular velocity of P, and translational velocity of P respectively.

The projection of P onto image plane is taken as $P_I = [u, v]$ (refer Fig. 4.3). From equations of similar triangles,

$$x = \frac{uz}{\lambda} \tag{4.15}$$

$$y = \frac{vz}{\lambda} \tag{4.16}$$

where, λ is the focal length of the camera. By substituting (4.15) and (4.16) into (4.14),

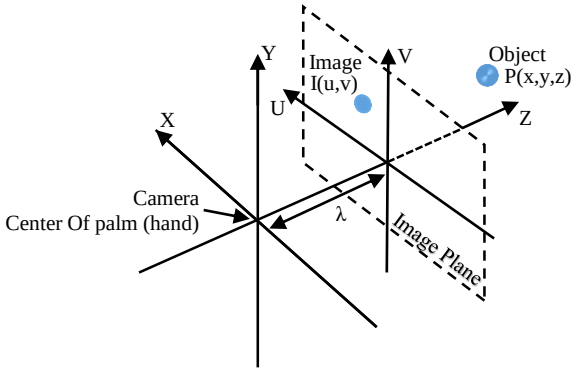


Figure 4.3: Perspective projection. Projection of an object in 3D space onto the camera plane. Image coordinates, (u,v) are extracted from the captured image and object position, (x,y,z) is the actual position of the object with respect to the Camera

$$[\dot{u}, \dot{v}]^T = J[V_x, V_y, V_z, \omega_x, \omega_y, \omega_z]^T \tag{4.17}$$

where,

$$J = \begin{pmatrix} \frac{\lambda}{z} & 0 & \frac{-u}{z} & \frac{-uv}{z} & \frac{(\lambda^2 u^2)}{\lambda} & -u \\ \\ 0 & \frac{\lambda}{z} & \frac{-v}{z} & \frac{-(\lambda^2+v^2)}{\lambda} & \frac{uv}{\lambda} & u \end{pmatrix} \tag{4.18}$$

J is known as the Image Jacobian. This Image Jacobian can be used for point to point image features in a monocular camera configuration [77]. As per (4.17), Image Jacobian relates differential changes of end effector position (velocity) and differential changes of images features. Jacobian, J requires the focal length (λ, a fixed parameter of the camera) and distance to the object from the camera, z for the calculation. z is measured using an US sensor attached to the palm of the prosthesis. Inverse of the Jacobian matrix can be used to estimate the desired end effector velocity (or position) from the image features. If Jacobian matrix is non-invertible, pseudo inverse is used [78]. The control law can be stated as follows,

$$\dot{q} = -KJ^{+}e(f) \tag{4.19}$$

where, $\dot{q}$, K, J^{+}, and $e(f)$ are end effector velocity screw ($[V_x, V_y, V_z, \omega_x, \omega_y, \omega_z]^T$), a constant gain, pseudo inverse of Jacobian, and feature error respectively. $e(f)$ can be given as (4.20).

$$e(f) = f_c - f_d \tag{4.20}$$

where, f_c and f_d are current and desired image features respectively. Current image features are the coordinates of centroid of the object. Desired image features are the center coordinates of the image frame.

The desired pose $q_{(t)} = [x, y, z, \alpha, \beta, \gamma]^T$ of the end effector (hand) is calculated according to (4.21) using $\dot{q}$ obtained from (4.19).

$$q_{(t)} = q_{(t-1)} + \dot{q} \times \delta T \tag{4.21}$$

where, $q_{(t)}$, $q_{(t-1)}$, and δT are desired pose, current pose, and sample time respectively.

Aforementioned pose is to move the end effector so that the object is in the middle of the image frame. Nevertheless, the end effector needs to align with the object, so that the object can be grasped by the hand of the prosthesis (see Fig. 4.4). This is achieved by combining the IBVS with the PBVS to come up with a 2-1/2D VSM. The required angle of rotation along the main axis of camera is calculated using the image. The desired pose of the end effector is transformed from the angle derived, along the camera axis [refer Fig. 4.5 and (4.22)].

$$\hat{q} = q_{(t)} \times T \tag{4.22}$$

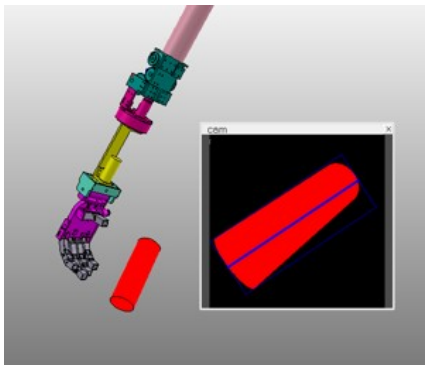


Figure 4.4: Misaligned object. The object needs to be aligned with the hand to grasp the object properly

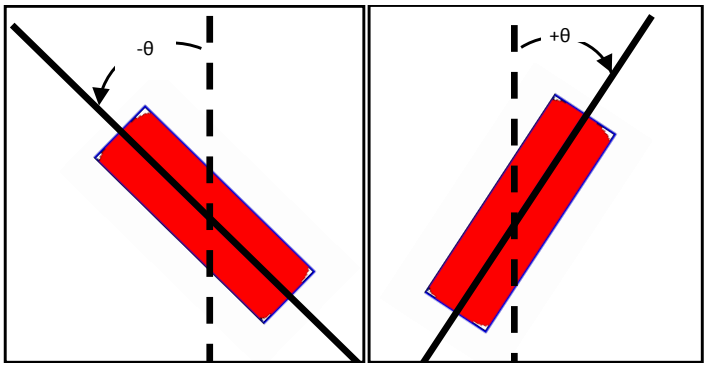


Figure 4.5: Misaligned object as seen by camera. Two images of the object as seen by the camera with the misaligned angle, θ marked on the images

where, $\hat{q}$ and T are pose after transformation and transformation matrix along the image axis by misaligned angle, θ respectively. The sign convention of θ is indicated in Fig. 4.5. Transformation along image axis (Z $axis$) can be stated as follows.

$$T = \begin{bmatrix} cos(\theta) & -sin(\theta) & 0 & 0 \\ sin(\theta) & cos(\theta) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.23}$$

The desired joint angles required to achieve the pose, $\hat{q}$ is derived using the ANN. The pose, $\hat{q}$ of the end effector is fed into the ANN to calculate desired joint angles. The VSM without the object reaching algorithm is shown in Fig. 4.6.

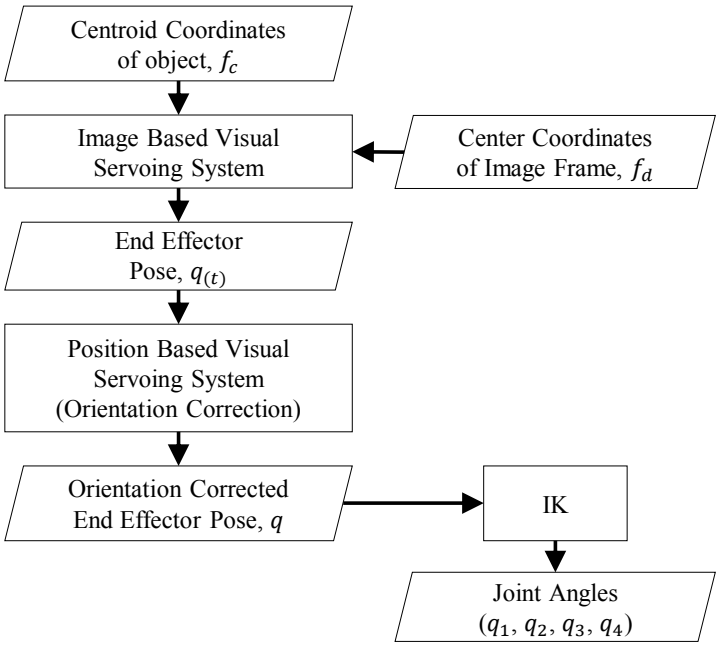


Figure 4.6: Visual servoing module. Coordinates of the centroid of object and center coordinates of the image are fed into the IBVS, which gives the hand pose required to center the object in the image frame. Orientation of the hand is corrected by the PBVS

The ANN is implemented as shown in Fig. 4.7. It consists of three layers: input layer, hidden layer, and output layer. Input layer consists of six input neurons to input the pose. ANN consists of 100 hidden neurons and 4 output neurons. Four output neurons outputs joint angles of robotic prosthesis in the following order: elbow FE, forearm SP, wrist URD, and wrist FE. Activation function in input and output layers are set to linear and in hidden layer it is set to sigmoid. Supervised back-propagation learning method is used to train the ANN using a data set generated using the kinematic model of robotic prosthesis. Several prosthesis joint angle combinations are generated by changing each and every joint by 10° intervals. Then the forward kinematics of these joint angle combinations are taken using the kinematic model. Pose generated by forward kinematics are used as inputs and respective joint angle combinations are used as outputs to train the ANN. Calculated joint angles from ANN are sent to the prosthesis.

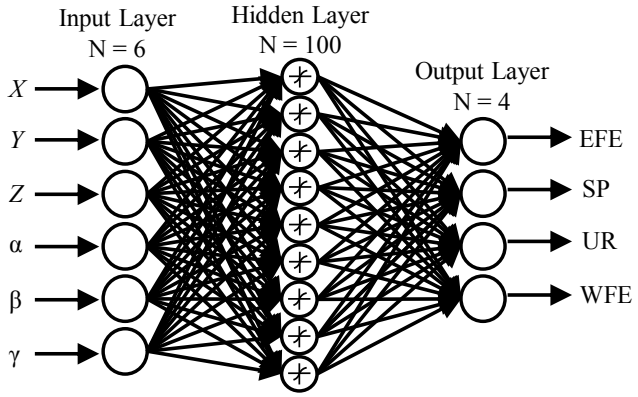


Figure 4.7: Artificial Neural Network (ANN) used for the Inverse Kinematics (IK)

The above method is capable of getting the object into the middle of the image plane and correct the orientation according to the position and orientation of the hand. However, prosthetic hand may not be able to reach towards the object while keeping the object in the middle of the image plane. In this regard elbow can be treated as the joint which most contributes in making the object closer to the hand. Thus an object reaching algorithm is introduced to adjust the elbow angle. The object reaching algorithm is shown in Algorithm 2. It measures the distance to object by rotating it in one direction, if the resultant distance is lower than the previous distance, further rotation towards that direction is performed. Moreover, if the resultant distance is greater than the previous distance, rotation in other direction is performed. These rotations are performed only if the difference in distances are above a specified threshold.

4.3 Integration of VSM and EBM

The overall path planning method of the prosthesis is shown in Fig. 4.8. The path planning method is built-up integrating the EBM and VSM. The EBM is initially used to move the prosthesis elbow joint according to the human motion intention. A voluntary isometric contraction in biceps brachii and triceps brachii is used to trigger the VSM. The VSM starts servoing towards the object if an

Algorithm 2 Object Reaching Algorithm

```
1: state = True ← place holder for two different states
2: Zto = 0 ← variable for old distance
3: Zt = 0 ← variable for distance
4: Dist = 0 ← variable for difference in distance
5: Ang ← Elbow Angle
6: L ← Proportional Constant
7: loop:
8: while Zt <= 50 do
9:     Visual Servoing
10:    Zto = Zt
11:    Zt ← Measured Depth from US sensor
12:    Dist = Zt - Zto
13:    if Dist > 0 then
14:        state = !state
15:    else
16:        state = state
17:    if state == True then
18:        Ang -= ( L x Dist )
19:    else
20:        Ang += ( L x Dist )
```

object is detected. The joints of prosthesis are controlled using the VSM. However, the elbow has two input angles from the EBM and the VSM. Therefore, a fusing filter is implemented as given in Algorithm 3 to control the prosthesis elbow. If the difference of two elbow angles produced by two modules (EBM and VSM) are above the changing threshold (Th), the fusion fusing filter goes into the stepping mode. In stepping mode, the elbow angle changes towards the angle derived from EBM in N steps. However, If the filter is not in the stepping mode the final elbow angle will take the angle from EBM. Both EBM and VSM calculations are running at 50ms intervals (20Hz). N and Th can be set based on the user experience.

4.4 Experimental Validation of the Path Planning Method

Experiments are carried out to validate the proposed path planning method. The EBM is evaluated using the fabricated prosthesis. The VSM and the fusion filter are evaluated using the simulation environment.

Algorithm 3 Fusion Filter

$state$ = True ← Place Holder for the VSM state
$n = 1$ ← Variable for incrementing from 1 to N
$D = 0$ ← Variable for difference between two elbow angles
Th ← Changing Threshold
N ← No of steps
Ang_{EFE} ← Calculated Elbow angle
Ang_{VS} ← Elbow angle from VSM
Ang_{EMG} ← Elbow angle from EBM
$D = Ang_{VS} - Ang_{EMG}$
loop:
while $state$ = True **do**
    if $|D| > Th$ **then**
        $n = 1$
    else
        $Ang_{EFE} = Ang_{VS} - Dn/N$
        if $n < N$ **then**
            n += *1*
        else
            $n = N$

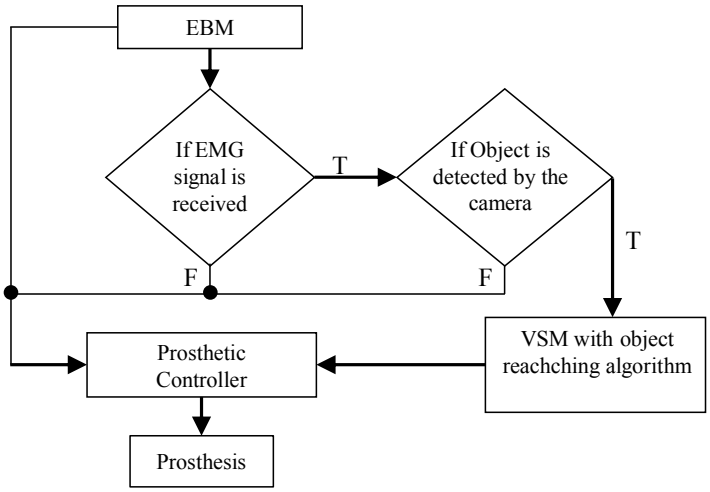


Figure 4.8: Overall path planning method of the prosthesis. Initially the elbow FE of the prosthesis is controlled with the EBM. When an object is detected by the camera and the EMG signal is received to reach towards the object VSM with object reaching algorithm activates and converges towards the object

4.4.1 Experimental Setup

The experimental setup is shown in Fig. 4.9. It consists of the fabricated prosthesis, EMG acquisition system (Bagnoli 16, Delsys), a personal Computer

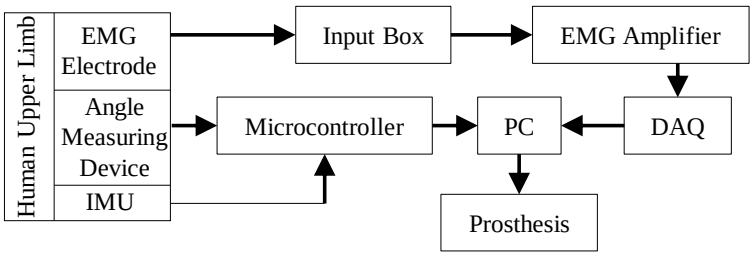


Figure 4.9: Experimental setup

(PC), IMU, and angle measuring device. The PC used has a 7th generation intel i7 processor, 16GB of RAM, nvidia quadro graphics card, and a 240 GB SSD.

EMG acquisition system is used to extract surface EMG signals from biceps brachii and triceps brachii of 8 healthy subjects. Two single differential EMG sensors are attached to the skin surface above biceps brachii and triceps brachii. Sensors are connected to the EMG amplifier through the input box. Amplified EMG signals are transmitted to the PC through a data acquisition (DAQ) card (NI-6220, National Instruments) and processed according to EBM. These signals are sampled at $2000Hz$ and band pass filtered to be within $50Hz$ to $450Hz$. The resultant elbow FE angle is calculated in a Matlab script according to EBM. The calculated elbow FE angle is fed into the prosthesis using a microcontroller (ATmega2560, Atmel) where the low level joint controllers are implemented. PC communicates serially (RS232) with the microcontroller at the prosthesis. EBM experiments are conducted on a real-time basis and subjects are asked to perform elbow FE motion without moving the shoulder. Elbow FE angles of the subject and the prosthesis are recorded with two angle measuring devices. These angle measuring devices are equipped with potentiometers and data are recorded using a microcontroller (ATmega2560, Atmel).

In simulations, the joint angles calculated in Matlab are fed into the simulation environment through remote Application Program Interface (API) functionality of V-REP. A virtual camera attached to the palm of the prosthesis is used to identify the target object. Images obtained from the camera are processed using an OpenCV based filter. The filter is capable of detecting objects and returning

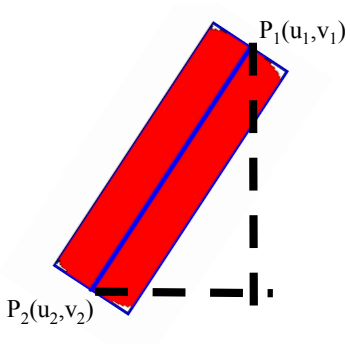


Figure 4.10: Detected two points for angle calculation. A bounding rectangle around the detected object is created and two points at the middle of the short edges of the bounding rectangle are selected

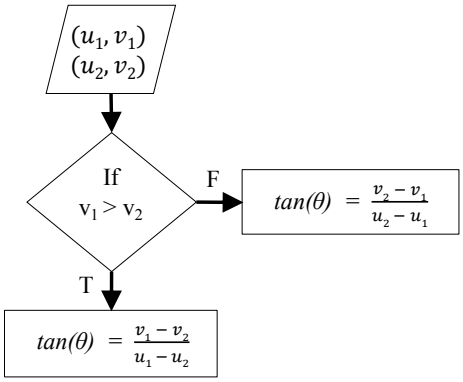


Figure 4.11: Calculation of angle, θ. Point at the top is selected by comparing v_1 and v_2. Hence the misaligned angle is calculated

the centroid image coordinates of the object. These centroid coordinates are processed according to (4.19) and (4.21).

The orientation correction process is performed immediately after visual servoing using the IBVS. The misaligned angle, θ is found by identifying two points in the image. These two points are at the two distal ends of the detected object. Those are the center points of the short edges of the constructed bounding rectangle (refer Fig. 4.10). The angle, θ is given as depicted in Fig. 4.11. This angle is used to calculate the desired pose, q using (4.22).

Moreover, the fusion filter is also evaluated using the simulation environment. resultant elbow angle calculated from the EBM is fused with the VSM using the fusion filter. The parameters N and Th are found using an user study.

4.4.2 Experiments and Results

The first experiment is conducted to evaluate the ANN used for the IK of the prosthesis. 1000 random combinations of joint angles are used for the experiment. Joint angle combinations are used to calculate the pose of the end effector using the kinematic model of the prosthesis. The poses are used as inputs to the ANN. Calculated joint angles from the ANN is compared with the original joint angles. Experimental procedure is shown in Fig. 4.12. The ANN was able to predict angles with a mean square error (MSE) less than 0.06. The results of the ANN based IK is depicted in Table 4.1. The MSE of angles are 0.0078, 0.0443, 0.0566, and 0.0444 for elbow flexion/extension, supination/pronation, wrist flexion/extension, and ulnar/radial deviation respectively. The MSE of position and orientation are 0.1731, 0.1115, 0.0572, 0.0.0439, 0.0751, and 0.0991 for x-coordinates, y-coordinates, z-coordinates, angle around x-axis (α), angle around y-axis (β), and angle around z-axis (γ) respectively.

Moreover, the ANN method of IK has been compared with other methods of obtaining IK. Damped least squares (DLS) method, which is based on robot Jacobian inversion and an optimization method based on inertia-point (IP) algorithm has been used for the comparison. The results are shown in Table. 4.2. Experiment is conducted for 100 poses within the reachable workspace. ANN method outperforms for the application on a prosthesis since it has no effect from singularities present due to trained pose that may result. Moreover, DLS and optimization methods try to find a pose which minimizes the error, which may find a local minima at joint limits and stuck there. Furthermore DLS and optimization methods resists shifting from one IK solution to other whereas ANN method outputs a solution based on the trained workspace. ANN can perform an IK calculation within $50ms$, whereas other two methods are more time consuming.

Secondly the EBM is evaluated. Fig. 4.13 shows the prosthesis and human angle variation for 3 subjects. In Fig. 4.13, (a1) and (a2) represents the same

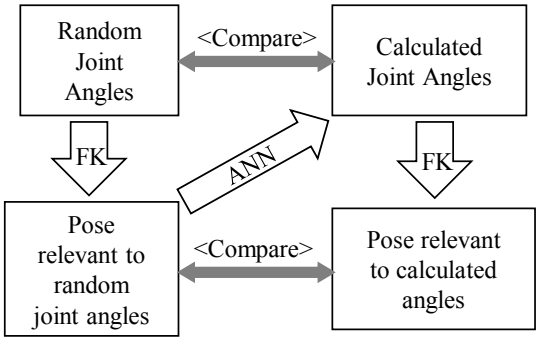


Figure 4.12: Experimental procedure for ANN based Inverse Kinematics (IK)

Table 4.1: Results of ANN based Inverse Kinematics (IK)

Parameter	MSE	Average Error	Maximum Error
$q1$	0.008219	0.000166	0.456126
$q2$	0.049166	-0.01161	0.665428
$q3$	0.057372	-0.01048	1.681989
$q4$	0.048098	0.019348	1.858260
x	0.192884	0.007915	2.389031
y	0.134373	0.012536	1.849927
z	0.062545	0.005552	1.155758
α	0.054377	-0.01048	0.995110
β	0.150753	-0.01519	2.325238
γ	0.154540	0.010858	3.023294

Table 4.2: Comparison of Inverse Kinematics (IK) methods

Algorithm	RMS Error of Position	Average Time Taken
ANN	1.65 mm	43 ms
IP Optimization	20.35 mm	59 ms
DLS	88.97 mm	127 ms

result. However, (a1) represents the real-time variation of the elbow angles, in (a2) the prosthesis angle is shifted to the left to match the human elbow angle eliminating the lag. Table 4.3 represents the RMSE, the shifted time or the time lag, and the RMSE after correcting the time lag. Mean row in Table 4.3 shows the mean values for 3 subjects. The prosthesis follows the desired motion of elbow with an RMSE of 10.87 degrees and percentage error of 7.44% (10.87/146). However, minor lagging behind the actual human limb can be observed due to the processing time incurred and also it takes 50ms to capture signals from the onset

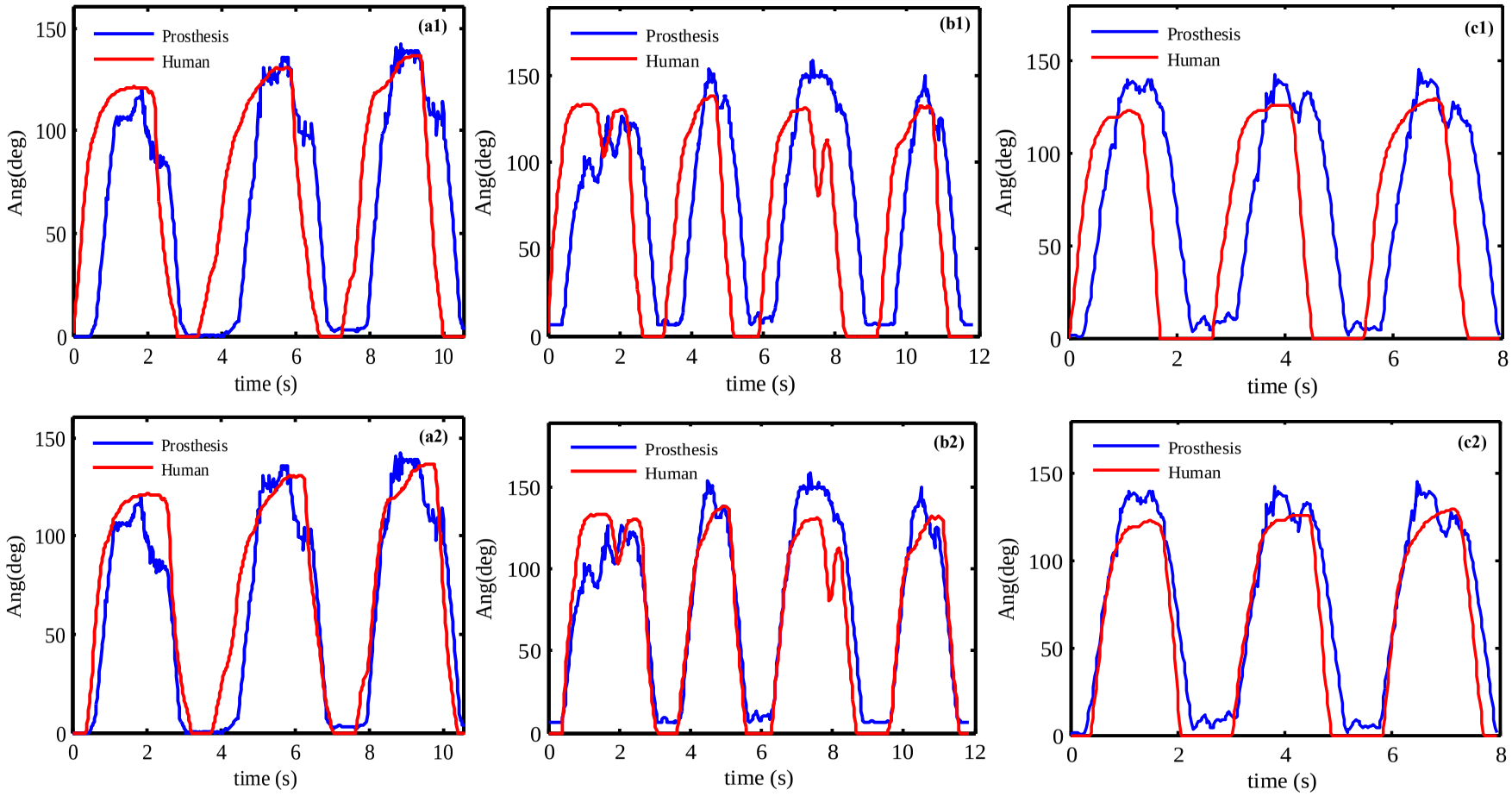


Figure 4.13: Comparison of the angle produced by the EBM with the human elbow angle for 3 healthy subjects

Table 4.3: Summary of EBM performance

Subject	**RMSE (Deg)**	**Time Shift (ms)**	**RMSE after shift (Deg)**
a	24.27	400	11.50
b	28.66	420	10.23
c	29.74	360	10.87
Mean	27.56	393	10.87

of motion. According to literature, a time lag of $300ms$ is acceptable for real-time operation [27]. Hence, a time lag of $393ms$ is acceptable and an amputee may get used to this small time lag as he/she uses the prosthesis.

In the third experiment, the IBVS is evaluated. A spherical object is randomly placed on the workspace making it visible to the camera. The convergence of the object towards the center of the image frame is observed while changing the shoulder angles from the IMU. The results are shown in Fig. 4.14. It can be seen from the results that the object moves out from the center of the image when shoulder angles are changed. The IBVS corrects prosthesis angles so that the object converges toward the center again. However, the IBVS alone cannot make the hand reach toward the object. Distance to the object (z) does not reduced to make the object closer to the hand.

In the fourth experiment, the IBVS with object reaching algorithm is evaluated. The convergence of the hand towards the object is observed. The results are shown in Fig. 4.15. It can be seen from Fig. 4.15 that the object converges towards the center of the image frame when the visual servoing is performed. Moreover, the object reaching algorithm makes the distance to the object minimum by changing the elbow angle as visual servoing is performed. In this process the distance to object (z) is reduced from S1 to S2 time interval [Refer Fig. 4.15(b)]. The elbow FE angle variation from S1 to S2 time interval to achieve this distance reduction can be seen in Fig. 4.15(c). From S2 time point onwards, the distance to object (z) is further reduced by changing the shoulder angles. The process where prosthetic hand reaches towards the object of interest by changing

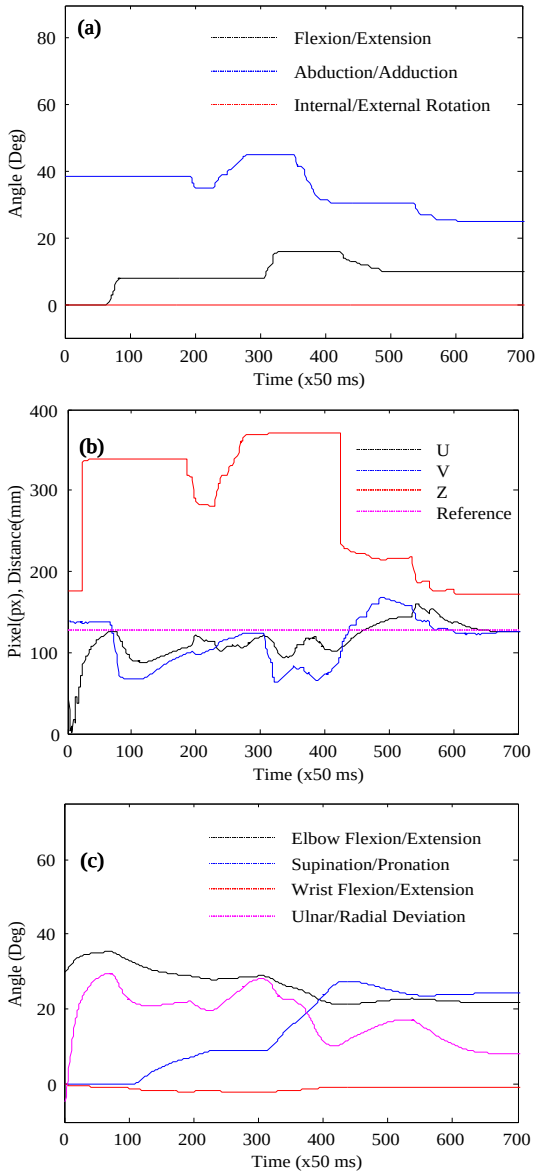


Figure 4.14: Reach-to-grasp path planning using IBVS, (a) Shoulder angle variation, (b) Image coordinates and distance to the object variation, and (c) Prosthesis joint angle variation

the elbow angle is depicted in Fig. 4.16. z reduces even below $50mm$ with the help of object reaching algorithm. The experiment is repeated for 10 different object positions. Hand paths for 3 of the object positions are shown in Fig. 4.17. Distance to the object from the hand at the end of visual servoing is $18mm$, $23mm$, and $28mm$ respectively for 3 paths shown in Fig. 4.17. As per the paths, the hand converges towards the object of interest with the aid of object reach-

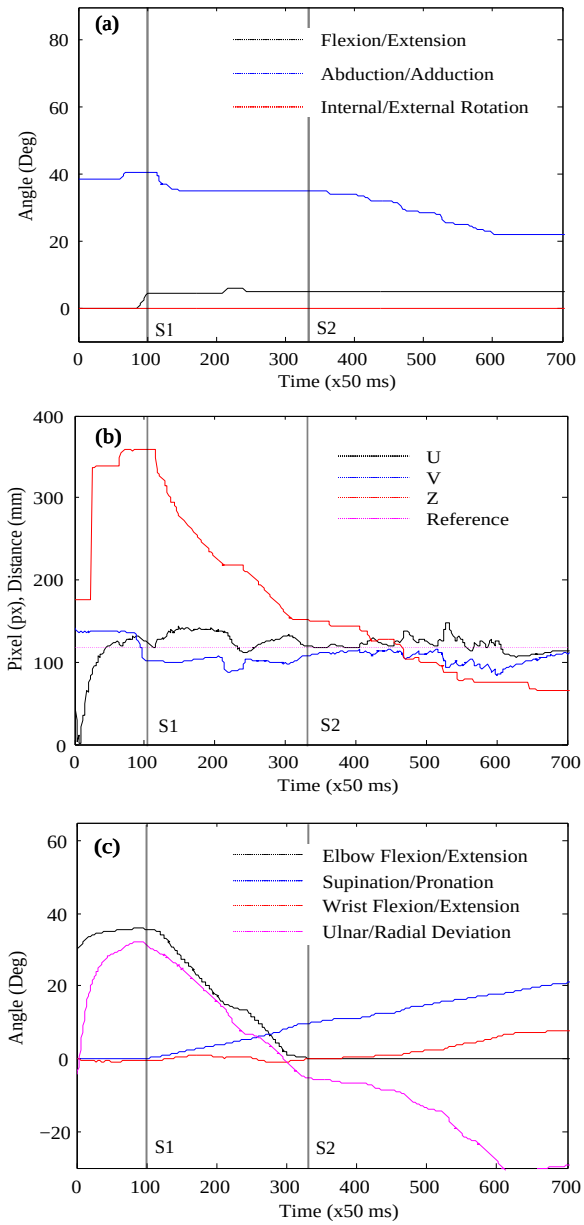


Figure 4.15: Reach-to-grasp path planning using IBVS with object reaching algorithm, (a) Shoulder angel variation, (b) Image coordinates and distance to the object variation, and (c) Prosthesis joint angle variation

ing algorithm. The paths are smooth without any sudden changes despite some curvatures.

Fifth experiment is the evaluation of PBVS. Only the PBVS is implemented and the resultant image features (position of the centroid and misaligned angle) are recorded. Results are shown in Fig. 4.18. At the beginning, a cylindrical

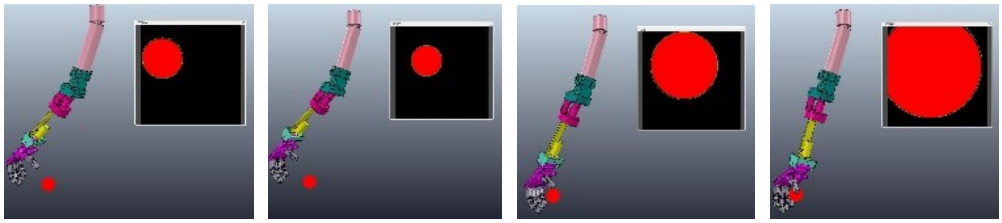


Figure 4.16: Reach towards the object of interest by changing the elbow angle using object reaching algorithm

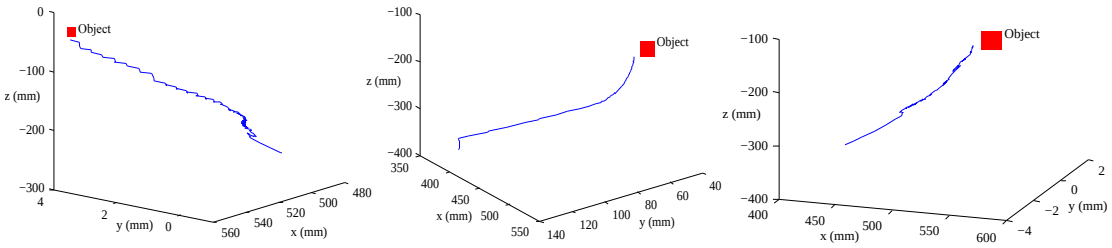


Figure 4.17: Hand trajectories for 3 different object positions when reach-to-grasp path planning with object reaching algorithm

object misaligned by $-10°$ is placed on the workspace. Fig. 4.18 shows that the proposed PBVS is capable of correcting the hand orientation so that the object is properly aligned. The PBVS can correct the misaligned angle effectively while keeping the object to the center of the image.

Sixth experiment is carried out to evaluate the overall VSM for reach-to-grasp motions of the prosthesis. The orientation of the object is changed between $-10°$ and $10°$ while changing the shoulder angles from IMU. The resultant joint angles from the VSM and the image features are recorded along with the orientation of the object and the shoulder angles. Object reaching algorithm is omitted since it may converge the hand towards the object, preventing further visual servoing. Results are shown in Fig. 4.19. The PBVS corrects the orientation soon after visual servoing is performed using IBVS. Orientation correction process causes the object to move out from the center which is then corrected by the IBVS in the next iteration. The results indicate that the proposed path planning method is capable of correcting the orientation of the object effectively. Moreover, the object stays within the middle of the image plane.

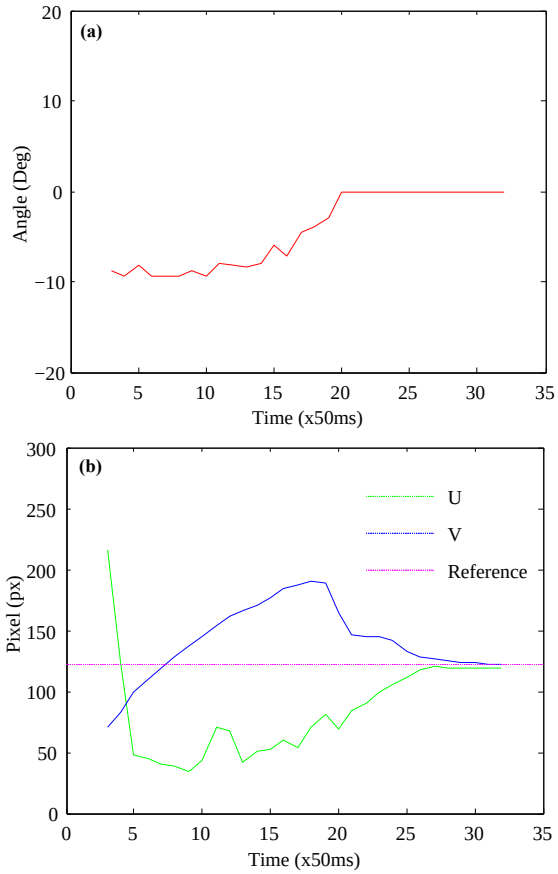


Figure 4.18: Correcting hand orientation using PBVS. (a) shows the misaligned angle and (b) shows the image coordinate variation as PBVS is performed

In the last experiment, the integration of VSM and EBM is evaluated. Two elbow FE angles from two modules are used as inputs to the fusing filter. A user study is conducted to decide on the number of steps, N and the changing threshold, Th for the fusion filter. US1, the first user study is conducted to find the changing threshold, Th. US2, the second user study is conducted to find the number of steps, N. The study is conducted using 20 subjects. In US1, changing threshold is set to different values and users are given the chance to select the best alternative for them. Users are asked to select the minimum elbow angle when the prosthesis is not required to follow the angle derived from the EBM. The results of US1 is shown in Table. 4.4. In US2, the number of steps, N is set to different values and the subjects are asked to select the best option for them. The subjects are asked to perform a higher elbow angle variation and the fusion

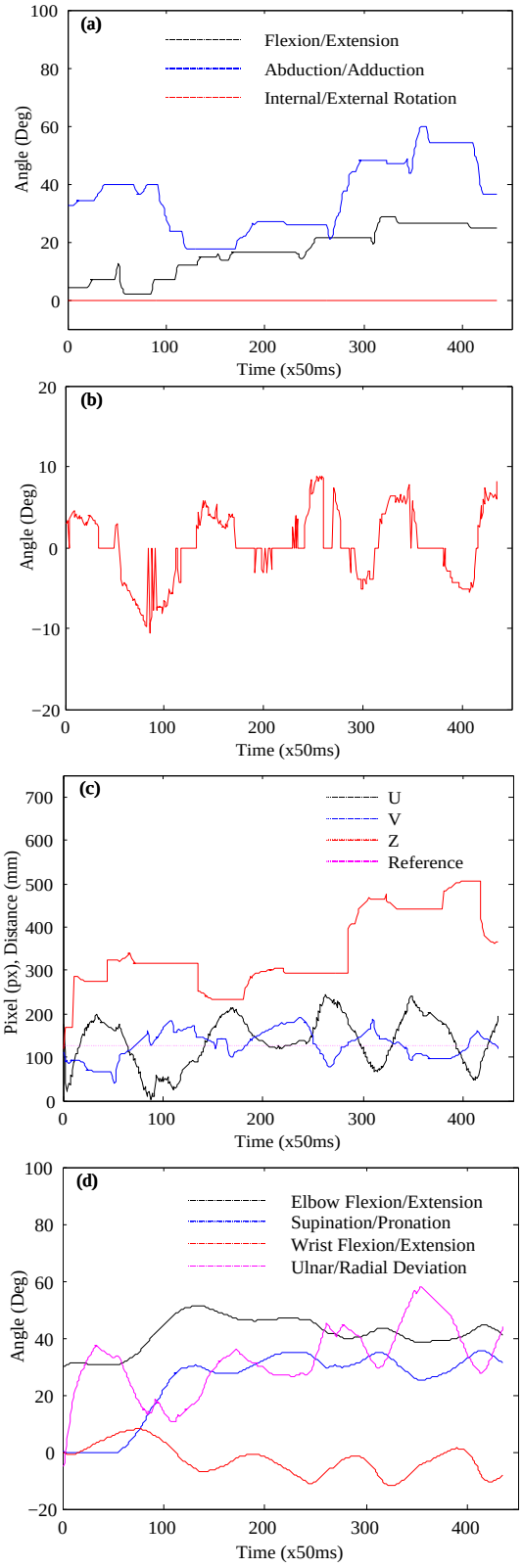


Figure 4.19: Reach-to-grasp path planning using overall VSM, (a) Shoulder angle variation, (b) Object orientation as seen by the camera, (c) image coordinate variation, and (d) Prosthesis joint angle variation

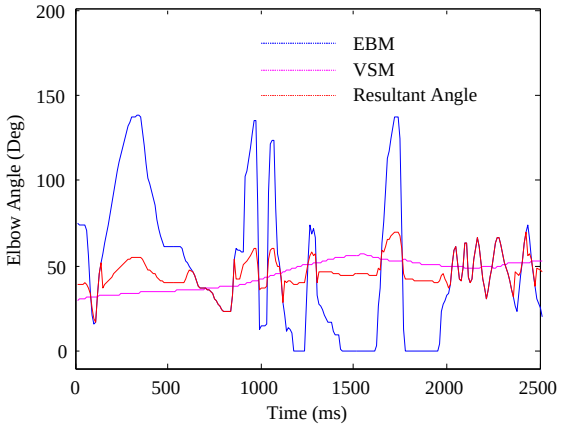


Figure 4.20: Resultant elbow FE angle obtained from integrating elbow FE angles derived from VSM and EBM

Table 4.4: Responses of User Study 1

Th (Deg)	**No of Responses**
4	1
6	2
8	3
10	8
12	4
14	2

filter changes the elbow angle towards the angle derived from the EBM based on the selected N. The responses are given in Table 4.5. According to the users' responses N and Th are selected, they are 5 and 10° respectively.

The results of the fusion filter are shown in Fig. 4.20. According to the results, elbow angle follows the VSM. However, when EBM produces angles varying slowly, the elbow angle follows the EBM. Nevertheless, if the EBM produces a sudden angle variation, only a part of that $(1/N)$ is reflected in the final elbow angle. If the EBM angle variation persists, the fusing filter drives the elbow angle towards the angle derived from EBM in N steps.

Table 4.5: Responses of User Study 2

N	No of Responses
2	1
3	2
4	4
5	7
6	3
7	2
8	1

4.5 Summary

This chapter proposed a vision based reach-to-grasp path planning method for trans-humeral prostheses. The proposed path planning method is capable of controlling the elbow of the prosthesis with EMG signals using the proposed EFPMB model. Furthermore, it used a 2-1/2D VSM to center the object relative to the prosthesis hand while aligning the hand with orientation of the object. An object reaching algorithm is proposed for the elbow joint to reach the prosthesis hand towards the object while reducing the distance to the object. Visual servoing is performed with the aid of an ANN based IK calculator. The ANN produces promising results for IK of the aforementioned 4 DOF prosthesis with a MSE of less than 0.06 in joint angles. Experimental results validated the effectiveness of the proposed vision based path planning method. Additionally, the path planning method was capable of converging towards the object while maintaining the controllability through human motion intention. When the human elbow angle was slowly changed, the prosthesis elbow followed the EBM. However, the angle produced by EBM does not directly affect the prosthesis elbow angle if EBM produces a high difference in angles.

Chapter 5

REACH-TO-GRASP PATH PLANNING BASED ON PATH TRACKING METHODS

The method discussed in the previous chapter helps the prosthetic hand to move towards the object, yet the path towards the object is not a straight line. In order to obtain human-like motions in prostheses, the prosthetic hand should be able to follow a straight line path with a bell shaped velocity profile. As a solution, this chapter proposes an improvement to the previous path planning method which makes the paths taken by the prosthesis straight line. The proposed method is based on two path tracking methods which are capable of taking the prosthetic hand on a straight line path.

First, the human motions are analyzed to validate the straight line nature of the human paths. Later the human like reach-to-grasp path planning method is proposed. The path planning method consists of a path generation module and a path tracking module. Path generation module generates a path towards the object of interest. Path tracking module carries the prosthetic hand on the generated path compensating shoulder motions as the shoulder motions are entirely performed by the wearer of the prosthesis.

5.1 Human Reach-to-grasp Motion Analysis

This study is conducted in order to understand how a natural UL performs reach-to-grasp motions. According to literature, the human hand moves in a straight line path with a bell shaped velocity profile [58, 59].

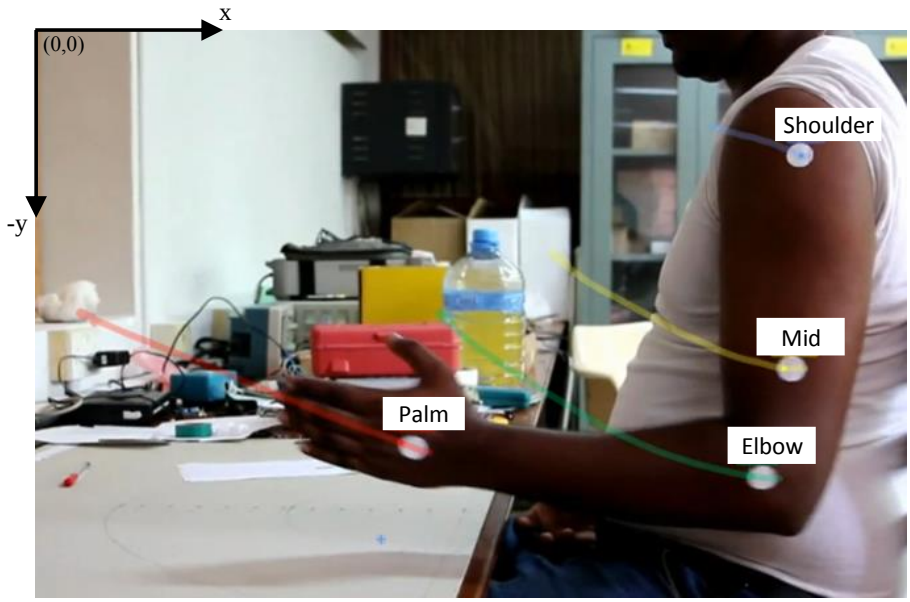


Figure 5.1: Capturing human motion while performing a reaching motion

Five human subject are asked to perform a motion towards a point in the space and the data are recorded. Several markers are attached to the hand, elbow, and shoulder of the UL of human subjects (see Fig. 5.1). The motions are recorded using a camera and analyzed using Kinovia: an open source motion analysis software. During the analysis, marker positions are tracked and recorded. Recorded marker position of the hand is used to derive the hand path and velocity profile.

The resultant paths for 3 attempts are shown in Fig. 5.2 with their velocity profiles. The straight line nature of these paths and end point errors (EPE) are shown in Table. 5.1. Straight line nature is given as the RMSE between the path taken by the hand and the fitted trend line to the path. The Mean RMSE for the paths was $3.1mm$. As per the results, human subjects are almost straight line with a bell shaped velocity profile. These results are used to evaluate the performance of the proposed path tracking methods.

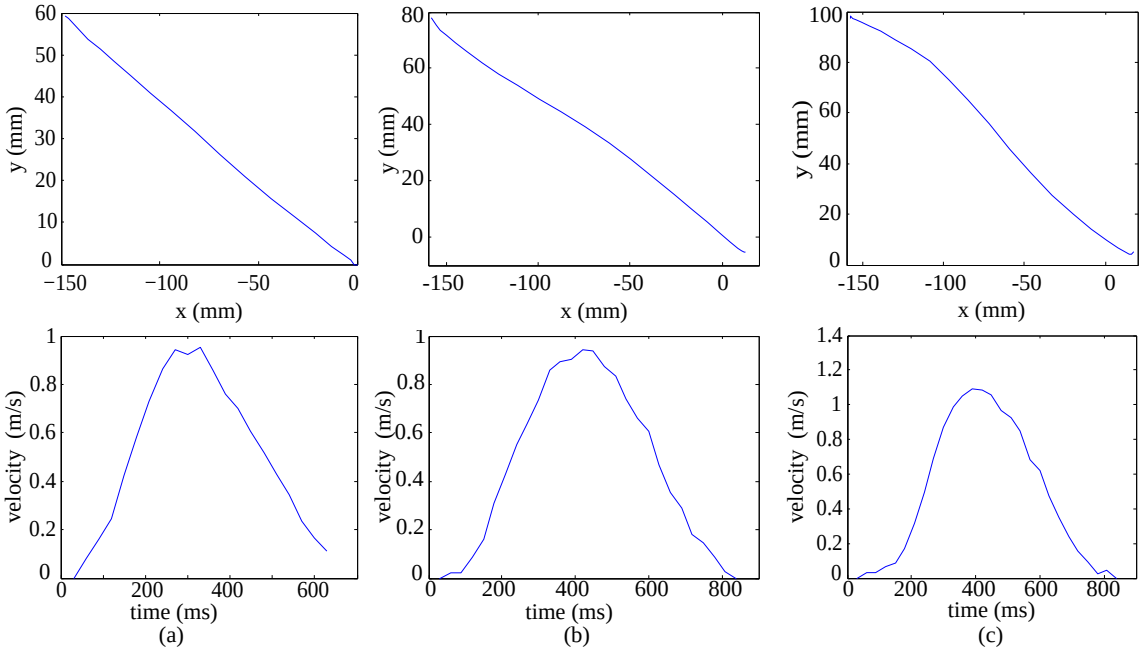


Figure 5.2: Human path taken by 3 subjects, (a) Path taken by subject 1, (b) Path taken by subject 3, and (c) Path taken by subject 4.

Table 5.1: Straight line nature of the paths taken by human hand

Person	**RMSE**	**EPE (mm)**
1	1.05	1.65
2	2.42	2.36
3	2.68	2.24
4	6.37	7.68
5	2.99	3.44
Mean	3.10	3.47

5.2 Proposed Reach-to-grasp Path Planning Method

The proposed path planning method consists of two modules: Path generation module and path tracking module. Path generation module generates a path from the current position of the hand to the destination position of the hand. Path tracking module follows the generated path to reach the destination. The destination position of the hand is the position of the object which needs to be grasped.

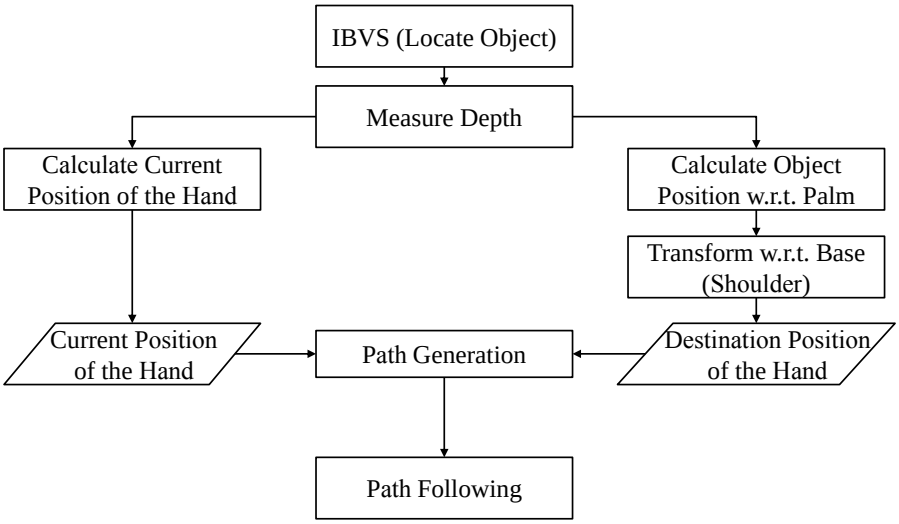


Figure 5.3: Proposed path generation module

5.2.1 Path Generation Module

In the path generation module, the object is located first. An IBVS algorithm is used to locate the object. After locating the object, current hand position w.r.t. the shoulder (base) is calculated using the kinematic model of the prosthesis and stump arm. Next, the object position w.r.t. the hand is derived. Object position w.r.t. the shoulder is calculated using transformation matrices. Finally, a path from the current position of the hand to the destination position is created. The proposed path generation module is shown in Fig. 5.3

Locating The Object

The prosthesis is set to a neutral position with following joint angles: Elbow FE:- 120 deg., SP:- 30 deg., UR:- 0 deg., and wrist FE:- -20 deg. These joint angles are selected to have the maximum visibility of the objects on a table (Refer Fig. 5.4). Moreover, the wearer can perform shoulder motions which improves the visibility of the objects on the table. Objects are positioned on the table. A camera is fitted on the hand of the prosthesis. Moreover, an US sensor is fitted closer to the camera to measure the distance to the object (refer Fig. 5.5). The

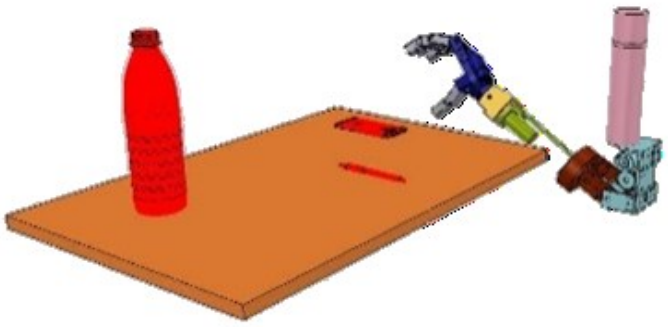


Figure 5.4: Initial position of the prosthesis

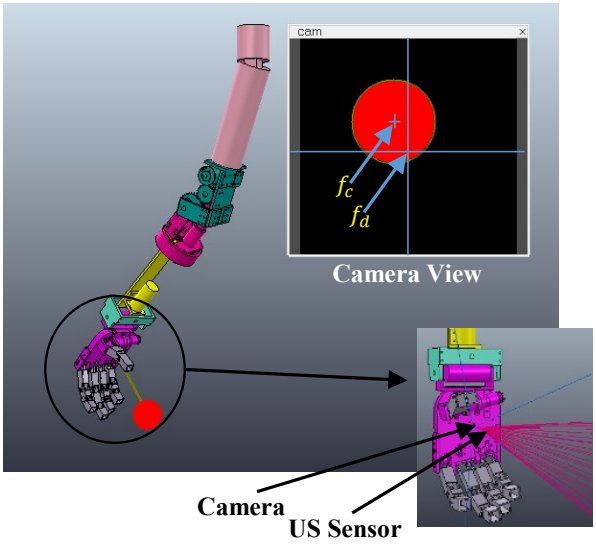


Figure 5.5: Attachment of camera and US sensor to the prosthesis

IBVS is used to locate the object and center the detected object relative to the hand.

Red coloured objects are used to eliminate the burden on image processing. The camera can detect the object on the table and return the center coordinates (f_c) of the detected object. An OpenCV based filter is used to detect the object. The image coordinate error that moves the object to the center of the image plane can be stated as (5.1).

$$e(f) = f_c - f_d \tag{5.1}$$

where, $e(f)$ and f_d are image coordinate error and desired image coordinates

respectively. The desired image coordinates (f_d) are the center coordinates of the image plane (Refer Fig. 5.5).

These image coordinate errors [$e(f)$] are used to calculate the difference in hand pose ($\dot{q}$) required to center the object in image plane. The control law can be stated as follows.

$$\dot{q} = KJ^{+}e(f) \tag{5.2}$$

where, K and J^{+} are proportional constant and pseudo inverse of the image jacobian respectively. The derivation of image jacobian can be found in chapter 4 [refer (4.18)].

The required hand pose to center the object in the image plane is calculated according to (5.3).

$$q_{(t)} = q_{(t-1)} + \dot{q} \times \delta t \tag{5.3}$$

where, $q_{(t)}$, $q_{(t-1)}$, and δt are the required hand pose, current hand pose, and sample time respectively.

The joint angles needs to achieve the required hand pose ($q_{(t)}$) are the IK of the prosthesis for the required pose. IK is calculated using the ANN proposed in chapter 4. The IBVS process is continued iteratively until the object gets to the center with a tolerance of 1/3 in both image axes.

After moving the object to the center of the image plane, the IBVS stops. The image coordinates of the object center and the distance to the object at this instance is used to derive the object position w.r.t. the hand.

Calculating Hand Position (Starting Point)

The prosthetic hand position w.r.t. the shoulder (base) is calculated after the IBVS process is completed. This is the starting point of the path where the hand starts to move towards the object from this point onwards. The current shoulder angles of the stump arm are captured using the IMU attached to the stump arm. The shoulder angle capturing process is explained in detail in Chapter 3 (Section 3.2.1).

Using the current shoulder angles, position of the end of the stump arm w.r.t. the shoulder (base) is calculated by using the kinematic model of the stump arm. Prosthetic joint angles after the completion of IBVS process is used to calculate the prosthetic hand position w.r.t. the end of the stump arm. The position of the prosthetic hand w.r.t. the shoulder (T_s) is calculated using these two positions.

Calculating Object/Destination Position (End Point)

Perspective projection has been used to derive the object position w.r.t. the prosthetic hand. According to Fig. 4.3, camera can see the object, $P(x, y, z)$. The projection of the object onto the camera frame is $I(u, v)$. By equations of similar triangles,

$$x = \frac{uz}{\lambda} \tag{5.4}$$

$$y = \frac{vz}{\lambda} \tag{5.5}$$

where z and λ are distance to the object and focal length of the camera respectively. z, is measured using the US sensor and λ is a fixed parameter for the camera. Hence, the object position w.r.t. the prosthesis hand is $[x, y, z]$. The object position w.r.t. the shoulder (base) can be calculated as (5.6). The positions

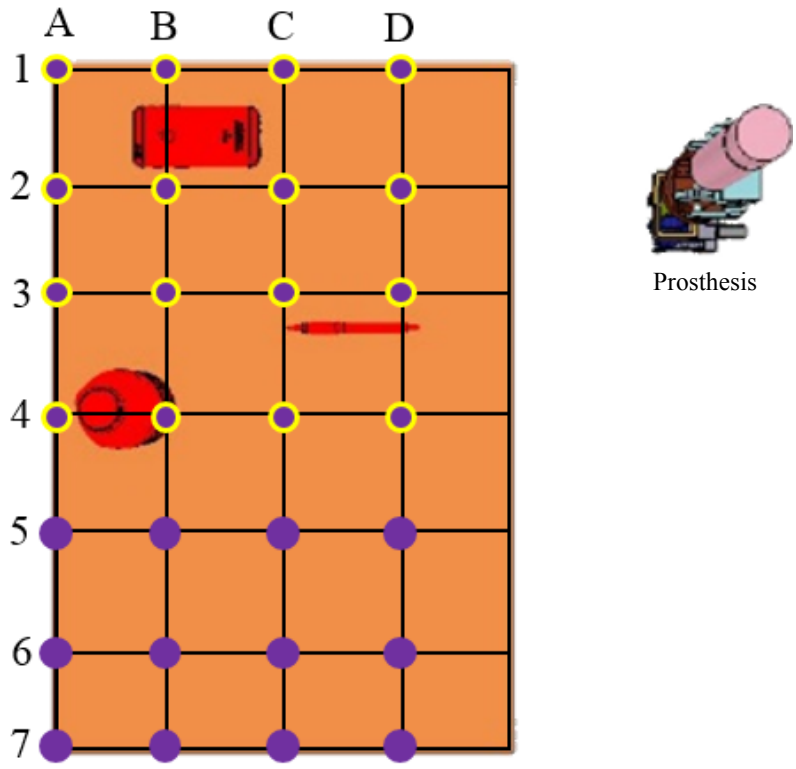


Figure 5.6: Experimental setup used to evaluate the path generation module - Plan view. Three objects: a bottle, a pen, and a mobile phone are placed on a table on 16 different positions from A1 to D4 (one by one placed on one position at a time). Path generation module create paths from the neutral position of the hand to the destination (object) position using the inputs from the camera and the US sensor fitted on the palm of the prosthesis.

are given as transformation matrices and required positions are extracted from these transformation matrices.

$$T_e = T_s \times [x, y, z, 1]^T \tag{5.6}$$

These two positions (T_e and T_s) are used to generate the path, from starting position (T_s) to the destination position (T_e). The generated path is sent to the path tracking module.

Experimental Evaluation of the Path Generation Module

Path generation capability of the path generation module is experimentally evaluated using the simulation environment. Three objects are placed on a table in 16 different positions. The paths are generated for these object positions from the neutral position of the hand. The experimental setup is shown in Fig 5.6. The objects used are a pen, a water bottle, and a mobile phone which are commonly available on a working table. Moreover, the objects are placed in 16 different positions from A1 to D4 which are circled in yellow colour in Fig. 5.6.

The starting and destination positions of the paths are found using the proposed method. The distance between the found destination positions and the nearest points on the surface of the objects are recorded using the simulation environment. These distances are depicted in Table. 5.2. As per the results, the predicted points are close to the objects with a mean less than 20mm. However, as the object size increases the distance error becomes minimum. After the hand reaches towards the object with a distance error less than 20mm, grasp planning and grasping can be done. Grasping and grasp planning are outside of the scope of this thesis.

The start and end positions of the paths generated by the path generation module for the mobile phone is depicted in Table 5.3. Object placement column shows where the mobile phone is actually placed. All positions are given relative to the shoulder (base). The end position will be a point on the surface rather than the centroid of the object. Hence, the comparison needs to be done allowing for tolerances for the object size. The tolerances are $x-axis$: +/-3.65, $y-axis$: +/-33.65, and $z-axis$:+/-69.05. According to the results, the end positions of the paths are almost similar to the position of the object with some variations along $x-axis$. This can be due to errors in measuring distance to the object (z) using the US sensor.

Table 5.2: Distance between detected positions and the object surface

Position	Phone (mm)	Pen (mm)	Bottle (mm)
1D	14.8	02.9	3.3
2D	02.4	02.1	1.6
3D	07.9	00.3	6.2
4D	03.6	-	-
1C	00.5	21.9	9.3
2C	07.8	16.5	7.7
3C	06.1	20.7	4.3
4C	01.1	15.5	2.8
1B	24.3	16.8	9.2
2B	12.4	17.0	7.9
3B	03.9	20.3	9.7
4B	07.0	39.4	7.3
1A	24.3	18.1	5.4
2A	18.7	18.7	6.9
3A	13.9	17.3	6.5
4A	12.8	25.8	0.3
Mean	10.09	16.88	5.89
SD	07.64	09.76	02.88

5.2.2 Path Tracking Module

The generated paths are tracked using a path tracking module. Two path tracking algorithms are proposed in this study: spatial path following method and MPC based path tracking method. These two methods are discussed in the subsequent sections.

5.3 Spatial Path Following Method

The prosthesis needs to synchronize with shoulder motions in order for the hand of the prosthesis to follow a given path. The shoulder motions are solely performed by the wearer. Based on the shoulder motions and the path that needs to be followed, respective poses of the prosthesis w.r.t. the end of stump arm (elbow) is derived. The prosthetic joint angles are calculated based on the derived

Table 5.3: Paths Generated by the path generator for mobile phone

	Start			End			Object Placement		
	x	y	z	x	y	z	x	y	z
1D	59.35	70.95	-286.25	254.22	137.50	-327.09	237.25	125	-325
2D	86.35	35.02	-290.60	236.52	0.450	-326.49	237.25	0	-325
3D	120.92	-15.83	-304.35	226.12	-112.81	-321.90	237.25	-125	-325
4D	135.54	-34.90	-303.64	230.50	-220.54	-317.45	237.25	-250	-325
1C	51.51	66.38	-291.94	240.49	132.52	-441.13	237.25	125	-450
2C	83.46	53.32	-294.81	248.89	-9.77	-443.69	237.25	0	-450
3C	97.84	8.16	-297.65	247.12	-130.04	-440.67	237.25	-125	-450
4C	118.87	-15.21	-301.16	238.82	-244.82	-436.89	237.25	-250	-450
1B	34.23	77.86	-289.29	209.89	121.13	-529.57	237.25	125	-575
2B	60.36	62.61	-287.08	221.78	15.59	-525.31	237.25	0	-575
3B	89.29	49.60	-278.89	245.08	-129.78	-570.41	237.25	-125	-575
4B	108.84	48.32	-272.73	248.14	-244.51	-566.27	237.25	-250	-575
1A	31.04	65.70	-279.20	209.86	122.52	-644.37	237.25	125	-700
2A	55.57	65.17	-279.58	215.53	9.79	-641.31	237.25	0	-700
3A	15.48	23.87	-295.81	220.26	-114.79	-661.30	237.25	-125	-700
4A	32.59	26.20	-295.84	253.94	-242.65	-698.14	237.25	-250	-700

pose of the prosthesis. The proposed method for path following is depicted in Fig. 5.7. The proposed path following method consists of shoulder pose estimator which is used to derive the current shoulder pose, prosthesis pose estimator which is used to derive the current prosthetic pose, shoulder pose predictor which is used to predict the shoulder pose for the next iteration, locus locator which is used to locate the current position (locus) of the prosthetic hand relative to the path, prosthesis pose generator which is used to generate prosthetic hand pose required for further advancements along the path, IK/FK generator which is used to calculate the forward and inverse kinematics of the prosthesis, and Min-Dist finder which is used to select the best possible pose and joint angles.

5.3.1 Shoulder Pose Predictor

According to the proposed method, shoulder motions for the next instance are estimated based on angular velocities and accelerations of the shoulder considering shoulder complex as three revolute joints. The angle of each joint is predicted

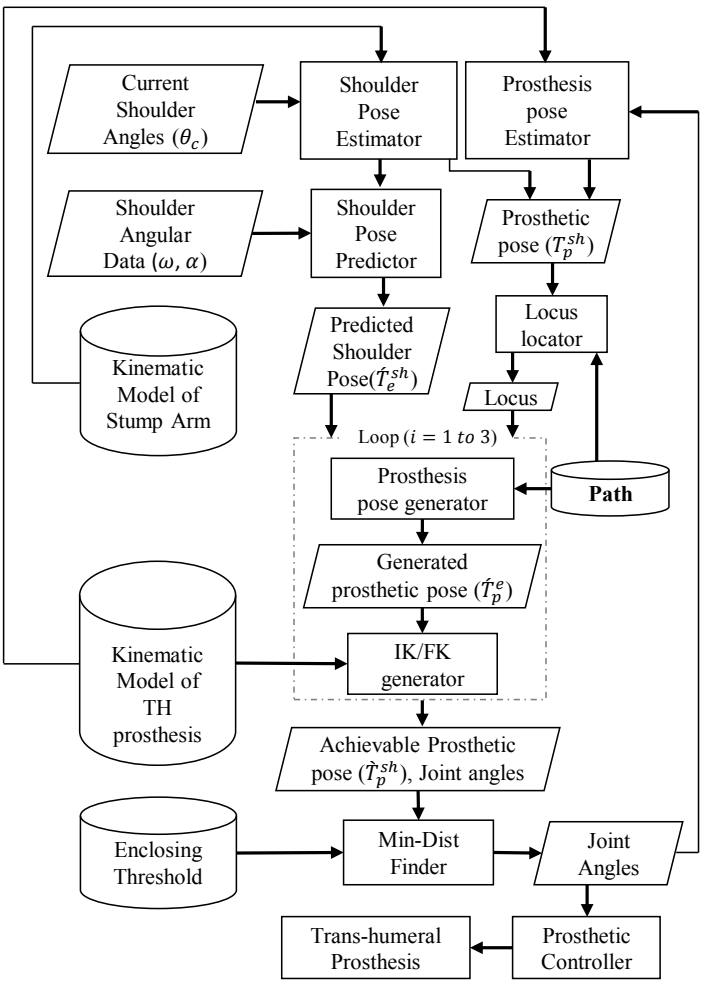


Figure 5.7: Proposed path following method

for the next instance according to (5.7) by neglecting higher order terms such as jerk and snap.

$$\theta_e = \theta_c + \omega t_s + \frac{1}{2}\alpha {t_s}^2 \tag{5.7}$$

where, θ_e, θ_c, ω, α, and t_s are predicted joint angle, current joint angle, angular velocity of respective joint, angular acceleration of respective joint, and sample time respectively.

The shoulder pose for the next instance, $\acute{T}_e^{sh}$ is derived using three estimated joint angles based on the kinematic model of the stump arm.

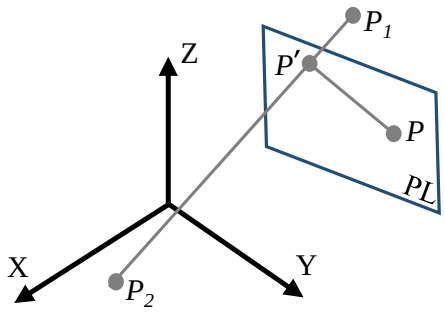


Figure 5.8: Linear path from P_1 to P_2

5.3.2 Path Following

Firstly, current pose of the prosthetic hand relative to the shoulder (T_p^{sh}) is derived. The current joint angles of the shoulder along with the kinematic model of the stump arm [Refer Fig. 3.16(c) and Table. 3.2] are used to calculate stump arm pose w.r.t. shoulder (base) (T_e^{sh}). Furthermore, pose of the prosthetic hand w.r.t. elbow (T_p^e) is calculated using the current joint angles of the prosthesis. Pose of the hand w.r.t. shoulder (T_p^{sh}) is extracted from the calculated transformation matrices using (3.7).

Next, the current position of the hand on the path needs to be derived. Locus locator is introduced in this regard. The shortest possible distance between the current hand position and the given path is considered as the distance to the path. The point on the path to which the hand has the shortest distance is considered as the current position (locus) on the path. If the path to be followed is considered to be a straight line in 3D space. The point (P') to which the path has the shortest distance is the locus on the path (see Fig. 5.8). The path from point P_1 to P_2 can be stated as (5.8).

$$\frac{x - P_{1x}}{P_{2x} - P_{1x}} = \frac{y - P_{1y}}{P_{2y} - P_{1y}} = \frac{z - P_{1z}}{P_{2z} - P_{1z}} = t \tag{5.8}$$

where, $[x, y, z]^T$, $[P_{nx}, P_{ny}P_{nz}]^T$, and t are arbitrary point on the path, point

$n(n = 1, 2)$, and time parameter relevant to the locus on the path respectively. t should be within 0 and 1.

The line connecting P and P' lies on a plane (PL) which is perpendicular to the path (P_1P_2). The direction of the path is the normal vector to the plane (PL). The normal vector can be stated as (5.9).

$$\underline{u} = \begin{pmatrix} P_{2x} - P_{1x} \\ P_{2y} - P_{1y} \\ P_{2z} - P_{1z} \end{pmatrix} \tag{5.9}$$

The equation of the plane which has point P, can be written as (5.10),

$$\underline{u}.R = \underline{u}.P \tag{5.10}$$

where, R is any point on the plane and can be treated as a point on the path. From (5.9), (5.10), and replacing R with P',

$$\begin{pmatrix} P_{2x} - P_{1x} \\ P_{2y} - P_{1y} \\ P_{2z} - P_{1z} \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} P_{2x} - P_{1x} \\ P_{2y} - P_{1y} \\ P_{2z} - P_{1z} \end{pmatrix} \cdot \begin{pmatrix} P_x \\ P_y \\ P_z \end{pmatrix} \tag{5.11}$$

where, $[P_x, P_y, P_z]^T$ is the current position of the prosthetic hand (P). By replacing $[x, y, z]^T$ using t, P_1, and P_2, the time parameter relevant to the locus (t) can be stated as (5.12).

$$t = \frac{(X.P_x + Y.P_y + Z.P_z) - (P_{1x} + P_{1y} + P_{1z})}{(X^2 + Y^2 + Z^2)} \tag{5.12}$$

where,

$$[X, Y, Z]^T = \begin{pmatrix} P_{2x} - P_{1x} \\ P_{2y} - P_{1y} \\ P_{2z} - P_{1z} \end{pmatrix} \tag{5.13}$$

The time parameter t is used to find the current locus (P'). Any possible points beyond the current locus on the path are checked for further advancements along the path. Three consecutive points related to time parameters $(t + 1/K)$, $(t + 2/K)$, and $(t + 3/K)$ are selected. In which, K is the number of steps that the path is divided into.

The prosthesis pose generator is used to generate the required pose of the prosthesis, $\acute{T}_p^e$ for aforementioned three points considering predicted shoulder pose, $\acute{T}_e^{sh}$. This can be stated as (5.14).

$$\acute{T}_p^e = (\acute{T}_e^{sh})^{-1} \times \acute{T}_p^{sh} \tag{5.14}$$

where, $\acute{T}_p^{sh}$ is a point on the path related to next movements $[(t + i/K), i = 1, 2, 3]$ along the path.

The IK for the required prosthetic hand pose ($\acute{T}_p^e$) gives the joint angles related to that pose. IK/FK generator is used to generate joint angles by taking IK. IK is calculated using the ANN [Refer Section 4.2 and Fig. 4.7 of Chapter 04]. The achievable pose ($\grave{T}_p^{sh}$) that the prosthesis can achieve relative to the required pose is derived by taking forward kinematics for the derived joint angles along with the predicted shoulder pose. Required poses ($\acute{T}_p^e$), achievable poses ($\grave{T}_p^{sh}$) and joint angles of achievable poses are sent to the Min-Dist finder (refer Fig. 5.7).

The point with minimum distance error is selected by the Min-Dist finder by comparing distances between all three pose pairs. Moreover, the maximum allowable error is defined by the enclosing threshold. If the minimum error, E_{min} is greater than the enclosing threshold, previous joint angles and pose are retained

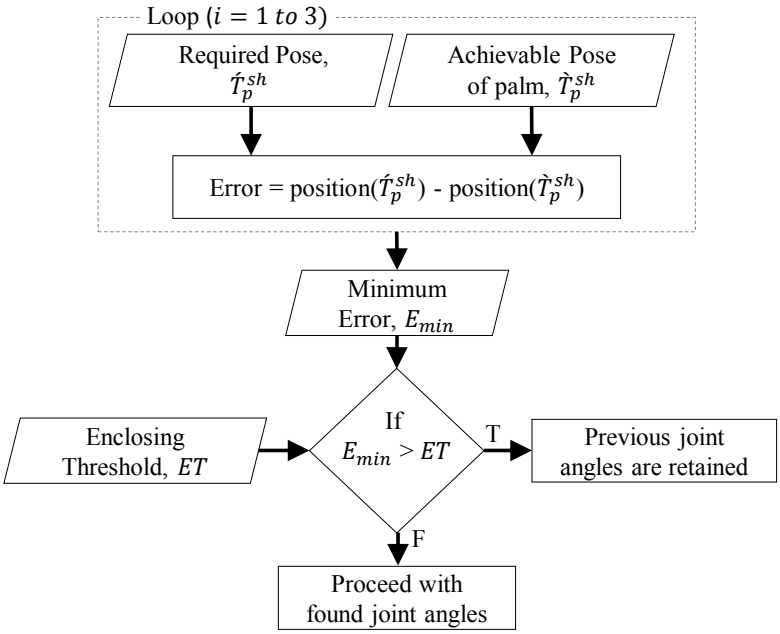


Figure 5.9: Operation of Min-Dist finder

without any changes (refer Fig. 5.9). Joint angles derived from the Min-Dist finder are sent to the prosthesis.

5.3.3 Experimental Evaluation of the Spatial Path Following Method

Several experiments are conducted to evaluate the effectiveness of the proposed path following method. The experiments are conducted using the simulation environment.

Experimental Setup

Experimental setup consists of an IMU (MPU6050), micro-controller (ATmega2560, Atmel), and a PC with V-REP and MatLab [68]. The experimental setup is depicted in Fig. 5.10. The angular motions of the shoulder are captured using the IMU. The IMU is capable of extracting shoulder FE and shoulder AAD (refer Chapter 3 Section 3.2.1). The angles are extracted using the accelerometer and gyroscope of IMU. Angular velocities, ω are the gyroscope readings around

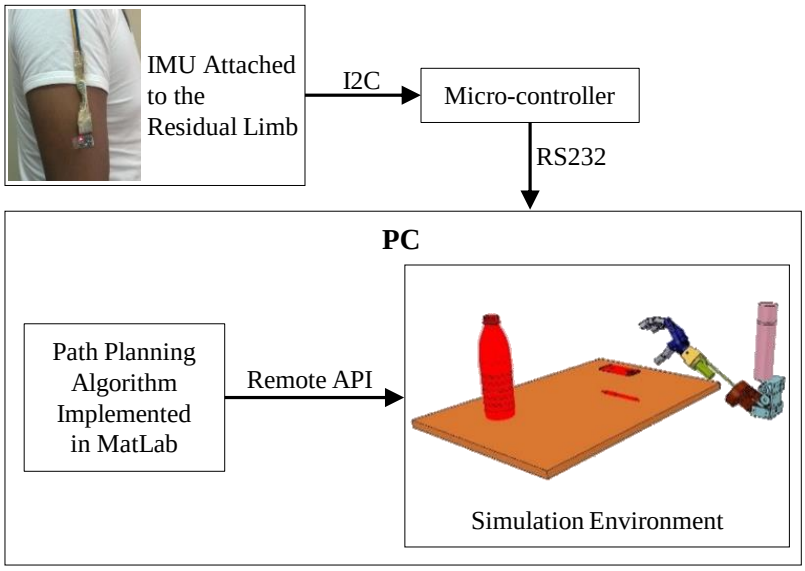


Figure 5.10: Experimental setup

respective axes of IMU, and angular accelerations, α are derived as shown in (5.15).

$$\alpha = \frac{\omega_{(t)} - \omega_{(t-1)}}{t_s} \tag{5.15}$$

where, $\omega_{(t)}$, $\omega_{(t-1)}$, and t_s are current angular velocity, previous angular velocity, and sample time respectively.

Captured shoulder data are fed into the PC using the micro-controller. Path following method is implemented using a MatLab script in the PC which calculates prosthetic joint angles based on the proposed method. Calculated joint angles are fed to the simulation environment through remote API functionality of V-REP [68].

Moreover, the paths of the prosthesis are compared with the paths of the human UL.

Table 5.4: Paths used for the experiment

Path	**Length**	**Starting point**			**End Point**		
	(mm)	x	y	z	x	y	z
1	146.5	254.35	-20.58	-327.33	321.54	-20.58	-457.48
2	189.0	88.42	-20.58	-291.90	66.36	-20.58	-479.56
3	233.6	254.35	-20.58	-327.33	191.29	86.69	-525.01
4	267.5	254.35	-20.58	-327.33	241.91	212.81	-457.48

Experimental Procedure

First, the performance of the shoulder pose predictor is evaluated. The shoulder angles captured using the IMU attached to the stump arm are compared with the predicted shoulder angles from (5.7). Predicted and actual shoulder angles are recorded.

Secondly, the path following ability of the proposed method for several paths has been tested. The experiments are conducted in two stages. In the first stage, path following ability of the proposed method without the shoulder pose predictor is evaluated. In the second stage, path following method with the shoulder pose predictor is evaluated. Table 5.4 depicts the starting and end points of the paths along with the length of the paths which have been used for the experiments. The starting and end points are stated w.r.t. frame 1 (X_1, Y_1, Z_1) as given in Fig. 3.16(c). Fig. 5.11 visualises these paths in 3D space. The points are selected by observing practical situations when reaching towards different objects. The paths used for the experiments are straight lines which are constructed based on (5.8). Moreover, the paths 1 and 2 are 2D planer paths.

The shoulder angles are given as a MatLab function when evaluating the path following method without the shoulder pose predictor. Captured human data is used to construct the MatLab function. Shoulder angles are recorded along with the prosthetic joint angles and locus while performing path following motions. Pose of the hand is calculated using stump arm and prosthetic joint angles. Hence, the path followed by the prosthetic hand is constructed.

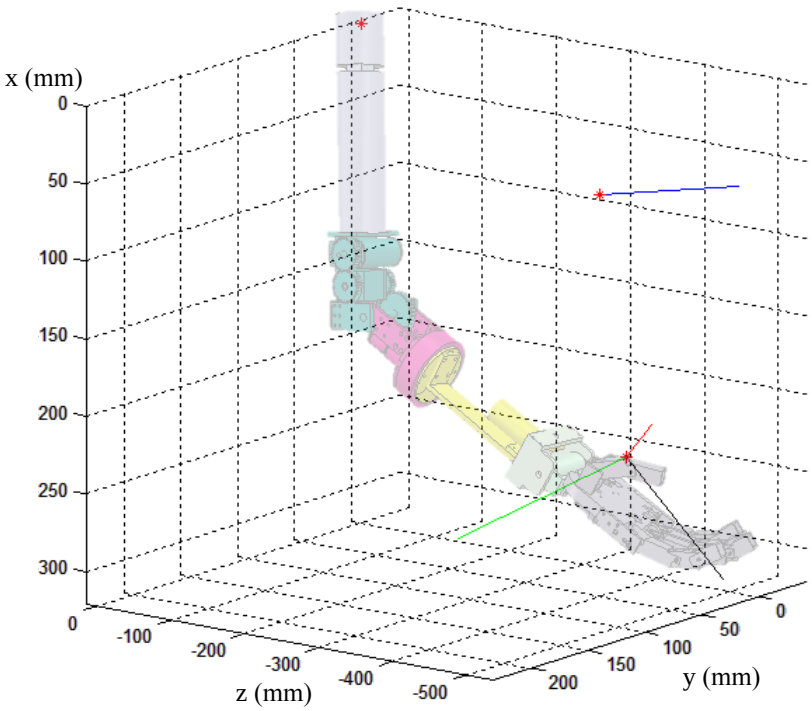


Figure 5.11: Paths used for the experiment. Path 1 is given in black colour, path 2 in blue colour, path 3 in red colour, and path 4 in green colour. Red star marks show the starting points of the paths. Paths 1 and 2 are 2D planer paths.

Velocity, V of the hand along the path is derived as (5.16).

$$V = \sqrt{{V_x}^2 + {V_y}^2 + {V_z}^2} \tag{5.16}$$

where,

$$V_k = \left(\frac{C_k - P_k}{t_s} \right) \tag{5.17}$$

where, V_k, C_k, P_k, and t_s are velocity along $k - Axis(k = x, y, z)$, k coordinate of current pose, k coordinate of previous pose, and sample time respectively.

Finally, the paths followed by the prosthesis is compared with the paths followed by the human UL [refer Fig. 5.2 and Table 5.1].

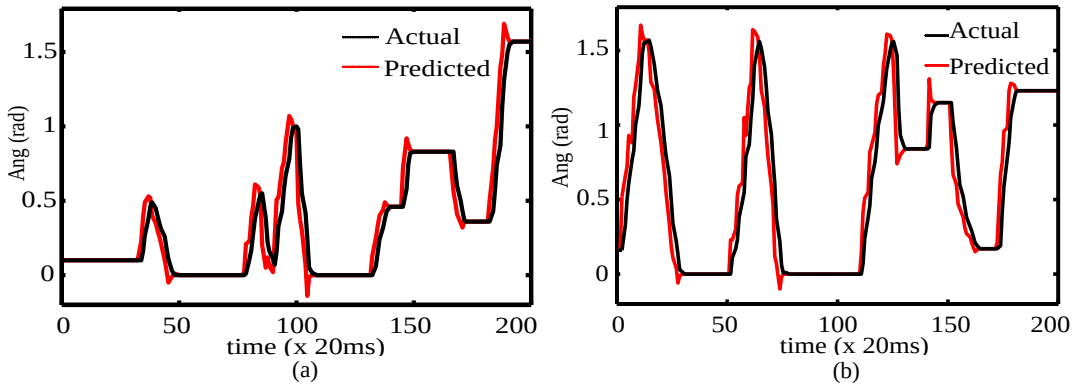


Figure 5.12: Results of shoulder pose predictor. (a) Shoulder AAD, (b) Shouder FE

Results and Discussion

In the first experiment the shoulder pose predictor is evaluated. The predicted shoulder angles are plotted along with the actual shoulder angles. Results of the shoulder pose predictor are depicted in Fig. 5.12. The shoulder pose predictor is capable of predicting shoulder AAD and shoulder FE with a RMSE of 3.67° and 5.16° respectively.

The results have some spikes and the prediction is slightly lagging behind (see Fig. 5.12). This can be due to the fact that the predictions are based on the current angular acceleration and angular velocity. Sudden changes in angular velocity and angular acceleration executed by the wearer to change the direction cannot be immediately reflected in the predictions. Moreover, the human motion intentions are reflected immediately in human motions as they are executed, where the predictor is not capable of extracting motion intentions as soon as they are generated in the human brain. Besides, the IMU is fitted on top of the skin. The changes in skin as motions are performed, are also affecting the extraction of shoulder motions.

Second experiment is performed to evaluate the path following method. Fig. 5.13 depicts the path following results. Furthermore, Table 5.5 depicts the RMSE of distance for each path followed by the prosthesis along with RMSE per unit

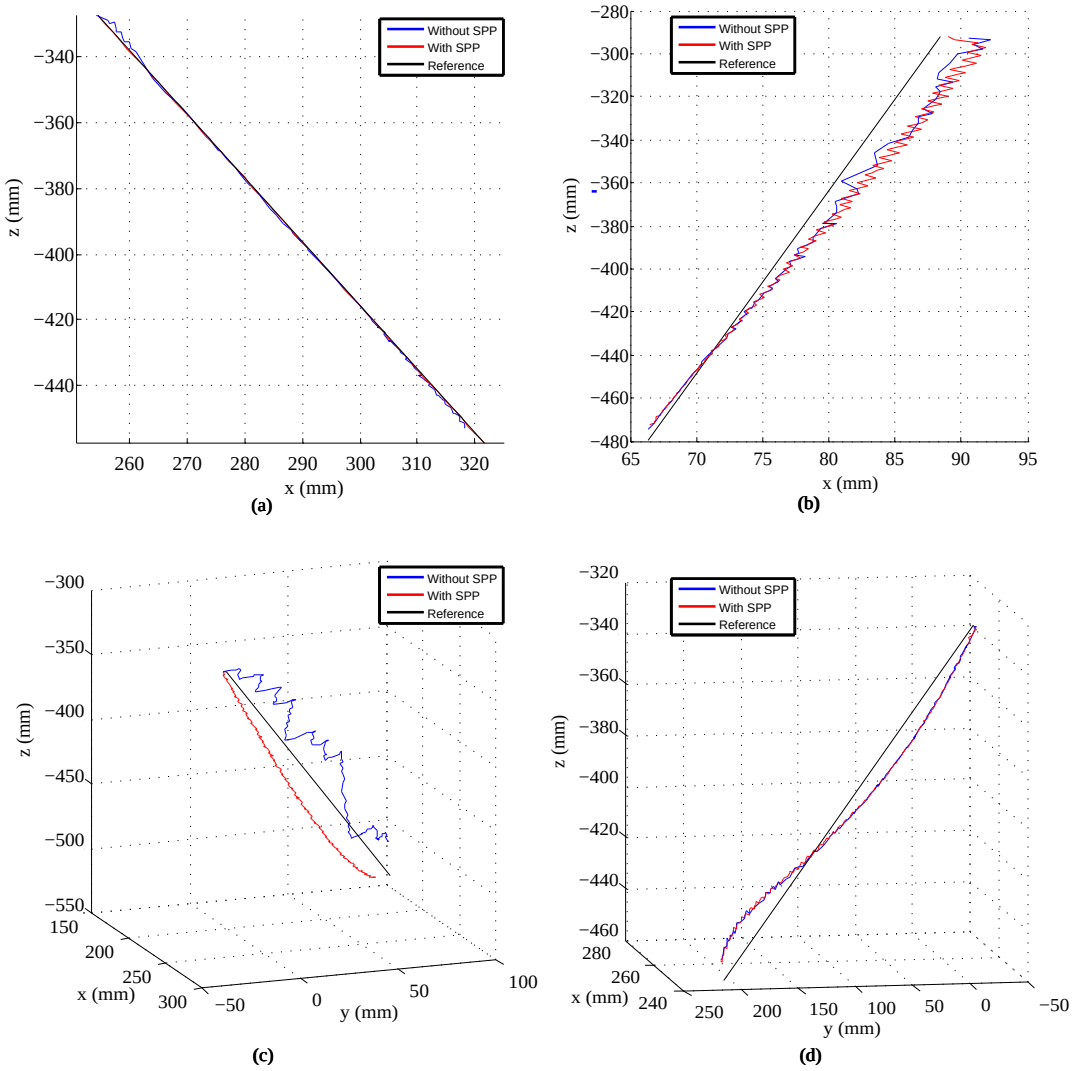


Figure 5.13: Path following results. Paths 1 and 2 are 2D planer paths hence, depicted as a 2D graph. SPP in the legend refers to the Shoulder Pose Predictor. (a) Path 1, (b) Path 2, (c) Path 3, and (d) Path 4

length of the path (RMSE/Length) for path following without the shoulder pose predictor. The distance between end points of the desired paths and the paths taken by the prosthesis are given in the last column of Table 5.5. Moreover, Fig. 5.14 shows the velocity profile of the hand when following path 1 by the prosthesis without the shoulder pose predictor. Table 5.6 depicts the RMSE of position, RMSE per unit length of path, and end point positional errors when path following is carried out with the shoulder pose predictor.

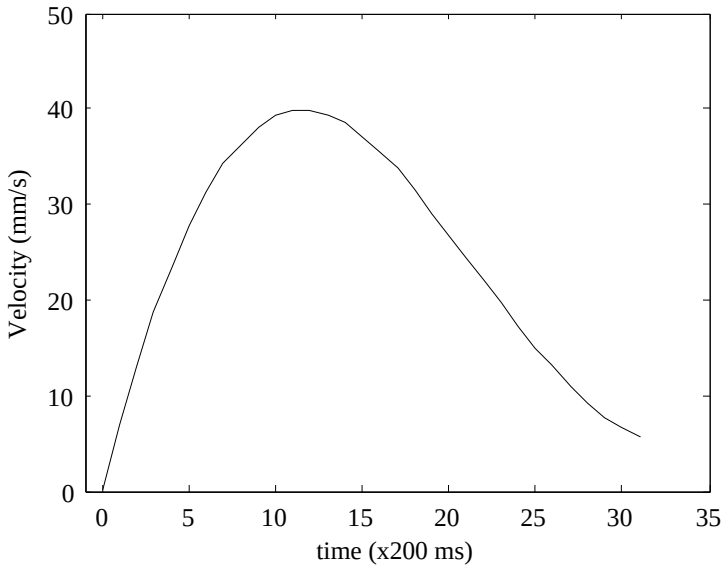


Figure 5.14: Velocity profile for the path 1 taken by the prosthesis

Table 5.5: Path following results without shoulder pose predictor

Path	**RMSE/Length**	**RMSE**	**End Points (mm)**
1	2.78 %	4.07	5.78
2	2.97 %	5.62	4.19
3	6.40 %	14.96	11.69
4	6.24 %	16.70	9.27

Table 5.6: Path following results with shoulder pose predictor

Path	**RMSE/Length**	**RMSE**	**End Points (mm)**
1	2.91 %	4.27	5.54
2	2.94 %	5.56	5.21
3	8.24 %	19.25	19.03
4	7.98 %	21.37	12.74

According to the results, the proposed method is capable of following a given path with a RMSE of less than 21.37 mm and RMSE per unit length of a path less than 8.5%. The path following results have some deviations from the original path. Errors of the shoulder pose predictor can be amplified at the prosthetic hand level causing variations. According to Table 5.5 and Table 5.6 the proposed method performs effectively without the shoulder pose predictor. The RMSE per unit length of the path has been reduced from 8% to 6%, for 3D paths.

Final experiment is the comparison of prosthetic paths produced by the proposed path following method with human UL paths. The human UL paths are not perfect straight lines and have an average RMSE of 3.10 (refer Table 5.1). Moreover, the velocity profiles (refer Fig. 5.2) are not perfectly bell shaped. From Fig. 5.2 and Table 5.1, it is clear that human paths are not perfect straight lines due to the kinematic limitations and limited joint angles of some joints. Hence, an RMSE of around 5 when following 2D paths can be considered as reasonable and human-like. The end points of the paths followed by the prosthesis are almost the end position of the desired paths with an error of around 5mm.

Moreover, the velocity profiles of 2D paths followed by the prosthesis are almost bell shaped with slightly higher acceleration at start (refer Fig. 5.14) which is almost similar to that of a human motion (refer Fig. 5.2). The slightly higher acceleration at the start may not affect the human-like nature of the paths followed by the prosthetic hand. However, the 3D paths have higher RMSEs and EPEs compared to 2D paths and these errors are slightly higher than that of the human UL motions.

5.4 Proposed MPC Based Path Tracking Method

Due to the higher errors and to overcome effect caused by the shoulder pose predictor this model predictive controller (MPC) based dynamic path tracking method has been proposed. The proposed method is depicted in Fig. 5.15 which consists of a MPC, shoulder matcher, path updater, and a Jerk remover. MPC generates an array of joint angle vectors which can take the prosthetic hand on the desired path. Shoulder matcher is used to match the current position of the stump arm with the prediction from the MPC. The joint angles for the prosthesis are selected based on the shoulder matcher. If the stump arm position has a higher variation, path updater is triggered and the reference path will be updated. The jerk remover sends the selected joint angles to the prosthesis, if

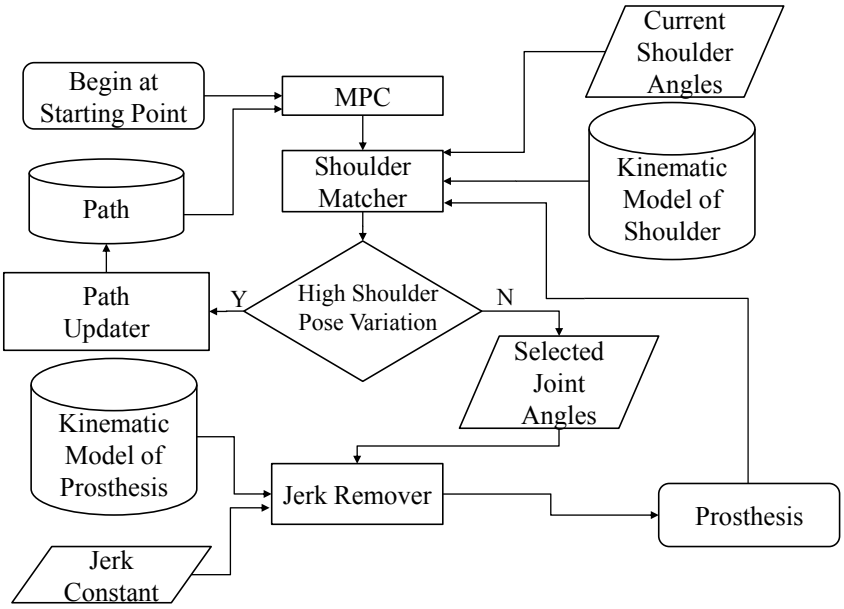


Figure 5.15: Dynamic path tracking method

the next movement is below a pre-specified error threshold. The path tracking problem is formulated using the model of the system.

5.4.1 Problem Formulation

For the hand to be taken on a reference path, all 7 joint movements from shoulder to the wrist will contribute (3 joints at shoulder, 2 joints at elbow, and 2 joints at wrist). Hence, the kinematics of the trans-humeral prosthesis and the stump arm should be considered to formulate a path tracking problem.

Prosthetic hand pose relative to the shoulder (T_p^{sh}) is a function of joint angles, q ($q1$ to $q7$) [Refer Section 3.3 of Chapter 03]. This can be stated as (5.18), where X is the pose of the prosthetic hand extracted from the transformation matrix in a vector form.

$$X = f(q) \tag{5.18}$$

Using $X(t)$ to describe the hand pose for respective joint angles of prosthesis

and stump arm at time t, the task is to find joint angles which can take $X(t)$ in a reference path $[X_r(t)]$ which can be described as (5.19).

$$X(t) - X_r(t) = 0 \tag{5.19}$$

In order to track the whole path, the problem can be expressed as below,

$$\sum_{t=start}^{end} \left\{ [X(t) - X_r(t)]^2 \right\} = 0 \tag{5.20}$$

5.4.2 Model Predictive Controller (MPC)

MPC is used to predict the future inputs (joint angles) of the prosthesis based on the current state (current pose of the prosthesis) and inputs. The shoulder joint angles cannot be controlled by the prosthetic controller, but these joint angles affect the pose of the hand. Hence, the MPC predicts all 7 joint angles ($q1$ to $q7$), including shoulder angles which makes the hand to be taken on the reference path.

Operation of the MPC used in this study is elaborated in Fig. 5.16. According to the figure the prediction of future inputs from $t+1$ to $t+p$ are done at time t. Prediction horizon is selected from the current point to the end of path since the prediction is done to optimize the entire path.

MPC uses the model of the system to obtain an optimal set of predicted inputs by minimizing an objective function. The objective function is constructed based on the system model given in (5.21) [refer (5.18)]. X, The prediction of path at time t can be stated as (5.22).

$$X(t) = f[q(t)] \tag{5.21}$$

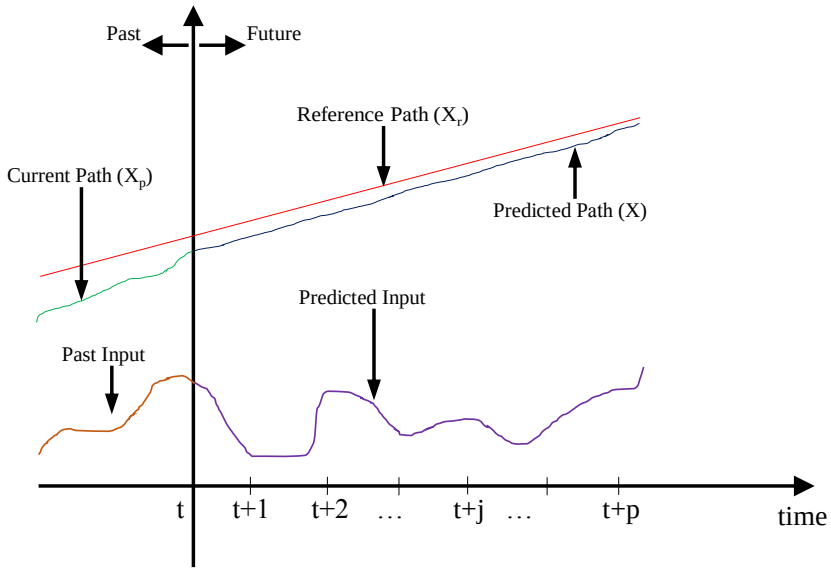


Figure 5.16: Operation of the Model Predictive Controller (MPC)

$$X(t+j+1|t) = f[q(t+j|t) + \delta q(t+j|t)] \tag{5.22}$$

where $X(t+j+1|t)$, $q(t+j|t)$, and $\delta q(t+j|t)$ are pose of the hand predicted for time $(t+j+1)$ predicted at time t, joint angles from previous prediction $(t+j)$ predicted at time t, difference in joint angles predicted at time t for time $(t+j)$ respectively.

As per the problem formulation, the error vector which needs to be minimized can be defined as follows.

$$\acute{X} = X - X_r \tag{5.23}$$

Hence, the objective function can be formulated as follows.

$$\phi(t) = \sum_{j=1}^{p} \acute{X}^T(t+j|t) Q \acute{X}(t+j|t) \tag{5.24}$$

where p and Q are prediction horizon and a constant greater than zero respectively.

Since, $\acute{X}$ is a function of q, the objective function is optimized to find the best possible joint angles, q. The optimization is done with the constraints on q.

$$q_{min} < q < q_{max} \tag{5.25}$$

where q_{min} and q_{max} are minimum an maximum joint angle vector for 7 joint angles respectively.

Resultant joint angles from the MPC is an array of q vectors. The size of the array depends on the number of samples (time points) that the path is divided into. The paths are divided into segments of 5mm length. i.e. path of 50mm in length has 10 samples.

5.4.3 Shoulder Matcher

Shoulder matcher selects the prosthetic joint angles based on the current stump arm position. IMU attached to the stump arm is used to capture current shoulder angles. Current stump arm position (P_c) is calculated using the kinematic model of the stump arm. The current stump arm position is compared with the positions generated from the predicted shoulder joint angles ($q1$ to $q3$): predicted stump arm position (P_p). The predicted stump arm position (P_p) which lies closest to the current stump arm position (P_c) is selected. The distance between these two positions (shoulder position error) are sent to the path updater. The prosthesis joint angles relevant to P_c are sent to the jerk remover.

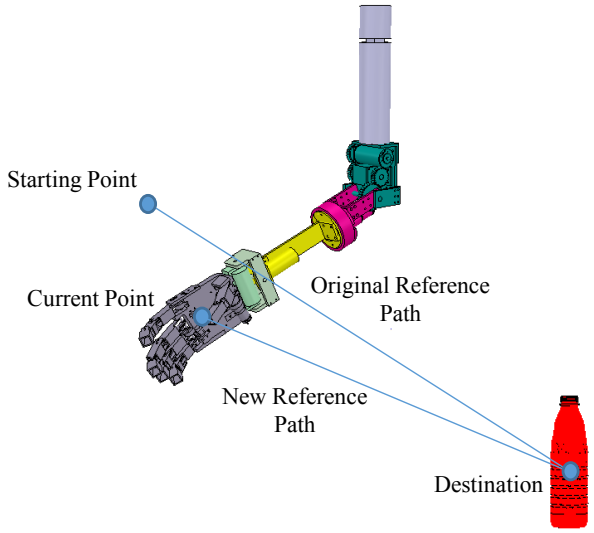


Figure 5.17: Path updating procedure

5.4.4 Path Updater

Path updater is used to update the reference path if the human stump arm motion varies significantly from the expected motion. This is measured using the shoulder position error. Hence, if the shoulder position error is above a pre-specified threshold, the reference path is updated. The new reference path is created starting from the current hand position (refer Fig. 5.17). The destination position remains unchanged.

5.4.5 Jerk Remover

The current shoulder pose and the prosthetic joint angles from the shoulder matcher are the inputs to this module. Using these inputs, the hand pose w.r.t. end of the stump arm, T_p^e is calculated. The hand pose w.r.t. the shoulder (base), T_p^{sh} is calculated as shown in (3.7). The jerk remover includes a locus locator which is used to locate the calculated pose (T_p^{sh}) on the desired path (Refer Section 5.3.2 for the detailed derivation of the Locus locator).

The distance between the locus and the calculated pose (T_p^{sh}) which is referred

to as locus error is compared with a threshold value. This threshold is given as jerk constant in Fig. 5.15. If the locus error is below the jerk constant, the prosthetic joint angles are sent to the prosthesis. If the locus error is higher than the jerk constant, the prosthetic joint angles are not changed towards the new joint angles. This module helps to eliminate the false predictions from the MPC and at the same time make the hand follow the reference path with minimum jerk.

5.4.6 Experimental Validation of the MPC Based Path Tracking Method

The proposed path tracking method is validated using several experiments. The results obtained are compared with human UL motions and the previous method. The path tracking experiments are conducted using able-bodied persons and the simulation environment.

Experimental Setup

Experimental setup consists of an IMU (MPU6050), micro-controller (Atmega 2560, Atmel), and a PC with V-REP and MatLab (Refer Fig. 5.10). Shoulder motions of the subjects are extracted using the IMU. Extracted shoulder angles are sent to the PC using the micro-controller. IMU communicates with the micro-controller using I2C protocol and micro-controller communicates with PC using serial (RS232) protocol. MPC based path tracking method is implemented on a MatLab script. The script outputs prosthetic joint angles for the current iteration by considering shoulder angles received through the IMU. These joints angles are sent to the simulation environment. PC communicates the joint angles to the simulation environment with the remote API capability of the V-REP.

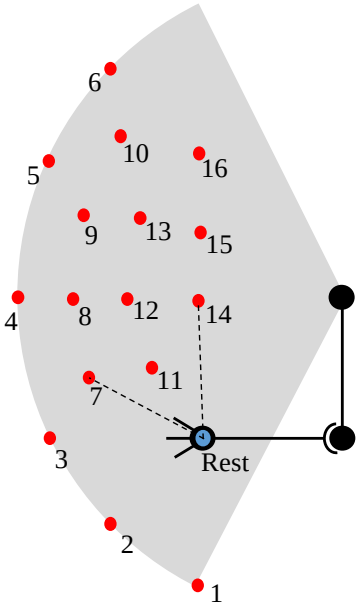


Figure 5.18: 2D paths used for the experiments. Paths are created from the Rest position to 16 different points lying in the same plane. Two paths are marked using dashed lines from the Rest position to point 7 and Rest position to point 14.

Experiments

In the first experiment the path prediction capability of the MPC is evaluated. The paths created during the path generation experiments are used in this regard (refer Fig. 5.6). The hand positions for the predicted joint angles from the MPC are calculated using the kinematic model of the prosthesis and the stump arm.

Secondly, the performance of the path tracking method for 2D planar paths has been evaluated. Sixteen predefined points lies in the same plane as the reference (rest) point (refer Fig. 5.18) are used for the experiments. Straight line paths are created from the rest point to these 16 points. The prosthesis needs to perform joint motions so that the hand moves from the rest point to a predefined point. Subjects with IMU attached to the shoulder are asked to perform the same motions. The shoulder joint angles along with the prosthetic joint angles have been recorded for all motions.

Table 5.7: 3D paths used for the experiment

Path	**Length**	**Starting point**			**End Point**		
	(mm)	x	y	z	x	y	z
1	277	197	-21	-325	156	235	-422
2	145	254	-21	-327	211	79	-423

Thirdly, performance of path tracking method for 3D paths have been evaluated. 2 randomly selected paths on the reachable workspace has been used. Starting and end points of the paths with their length are given in Table. 5.7. These coordinates are given with reference to frame 01 as shown in Fig. 3.16. As per the previous experiment, the subjects are fitted with an IMU and asked to perform the same motions. In this experiment also joint angles of the prosthesis and shoulder joint angles are recorded.

Results and Discussion

According to the first experiment, the hand poses are plotted against the reference path for the mobile phone for the object placements from C1 to C4 and D1 to D4 (refer Fig. 5.6). The prediction results are shown in Fig. 5.19 and Fig. 5.20. Summery of path prediction results are shown in Table 5.8. According to the results, prediction for the paths C1 to C3 and D1 to D3 are almost similar to the reference path. However, predictions for paths C4 and D4 deviates from the reference path when its closer to the destination position. This is as a result of the kinematic restrictions of the UL, the prediction takes the best possible path which can be kinematically achieved by the prosthesis and the stump arm. In order to reach towards these positions (C4 and D4), the human trunk also needs to perform some motions which are not considered in this study.

In the second experiment when following the paths, the hand paths of the prosthesis with their velocity profiles have been calculated using the kinematics of stump arm and the prosthesis. The RMSE of these paths w.r.t. the reference path and the EPE is calculated for all the paths. Fig. 5.21 shows the paths taken

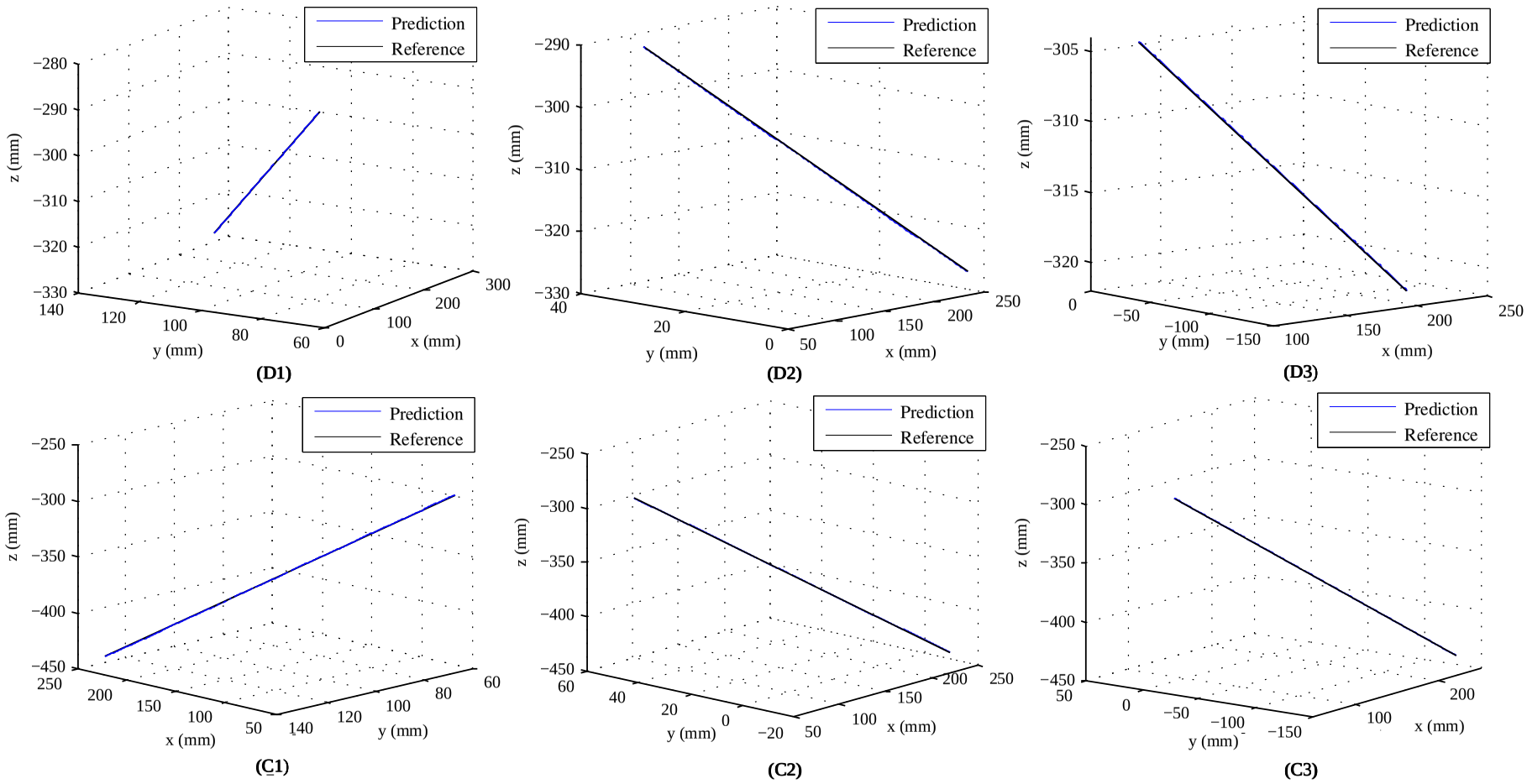


Figure 5.19: MPC Prediction results for the object placements C1, C2, C3, D1, D2, and D3

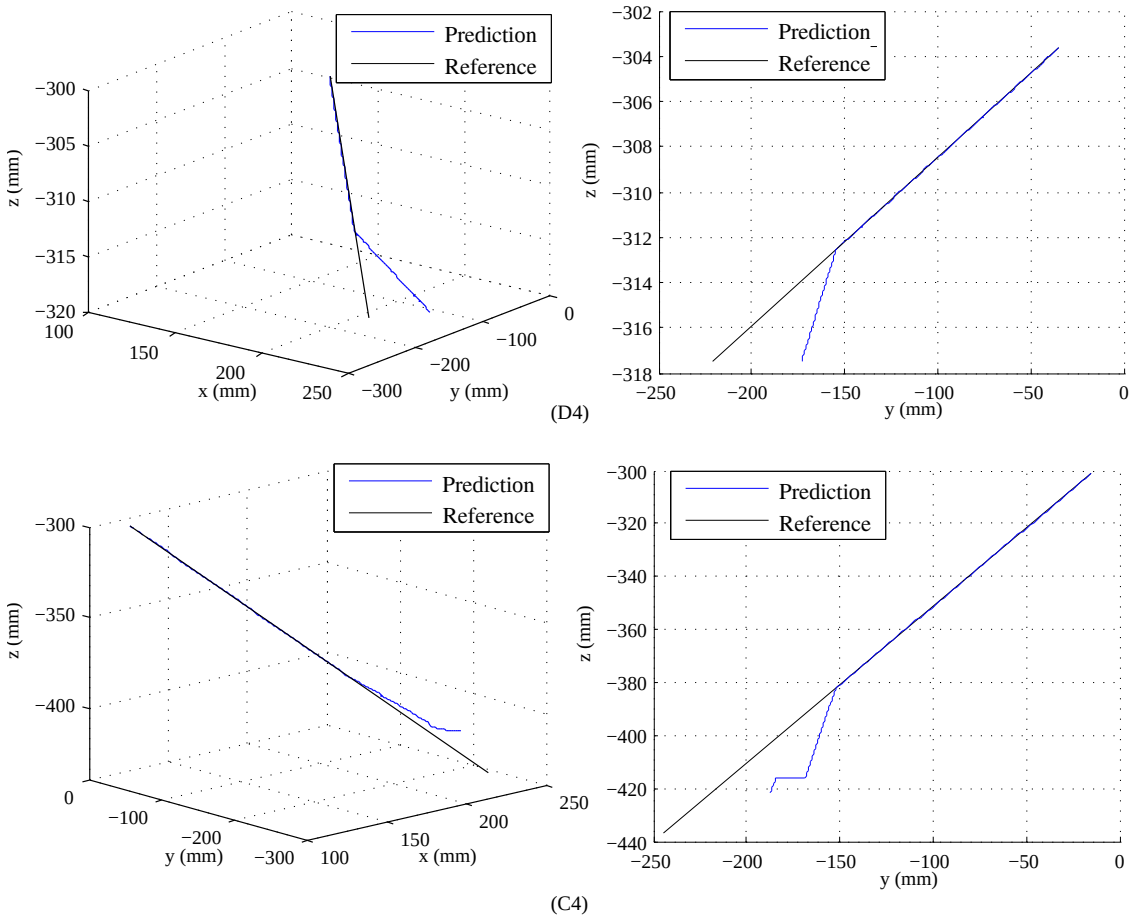


Figure 5.20: MPC Prediction results for the object placements C4 and D4

by the prosthesis and their velocity profiles for 3 different paths (paths 12, 9, and 10 respectively according to Fig. 5.18). Table 5.9 depicts the length of the paths, RMSE, and EPE for all the paths taken by the prosthesis. Moreover, the mean and the standard deviation (SD) are given at the end of the Table.

According to the results, the proposed method is capable of tracking a given path with a mean RMSE of 3.21 mm and a mean EPE of 1.92 mm. Compared to human paths (mean RMSE = 3.10 mm and mean EPE = 2.42 mm) which lies in the same range of errors. Moreover, this method outperform compared to the method proposed in the previous section. The RMSE reported in the previous method for 2D paths was 4.27 mm and 5.56 mm, also the EPE reported was 5.54 mm and 5.21 mm. Moreover, the velocity profiles resulting from the proposed method are more bell shaped compared to velocity profiles reported in

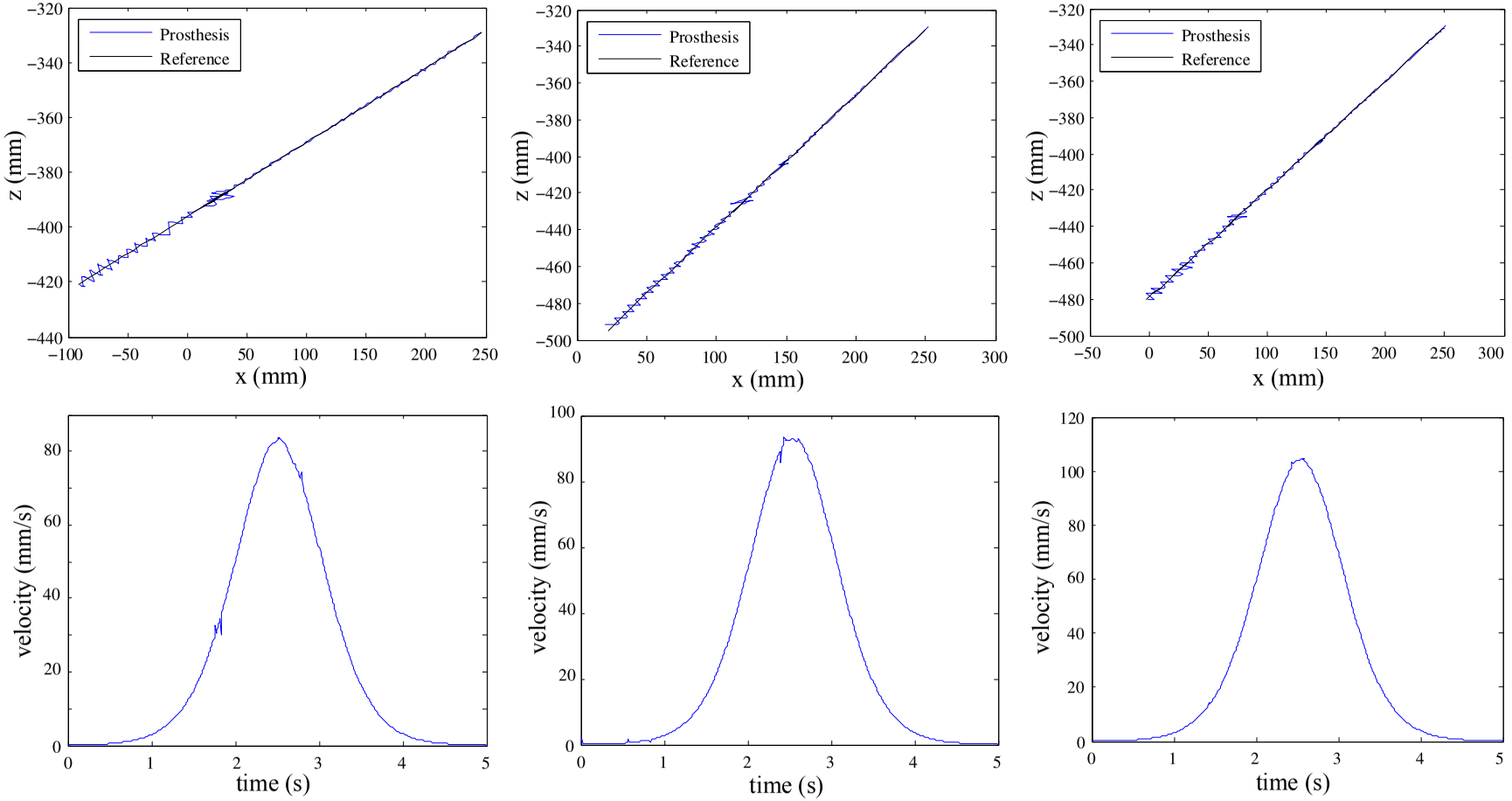


Figure 5.21: Paths taken by the prosthesis

Table 5.8: Summary of path prediction

Path	RMSE (mm)	EPE (mm)
D1	1.04	1.04
D2	0.78	0.78
D3	0.71	0.71
D4	12.88	37.04
C1	1.23	1.24
C2	1.14	1.14
C3	1.23	1.23
C4	15.49	35.74
Mean without D4 and C4	1.03	1.02
SD without D4 and C4	0.21	0.21
Overall Mean	4.32	9.87
Overall SD	5.74	15.32

Table 5.9: Summary of 2D path tracking

Path	**Length (mm)**	**RMSE (mm)**	**EPE (mm)**
1	161.37	2.77	3.4
2	138.91	2.98	1.29
3	170.36	2.98	3.25
4	322.44	4.66	1.55
5	431.75	4.09	1.61
6	476.07	3.92	3.16
7	165.52	2.23	2.44
8	280.36	3.95	3.71
9	291.89	3.42	1.03
10	424.15	3.39	0.192
11	104.50	1.21	0.58
12	248.33	3.08	2.13
13	350.13	3.34	1.8
14	177.16	2.99	1.93
15	395.08	2.96	1.17
16	449.17	3.36	1.39
Mean		3.208	1.915
SD		0.793	1.035

the previous method in which, the velocity profiles had a higher acceleration at the beginning.

Third experiment is the evaluation of tracking performance for 3D paths. The paths taken by the prosthetic hand is shown in Fig. 5.22. Prosthesis is able to

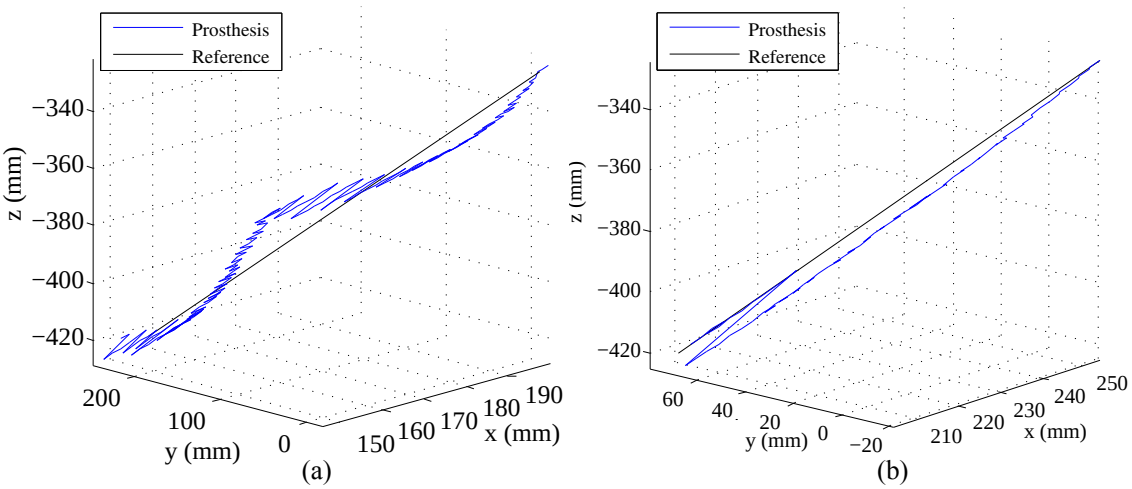


Figure 5.22: Paths taken by the prosthetic hand when following 3D paths. (a) path 1. (b) path 2.

track the given 3D paths with an RMSE of 7.31 mm and 7.76 mm respectively (Table 5.4). The EPE for the two paths are 3.05 mm and 2.45 mm respectively. Where the reported RMSE and EPE in the previous method was around 20 mm and 15 mm respectively which again outperforms compared to the previous method.

5.5 Comparison of Two Path Following Methods

Two path following methods: spatial path following method and MPC based path tracking method have been statistically compared with the human motion. The comparison has been done for the 2D paths. Path following results of two methods given in Table 5.5 and Table 5.9 are used for the comparison. These paths are compared with the human paths which are given in Table 5.1. Three T-Tests has been carried out: first one is between human path and the spatial path following method, second one is between the human path and the MPC based path tracking method, and third one is between two path following methods. Hypothesis for the T-tests has been selected as follows,

Table 5.10: T-Test results

	Human Vs. SPFM	Human Vs. MPFM	SPFM Vs. MPFM
h	H_0	H_0	H_a
p	0.2849	0.8586	0.0118

$$H_0 : \mu_1 = \mu_2 \tag{5.26}$$

$$H_a : \mu_1 \neq \mu_2 \tag{5.27}$$

where, H_0, H_a, μ_1, and μ_2 are null hypothesis, alternative hypothesis, mean of population 1, and mean of population 2 respectively.

The outcomes of the T-Tests performed at 95% confidence level are shown in Table. 5.10 in which Human, SPFM, and MPFM are referred to human paths, paths resulted from spatial path following method, and paths resulted from MPC based path tracking method respectively.

According to the T-Test, two path following methods compared to the human paths resulted in H_0 where the null hypothesis can not be rejected. Hence, there is no statistically significant difference between compared two populations. This can be concluded as two path following methods did not have statistically significant variations from the paths taken by the human.

Besides that, T-Test between SPFM and MPFM resulted in H_a. Which can say that there is a significant difference between two paths. Since mean RMSE of MPFM is less than the mean RMSE of SPFM and T-test resulted in accepting alternative hypothesis, there is a significant reduction of mean RMSE in MPFM compared to SPFM. This can be concluded as there is a statistically significant improvement in the MPC based path tracking method compared to the spatial path following method.

5.6 Summary

This chapter proposed a reach-to-grasp path planning method with two path following methods for trans-humeral prostheses. The proposed path planning method consists of a path generator which is used to generate a path towards the destination object. Path following methods are used to take the prosthetic hand on the generated path.

The path generator, first makes the object to the center of the hand using an IBVS. The hand position is derived after the IBVS process is completed. The object position w.r.t. the hand is calculated and hence, object position w.r.t the shoulder is calculated. A straight line path from the current position of the hand to the object position is generated with calculated positions.

The proposed first path tracking method is capable of following a given path compensating shoulder motions. The current position (locus) on the path is considered when predicting motions for the next iteration. The path following method is capable of adhering to the path with a RMSE per unit length of path about 2% for 2D paths with a bell shaped velocity profile. For 3D paths, RMSE for unit length of path lies around 6% when considering only the path following without the shoulder pose predictor. However, the accuracy reduced with the shoulder pose predictor leading to a RMSE per unit length about 8% for 3D paths. Hence, the shoulder angle prediction has a major impact on the path following ability of the proposed method requiring improvements.

To overcome the burden from the shoulder pose predictor and higher errors reported in the first method, the second method is proposed. The proposed second path tracking method is the MPC based path tracking method for trans-humeral prostheses to track a straight line path at hand of the prosthesis. This method was capable of adhering to the path by considering the shoulder motions which were performed by the wearer. The MPC can predict a joint angle matrix

which is able to take the prosthetic hand on the path. Shoulder matcher is used to match the current stump arm position with the predicted shoulder position derived using predicted shoulder angles, and selects the best possible joint angles for the prosthesis. The reference path is updated if the position error between the prediction and the current shoulder positions are high. This method was capable of tracking 2D paths with a mean RMSE of 3.21mm and 3D paths with a mean RMSE of 7.54mm. For human paths the mean RMSE was around 3.10mm. Therefore, the proposed path planning method is capable of generating paths towards an object of interest and follow the generated path by the prosthetic hand in a human-like nature.

Statistical results verified the human like nature of the two path following methods. Moreover, MPC based path tracking method has a statistically significant improvement compared to the spatial path following method.

Chapter 6

CONCLUSION

This chapter includes the conclusions of the thesis together with a discussion and the future directions of the study. This study addresses the issues related to controlling trans-humeral upper limb (UL) prostheses. Specifically the issues related to planning and performing reach-to-grasp motions. Two path planning methods has been proposed in chapters 4 and 5. This study was motivated by the increase in need for more reliable and functional prostheses which can mimic human-like UL motions. The major contributions of the thesis are:

- Proposal of an EMG-force proportional and moment balance (EFPMB) model for elbow motion prediction.
- Proposal of a vision aided path planning method for reach-to-grasp tasks of the trans-humeral prosthesis based on a 2-1/2D method of visual servoing.
- Proposal of a path following method for the trans-humeral prosthesis, compensating shoulder motions.
- Proposal of an improved path following method based on a MPC for the trans-humeral prostheses.
- Development of an improved vision based path planning method for reach-to-grasp motions of the trans-humeral prosthesis using proposed path following methods.

6.1 Conclusion

Most of the trans-humeral prosthetic controllers available in literature are developed to control prostheses at joint level. In those prostheses, each joint is controlled separately using biological signals such as EMG or external sensory inputs. Moreover to perform a task, each joint should be controlled separately. This study aimed at developing a task level prosthetic controller. A person may perform the same task differently which may require different motions of the joints of the prosthesis. This means the paths taken by the prosthesis varies even for the same task. Therefore giving path planning capability is required to advance research in task level prosthetic control. As a solution, this thesis propose a reach-to-grasp path planning method. The proposed method is capable of identifying the object intended by the user and reach towards the object in a way that human limbs are performing reach-to-grasp motions. A vision sensor has been used as an added sensor to identify the objects in the workspace and to extract spatial information related to the object intended by the user.

The thesis consists of six chapters: Introduction, Literature Review, The Trans-humeral Prosthesis Used to Evaluate the Path Planing Method, Reach-to-Grasp Path Planning Based on a 2-1/2D Method of Visual Servoing, Reach-to-Grasp Path Planning Based on path Tracking Methods, and the Conclusion. Contents of the chapters are briefly stipulated as follows.

First chapter explains the motivation behind the work carried out in this thesis with a brief introduction to the thesis.

Second chapter reviews the available literature about prosthetic controllers with background information about UL bio-mechanics. The bio-mechanics of the human UL has been discussed with major joint complexes and their motions. The kinematics of the human UL has been discussed with the intention of achieving human-like motions. In addition the use of biological signals in controlling pros-

theses has been discussed with their drawbacks. The use of EMG as a control input and the availability of signals for the use of controlling purposes has been reviewed. Furthermore, the use of external sensors and using vision as an added sensor has been reviewed with their merits and demerits. Besides, reach-to-grasp paths of a UL and path planning methods has been discussed with the intention of developing task level prosthetic controllers.

Third chapter explains the design and development of the simulation environment and the 5 DOF trans-humeral prosthesis. Simulation environment consists of a virtual prosthesis and a shoulder. The shoulder is capable of actuating according to a stump arm of an amputee using an IMU attached to the stump arm. The virtual prosthesis can be actuated accordingly using prosthetic control algorithms. The hand of the virtual prosthesis is fitted with a camera and an US sensor. Inputs from the camera and the US sensor have been used when implementing reach-to-grasp path planning algorithms. Fabricated prosthesis has the same design as the virtual prosthesis used in the simulation environment which actively powers 5 DOF: elbow FE, forearm SP, wrist FE, wrist URD, and compound motion of the hand. Low level controllers of the prosthesis are implemented using micro-controllers. These micro-controllers can communicate with a PC, where the prosthetic control algorithms are implemented. The prosthesis is capable of generating a range of motions (ROM) similar to a natural human UL. Moreover, the kinematics and the workspace of the prosthesis are almost similar to that of a natural limb despite some link offsets.

Fourth chapter propose the vision aided reach-to-grasp path planning method based on a 2-1/2D method of visual servoing. An EMG force proportional moment balance (EFPMB) model is proposed in this chapter. The EFPMB model is capable of controlling the elbow motion of the prosthesis effectively using EMG signals of biceps brachii and triceps brachii. The model is capable of achiving a accuracy of 92% and a real time delay less than 400ms. Moreover, the proposed method uses a 2-1/2D method of visual servoing to center the object relative to

the prosthetic hand while aligning the hand with the orientation of the object. An object reaching algorithm is proposed for the elbow joint to reach the prosthetic hand towards the object while reducing the distance to the object. Visual servoing is performed with the aid of an ANN based inverse kinematic calculator. The ANN produces promising results for inverse kinematics of the prosthesis with a Mean Square Error (MSE) of less than 0.06 in joint angles. EMG based module and the visual servoing module is integrated using a fusion filter to get an resultant elbow joint angle. Experimental results validated the effectiveness of the proposed hybrid vision based path planning method. Additionally, the path planning method was capable of converging towards the object while maintaining controllability through human motion intention.

Fifth chapter consists of the improved reach-to-grasp path planning method based on path tracking methods. This method is a extension of the previous method and used to overcome the drawbacks present in the previous method. The proposed path planning method consists of two modules: path generation module and a path tracking module. Path generation module is used to generate the required path towards the object of interest with the aid of vision feedback. An image based visual servoing (IBVS) system is used to center the object w.r.t. the hand. Then the position of the object is derived using the perspective projection. The path from the current position of the hand to the destination position of the hand is generated using these captured positions. Two path tracking methods are proposed in order to track generated paths. The first path tracking method is capable of following a given path compensating shoulder motions. The current position (locus) on the path is considered when predicting motions for the next iteration. The path following method is capable of adhering to the path with a root mean square error (RMSE) per unit length of path around 2% for 2D paths with a bell shaped velocity profile. For 3D paths, RMSE for unit length of path was around 6%. The second method is the path tracking method using a model predictive controller (MPC). This method was capable of adhering to the path by considering the shoulder motions which were performed by the wearer. The MPC

can predict a joint angle matrix which is capable of taking the prosthetic hand on the path. Shoulder matcher is used to match the current stump arm pose with the predicted shoulder pose derived using predicted shoulder angles and selects the best possible shoulder pose and angles. Prosthetic joint angles related to the selected shoulder angles are retrieved and sent to the prosthesis. Moreover, the reference path is updated if the shoulder motions are not as planned. The proposed method is capable of tracking a given path with a mean RMSE of 3.21 mm and mean EPE of 1.92 mm. Compared to human paths (mean RMSE = 3.10 mm and mean EPE = 2.42 mm) which lies in the same range of errors. Statistical analysis verified the effectiveness of the proposed path tracking methods.

6.2 Discussion and Future Directions

The work presented in this thesis addresses the issue of reach-to-grasp path planning of a trans-humeral prosthesis. The proposed path planning method is discussed for further improvements. Moreover, the future directions towards a human like prosthetic controller has been elaborated as ways of extending the work presented.

1. The fabricated 5 DOF prosthesis has been fitted on a mannequin rather than on an actual amputee. Hence, fitting methods for an amputee should be developed and tested. Moreover, the EMG acquisition system used during the research is a desktop system which is not mobile and cannot be used as a stand-alone mobile system with the prosthesis. Hence, the controllers and the EMG acquisition system needs improvements before making the prosthesis fitted on an amputee and taken out of the laboratory.

2. The proposed EFPMB model to control elbow FE using biceps brachii and triceps brachii have been evaluated using healthy subjects. However almost similar EMG signal can be expected from amputees, the proposed EFPMB

should be evaluated using amputees. The proportional gains can drastically change when evaluating the EFPMB with amputees.

3. In this research, the focus is towards the vision based path planning and visual servoing. Hence, the image processing has not been considered in depth and red coloured objects are used. The objects are identified directly from the colour itself. However, in order to apply this method on a real prosthesis, image processing algorithms should be developed to identify and differentiate real world objects. Hence, further research in the field of image processing needs to be done in identifying and differentiation real world objects that a normal human uses in activities of daily living (ADL).

4. This study is conducted only for reach-to-grasp tasks of a trans-humeral prosthesis. In order to achieve full functionality from the prosthesis other tasks such as grasping and grasp planning, manipulation of objects after grasping, and tasks which does not require grasping should also be able to be performed by the prosthesis. Hence, further research in developing task planning methods for all tasks needs to be carried out and integrated with the reach-to-grasp task planning method.

5. This study proposed path planning methods for the prosthetic hand of a trans-humeral prosthesis to be taken on a predefined path. The proposed method is capable of compensating shoulder motions and adapting the prosthetic joints accordingly. This principle can be used on any manipulator robot where the base moment is unknown and the end effector should be taken on a predefined path. However, this requires the base movements to be continuously measured. Moreover, the proposed method needs to undergo changes to be used on a robot manipulator with unknown base movements.

LIST OF PUBLICATIONS

Articles Accepted/Submitted in Refereed International Journals

1. **D.G.K. Madusanka**, R.A.R.C. Gopura, Y.W.R. Amarasinghe and G.K.I. Mann, "Hybrid Vision Based Reach-to-Grasp Task Planning Method for Trans-Humeral Prostheses," in *IEEE Access*, vol. 5, pp. 16149-16161, 2017. doi: 10.1109/ACCESS.2017.2727502

2. **D.G.K. Madusanka**, R.A.R.C. Gopura, Y.W.R. Amarasinghe and G.K.I. Mann, "Vision aided reach-to-grasp path planning of a trans-humeral prostheses based on a MPC based path tracking method," *Transaction on Human-Machine Systems (under review)*

3. **D.G.K. Madusanka**, R.A.R.C. Gopura, Y.W.R. Amarasinghe, and G.K.I. Mann, MPC based Spatio-temporal Path Tracking Method for Trans-humeral Prostheses, *IEEE Access (under review)*

International Conferences

1. **D.G.K. Madusanka**, L.N.S. Wijayasingha, K. Sanjeevan, M.A.R. Ahamed, J.C.W. Edirisooriya, and R.A.R.C. Gopura, A 3DOF Transtibial Robotic Prosthetic Limb, in *Int. conf. on Inform. and Automation for Sustainability*, Colombo, Sri Lanka, pp. 1-6, 2014.

2. **D.G.K. Madusanka**, L.N.S. Wijayasingha, R.A.R.C. Gopura, Y.W.R. Amarasinghe, and G.K.I. Mann, A Review on Hybrid Myoelectric Control Systems for Upper Limb Prosthesis, in *Moratuwa Engineering Research Conference*, Colombo, Sri Lanka, pp. 136-141, 2015.

3. **D.G.K. Madusanka**, R.A.R.C. Gopura, Y.W.R. Amarasinghe, and G.K.I. Mann, Simulation Environment for simulating transhumeral Prosthetic Control Algorithms, in *Int. conf. on Emerging Trends in Mechanical Engineering*, cochin, India, pp. 190-196, 2015.

4. C.L. Semasinghe, J.L.B. Prasanna, H.M. Kandamby, R.K.P.S. Ranaweera, **D.G.K. Madusanka**, and R.A.R.C. Gopura, Transradial Prostheses: Current Status and Future Directions, in *Manufacturing & Industrial Engineering Symposium*, Colombo, Sri Lanka, pp. 1-7, 2016.

5. **D.G.K. Madusanka**, R.A.R.C. Gopura, Y.W.R. Amarasinghe, and G.K.I. Mann, IBVS And EMG based Reach-to-Grasp Task Planning Method for a Trans-humeral Prosthesis, in *IEEE/SICE International Symposium on Systems Integration*, Sapporo, Japan, pp. 447-452, 2016.

6. R.A.M. Abayasiri, **D.G.K. Madusanka**, N.M.P. Arachchige, A.T.S. Silva, and R.A.R.C. Gopura, MoBio: A 5DOF Trans-humeral Prosthesis, in *IEEE Conference on Rehabilitation Robotics*, London, UK, pp. 1627-1632, 2017.

7. **D.G.K. Madusanka**, R.A.R.C. Gopura, Y.W.R. Amarasinghe, and G.K.I. Mann, Spatial Trajectory Following Scheme for a Trans-humeral Prosthesis, in *IEEE International Conference on Robotics and Biomimetics*, Macau SAR, China, 2017.

REFERENCES

[1] "Dictionary.com unabridged," Jan 2017.

[2] E. Vanderwerker, "A brief review of the history of amputations and prostheses," *Inter-Clinic Information Bulletin*, vol. 15, no. 5, pp. 15–16, 1976.

[3] J. Finch, "The ancient origins of prosthetic medicine," *The Lancet*, vol. 377, no. 9765, pp. 548–549, 2011.

[4] P. D. R. Project and N. R. C. U. C. on Artificial Limbs, *Report to National Research Council, Committee on Artificial Limbs: On Fundamental Studies of Human Locomotion and Other Information Relating to Design of Artificial Limbs, Covering the Period from September 1945 Through June 1947*, vol. 2. publisher not identified, 1947.

[5] N. Wiener, *Cybernetics or Control and Communication in the Animal and the Machine*, vol. 25. MIT press, 1961.

[6] R. Meier and D. Atkins, "Functional restoration of adults and children with upper extremity amputation," *Res Trends for the Twenty-First Century*, vol. 30, no. 30, pp. 353–360.

[7] L. E. Pezzin, T. R. Dillingham, E. J. MacKenzie, P. Ephraim, and P. Rossbach, "Use and satisfaction with prosthetic limb devices and related services 1," *Archives of Physical Medicine and Rehabilitation*, vol. 85, pp. 723–729, May 2004.

[8] S. Amsuss, P. Goebel, N. Jiang, B. Graimann, L. Paredes, and D. Farina, "Self-Correcting Pattern Recognition System of Surface EMG Signals for

Upper Limb Prosthesis Control," *IEEE Transactions on Biomedical Engineering*, vol. 61, pp. 1167–1176, Apr. 2014.

[9] J. Liu, "Adaptive myoelectric pattern recognition toward improved multifunctional prosthesis control," *Medical Engineering & Physics*, vol. 37, pp. 424–430, Apr. 2015.

[10] A. L. Edwards, M. R. Dawson, J. S. Hebert, C. Sherstan, R. S. Sutton, K. M. Chan, and P. M. Pilarski, "Application of real-time machine learning to myoelectric prosthesis control: A case series in adaptive switching," *Prosthetics and Orthotics International*, Sept. 2015.

[11] S. Amsuess, I. Vujaklija, P. Gobel, A. Roche, B. Graimann, O. Aszmann, and D. Farina, "Context-Dependent Upper Limb Prosthesis Control for Natural and Robust Use," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 24, no. 7, pp. 744–753, 2016.

[12] "Organization of the Human (Structure and Function) (Nursing) Part 1."

[13] C. A. Oatis, *Kinesiology: The Mechanics and Pathomechanics of Human Movement*. Lippincott Williams & Wilkins, second ed., Feb. 2008.

[14] D. C. Boone and S. P. Azen, "Normal range of motion of joints in male subjects," *The Journal of Bone and Joint Surgery. American Volume*, vol. 61, pp. 756–759, July 1979.

[15] R. Gopura, *Development and Control of Upper-Limb Exoskeleton Robots*. PhD thesis, Saga University, Japan, 2009.

[16] F. H. Martini, M. J. Timmons, and R. B. Tallitsch, *Human Anatomy*. Boston: Pearson, 7 edition ed., Jan. 2011.

[17] 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 *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pp. 4169–4175, IEEE, 2011.

[18] A. Freivalds, *Biomechanics of the Upper Limbs: Mechanics, Modeling and Musculoskeletal Injuries, Second Edition*. CRC Press, Feb. 2011.

[19] P. Corke, "A Simple and Systematic Approach to Assigning Denavit — Hartenberg Parameters," *IEEE Transactions on Robotics*, vol. 23, pp. 590–594, June 2007.

[20] P. Corke, *Robotics, Vision and Control: Fundamental Algorithms in MATLAB*. Springer Science & Business Media, Nov. 2011.

[21] A. E. Schultz and T. A. Kuiken, "Neural Interfaces for Control of Upper Limb Prostheses: The State of the Art and Future Possibilities," *PM&R*, vol. 3, pp. 55–67, Jan. 2011.

[22] D. Farina, N. Jiang, H. Rehbaum, A. Holobar, B. Graimann, H. Dietl, and O. Aszmann, "The Extraction of Neural Information from the Surface EMG for the Control of Upper-Limb Prostheses: Emerging Avenues and Challenges," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 22, pp. 797–809, July 2014.

[23] C. J. Bell, P. Shenoy, R. Chalodhorn, and R. P. Rao, "Control of a humanoid robot by a noninvasive braincomputer interface in humans," *Journal of neural engineering*, vol. 5, no. 2, p. 214, 2008.

[24] J. Kubnek, K. J. Miller, J. G. Ojemann, J. R. Wolpaw, and G. Schalk, "Decoding Flexion of Individual Fingers Using Electrocorticographic Signals in Humans," *Journal of neural engineering*, vol. 6, Dec. 2009.

[25] A. Merlo and I. Campanini, "Technical aspects of surface electromyography for clinicians," *Open Rehabilitation Journal*, vol. 3, pp. 98–109, 2010.

[26] R. A. R. C. Gopura, D. S. V. Bandara, J. M. P. Gunasekara, and T. S. S. Jayawardane, "Recent trends in EMG-Based control methods for assistive robots," in *Electrodiagnosis in New Frontiers of Clinical Research* (H. Turker, ed.), ch. 12, pp. 237 – 268, InTech, 2013.

[27] M. Asghari Oskoei and H. Hu, "Myoelectric control systems — A survey," *Biomedical Signal Processing and Control*, vol. 2, pp. 275–294, Oct. 2007.

[28] B. Hudgins, P. Parker, and R. Scott, "A new strategy for multifunction myoelectric control," *IEEE Transactions on Biomedical Engineering*, vol. 40, pp. 82–94, Jan. 1993.

[29] F. H. Chan, Y.-S. Yang, F. K. Lam, Y.-T. Zhang, and P. A. Parker, "Fuzzy EMG classification for prosthesis control," *Rehabilitation Engineering, IEEE Transactions on*, vol. 8, no. 3, pp. 305–311, 2000.

[30] A. J. Young, L. H. Smith, E. J. Rouse, and L. J. Hargrove, "Classification of Simultaneous Movements Using Surface EMG Pattern Recognition," *IEEE Transactions on Biomedical Engineering*, vol. 60, pp. 1250–1258, May 2013.

[31] J.-U. Chu, I. Moon, and M.-S. Mun, "A Real-Time EMG Pattern Recognition System Based on Linear-Nonlinear Feature Projection for a Multifunction Myoelectric Hand," *IEEE Transactions on Biomedical Engineering*, vol. 53, pp. 2232–2239, Nov. 2006.

[32] A. D. Chan and G. C. Green, "Myoelectric control development toolbox," in *Proceedings of 30th Conference of the Canadian Medical & Biological Engineering Society*, vol. 1, pp. M0100–1, 2007.

[33] L. Hargrove, G. Li, K. Englehart, and B. Hudgins, "Principal Components Analysis Preprocessing for Improved Classification Accuracies in Pattern-Recognition-Based Myoelectric Control," *IEEE Transactions on Biomedical Engineering*, vol. 56, pp. 1407–1414, May 2009.

[34] K. Englehart and B. Hudgins, "A robust, real-time control scheme for multifunction myoelectric control," *IEEE Transactions on Biomedical Engineering*, vol. 50, pp. 848–854, July 2003.

[35] M. Oskoei and H. Hu, "Support Vector Machine-Based Classification Scheme for Myoelectric Control Applied to Upper Limb," *IEEE Transactions on Biomedical Engineering*, vol. 55, pp. 1956–1965, Aug. 2008.

[36] S. Herle, S. Man, G. Lazea, and P. Raica, "Myoelectric Control Strategies for a Human Upper Limb Prosthesis," *Journal of Control Engineering and Applied Informatics*, vol. 14, no. 1, pp. 58–66, 2012.

[37] T. Lenzi, S. M. M. De Rossi, N. Vitiello, and M. C. Carrozza, "Proportional EMG control for upper-limb powered exoskeletons," in *Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 628–631, IEEE, 2011.

[38] John R. Zenie, "Prosthetic Options for Persons with Upper-Extremity Amputation," in *Orthotics & Prosthetics in Rehabilitation*, pp. 795–813, Missouri: Elsevier Saunders, 3 ed., 2013.

[39] L. Philipson and R. Srbye, "Myoelectric elbow and hand prosthesis controlled by signals from 2 muscles only, in a 9 year old girl," *Prosthetics and Orthotics International*, vol. 5, pp. 29–32, Apr. 1981.

[40] C. L. Pulliam, J. M. Lambrecht, and R. F. Kirsch, "EMG-Based Neural Network Control of Transhumeral Prostheses," *Journal of rehabilitation research and development*, vol. 48, no. 6, pp. 739–754, 2011.

[41] C. Antfolk, C. Cipriani, M. Controzzi, M. C. Carrozza, G. Lundborg, B. Rosn, and F. Sebelius, "Using EMG for real-time prediction of joint angles to control a prosthetic hand equipped with a sensory feedback system," *Journal of Medical and Biological Engineering*, vol. 30, no. 6, pp. 399–406, 2010.

[42] J.-U. Chu, I. Moon, Y.-J. Lee, S.-K. Kim, and M.-s. Mun, "A Supervised Feature-Projection-Based Real-Time EMG Pattern Recognition for Multifunction Myoelectric Hand Control," *IEEE/ASME Transactions on Mechatronics*, vol. 12, pp. 282–290, June 2007.

[43] M. Siomau and N. Jiang, "Myoelectric Control of Artificial Limb Inspired by Quantum Information Processing," *Physica Scripta*, vol. 90, Mar. 2015.

[44] T. A. Kuiken, L. A. Miller, R. D. Lipschutz, B. A. Lock, K. Stubblefield, P. D. Marasco, P. Zhou, and G. A. Dumanian, "Targeted reinnervation for enhanced prosthetic arm function in a woman with a proximal amputation: a case study," *The Lancet*, vol. 369, pp. 371–380, Feb. 2007.

[45] D. C. Tkach, A. J. Young, L. H. Smith, E. J. Rouse, and L. J. Hargrove, "Real-time and offline performance of pattern recognition myoelectric control using a generic electrode grid with targeted muscle reinnervation patients," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 22, no. 4, pp. 727–734, 2014.

[46] T. A. Kuiken, G. Li, B. A. Lock, R. D. Lipschutz, L. A. Miller, K. A. Stubblefield, and K. B. Englehart, "Targeted muscle reinnervation for real-time myoelectric control of multifunction artificial arms," *JAMA*, vol. 301, pp. 619–628, Feb. 2009.

[47] T. A. Kuiken, P. D. Marasco, B. A. Lock, R. N. Harden, and J. P. A. Dewald, "Redirection of cutaneous sensation from the hand to the chest skin of human amputees with targeted reinnervation," *Proceedings of the National Academy of Sciences*, vol. 104, pp. 20061–20066, Dec. 2007.

[48] G. Pfurtscheller, B. Z. Allison, C. Brunner, G. Bauernfeind, T. Solis-Escalante, R. Scherer, T. O. Zander, G. Mueller-Putz, C. Neuper, and N. Birbaumer, "The Hybrid BCI," *Frontiers in Neuroscience*, vol. 4, Apr. 2010.

[49] T. D. Lalitharatne, K. Teramoto, Y. Hayashi, and K. Kiguchi, "Towards Hybrid EEG-EMG-Based Control Approaches to be Used in Bio-robotics Applications: Current Status, Challenges and Future Directions," *Paladyn, Journal of Behavioral Robotics*, vol. 4, no. 2, pp. 147–154, 2013.

[50] M. C. Carrozza, A. Persichetti, C. Laschi, F. Vecchi, R. Lazzarini, V. Tamburrelli, P. Vacalebri, and P. Dario, "A novel wearable interface for

robotic hand prostheses," in *9th International Conference on Rehabilitation Robotics*, pp. 109–112, Aug 2005.

[51] A. Fougner, E. Scheme, A. Chan, K. Englehart, and O. Stavdahl, "A multi-modal approach for hand motion classification using surface EMG and accelerometers," in *Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 4247–4250, Aug. 2011.

[52] K. Ramnath, "A Framework for Robotic Vision-based Grasping Task," *Project report, The Robotics Institute, Carnegie Mellon University*, 2004.

[53] D. Klisic, M. Kostic, S. Dosen, and D. B. Popovic, "Control of prehension for the transradial prosthesis: Natural-like image recognition system," *Journal of Automatic Control*, vol. 19, no. 1, pp. 27–31, 2009.

[54] M. trbac and M. Markovi, "Stereovision system for estimation of the grasp type for electrotherapy," *Serbian Journal of Electrical Engineering*, vol. 8, no. 1, pp. 17–25, 2011.

[55] M. Gardner, R. Woodward, R. Vaidyanathan, E. Burdet, and B. C. Khoo, "An unobtrusive vision system to reduce the cognitive burden of hand prosthesis control," in *International Conference on Control, Automation, Robotics, And Vision*, pp. 1279–1284, Dec. 2014.

[56] S. Došen, C. Cipriani, M. Kostić, M. Controzzi, M. C. Carrozza, and D. B. Popović, "Cognitive vision system for control of dexterous prosthetic hands: Experimental evaluation," *Journal of NeuroEngineering and Rehabilitation*, vol. 7, no. 1, p. 42, 2010.

[57] S. Doen and D. B. Popovi, "Transradial Prosthesis: Artificial Vision for Control of Prehension," *Artificial Organs*, vol. 35, pp. 37–48, Jan. 2011.

[58] T. Flash and N. Hogan, "The coordination of arm movements: an experimentally confirmed mathematical model," *The Journal of Neuroscience: The*

Official Journal of the Society for Neuroscience, vol. 5, pp. 1688–1703, July 1985.

[59] Y. Uno, M. Kawato, and R. Suzuki, "Formation and control of optimal trajectory in human multijoint arm movement," *Biological Cybernetics*, vol. 61, pp. 89–101, June 1989.

[60] W. D. I. G. Dasanayake, R. A. R. C. Gopura, V. P. C. Dassanayake, and G. K. I. Mann, "Estimation of Prosthetic Arm Motions using Stump Arm Kinematics," in *International conference on Information and Automation for Sustainability*, pp. 1–6, Dec. 2014.

[61] C. Wang, Y. Zhao, Y. Chen, and M. Tomizuka, "Nonparametric statistical learning control of robot manipulators for trajectory or contour tracking," *Robotics and Computer-Integrated Manufacturing*, vol. 35, pp. 96–103, Oct. 2015.

[62] D. Heck, A. Saccon, N. van de Wouw, and H. Nijmeijer, "Guaranteeing stable tracking of hybrid position-force trajectories for a robot manipulator interacting with a stiff environment," *Automatica*, vol. 63, pp. 235–247, Jan. 2016.

[63] F. Bouakrif and M. Zasadzinski, "Trajectory tracking control for perturbed robot manipulators using iterative learning method," *The International Journal of Advanced Manufacturing Technology*, pp. 1–10, Mar. 2016.

[64] S. K. Kundu, K. Kiguchi, and E. Horikawa, "Design and control strategy for a 5 dof above-elbow prosthetic arm," *Int. J. Assist. Robot. Mechatron*, vol. 9, pp. 61–75, 2008.

[65] C. Piazza, C. D. Santina, M. Catalano, G. Grioli, M. Garabini, and A. Bicchi, "Softhand pro-d: Matching dynamic content of natural user commands with hand embodiment for enhanced prosthesis control," in *IEEE International Conference on Robotics and Automation*, pp. 3516–3523, May 2016.

[66] P. Helliwell, "Biomechanics of the upper limbs: Mechanics, modeling, and musculoskeletal injuries," 2007.

[67] C. C. Wei and J. F. Lin, "Kinematic analysis of the ball screw mechanism considering variable contact angles and elastic deformations," *Journal of Mechanical Design*, vol. 125, no. 4, pp. 717–733, 2003.

[68] E. Rohmer, S. P. N. Singh, and M. Freese, "V-REP: A versatile and scalable robot simulation framework," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1321–1326, Nov. 2013.

[69] D. S. V. Bandara, R. A. R. C. Gopura, K. T. M. U. Hemapala, and K. Kiguchi, "Upper Extremity Prosthetics: Current Status, Challenges and Future Directions," in *International Symposium on Artificial Life and Robotics*, pp. 875–880, Jan. 2012.

[70] P. Shenoy, K. J. Miller, B. Crawford, and R. P. N. Rao, "Online Electromyographic Control of a Robotic Prosthesis," *IEEE Transactions on Biomedical Engineering*, vol. 55, pp. 1128–1135, Mar. 2008.

[71] E. Scheme and K. Englehart, "Electromyogram pattern recognition for control of powered upper-limb prostheses: State of the art and challenges for clinical use," *Journal of rehabilitation research and development*, vol. 48, no. 6, p. 643, 2011.

[72] D. Staudenmann, I. Kingma, A. Daffertshofer, D. F. Stegeman, and J. H. van Dieën, "Improving emg-based muscle force estimation by using a high-density emg grid and principal component analysis," *IEEE Transactions on Biomedical Engineering*, vol. 53, no. 4, pp. 712–719, 2006.

[73] A. L. Hof, "The relationship between electromyogram and muscle force," *Sportverletzung Sportschaden: Organ Der Gesellschaft Fur Orthopadisch-Traumatologische Sportmedizin*, vol. 11, pp. 79–86, Sept. 1997.

[74] L. Aggarwal, K. Aggarwal, and R. J. Urbanic, "Use of Artificial Neural Networks for the Development of an Inverse Kinematic Solution and Visual Identification of Singularity Zone(s)," *Procedia CIRP*, vol. 17, pp. 812–817, 2014.

[75] E. Oyama, N. Y. Chong, A. Agah, and T. Maeda, "Inverse kinematics learning by modular architecture neural networks with performance prediction networks," in *IEEE International Conference on Robotics and Automation*, pp. 1006–1012, May 2001.

[76] B. Nelson and P. Khosla, "Increasing the tracking region of an eye-in-hand system by singularity and joint limit avoidance," in *IEEE International Conference on Robotics and Automation*, pp. 418–423, May 1993.

[77] F. Nadi, V. Derhami, and M. Rezaeian, "Visual servoing control of robot manipulator with Jacobian matrix estimation," in *RSI/ISM International Conference on Robotics and Mechatronics*, pp. 405–409, Dec. 2014.

[78] G. D. Hager, "A modular system for robust positioning using feedback from stereo vision," *IEEE Transactions on Robotics and Automation*, vol. 13, no. 4, pp. 582–595, 1997.

Appendix A

TIME DOMAIN FEATURES

Time domain features used to classify EMG signals in pattern recognition based prosthetic control systems.

A.1 Root Mean Square (RMS)

$$RMS = \sqrt{\frac{1}{N}\sum_{i=1}^{N} x_i^2} \tag{A.1}$$

A.2 Mean Absolute value (MAV)

$$MAV = \frac{1}{N}\sum_{i=1}^{N} |x_i| \tag{A.2}$$

A.3 Mean Absolute Value Slope (MAV Slope)

$$MAVS = MAV_{k+1} - MAV_k \tag{A.3}$$

A.4 Zero Crossings (ZC)

$$\{x_i > 0 \text{ and } x_{i+1} < 0\} \text{ or } \{x_i < 0 \text{ and } x_{i+1} > 0\} \text{ and } |x_i - x_{i+1}| \geq \epsilon \tag{A.4}$$

A.5 Slope Sign Change (SSC)

$$\{x_i > x_{i-1} \text{ and } x_i > x_{i+1}\} \text{ or } \{x_i < x_{i-1} \text{ and } x_i < x_{x+1}\} \tag{A.5}$$

and

$$|x_i - x_{i+1}| \geq \epsilon \text{ or } |x_i - x_{i-1}| \geq \epsilon \tag{A.6}$$

A.6 Waveform Length (WL)

$$WL = \sum_{i=1}^{N-1} |x_{i+1} - x_i| \tag{A.7}$$

Appendix B

IMPLEMENTATION OF SHOULDER CONTROLLING ALGORITHMS OF SIMULATION ENVIRONMENT (IMU TO V-REP)

B.1 Micro-controller Program

```
double acc = 9.81/16383.5; // calibration parameters
double an = 250.0/32767.0; // calibration parameters

void setup() {
Wire.begin();           // join I2C bus
Serial.begin(9600);     // initialize serial communication
accelgyro.initialize(); // initialize device
}

void loop() {
// read raw accel/gyro measurements from device
accelgyro.getMotion6(&az, &ax, &ay, &gz, &gx, &gy);

// calbration zxy
axx = ax*acc + 0.05;
ayy = ay*acc*0.98 + 1.35;
azz = az*acc - 0.4;
```

```
gxx = gx*an;
gyy = gy*an;
gzz = gz*an+3.32;

// angle calculation from accelerometer
angX = atan(axx/sqrt(ayy*ayy + azz*azz))*57.2957795;
angY = atan(ayy/sqrt(axx*axx + azz*azz))*57.2957795;

//complementary filter
angXX = 0.98*(angXX + gxx*0.000125) + 0.02*angX;
angYY = 0.98*(angYY + gyy*0.000125) + 0.02*angY;

// calculated angles
angXXX = abs(angXX + 90); // Map -90 to 90 into 0 to 180
angYYY = abs(angYY + 90);
}
```

B.2 LUA Code in V-REP Used to Control Shoulder

```
angSFE    = simGetJointPosition(shFE)*57.2957795131
angSAD    = simGetJointPosition(shAD)*57.2957795131

erSFE_p = erSFE // Previous P error
erSAD_p = erSAD

// x and y are read from the serial port
erSFE = x - angSFE              // set P error
erSAD = 0 - z - angSAD

eriSFE = eriSFE + erSFE         // set I error
eriSAD = eriSAD + erSAD

p_SFE = erSFE*kp                // P term
p_SAD = erSAD*kp

d_SFE = (erSFE - erSFE_p)*kd    // D term
d_SAD = (erSAD - erSAD_p)*kd

i_SFE = eriSFE*ki               // I term
i_SAD = eriSAD*ki

pid_SFE = p_SFE + d_SFE + i_SFE // PID term
pid_SAD = p_SAD + d_SAD + i_SAD

if (pid_SFE>0.01) then          // Set joint velocities
simSetJointTargetVelocity(shFE,pid_SFE)
elseif (pid_SFE<-0.01) then
```

```
simSetJointTargetVelocity(shFE,pid_SFE)
else
simSetJointTargetVelocity(shFE,0)
end

if (pid_SAD>0.01) then
simSetJointTargetVelocity(shAD,pid_SAD)
elseif (pid_SAD<-0.01) then
simSetJointTargetVelocity(shAD,pid_SAD)
else
simSetJointTargetVelocity(shAD,0)
end
```

Appendix C

PROSTHESIS JOINT CONTROL IN SIMULATION ENVIRONMENT USING UI

```
angEFE   = simGetJointPosition(elbowFE)*57.2957795131
angSP    = simGetJointPosition(supPro) *57.2957795131
angUR    = simGetJointPosition(ulnarR) *57.2957795131
angWFE   = simGetJointPosition(wristFE)*57.2957795131

slEFE   = simGetUISlider(manualCt,3)
slSP    = simGetUISlider(manualCt,4)
slUR    = simGetUISlider(manualCt,5)
slWFE   = simGetUISlider(manualCt,6)

erEFE   = slEFE * 150/1000 - angEFE
erSP    = slSP  * 180/1000 - (angSP  + 180)
erUR    = slUR  * 080/1000 - (angUR  + 40 )
erWFE   = slWFE * 180/1000 - (angWFE + 90 )

if (erEFE>2) then
simSetJointTargetVelocity(elbowFE,erEFE*0.1)
elseif (erEFE<-2) then
simSetJointTargetVelocity(elbowFE,erEFE*0.1)
else
simSetJointTargetVelocity(elbowFE,0)
end
```

```
if (erSP>2) then
simSetJointTargetVelocity(supPro,erSP*0.1)
elseif (erSP<-2) then
simSetJointTargetVelocity(supPro,erSP*0.1)
else
simSetJointTargetVelocity(supPro,0)
end

if (erUR>2) then
simSetJointTargetVelocity(ulnarR,erUR*0.1)
elseif (erUR<-2) then
simSetJointTargetVelocity(ulnarR,erUR*0.1)
else
simSetJointTargetVelocity(ulnarR,0)
end

if (erWFE>2) then
simSetJointTargetVelocity(wristFE,erWFE*0.1)
elseif (erWFE<-2) then
simSetJointTargetVelocity(wristFE,erWFE*0.1)
else
simSetJointTargetVelocity(wristFE,0)
end
end
```

Appendix D

IMPLEMENTATION OF LOW LEVEL PID CONTROLLERS OF THE PROSTHESIS

Implementation of Low Level PID controller in micro-controller used to control supination/pronation motor. all other DC motor controller are implemented in the same way except for hand motor.

```
void setup(){                  //initializing
pinMode(Motorline11,OUTPUT);  // Set output pins to H-bridge
pinMode(Motorline12,OUTPUT);
pinMode(lmtSt,INPUT);          // set input pin for limit switch
EncM1.write(0);                // Initialize encoder
Limit = digitalRead(lmtSt);
Minit();                       // function to initialize joint
Serial.begin(9600);            // initialize serial communication
}

void loop(){
if (Serial.available() > 0) { // read serial port for joint angle
inByte = Serial.read();
}
position1 = (uint8_t)inByte;   // set joint angle from serial comm.
EncreadP1 = EncM1.read();      // read encoder for current joint angle
setP1 = position1*180.0/255.0; // mapping the positions from serial
```

```
nP1 = EncreadP1*360.0/2580.0;  // mapping the position from encoder
erPP1 = erP1;                  // Previous proportional error
erP1 = (setP1-nP1)*255.0/180.0;// Proportional error
erD1 = erP1-erPP1;             // Derivative error
erI1 += erP1;                  // Integral error
pTerm1 = kp1*erP1;             // P of PID
dTerm1 = kd1*erD1;             // D of PID
iTerm1 =  ki1*erI1;            // I of PID
pidTerm1 = (int)(pTerm1 + iTerm1 + dTerm1); // calculated PID value

if(pidTerm1>255){              // fit PID value into limits
pidTerm1 = 255;
}
else if(pidTerm1 <-255){
pidTerm1 = -255;
}

if(pidTerm1 > 1){              // foreward rotation of motor
Mfwd(pwm1,Motorline11,Motorline12,pidTerm1);
}
else if(pidTerm1 < -1){        // backward rotation of motor
Mrev(pwm1,Motorline11,Motorline12,-pidTerm1);
}
else {                         // motor stop
Mstop(pwm1,Motorline11,Motorline12);
}

}
```

```
// motor initialization function
void Minit(){
while (Limit == 0)
{
Limit = digitalRead(lmtSt);
Mrev(pwm1,Motorline11,Motorline12,50);
}
EncM1.write(0);
}

// motor reverse function
void Mrev(int EN , int IN1 , int IN2 , int spd){
analogWrite(EN , spd);
digitalWrite(IN1 , LOW);
digitalWrite(IN2 , HIGH);
}

// motor foreward function
void Mfwd(int EN , int IN1 , int IN2 , int spd ){
analogWrite(EN , spd);
digitalWrite(IN1 , HIGH);
digitalWrite(IN2 , LOW);
}

// motor stop function
void Mstop(int EN , int IN1 , int IN2){
digitalWrite(EN, LOW);
digitalWrite(IN1,HIGH);
digitalWrite(IN2,HIGH);
}
```

Appendix E

MATLAB SCRIPTS USED IN 2-1/2D VISUAL SERVOING BASED PATH PLANNING METHOD

E.1 IBVS Implementation

```
% u and v are the current image coordinates
% zz is the distance to the object

if(~isempty(tt0) && tt0(3)~=0) % update errors if got an image frame
ex  = u - 127.5;
ey  = v - 127.5;

% image feature jacobian
jaco =  [  fl/zz 0 -u/zz -u*v/fl (fl^2+u^2)/fl -v;
           0 fl/zz -v/zz -(fl^2+v^2)/fl u*v/fl u ];

% psudo inverse of jacobian
invJaco = transpose(jaco)*inv(jaco*transpose(jaco));

qDot = gK*invJaco*[ex;ey];
qPos = qPos + qDot;
errs = net(qPos);               % calculate joint angles from NN
% send angles to v-rep
```

E.2 PBVS Implementation

```
% (l1x, l1y) and (l2x, l2y) are coordinates of the
% mid points of the bounding rectangle

if(l1y>l2y)         % find the point at the top
yy = l1y - l2y;
xx = l1x - l2x;
else
yy = l2y - l1y;
xx = l2x - l1x;
end

ang = atan2(xx,yy); % misaligned angle
if(abs(ang)<0.04)   % if ang is small
ang = 0;
end

% orientation correction due to t5 and th
pu = -sin(t5);
pv = 0;
pw = cos(t5);
th = ang;

Tr = [ pu^2+(1-pu^2)*cos(th) pu*pv*(1-cos(th))-pw*sin(th)
       pu*pw*(1-cos(th))+pv*sin(th) 0;
       pu*pv*(1-cos(th))+pw*sin(th) pv^2+(1-pv^2)*cos(th)
       pv*pw*(1-cos(th))-pu*sin(th) 0;
       pu*pw*(1-cos(th))-pv*sin(th) pv*pw*(1-cos(th))+pu*sin(th)
       pw^2+(1-pw^2)*cos(th) 0;
       0 0 0 1];
```

```
if(th~=0)          % if th is not zero, correct the orientation
qq = qq*Tr;
end

% qPos is the pose extracted from qq

errs = net(qPos); % camculate joint angles from ANN

% send angles to v-rep
```

E.3 Reaching Algorithm Implementation

```
if(zz>50 && abs(zz-zzOld)>1 && th==0)

if(state == 1 && errs(1) > gd)
 errs(1) = errs(1) - gd*(zz-zzOld);
 if(errs(1)>0 && errs(1)<146)
  % send errs(1) to V-REP
 end
 if(zz<zzOld)
  state = 1;
 elseif(zz>zzOld)
  state = 2;
 end
end

if(state == 2 && errs(1) < (146 - gd))
 errs(1) = errs(1) + gd*(zz-zzOld);
 if(errs(1)>0 && errs(1)<146)
  % send errs(1) to V-REP
 end
 if(zz<zzOld)
  state = 2;
 elseif(zz>zzOld)
  state = 1;
 end
end
```