Authorization

Kinematics of Robot-Human Upper Body Interaction in Virtual, Augmented and Real Environment

by

Omid Heidari

A dissertation

submitted in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy in the Department of Mechanical Engineering

Idaho State University

December 2019

Committee Approval

To the Graduate Faculty:

The members of the committee appointed to examine the thesis of OMID HEIDARI find it satisfactory and recommend that it be accepted.

....
Alba Perez Gracia, PhD, Associate Professor
(Major Adviser)

....
Marco P. Schoen, PhD, P.E., Professor
(Co-Adviser)

....
Anish Sebastian, PhD, Assistant Professor
(Committee Member)

....
Nina Robson, PhD, Assistant Professor
(Committee Member)

....
Alexander Urfer, PT., PhD, Professor
(Graduate Faculty Representative)

This dissertation is dedicated to my dear mom, Sodabeh.

Acknowledgments

I would like to thank my adviser Professor Alba Perez Gracia who has given me unparalleled direction, support, and guidance throughout my education and dissertation. Above all else, I thank her for believing in me even when I didn't believe in myself. Many thanks to the Department of Mechanical Engineering at Idaho State University for providing me with limitless opportunities to pursue my areas of interest. I pay special thanks to Professor Marco Schoen who not only offered me professional support but helped to guide me through the whimsical twists and turns of academic life. I would also like to acknowledge and thank members of the Measurement and Control Engineering Research Center; especially Dr. Abhijit Makhal whose ambition and professional breadth helped to spark my interest in many fascinating areas of robotics. I am thankful to Dr. Yimesker Yihun whose knowledge in kinematics greatly improved my understanding of robotic exoskeletons. My thanks to Professor Alex Urfer whose expertise in Physical Therapy and Physiology greatly enhanced my knowledge and understanding of medical and rehabilitation robotics. I express my deepest and most heartfelt appreciation to my dear parents and siblings for their support, words of encouragement, motivation, and love from afar. Finally, I thank my wife Andrea for tolerating many late work nights and always supporting and encouraging me.

This work is supported by the National Science Foundation under Grant No. CBET-1403688 and Idaho Global Entrepreneurial Mission (IGEM) grant 2019. The content is solely the authors' responsibility.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Kinematics of Robot-Human Upper Body Interaction in Virtual, Augmented and Real Environment

Dissertation Abstract–Idaho State University (2019)

A correct kinematics model for human upper limb that gives all the information about the workspace and relation between joints and links at any configuration is a fundamental step for human motion integration in real/augmented environment. This information can also be used for designing external tools that can improve human capabilities. The augmented environments include a spectrum of research areas such as exoskeletons, medical robots, prosthetic limbs, robotic manipulators and humanoids. This dissertation aims at studying the upper limb of human body in terms of kinematics and using this knowledge to construct, augment and simulate the limb counterparts in real and virtual worlds. This work begins with the description of fundamental issues in kinematics of human upper limb which includes the joints types and numbers as well as the workspace and its dimension. By point synthesis approach, a parallel robot to simulate the human shoulder complex workspace is suggested. Following a task-based approach to design a mechanism to simulate shoulder complex motions, a method to design exoskeleton as a systematic way is introduced. This work investigates the possibility of the use of virtual reality (VR) in rehabilitation in order to find out the effects of an immersing environment consisting of intractable objects on healthy and unhealthy patients. Moreover, a VR application to do dimensional synthesis for serial robots and hand structures is created. This enables the robot designers to modify the design parameters in real-time so a better model is achieved. As the last part of this research, AR is going to be investigated in an attempt to bridge between Robotics and augmented world where the users can perceive the robot's state, task and information in a safer, faster and natural way. The outcome of this research will suggest a model for human upper limb with a more precise mechanism for shoulder complex. The development of new solutions in virtual reality may provide a comprehensive platform for robot designers as well as an immersive environment for physical therapy purposes. Finally, the assessment of the work done and the description of the research to be completed are included.

Key Words: Kinematics, Human Shoulder Complex, Parallel Manioulators, Virtual Reality, Exoskeleton, Augmented Reality.

Chapter 1: Introduction

The goal of this research is to create an optimal methodology to identify, simulate and augment human limb motions, consisting of mathematical tools and programming environments. This is a key aspect of the development of physical and virtual human-machine interaction, from exoskeleton to immersive avatars. Having an incomplete or inaccurate model of this interaction leads to sub-optimal solutions and in some cases rejection by humans of the combined systems.

The primary difficulty of development of such methodology is modeling the human joint correctly. Due to soft contacts or even in some cases no contact between bones, human joints are not necessarily known in robotics realm which yields in estimating their types and degrees of freedom. Capturing precise data of limbs motions is another source of complexity which exacerbates the difficulty. Another problem is adjusting the constructed mechanism to different human anatomical parameters. In addition, human body constraints puts some restrictions to the design as there is always a chance of invading or intersecting the human body. On the other hand, in virtual medium, the proposed model needs to respond fast and accurately to the changes of its surroundings. Interacting with objects through virtual limbs needs a fast, real-time and realistic physics engine in the virtual world. Finally making a model that is not limb-specific and can be used as a template for other parts of the human body is necessary and at the same time challenging.

In this dissertation, some of these issues are going to be addressed and an applicable model will be introduced. In the first part, the focus is on the analysis of upper-limb anatomically and kinematically including the joint complexes of the hand, arm and shoulder. The arm has been considered as a serial chain of revolute joints in most of the references and a common link connecting couple of serial chains is simply the popular mechanism among the researchers to model the hand. But the shoulder complex is one of the most sophisticated regions of the human body which provides vast ranges of motions due to its redundant structure while serving a pivotal role in all arm movements. Despite the difficulty of the problem, researchers have been trying to find a mechanism which mimic the shoulder workspace. Having a correct kinematic model of the upper-limb gives the opportunity to explore any possible synergy among joints values where the relations between joints is studied. The methodology utilized throughout this dissertation is to focus more on the workspace of an specific biological mechanism instead of its anatomy to design correspondent robotic systems. This perspective leads to a task-based design method where robotic systems are designed for specific tasks and workspace.

The second part of this dissertation is dedicated to investigating the applications and uses of virtual reality and augmented reality technology in rehabilitation robotics, exoskeletons, kinematics synthesis and industrial robots. The goal of the author is to bridge bewteen two new technologies, i.e. robotics and VR/AR. VR is aimed to develop and test a novel system for the training of the human arm of post-stroke patients. Fast, accurate and customized modeling and embedding of the arm kinematics is essential for successful perception and training which can be gained in an immersive environment with interactive virtual objects. AR also can improve the interaction between human and robots by adding extra information and objects to the real world explaining robot information and intentions as well as programming and commanding desired tasks and motions.

1.1 Human Upper-Limb Kinematics Model

We can divide human upper-limb complex into three different systems: Shoulder, arm and hand. The workspace and the degrees of freedom (DOF) of the human shoulder has always been a difficualt question to answer among researchers. A minimal 4-DOF parallel linkage with spherical and translational joints has been considered in [IEF$^+$13]. Based on [LS03], the shoulder is a 5 DOF mechanism containing a universal joint with a slider along the clavicle which is dependent to the univeral joint plus the GH pair with 3 DOFs. Having the shoulder analyzed anatomically and biomechanically, Thomas et al., [TMM05] studied the glenoid motion relative to torso which yields 2 DOFs perpendicular sliders. This model was simplified into only z trnaslational direction in [JB13]. In [Ton05], a 3-DOF planar mechanism was claimed to mimic scapulathoracic articulation. A serial manipulator with two revolute joints and one prismatic (RRP) all perpendicular on each other is modeled in [KTL07] to provide depression/elevation and retraction/protraction of the shoulder.

GH is kinematically and anatomically accepted as a spherical pair by many researches due to its ball-and-socket shape and functionality since [CE14] and approved later by [HSC$^+$90] and [Vee00]. However, there is a small relative motion, about a few millimeters, between the humerus head and the glenoid which contradicts the assumption of being a spherical joint ([MBP+12], [MRL+14], [CCK+14], [TPLC+16]). In this work, we consider this relative motion as part of the system (black box) and focus on the location of humeral head representing the location of GH joint. As a matter of fact, the duty of the shoulder complex is locating the humeral head. If we can find a mechanism that can produce the same locations for humeral head, then we meet our goal.

On the other hand, elbow and wirst are not as complicated. The anatomy and workspace of elbow and wrist have led many researchers to choose two revolute joints ([BRBFM00], [KMB+12], [BRZ+16]). In this dissertation also, we consider the flexion-extension of the elbow and the pronation-supination of the forearm as two non-intersecting revolute joints. Their particular location and relative orientation should be calculated by the dimensional synthesis process. Similarly, at the wrist, the flexion-extension and radial-ulnar deviation as two intersecting revolute joints. The kinematic synthesis for elbow and wrist for spatial robots is well developed as they are considered serial chains; however, less work has been done so for robotic hands. The design of robotic hands has been attempted in some different ways [MSVR10], from underactuated hands [Oea14], [QSNS14] to anthropomorphic hands [CCC14]. In particular, the design of single underactuated fingers has drawn more attention, see [RAS14] and [CZ15]. The design of multi-fingered robotic hands for simultaneous tasks of the fingertips has been extensively studied in [HPG16], [TPGP16] and [SSPG14a]. Other approaches include workspace optimization [BD15] for multi-fingered robotic hands.

After having a correct model for the human mechanism, an interesting topic is to find the synergy in different motions created by that mechnism. Synergy is a topic of interest as it helps us understand the limb motions better which can be used for analyzing stroke subjects and malfunctioned limb. Stages of recovery from stroke can progress from flaccidity, to uncontrolled movement within basic limb synergistic patterns, to gradual gain of control of these movements, then progression out of these stereotypic compulsory patterns to a more functional motion [Bru66]. After the recovery phase, some motion synergies are observed in the patients, typically a flexor or extensor synergy. Motion synergies have also been observed in healthy patients. The difference seems to be that, while the healthy patient is able to perform a richer variety of motions by combining primitives or isolating degrees of freedom, it is not possible for some stroke patients to decouple the synergies. These synergies have been observed repeatedly in patients [O'S14], however the characterization of those as compared to the synergies obtained in healthy patients using model reduction tools [WF04] has not been done before. Given a kinematic model for the limb, human synergies have been widely observed ([BGS11], [GSMP13], [MGE01], [SFS02]). In particular, human synergies, also called motion primitives, have been widely studied for human grasping using PCA analysis ([WF04], [GSMP13], [CHJ+14]).

Regardless of the methodology used in the design process, topology or synergy, the foremost step is to collect the correct data of human body motions. Human body motion capture is used for many applications such as character animation, sports and biomechanical analysis. It focuses on simultaneously estimating the relative position and orientation of the different body segments and estimating the absolute position of the body. To have the motion captured, different set of reflective markers should be placed at key points on the body and the motion of them are tracked using a set of high-resolution cameras around the subject. This is a vision-based technology [KHS14] which can sense human motions and is usually referred to as optical motion capture. If properly marked, it is possible to capture the motion of each degree of freedom or a group of degrees of freedom at each joint. Optical motion capture systems ([SPS+11], [LPM]) are among the most widely used in the industry today. Vicon™ is one of the most popular optical motion capture system among researchers and commercial applications. These systems are popular due to their accuracy; their major disadvantages are cost, portability, and intrusiveness. Optical systems require indoor setups that typically cost between tens and hundreds of thousands of dollars. Having the Cartesian coordinates of the markers, triangulation is used to recover the 3D position of these markers in space which are used to fit a skeletal model to the observed motions. Because of different sources of noises in the captured data, filtering and smoothing are essential parts of having clean data. [Wol86] is one of the original work for filtering and smoothing where n noisy data points is determined from the data by means of the Generalized Cross-Validation (GCV) or predicted Mean-Squared Error (MSE) criteria.

1.2 General Task-Based Methodology

Extending the concept used in finding the right mechanism to mimic shoulder motions, exoskeletons in general can be designed by a task-based approach where the workspace or a subspace of that is considered. Exoskeleton attach directly to a human to augment the abilities of the user. Some of the application include medical monitoring and intervention, strenuous and repetitive work, dangerous jobs and military missions ([FA06], [RGM+07], [SG10]). For the purpose of industrial and medical application, robotic exoskeletons were studied in the late 1960s and 1970s ([Mos67], [Clo65], [SS73]). Exoskeletons were also designed to enhance the strength of humans ([BK90], [KM91]). Currently, many exoskeleton robots are proposed/designed for the rehabilitation, haptic interaction, and power augmentation purpose ([KKF03], [GO06], [IKTC14], [BVZ+15]). For more details of recent robotic exoskeleton developments and applications see [Bog15]. An ideal exoskeleton should generate natural motions within the workspace of the human limb without causing vibrations or sudden motion changes and without adding extra load or burden on the user. All these considerations make the design of robotic exoskeletons difficult. The prevailing notion with exoskeleton research and development is that ideally they should be able to reproduce every motion that the human operator is capable of. The common approach in exoskeleton design is to attempt to align each robotic joint axis with its human counterpart (e.g. a rotational joint for the elbow)([CLZ+14], [KYH+14]). For example, the surmised ideal upper extremity exoskeleton should have the same workspace as the human arm. To this end, researchers continue to increase the degrees-of-freedom (DOFs) of their exoskeletons. In the case of the human arm, excluding the hand and fingers but including the wrist, this is 7 DOFs (3 at the shoulder, 1 at the elbow, 1 at forearm, 2 at wrist). Adding the hand and fingers adds additional DOFs. As the DOFs increase, so does the complexity, size, weight, and cost of the exoskeleton. Furthermore, good alignment is often difficult and the distances between joints must be adjustable to accommodate the variances of human limbs.

A major practical challenge to the comfort and usability for exoskeletons is the need to avoid misalignment of the exoskeletal joints with the corresponding human joint. Alignment disparities are difficult to prevent due to large inter-user variability, and can create large stresses on the attachment system and underlying human anatomy([SKS+14], [CBHG16]). An alternative to the rigid exoskeleton is the *soft* exoskeleton, where mechanical joint axes are omitted and the human limb itself serves as the mechanical structure. Actuators are attached between limb segments to augment the strength of the human subjected. This produces a lighter and less obstructive exoskeleton, but its premier application may be fatigue mitigation; in the case of human augmentation, either for rehabilitation or industrial applications, power amplification requires excessive joint and bone loading. For some human joints, such as the elbow, the motion of the joint can be fairly accurately reproduced with a common robotic joint (in this case the revolute, or rotating joint). In other cases, however, such as movement of the shoulder through scapula and clavicle articulation, the human body follow a complex motion that difficult to reproduce with combination of revolute and prismatic joints. Traditionally, exoskeletons are designed so that they try to align with the human joint axes of motion ([BKB10], [BJ04], [KH02]). This assumes that the location of the axis can be accurately known, and that such a fixed axis exists for the range of motion of the joint or set of joints, which is not always the case. A clear example of complex kinematic modeling is the thumb, for which precise detection methods such as MRI segmentation ([SvdS10], [Rus12]) show that considering fixed rotational axes, especially for the CMC joint, is not a good approximation; see also [CGWH10]. Similarly, the human shoulder follows a complex motion that its center of rotation changes with its motion [KKF03] which makes the alignment joints of the human with the exoskeleton more difficult as the location of complex human joints changes.

One of the hypothesis of this work is that for many applications, a complete recreation of the human workspace may be unnecessary and suboptimal. In fact, a reduced, sub-workspace may be the best solution. For rehabilitation of the upper extremity, for example, it is likely that a properly designed robotic exoskeleton could achieve a large percentage of clinically desired motions with a greatly reduced workspace. In the proposed design methodology, the process is divided into three stages. The first stage uses motion capture to record the kinematics associated with the desired task or set of tasks. The complexity of the data is then reduced through optimization to a workspace that properly characterizes the desired task(s). The goal is to accurately represent the design motions with an optimal set of joints and actuators. This may be thought of a curve-fitting (regression); we are looking for the proper type and number of basis-functions (joints, in this case) to reproduce the desired motions without over fitting the data. The second stage uses dimensional kinematic synthesis in order to create an articulated system able to follow a specified motion [HRPGK16]. This stage defines, given the type and number of joints and the loops of the mechanism, the relative position between the joints; this specifies the workspace of the mechanism. Several methods exist for the dimensional kinematic synthesis of linkages. Geometric constraints imposed by the joints can be used to define design equations [MS10]; robot kinematics equations to reach a set of positions can be stated and solved for both the joint variables and the structural variables ([LM04], [HDV14]). In our research, we follow [PGM06]. It is important to notice that any dimensional synthesis method used for the second stage can be used to provide the input data (the joint axes and their connectivity) for the third stage. The third stage deals with the optimization of the links to satisfy a set of performance requirements. Many of these additional performances, such as motion smoothness, obstacle avoidance, force transmission, or physical dimensions to name a few, are fully or partially independent of the kinematic task. The optimization stage has been successfully developed, implemented and tested in several mechanism designs. It is a general method that can be used to optimize different topologies; such as serial chain, closed linkages, linkages with tree structure and hybrid mechanisms. The output from the optimization algorithm is used for CAD implementation. This helps to have a 3D visualization and simulation of several candidate solutions. The CAD model also used to check the response of different actuators and their placement in the mechanism.

1.3 Virtual Reality for Rehabilitation

Recent research has shown that VR environments enhance recovery after spinal cord injury [VBK+13], and that action-observation training improves upper-limb function in children with unilateral cerebral palsy [SFC+13]. utilizing this technology, VR, will allow us to exploit the visual link between action-perception and action-production and develop more effective retraining protocols in controlled clinical environments. Human imaging work (PET, fMRI) has revealed a mirror-neuron network (pre-motor cortex, parietal lobe, temporal lobe) that supports our ability to learn through action imitation and action observation ([FFG+05], [CMGG+04], [CCF11]). This direct link between human visual perception and human action execution is diminished [PBFH05] or disappears when non-anthropomorphic motion is observed (see [UMCA06], [TB08], [OMRP07]). Research over couple of past years has started to investigate how action-observation protocols linked to the mirror neuron system may benefit recovery of function after stroke and enhance clinical training protocols. Some promise regarding the use of action-observation as a means to tap into the mirror neuron system in the clinic have come from training protocols that use video to help patients mimic activities of daily living [ESS+07] and virtual reality systems that transfer the motion of the patient's real arm to a set of virtual arms in real time ([ESP+07], [PHB12]).

1.4 Virtual Reality in Robotics System Design

A virtual reality environment for robotic design is desirable for several reasons related to human perception of spatial positions and spatial motion. Among those reasons spatial task definition, candidate selection and performance assessment are to be highlighted in this work. Defining a task with spatial (3D) motion is difficult for the designer because we lack the graphical aid tools that are available for planar motion. This difficulty of defining three-dimensional motion tasks for rigid bodies is well known [KVL02]; as a consequence, a couple of techniques have become popular. Teach pendant is the task definition process in which the robot is manually guided through the task and the motion is being recorded as robot joint angles. In this area, progress is being made in the use of virtual and augmented reality [CPG+15]. Sensing the desired motion as performed by a moving element is done using motion capture with infrared cameras. This method provides realistic sets of data points for human motion; however, noise from sensors and other problems such as relative movement of marker with respect to the underlying bone due to skin deformation need to be addressed. Also the excessive richness of the data should taken to account. Small motions in the many degrees of freedom of the human body are captured even when the subject is trying to perform a single motion [VUPGK06]. Those mini-motions need to be identified in the dataset and possibly eliminated. Another issue is the dealing with excess of information, commonly with hundreds of data points per seconds. Using all those points leads to a costly optimization process, while in downsampling we may be missing key parts of the motion while keeping uninteresting segments.

Usually the number of solutions that can be created for a given task is an open set, and ranking all possible designs is costly and many times lacks objective meaningful measures. Limiting the search space for designs is complicated because of the very nonlinear nature of the problem, which makes the selection of this search space difficult. VR provides an interactive and natural environment to do this, see [BV17]. Finally, once a design is selected, its performance needs to be assessed. VR offers the possibility of a quick and intuitive assessment of the robot performance. Automatic environment modeling using VR is used for teaching by demonstration of grasping actions [AC07], and for mechanism motion [BCM07]. VR was applied in kinematic design in [KVL02] and [LVK02].

1.5 Augmented Reality for Robot Interaction

Collaboration among team members in different activities elementally relies on how much each individual can perceive and anticipate other team members' behavior and action [KFBW05]. Robots are becoming more and more popular in different aspects of industrial applications. For safety purposes, robots are usually working in cells and isolated spaces where no human can enter without stopping the process. However, if we have a clear, precises collaboration and communication with the robot, then we can add that to the team as if it is a teammate. This will increase the performance and outcome of the whole team. In human-human interactions, many signs are used to convey the intentions but in robot-human interactions, there are less available cues to predict and collaborate. In recent work, researchers have tried to add some features to improve the robot-human interaction; in [DLS13], legible motions were introduced and investigated to have clear robot intentions. Expressive motion primitives were developed in [SMF14] and projector-based systems has been used in ([AMMB16], [CAKL15], [WIM⁺15])to provide extra information about the robot and the system. In [BRV16], light signals were used as explicit cues. The main goal is to have a clear interaction with robots where the other teammates or users can feel comfortable to work with and around the robots. Augmented reality technology as it is obvious by its name augments the real world by virtual objects so we, human, can have a better understanding of our surroundings. Therefore, AR can be a perfect platform to augment the perception of the robots for human and users. AR has been considered for human-robot interactions for aerial robot in [WHLS18].

Moreover, there are situations where because of hazardous materials, burden of the work or high precision but small autonomy, a robot that can be remotely operated is needed. As a result, teleoperarion is another paradigm for human-robot communications. Some of the applications are medical and surgerical purposes ([NMG04], [TMFD08]), aerial robots [PRC+17] and space exploration [MLMON07]. A great deal of research has been done on human performance issues with regard to robotic teleoperation interfaces, mixed teleoperation and supervisory control systems [CHB07]. Teleoperating robot and information exchange between human and robots through AR has been considered in prior work ([DCMP03], [GBCC08], [MZDG93], [RBTB16], [TRK+17]) . Recently it has been verified that using AR significantly improves objective measures of teleoperation performance and speed while reducing crashes [HWS18].

Augmented reality seems to have a lot of capabilities in different respects; collaborative environment, teleoperating, reducing the gap between simulation and implementation by enabling the prototyping of algorithms, system information, assembly and manufacturing process ([MKM+16], [MKKM16], [HMS+15], [EGL16]). In this dissertation, the author aims at investigating the possibilities of AR uses in Robotics in general and in particular creating an AR platform for ABB robots where both the users and designers can enhance their interaction with robots.

1.6 Research Goals

In the previous sections, we have seen that modeling human joints is a challenging process because of soft contact between bones. Also capturing the data of human limbs is another difficulty due to skin motion that creates unwanted relative motions in the markers data. Furthermore, modeling the human limbs based on how they look and their anatomy does not create working and satisfying results. On the other hand, emerging technology like VR and AR are to be discovered for the possibility of creating a more natural and intractable relations between human and robots. This dissertation aims to address these challenges; so the goals of this research is structured in the following objectives:

- Find a correct kinematic model of human upper limb including shoulder complex to identify its motions.
- Expand the human upper limb capabilities by introducing task-based dimensional kinematics synthesis for exoskeletons.
- Implement virtual version of human upper limb for rehabilitation and physical therapy.
- Utilize virtual model of human upper limb for robotics design and programming.
- Enhance human upper limb capacities by Augmented Reality for robot-human interaction purposes.

The following chapters describe the research already done and the proposed research to complete the goal.

Chapter 2: Kinematics of Human Upper-Limb

Human upper limb consisting of shoulder complex, arm and hand are studied in this chapter. The aim is to understand the anatomy of theses systems first, and then see how the motions of each of these parts can be captured or calculated if they are physically unreachable. The next step is the workspace of these limbs so that we can find comparably good candidates (mechanical mechanisms) whose DOF and space in which they work in is similar. In the end, having the correct kinematics model of the upper limb, we can try to find any possible synergy among shoulder, arm or hand joints.

2.1 Human Upper Limb Anatomy

The first part of human upper limb is the shoulder girdle. This complex is one of the most complicated parts of the human body which provides sophisticated motions due to its redundant structure. It plays a significant role on hand and arm motions and must be analyzed kinematically correctly to have the modeled robotic arm working properly as the real one.

The shoulder girdle is a set of joints that can provide complex motions due to its combination of serial and parallel structures. It comprises three bones, clavicle, scapula, and humerus, attached serially to the thorax as a base. These bones are connected by the sternoclavicular, acromioclavicular, and glenohumeral joints, as shown in Figure 2.2.

Clavicle contains two curves (Figure 2.3); the first one is 2/3 of the whole length, medial, which has a convex curve and the other one is the lateral on the remained 1/3 of the length having a concave curve if it is seen from front view. This bone is the only one that stays naturally horizontally in the human body. The lateral end of this bone articulates with the acromion process of scapula and the medial end connects to the sternum. In medial side, clavicle is like a rounded shaft whereas on the both ends, it is flattened. Clavicle carries some part of the weight of upper limb and increases the functional efficiency of the upper limb greatly because it is laterally away from the body.

The humerus is the biggest bone in the upper limb (Figure 2.4, on the left). Humerus movements are vital for different tasks of the arm such as typing, holding, pushing and lifting. In terms of structure, the humerus functions as a link between the scapula and the elbow. At the elbow joint, it links to the two bones in the lower arm. One can divide the humerus into three segments. The rounded head

Figure 2.1: Human Shoulder.

Source: www.menshealth.com

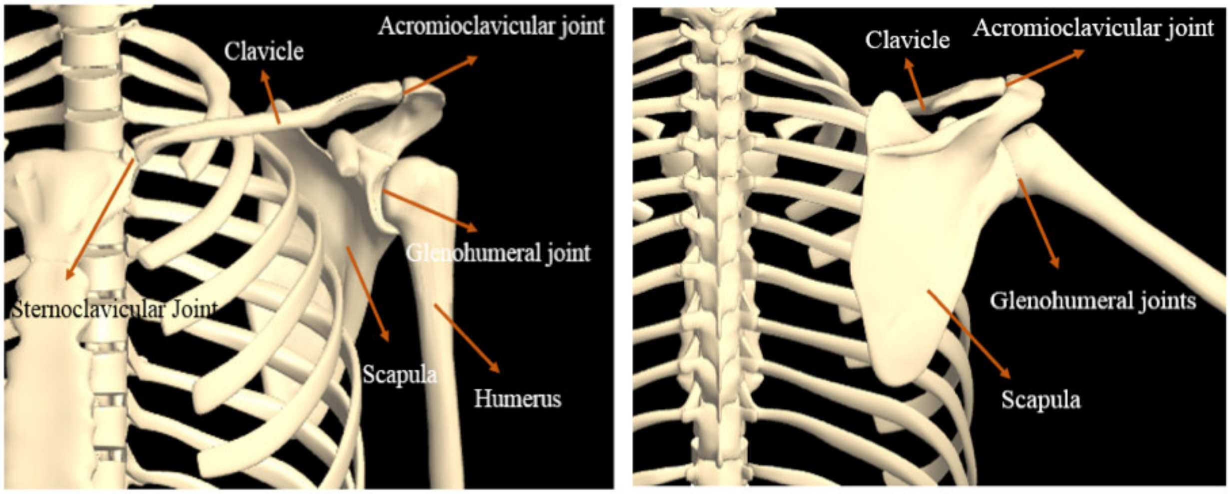


Figure 2.2: Human Shoulder Girdle. (left) front view. (right) back view.

Source: Constructed with *Kineman*

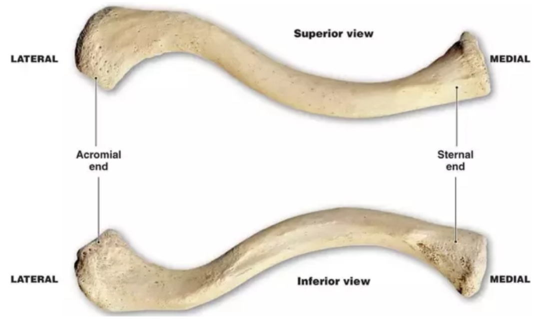


Figure 2.3: Clavicle.

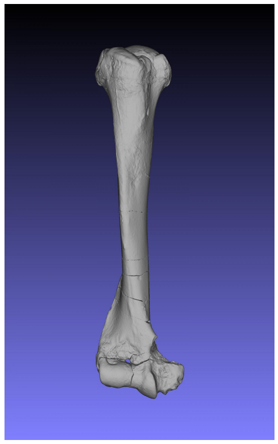
Source: wp.stu.ca

at the top, the shaft of the humerus which has a cylindrical shape and its bottom part which articulates forearm. Due to its structure and physiology, the role of humerus in entire arm movements is in fact very crucial [Winb].

The scapula (shoulder blade) is a flat triangular-shaped bone at the back of the trunk and glides over the surface of ribs two to seven (Figure 2.4, on the right). Indeed, there is no bony contact between scapula and thorax which provides quite large free motions in different directions.

The human shoulder girdle comprises four joints between the aforementioned bones above (see Figure 2.5). The first joint is Sternoclavicular joint which is between the sternum and clavicle. There is no muscle directly acting on this joint (passive) but almost any movement in the shoulder complex impacts on it.

Nevertheless, this joint reciprocate for the scapula motions. The second joint is Acromioclavicular (AC) which links the scapula to the clavicle. And the third joint is Glenohumeral joint. The GH joint facilitates the articulation between the glenoid and the head of the humerus. It is commonly accepted, both for anatomic and kinematic purposes, to consider the GH joint as a spherical (ball-and-socket) joint. It allows rotational motions of the arm around shoulder region as well as small translational movement towards or away from the body. Because of this joint humerus can move up and down vertically too. This joint creates an important rhythm where it allows the glenoid to compensate for the varying movements of the humerus head. This rhythm will be investigated more in details in the fol-



(a) Humerus.

(b) Scapula.

Figure 2.4: (left) humerus of human upper limb. (right) scapula of human upper limb

Sources: www.eurekalert.org, www.cgtrader.com

lowing sections for an attempt to find GH joint locations.

Finally, Scapulothoracic joint which helps scapula glide over a curved surface on the rib cage. This joint is not an anatomical joint because it does not have ligaments or capsules at the place of bones contacts. The connection is held by couple of muscles pressing the scapula against thorax, creating kinematically a planar joint gliding over the rib cage.

The next segment of the upper limb is the elbow (Figure 2.6). The elbow joint is between the arm and the forearm. It basically articulates three bones: the humerus of the arm and the radius and the ulna of the forearm. The elbow joint provides the flexion and extension of the forearm relative to the upper arm, as well as rotation of the forearm and the wrist.

The structure and bones of the wrist and hand create flexibility and the possibility of manipulation of varied objects in vast ranges of ways. Each hand contains 27 distinct bones (Figure 2.7). The movements of bones in wrist and hand are supported by forearm's ulna and radius which provide the supination and pronation motions of the hand ([Wina]). There are eight small carpal bones in the wrist that are firmly bound in two rows of four bones each. These bones mass is called the carpus. The carpus is rounded on its proximal end, where it articulates with the ulna and radius at the wrist.

Wrist plays an important role between the forearm and palm, making rotational movements possible

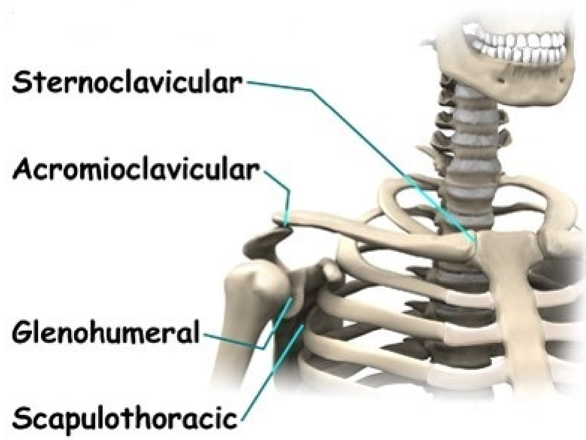


Figure 2.5: Shoulder Joints

Source: www.eorthopod.com

for hand around many different directions. Each finger extending from palm are moved by tiny muscles in the hand which provides stretching, compressing, and folding for the palm. There are 3 phalanges in each finger except for the thumb that has only 2. The phalanges are long bones that create hinge between each other. Phalanges that connect to the metacarpals at the bottom of the fingers are recognized as the proximal and the ones at the end of each finger are called the distal phalanges.

2.2 Human Motion Capture

To study human body motions, the data of the limbs movements should be captured somehow. One of the best ways to do so is through optical motion capture systems where reflective markers on the particular parts of the body are detected by cameras and their Cartesian coordinate information are stored. Vicon™ is one of the most popular systems to capture human motions which was used in this work as well. This system is designed to capture markers locations with adjustable rate (in the unit of frames per second) which yields thousands of data points. In the present study as shown in Figure 2.8, Bonita cameras are set up around the subject to provide a good workspace to capture targeted motions. The spatial position of each marker is tracked with respect to the world frame which is set up during the calibration stage.

Figure 2.6: Elbow.

Source: www.nottinghamshoulders.com

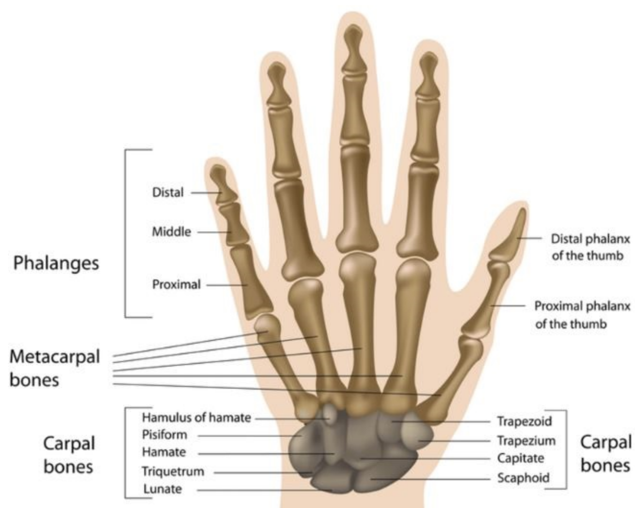


Figure 2.7: Wrist and Hand.

Source: [IRKS17]

Figure 2.8: Motion capture setup.

When using optical motion capture systems, there are some pros and cons. Since the capturing process has to be done in the laboratory it makes it difficult to have the outdoor motion tracked and maybe the body mounted cameras are better system to record the outdoor motions. Also the use of markers may cause the data to be off from the accepted value because of the non-rigidity of the human body. The distance of the markers to the center of rotation on the rigid body and center of the capture volume, speed of the motion and system calibration and not-optimized capture volume can be all different sources of error for a capture using this system. That being said, motion capture system is still one of the best choice for researchers to employ since it is accurate and can be applied to many situations involving motion capture data acquisition.

Depending on the purpose of capturing the data of each limb, there are two different strategies in terms of arrangement of the markers. One single marker provides the position of one spot of the limb while if we need to capture the full rigid-body information of each limb, i.e. position and orientation, a frame having 3 markers mounted on (L-frame) should be used (see Figure 2.9).

The calculation process to obtain three vectors perpendicular to each other, representing the moving frame of each limb, is as follows:

- Let the positions of three markers on the L-frame in the world frame be $\mathbf{m}_1$, $\mathbf{m}_2$ and $\mathbf{m}_3$.

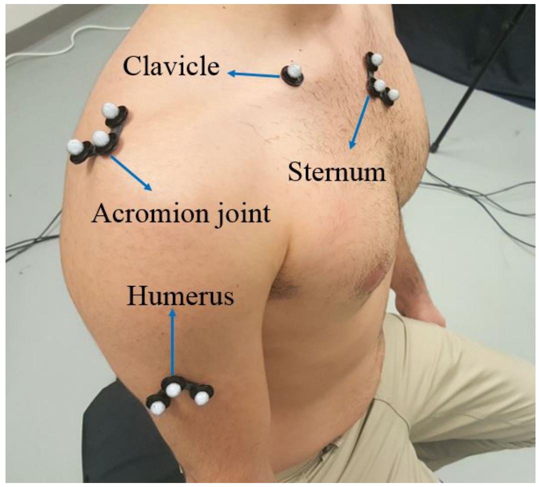


Figure 2.9: The spots of different markers on the subject's body.

- Calculate the vectors connecting the middle marker to the other two: $\mathbf{p}_1 = \mathbf{m}_2 - \mathbf{m}_1, \quad \mathbf{p}_2 = \mathbf{m}_3 - \mathbf{m}_1, \quad \mathbf{p}_3 = \mathbf{p}_1 \times \mathbf{p}_2$.
- Calculate the unit vector of $\mathbf{p}_1$: $\quad \mathbf{u}_1 = \frac{\mathbf{p}_1}{|\mathbf{p}_1|}$.
- Find the vertical element of $\mathbf{p}_2$ which is perpendicular to $\mathbf{p}_1$: $\quad \mathbf{p}_{2n} = \mathbf{p}_2 - (\mathbf{p}_2 \cdot \mathbf{u}_1)\mathbf{u}_1$
- Normalize $\mathbf{p}_{2n}$ to obtain the unit vector of the second perpendicular vector: $\quad \mathbf{u}_2 = \frac{\mathbf{p}_{2n}}{|\mathbf{p}_{2n}|}$.
- Find the vertical element of $\mathbf{p}_3$ which is perpendicular to both $\mathbf{p}_1$ and $\mathbf{p}_2$: $\quad \mathbf{p}_{3n} = \mathbf{p}_3 - (\mathbf{p}_3 \cdot \mathbf{u}_1)\mathbf{u}_1 - (\mathbf{p}_3 \cdot \mathbf{u}_2)\mathbf{u}_2$.
- Calculate the unit vector of the third perpendicular vector: $\quad \mathbf{u}_3 = \frac{\mathbf{p}_{3n}}{|\mathbf{p}_{3n}|}$

Applying the above steps, the L-frames shown in 2.9 are converted to kinematic moving frames containing rigid body position and orientation information (See Figure 2.10).

This whole process of making the kinematic frames and calculating the position and orientation of each limb is done in a MATLAB tool box created by the author (See Appendix A). This tool box reads the output of Vicon$^{\text{TM}}$ with software Nexus$^{\text{TM}}$ and plot the kinematics frames, in desired coordinate frame, for each data point captured by the system.

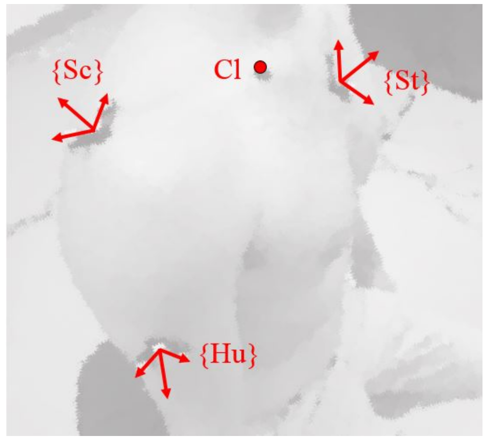


Figure 2.10: The kinematics frames for each limb considered as rigid body. $\{\mathrm{St}\}$ is the frame containing kinematics information of Sternum. Likewise, $\{\mathrm{Sc}\}$ and $\{\mathrm{Hu}\}$ represent Scapula and Humerus respectively. Cl is a single marker only containing the position information.

2.3 Kinematics Synthesis

Having the correct data of human limbs motions, we can go to the next step which is kinematics design process. The design of any mechanical system in 3D needs to consider kinematic, dynamic and other requirements. In a first stage, the kinematic design will determine some basic properties of the system regarding the mechanical structure, usually denoted the *topology*, which determines the number and type of joints and their connectivity. A second stage will consider dynamic and other performance requirements to shape the links connecting the joints as well as the links that will interact directly with the environment. Finally in a third stage the system needs to be instrumented and actuated. These three stages are not necessarily consecutive.

In this work we focus on the kinematic design stage. Kinematic synthesis, the process of creating a mechanical system for a given motion task, can be used in order to select and size a topology as a candidate hand design. The input for the kinematic synthesis will be a desired motion for the system as well as other performance requirements, and the output will be an articulated system in which basic dimension are defined. Fig. 2.11 shows the kinematic design process.

The kinematic chain is designed for a desired task, the intended motion of the elements of the hand

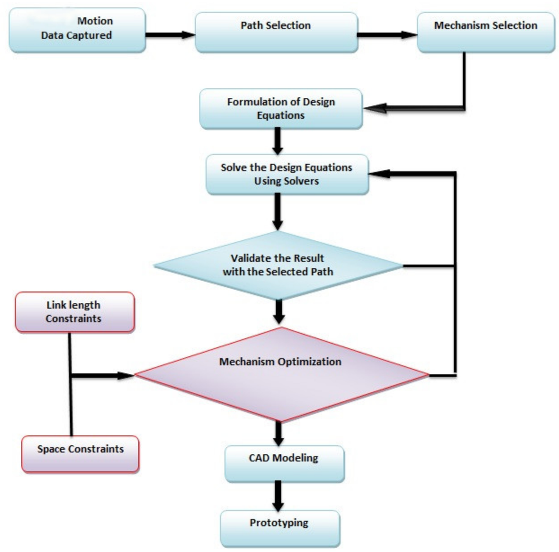


Figure 2.11: Flowchart for the kinematic design process.

whose interaction with the environment is of interest, usually the fingertips or end-effector. For a multi-fingered hand, a *simultaneous* motion of all fingertips or surface contacts, which could be any limb of the hand, is to be defined. Given the task, the type synthesis or structural synthesis, seeks to select the best topology by enumeration, selection and ranking of the kinematic chain to be used as candidate designs. It includes the selection or calculation of the number of fingers, number of common joints and for each serial chain making each branch, as well as the type of joints to be used. In the case of simultaneous end effectors, the solvability of the overall chain needs to be calculated [MPG14].

In the dimensional synthesis stage, the position of the joint axes are to be calculated, for the selected solvable topology and for the desired kinematic task. The output of the process is the position of the joint axes at a reference configuration, which is equivalent to the set of parameters defining the relative location and orientation between adjacent joints. At the end of the process, a prototype design is obtained that needs to be analyzed and simulated for performance fullfilment.

2.3.1 Design of Serial Robots

Most of the robots used for manipulation and inspection consist or have a subsystem consisting of a series of joints leading to an end-effector, which is known as a serial robot or a robotic arm. The design of a robotic arm is usually done by creating a serial chain that allows positioning the end effector at a desired set of positions, with a final wrist for orientation and a gripper-like attachment for grasping and manipulation.

A more systematic way of designing serial robots will follow the kinematic design methodology, in which the robot topology will be selected based on the desired task. A kinematic task is usually defined as a set of finitely-separated positions. Velocities, or even accelerations, can be defined at some of those positions. The selection of the type and number of joints of the arm is usually done based on the desired mobility of the robot in order to reach all positions of the task or workspace.

Dimensional kinematic synthesis seeks to find the position of the joint axes for a given topology and given task. Consider the direction $\mathbf{s}$ and moment $\mathbf{s}^{\circ}$ of a line representing a joint axis expressed in Plucker coordinates $\mathrm{S} = \mathbf{s} + \varepsilon\mathbf{s}^{\circ}$. By adding the pitch $p = t/\varphi$ to this line, it can be transformed onto a unit screw $\mathrm{J} = \mathbf{s} + \varepsilon(\mathbf{s}^{\circ} + p\mathbf{s})$ which combines the translational and rotational motion along the joint axis. Then transformation can be expressed as the representation-agnostic exponential,

$$S(\varphi, t) = e^{\mathrm{J}} \quad . \tag{2.1}$$

If $t = 0$, the notation denotes a revolute joint and in the case of $\varphi = 0$, it represents the prismatic joint. Product of these exponentials can explain the relative transformation between each target pose and a reference pose, P_{target}. These form the synthesis equations shown in Eq.(2.2), which have to be solved simultaneously.

$$S_1(\vartheta_1^i)S_2(\vartheta_2^i)\ldots S_n(\vartheta_n^i) = P_{target}^i, \quad i = 1, \ldots, m, \tag{2.2}$$

where n is the degrees of freedom or number of joints in the serial manipulator and m is the number of desired poses. Solving these equations requires powerful numerical solvers due to their high degrees and nonlinearity.

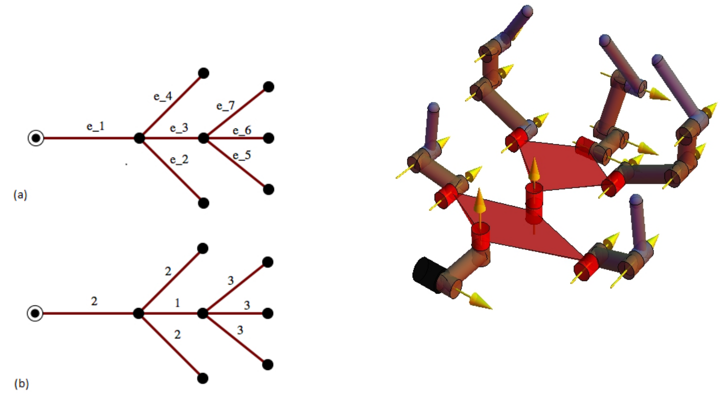


Figure 2.12: A five-fingered, two-palm hand topology. (a) indicates the numbering of the edges and (b) indicates the number of joints for each edge. Below: A kinematic sketch of the hand.

2.3.2 Design of Robotic Hands

Robotic hands are mechanical linkages with a tree topology (see [TPPG18] for more details). A tree topology for a kinematic chain has a set of common joints spanning several chains and ending in multiple end-effectors [Sel04], the fingertips in the case of a hand. A *branch* of the hand is defined as a serial chain connecting the root node to one of the end-effectors, and a *palm* is a link that is ternary or above. The tree topology can be represented as rooted a tree graph, see [Tsa01]. See a tree topology in Fig. 2.12.

Given the large number of topologies available for multi-fingered hands, it is necessary to derive some criterion to select the most suited topologies for a given task. Selection on the candidate topologies can be based on the number of positions of the task, the number of fingertips, the hand complexity, and the solvability, that is, the ability of the topology for being synthesized. Algorithms have been developed [TPGP16] to find solvable topologies for a defined task.

Dimensional kinematic synthesis is performed in a similar fashion as the synthesis or arm manipulators. Consider a hand topology with b branches and a total of n_e joint axes S_i, and create the sets of ordered indices B_j of joints belonging to the serial chain starting at the root and ending on end-effector j, for $j = 1, \ldots, b$. Given a simultaneous task for each fingertip, characterized by a set of m_p finite positions $\hat{P}^b_{1k}$ and m_v velocities $\dot{\mathrm{P}}^b_k$, kinematic synthesis is applied by equating the forward kinematics equations of each branch to the relative displacement of the corresponding fingertip. Similarly, velocities can be defined for some of those task positions,

$$\mathbf{F}(\mathrm{S}, \Delta\vartheta, \dot{\vartheta}) = \begin{cases} \hat{P}^{j}_{1k} - \prod_{i \in B_j} e^{\frac{\Delta\,{}^{k}_{i}}{2}\mathrm{S}_i}, & k = 2, \ldots, m_p \\ & j = 1, \ldots, b \\ \dot{\mathrm{P}}^{j}_{k} - \sum_{i \in B_j} \dot{\vartheta}^{k}_{i} \mathrm{S}^{k}_{i}, & k = 1, \ldots, m_v \\ & j = 1, \ldots, b \end{cases} \tag{2.3}$$

where S^k_i is the i^{th} joint axes when moved to position k. This yields a total of $6(m-1)b$ independent equations to be simultaneously solved. Again, this set of equations needs to be solved using numerical methods to obtain a kinematic design.

2.4 Human Motion Synergy

Motion synergies are principal components of the movement, obtained as combinations of joint degrees of freedom, that account for common postures of the human body. These synergies are usually obtained by capturing the motion of the human joints and reducing the dimensionality of the joint space with techniques such as Principal Component Analysis (PCA). For the purpose of identifiyng angular synergies, the model needs to be able to capture the main degrees of freedom while adapting to the user's dimensions. This can be accomplished by selecting a set of kinematic chains with similar motion complexity and performing dimensional synthesis in order to select the best fitted chain and to adapt it to the subject.

The goal is to create an experimental procedure to compare synergies for stroke patients and for healthy subjects, and to provide the results of pilot data comparing stroke and non-stroke patients. In order to do so, a low-error kinematic topology is selected for the upper limb and fitted to each subject individually using kinematic synthesis techniques. The motion is captured using a Vicon™system, and inverse kinematic techniques are used to calculate the joint variables for every joint of the model. Those joint angles are then used to compute motion primitives using PCA techniques.

The PCA technique allows identifying patterns in the data and those can be used to reduce its dimensionality without much loss of information. Consider a k-dimensional vector $\mathbf{o}^i_T$ containing all joint variables defining the upper-limb configuration for frame i. For a set of m observations in a k-dimensional space, the $m \times k$ matrix $[O]$ contains the angles for all the frames. We use the standard technique of creating a new set of observations as the errors $\mathbf{o}_e^i$ from the mean value $\mathbf{o}_A$ along an entire motion,

$$\mathbf{o}_A = \frac{\sum_{i=1}^{m} \mathbf{o}_T^i}{m}. \tag{2.4}$$

The new matrix of observation errors $[O_e]$ is used to compute the covariance matrix $[C]$,

$$[C] = [O_e]^T[O_e], \tag{2.5}$$

from which the eigenvalues λ_i and eigenvectors $\mathbf{u}_i$, $i = 1, \ldots k$, are calculated. The percentage of the data explained is controlled by the number r of eigenvalues selected,

$$\frac{\lambda_1 + \ldots + \lambda_r}{\lambda_1 + \ldots + \lambda_k} \geq p. \tag{2.6}$$

When used to obtain a smaller-dimensional set of reconstructed data, the weights of the data vectors with respect to the r-dimensional eigenvector basis are calculated. Consider the $k \times r$ matrix $[M]$ where the columns are the selected eigenvectors, the matrix of weights is

$$[W] = [O_e][M], \tag{2.7}$$

and the data can be reconstructed as

$$\tilde{\mathbf{o}}^i = \mathbf{o}_A + \sum_{j=1}^{k} w_{ij}\mathbf{u}_j. \tag{2.8}$$

Using this method the relation between angles along a trajectory for a particular motion can be identified.

2.5 Kinematics Model I

In order to calculate the motion of each degree of freedom in the upper limb, a good kinematic model is needed. Models are usually anatomically correct or are simplified based on robot kinematics [SK08]. There is a great variety of simplified kinematic chains to represent the human skeleton motion, ranging from simplest, consisting of spherical joints in all joints, to more complicated combinations of revolute, prismatic and spherical joints. In the kinematic model, an additional aspect to consider is the fitting of the model to the human subject, that is, the correct location and orientation of each joint in order to replicate the human motion. This is sometimes accomplished by measuring anatomic features with different degrees of accuracy. Other possibility, used in this work, is to apply kinematic synthesis techniques to size the kinematic chain [VUPGK06].

The upper-limb kinematics includes the joint complexes of the hand, wrist, forearm, elbow, and shoulder. In this work, different kinematic models in the literature have been reviewed, and a serial chain, where it starts from sternum and ends at the wrist, is considered and designed for the upper arm; the hand is skipped in this model and can be designed later on. A three degree of freedom (3-DOF) linkage containing two revolute joints and one prismatic joint has been chosen to simulate the shoulder motion. A spherical joint represents the Glenohumeral (GH) joint; the elbow and ulna-radius rotations are represented by two revolute joints and the wrist is modeled with two revolute joints. The hand has a tree structure and branches into the individual phalanges, with a 2-dof MCP joint and single revolute joints for the rest of the phalangeal joints.

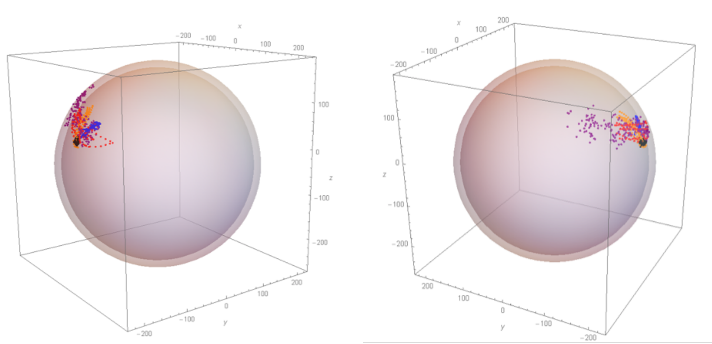


Figure 2.13: Approximately spherical motion of the marker placed at the end of the clavicle.

2.5.1 The Shoulder Complex

It is commonly accepted, both for anatomic and kinematic purposes, to consider the GH joint as a spherical joint. Then locating the centre of the GH joint suffices to model the motion from thorax to upper arm. In order to asses the kinematic model that can best fit the displacement of the GH center, motion capture data have been collected for 5 different motions and for a total of 620 points. Figure 2.13 shows that the marker at the end of the clavicle has an approximate spherical motion, with an error of approximately 15 mm.

The shoulder complex kinematics is modeled using a serial chain, consisting of a universal joint (U) to obtain the spherical motion, followed by a prismatic (P) joint to adjust the radius of the sphere for minimum error. Figure 2.14 shows the UP-chain kinematic model of the shoulder.

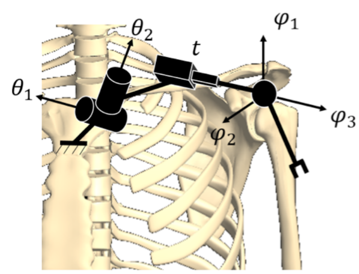


Figure 2.14: Kinematic model for the shoulder.

2.5.2 Elbow and Wrist Joints

We consider the flexion-extension of the elbow and the pronation-supination of the forearm as two nonintersecting R joints. Their particular location and relative orientation is given by the dimensional synthesis process explained below. Similarly, at the wrist we consider the flexion-extension and radial-ulnar deviation as two intersecting R joints. This is shown in Figure 2.15.

2.5.3 Hand complex

The kinematic model of the hand follows a tree structure, where each finger is a serial chain spanning from a rigid palm. Fingers are modelled as serial 4R chains, in which the distal chain consists of parallel

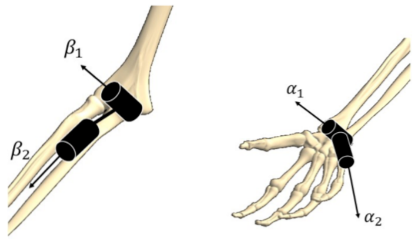


Figure 2.15: Kinematics model for the elbow, forearm and wrist.

joints, following the human anatomy, while the first R joint models the finger abduction-adduction, as can be seen in Figure 2.16.

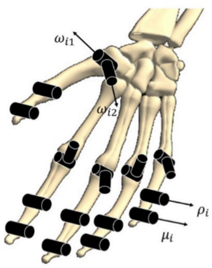


Figure 2.16: Kinematic model for the hand.

2.5.4 Generalized Inverse Kinematics

The kinematic model of the arm allows reducing the information of a given arm motion to a set of *joint variables*, which determine the rotation or translation associated to each joint of the model. The generalized inverse kinematics uses dimensional synthesis to fit the kinematic chain to the particular dimensions of the subject on a first step [VUPGK06], to calculate then the joint variables associated to each of the joints of the fitted model. Both the fitting and the inverse kinematics are calculated from the motion.

In the first step, point synthesis is used to calculate the UP shoulder chain and joint variables by equating the motion of the UP chain to the point locations of the marker, P_{GH},

$$S_1(\vartheta_1^i)S_2(\vartheta_2^i)S_3(t^i) = P_{GH}^i, \quad i = 1, \ldots, m, \tag{2.9}$$

where S_1 and S_2 are perpendicular and intersecting lines, and only the translational part of the displacement is considered.

After this, the center of the GH spherical joint is computed. Any 3 perpendicular and intersecting rotation axes can be used to create the spherical joint; joint axes are $P_i = \mathbf{e}_i + \varepsilon\mathbf{g} \times \mathbf{e}_i$, where unit basis vectors $\mathbf{e}_1$, $\mathbf{e}_2$ and $\mathbf{e}_3$ are used for the direction, intersecting at a common point $\mathbf{g}$, and with rotation angles φ_1, φ_2 and φ_3.

The displacement of the spherical joint, $H(\varphi_1^i, \varphi_2^i, \varphi_3^i)$ is equated to each pose of the upper arm, D_{sh}, in order to find the intersection point of the three axes and the rotation angle about each axis,

$$H(\varphi_1^i, \varphi_2^i, \varphi_3^i) = H_1(\varphi_1^i)H_2(\varphi_2^i)H_3(\varphi_3^i) = D_{sh}^i, \quad i = 1, \ldots, m. \tag{2.10}$$

Similarly for the elbow and pronation-supination of the forearm, consider two revolute joints of axes $B_1 = \mathbf{b}_1 + \varepsilon\mathbf{n}_1$ and $B_2 = \mathbf{b}_2 + \varepsilon\mathbf{n}_2$. The wrist model consists also of two revolute joints of axes A_1 and A_2 as shown in Figure 2.15. The precise location of these joints is calculated by equating the model to the motion of the palm with respect to the proximal arm, D_{wr},

$$B_1(\beta_1^i)B_2(\beta_2^i)A_1(\alpha_1^i)A_2(\alpha_2^i) = D_{wr}^i, \quad i = 1, \ldots, m, \tag{2.11}$$

and adding the extra constraints of the axes being perpendicular and intersecting according to the model.

The hand model for each finger can be solved separately. Each finger is based on a URR chain, where the universal (U) joint corresponds to the metacarpophalangeal joints (MCP) human joint and each phalangeal joint is a revoute (R) joint, all of them being parallel. Those degrees of freedom with small or coupled motion have been simplified in the hand model. The markers allow separate solutions for the motion of the MCP joint, $\hat{D}_{Ui}$, and the two phalangeal joints, $\hat{D}_{RRi}$. Consider $\hat{D}_{phx-i}$ and $\hat{D}_{fing-i}$ the motion of the proximal phalanx and fingertip with respect to the previous limb, respectively. Dimensional synthesis equations can then be created and solved separately,

$$O_1(\omega_{j1}^i)O_2(\omega_{j2}^i) = \hat{D}_{ph-j}^i, \quad j = 1, \ldots, 5, \quad i = 1, \ldots, m, \tag{2.12}$$

$$M_1(\mu_{j1}^i)M_2(\mu_{j2}^i) = \hat{D}_{fing-j}^i, \quad j = 1, \ldots, 5, \quad i = 1, \ldots, m, \tag{2.13}$$

In order to minimize the effect of the error in the data, an approximate solution is computed by using an excess of positions. For the dimensional synthesis part, this is accomplished using a constrained optimization algorithm, in which the constraints are the Plucker coordinates of the joint axes, which must be exactly enforced.

Once the joint axes are found to a satisfactory error, these are used to compute the inverse kinematics for the overall motion of the limb. For the inverse kinematics, the same set of equations (Eq.(2.9) to Eq.(2.13)) are used, where the screw axes are now known and only the joint variables appear as unknown. For the simplest chains, the dimensional synthesis and inverse kinematics steps can be solved simultaneously.

2.5.5 Experimental Setup and Data Acquisition

The results presented here correspond to two subjects, one healthy subject (HS) and one stroke subjects (SV). Stroke victim 1 (SV1) had a stroke within a year of capture, and underwent physical therapy. The physical effect of the stroke was the partial paralysis to the left side. Several movements are to be studied in order to recruit all the joints of the body within daily tasks. In particular, movements targeted in this study are:

1. Cross grasping: opening of a door.
2. Power grasp with flexion: grasping a glass and drinking.
3. Extended arm, two finger point: pointing in front of the subject. Arm in right angle with the upper arm parallel to the torso and is held out with the thumb up. Full open-close motion.

All subjects were asked to perform motions 1 to 3 above. Five trials were recorded for each motion and each subject, the time for each trial depending on the subject; some trials were as short as 8 seconds and some as long as 43 seconds. The resulting set of points for both subjects is presented in Figure 2.17.

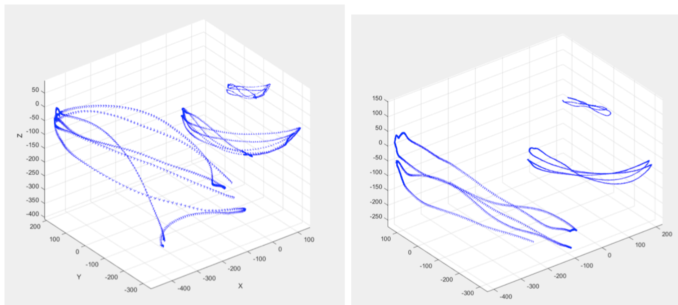


Figure 2.17: Motion Trajectories for Shoulder, Upper Arm and Palm. Both HS2 subject (left) and SV1 subject (right) for the Open Door motion.

The motion is captured using the Vicon™ system, with software Nexus™ 2.2. Vicon Nexus is designed to follow and record the spatial coordinates of markers placed on human subjects.

In the setup for capturing upper body motion, up to eight Bonita cameras are placed around the designed area of focus, see Figures 2.18 for an example. Calibration is performed and a model is created to be used within the Nexus software. Markers placed in the subject need to capture the full rigid-body motion of each limb. In order to do this, L-frames are placed on each limb when possible.

Figure 2.18: Mmotion Capture Setup

The Vicon system captures the position coordinates of each marker with respect to the reference frame set up during the calibration stage. The system works at frame rate of 120 frames per second, so the normal capture of a motion lasting seconds to minutes yields thousands of data points. For the hand, only three fingers are tracked due to constraints on the minimum distance between markers.

The resulting datasets contain between a maximum of 1019 frames (Open Door, SV1 subject) to a minimum of 703 frames (Pointing motion, HS2 subject). The Cross Grasping motion and the Power Grasp with Flexion motions were captured for a healthy subject and a stroke victim, and the Extended arm motion was captured and analyzed for the healthy subject only.

The point coordinates of the markers are converted to rigid-body displacements, according to the algorithm explained in 2.2, in those cases in which three markers are available for the limb.

For the synthesis phase, 114 poses (HS2) and 68 poses (SV1) are used to locate the kinematic chain of the shoulder. The GH spherical joint location was computed with 63 point positions (HS2) and 51 point positions (SV1). Finally, the elbow-wrist chain was dimensioned with 60 positions (HS2) ad 39 positions (SV1).

The dimensional synthesis stage yields a good fitting between the performed task and the trajectory of the synthesized upper-limb kinematic chain, as shown in Figure 2.19.

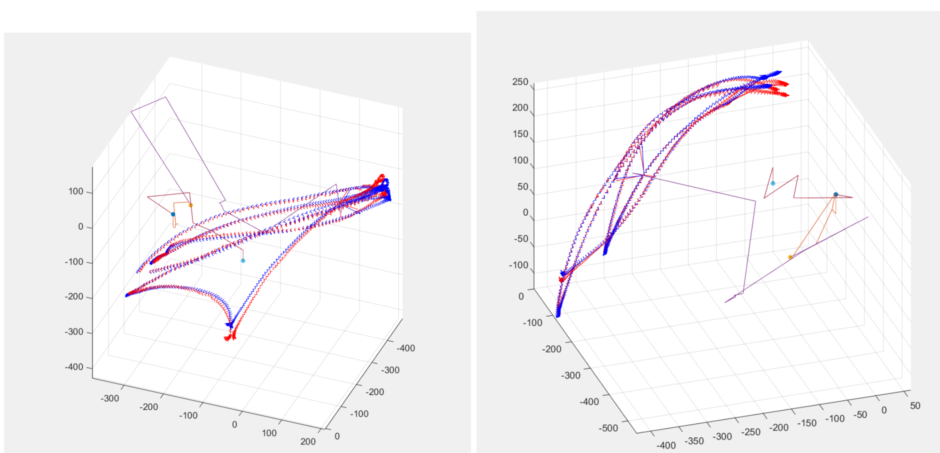


Figure 2.19: Task Trajectories. Open Door motion in the upper figure and Drinking in the lower figure, original (red) and performed by the kinematic chain (blue). The kinematic chain is also shown starting at the sternum (blue dot). Healthy patient.

In Figure 2.19, the inverse kinematics to reach each frame has been calculated as an unconstrained minimisation problem for the distance between the target displacement and the displacement of the kinematic chain. The results are obtained by computing the inverse kinematics for each limb, that is, sternum to upper-arm and upper-arm to palm, in order to make sure that not only the final motion is accurate but also the pose of the arm along the motion.

2.5.6 Calculation of Synergy

Human motion synergies happen when a reduced set of variables can be used to generate most of the arm common configurations and standard motions. Basically, they can be identified as common combinations of angles that appear along different actions or repetitions of the same action. These synergies can be calculated using histograms for the joint angles, and quantified using PCA techniques among other methods.

A problem of measure [MD95] appears when trying to combine angular joint variables, corresponding to revolute joints, with linear joint variables for preismatic joints. In our model one prismatic joint is included. In order to obtain a homogeneous formulation, this translation is considered as a rotation of angle γ along the z-direction in a 4D space, see [AM01] for the development of this formulation.

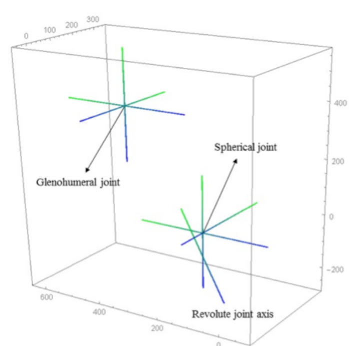


Figure 2.20: Kinematic Chain Results for the Shoulder Complex, Healthy Subject.

The dimensioning of the upper-arm kinematic chain was performed with 27 positions for the shoulder complex and combined with the inverse kinematics, for the total number of motion points, for the rest of the joints. The solution for the shoulder is shown in Figure 2.20.

The dimensional synthesis procedure to fit the kinematic chain to the subject yields good results. The trajectories of the arm can be replicated with good accuracy by performing approximate inverse kinematics using the results of the synthesis step. The maximum error between the desired pose and the position of the chain is less than 1mm for the position. This can be also observed in Figure 2.19 for the HS2 subject and in Figure 2.21 for the SV1 subject.

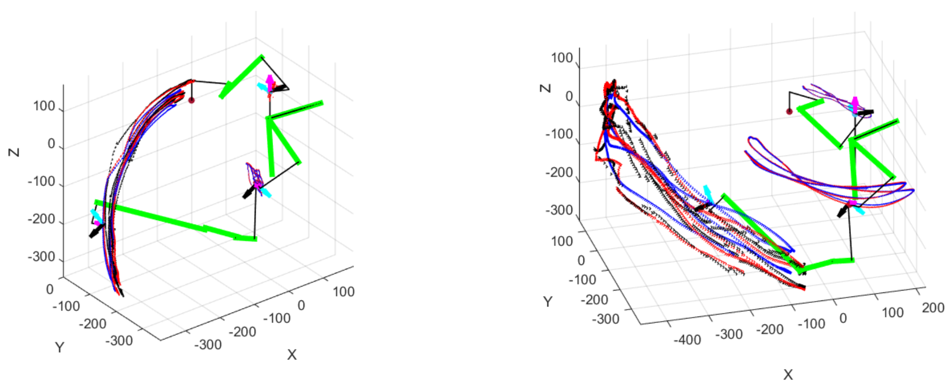


Figure 2.21: Task Trajectories. Open Door motion in the upper figure and Drinking in the lower figure, original (red) and performed by the kinematic chain (blue). The kinematic chain is also shown starting at the sternum (blue dot). SV1 subject.

The results from the inverse kinematics are used to create combined vectors of joint variables, $\boldsymbol{o}^i_T$, where i corresponds to the time frame, whose dimension can be up to 21 when considering arm plus three-fingered hand. In this study only 10 degrees of freedom are considered, from sternum to carpal area.

Principal component analysis is used to identify the relation between angles at different joints along a single task and considering repetitions. The eigenvalues are presented in Table 2.1 for both the HS2 and SV1 subjects and for the Open Door and Drinking tasks.

It is found that most of the degrees of freedom are needed for a faithful reconstruction of the motion. For the Open Door and Drinking motions, 8 eigenvalues were used to reconstruct the data (Figure 2.22), while for the pointing motion, 7 eigenvectors are used.

The eigenvectors corresponding to the main eigenvalues are presented in Table 2.2 for HS2 and in Table 2.3 for SV1, both for the Open Door motion.

Table 2.1: Eigenvalues for Open Door and Drinking Tasks

Motion	HS2 (*10³)	SV1 (*10³)	Motion	HS2 (*10³)	SV1 (*10³)
Open Door	6.8011	11.470	Drinking	3.243	4.483
	4.519	4.960		1.650	2.612
	3.931	2.579		0.993	1.083
	2.969	1.808		0.420	0.833
	1.424	0.806		0.277	0.251
	0.766	0.556		0.114	0.025
	0.495	0.098		0.003	0.002
	0.036	0.031		0.002	0.0006
	0.010	0.012		0.0002	0.0002
	0.004	0.003		0.0000	0.0000

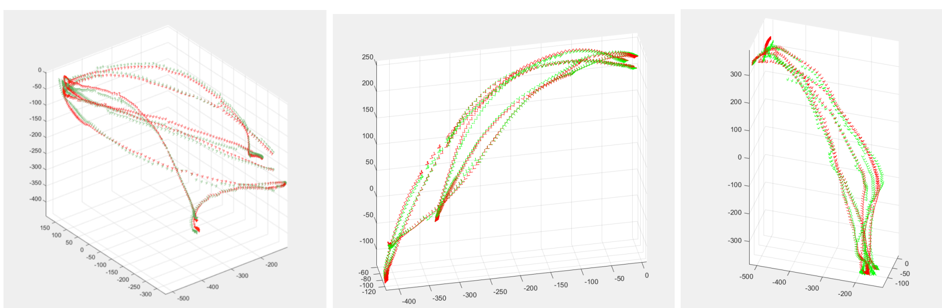


Figure 2.22: PCA Reconstruction. From left to right and top to bottom: Open Door motion (8 eigenvectors), Drinking motion (8 eigenvectors), and Pointing motion (7 eigenvectors). In all cases, red trajectory is the original task.

2.5.7 Result Discussion

The results presented here are preliminary and used to verify each step of the methodology and as a first indication of the phenomenology. In order to obtain conclusive results, more patients need to be added to the study, both healthy subjects and stroke patients. Also different options in how angular relations are calculated can be used to identify different trends in the data.

The sizing of the kinematic model yielded good results, with a small error in the overall minimized distance between motions and in the overall arm configuration. The inverse kinematics calculations yielded smooth angular trajectories, with enough accuracy to be used to identify patterns in the data.

Table 2.2: Main Eigenvectors for Open Door Motion, HS2.

μ_1	μ_2	μ_3	μ_4
-0.0057	-0.0945	0.0493	0.1046
-0.0234	0.0244	-0.0222	-0.0575
-0.4596	-0.8289	-0.1565	-0.2377
-0.8412	0.3944	0.2460	0.1972
-0.0340	-0.0301	0.0185	0.0427
0.1043	0.0477	-0.0628	-0.1134
-0.1371	0.2014	-0.3558	-0.3428
0.0097	-0.2132	0.7489	-0.0392
-0.1479	0.2356	-0.0548	-0.7136
0.1669	0.0564	0.4661	-0.4974

Table 2.3: Main Eigenvectors for Open Door Motion, SV1.

μ_1	μ_2	μ_3	μ_4
0.1200	-0.0247	0.0640	-0.0355
-0.5496	-0.3388	0.3072	-0.6920
0.0063	-0.0476	0.1627	0.1156
0.6506	-0.6284	-0.1964	-0.3143
0.0172	0.0116	-0.0027	-0.0113
0.0913	-0.0073	-0.0610	0.0278
-0.1798	0.2595	-0.8523	-0.3574
0.0387	0.2921	0.1435	-0.1328
-0.1050	0.0114	-0.0178	0.0126
0.4547	0.5783	0.2923	-0.5111

PCA applied to overall motions can be used to reduce the dimensionality of the dataset, with applications in motion simulation where real time results are required. The data reduction analysis shows that precise reconstruction requires most of the eigenvectors. For the studied motions, between 70% and 80% of the variables are required for fine reconstruction.

Muscle atrophy, the wasting or loss of muscle tissue, is a common effect of those who have suffered a stroke. Due to this and the loss of the brain cells due to the stroke, movement in a stroke victim is known to demonstrate more primitive patterning. The stroke victim in this study had limited hand motion, providing limited information for the limbs beyond the wrist; however many of the fingers show a synergistic power position.

In this study we observe some differences and similarities in the synergies of the stroke and non-stroke subjects. The analysis of the main eigenvectors shows that for the HS2 subject, the main component correspond to a synergistic combination of elbow joint and wrist flexton-extension. The second eigenvector combines the same two angles with different signs, and it may be either different parts of the motion or corresponding to the opening and closing parts of the motion. The third eigenvector shows a combination of shoulder and GH joint motion, and the fourth one shows similar components with some sign differences, which may be because of the reasons stated above. For the SV1 subject, it can be observed that the main eigenvectors combines shoulder motion with elbow and ulna-radius rotation. The same phenomenon of similar eigenvectors with different signs is observed for the SV1 subject. The third eigenvector is a combination of GH rotation with wrist motion. In general it is observed that the stroke victim uses the shoulder joints in synergy with the elbow and wrist more significantly than the healthy subject. Dividing the motion in smaller segments and separating the two phases (forward and backward) of each trial may help identifying other synergies and explaining the appearance of repeated eigenvectors with different signs.

Another factor that may be driving the range of the eigenvalues is the different range of motion of different joints for the given tasks. Big values of the angles may yield higher eigenvalues, and because of this, synergies of the shoulder-GH complex may not be directly comparable to those of the wrist or fingers. A scaling may help in this case.

The preliminary results presented here for one stroke and one healthy patients show that this methodology may be useful for quantifying differences between healthy and stroke patients. More subjects and motions are going to be analyzed in order to fully identify the synergies.

2.6 Kinematics Model II

Kinematics Model I assumes that the shoulder complex has 3 DOF without any constraints in the joint space. However, this assumption might not be true as different researchers has suggested different DOF for this mechanism; the complexity of the shoulder structure also makes it hard to easily decide about the workspace and joints numbers and types. The goal of designing kinematics model II, however, is to investigate the mechanisms that has the best candidate to simulate shoulder motions more precisely. The rest of the model including arm, elbow, wrist and hand are the same. The shoulder plays a significant role on hand and arm motions, however its kinematic modelling using standard robotic tools is not simple, this is the main reason to design Kinematics Model II.

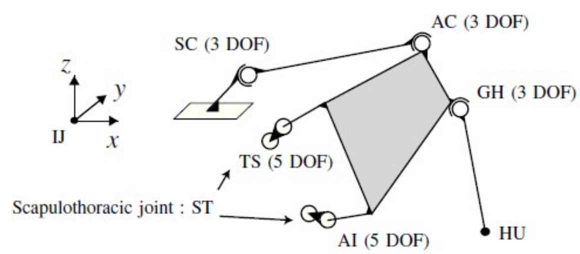


Figure 2.23: A parallel mechanism for shoulder with 4 DOF, [IEF+13].

As shown in Figure 2.2, the sternoclavicular joint connects the clavicle to the thorax. The other end of the clavicle is attached to the scapula through the acromioclavicular joint. These two joints can be considered as spherical sharing one degree of freedom, so that they can be simplified as a universal joint at the sternum and a spherical joint at the acromion. Moreover, the scapulothoracic joint, a compliant connection, exists between the scapula and the thorax allowing non-planar translational movement of scapula, when the underside of the scapula glides over the surface of the ribcage. This connection has not been considered as a joint anatomically because the bones are not connected directly to each other. A number of muscles act between the scapula and the ribcage pressing the scapula against the thorax, which makes gliding motions of scapula over the ribcage possible.

In [IEF+13] a minimal set of coordinates to describe the motion of shoulder is investigated. The proposed mechanism has 4 DOF in which Sternoclavicular, Acromioclavicle, and Glenohumeral joints are modeled as spherical joints. And two end points of the scapula's medial border (TS & AI) are modeled as two spherical slider joints being able to slide on the ribcage (see Figure 2.23). High load-carrying capacity and significantly small motion range within the shoulder girdle comparing to glenohumeral joint are properties which makes one choose a simple single-loop parallel mechanism. In this mechanism, shoulder has 4 DOf of which one is idle; GH has 3 DOF which makes it totally to have 7 DOF.

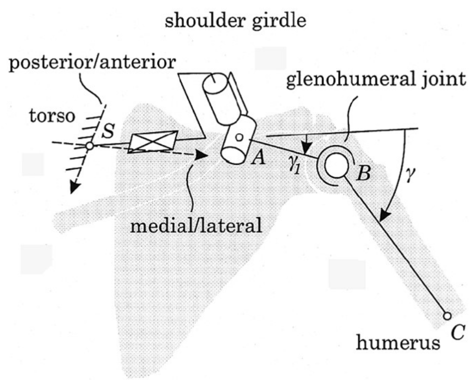


Figure 2.24: A serial manipulator with additional translation which is a dependent coordinate and can be treated as a function of the rotation angles, [LS03].

However, in some references serial mechanisms has been considered instead. In [LS03], the girdle (Clavicle, Scapula, Torso) is being considered as 2 degrees of freedom joints plus the GH joint as the spherical one which makes it to have 5 DOF. The shoulder girdle is substituted by a simplified kinematic equivalent consisting of a universal joint with two rotations intersecting on clavicle (Figure 2.24). In this proposed mechanism, an additional translation is introduced which is a dependent coordinate, as a function of the rotation angles.

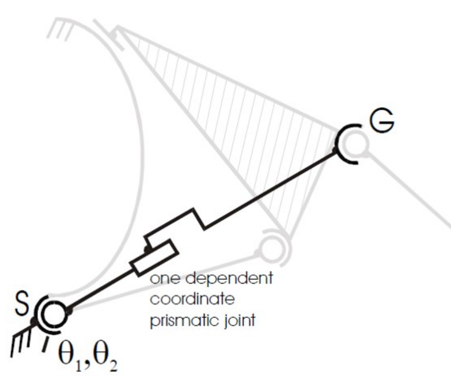


Figure 2.25: A serial manipulator with three DOF, among which one is dependent, [Ton05].

In [Ton05], the scapulathoracic articulation is modeled by a three DOF planar joint (See Figure 2.25). An equivalent model composed of two aligned links, the first one is joined to the sternum by a universal joint of variables ϑ_1 and ϑ_2 , the second one being joined to the first by a prismatic joint dependent on ϑ_1 and ϑ_2.

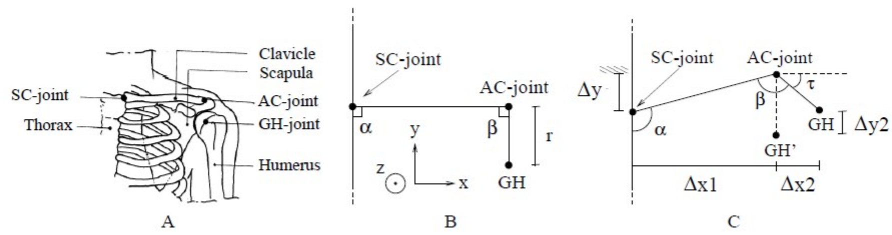


Figure 2.26: A serial manipulator with two DOF, [TMM05].

In [TMM05], by analyzing the outcome of advanced biomechanical models, they observed that the primary motion of the glenoid with respect to the torso is: rotations around the z and y axes, and vertical and horizontal displacements along the y and x axes. As it is depicted in Figure 2.26, the two rotations can be governed by increasing the range of GH-joint while the two displacements can be modeled by two prismatic joints each having one DOF. Thus, two translational degrees of freedom for shoulder girdle is being considered in this research.

In [JB13], however, they consider only the z direction. Meaning the scapula, based on their assumption, has just one degree of freedom, Figure 2.27.

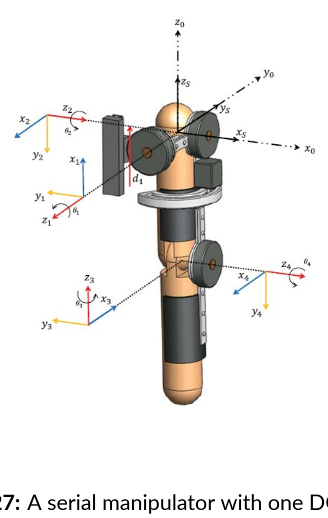


Figure 2.27: A serial manipulator with one DOF, [JB13].

However, a mechanism that looks like a shoulder is not necessarily a good replacement to mimic the same motion and provide the same workspace of the shoulder. So a better approach may be to simulate the shoulder complex as a black box, without trying to replicate the anatomy. This is what we describe in this work, we take the output of this black box and analyze it in an attempt to find its workspace, the mechanism which has the same workspace is the right candidate. Nevertheless, the first step is to determine the correct output of this system. The output or the duty of the shoulder is to locate Glenohumeral joint but reaching to this joint and capturing its motion directly is impossible. Thus, an accurate estimation of GH from marker data on the skin is crucial to analyze the shoulder kinematics and dynamics and have the correct output of the system. The process basically starts with finding the locations of the GH for different motions vertically and horizontally for a couple of subjects. Then try 1 and 2 DOF different serial and parallel mechanisms for point synthesis in order to investigate any relation between the dimensions of the shoulder workspace. A sample of all the targeted motions together should be considered to do this point synthesis. The type of mechanism that is performing the the shoulder motion more precisely for more subjects is the best candidate.

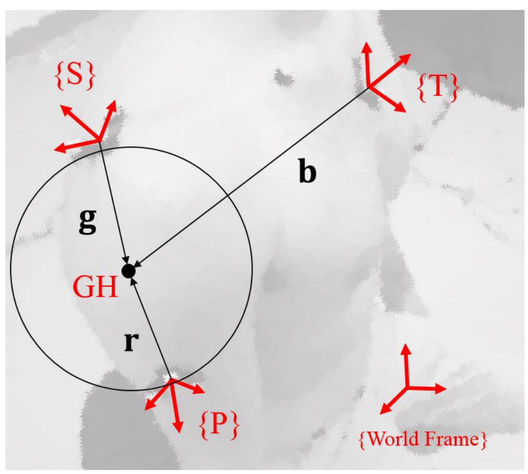


Figure 2.28: The frames used in the kinematics process, $\mathbf{b}$, $\mathbf{g}$ and $\mathbf{r}$ are the GH location vector relative to and expressed in frame {T}, {S} and {P} respectively.

Throughout the process for kinematics calculations in this chapter, the frames described in Figure 2.28 will be used. {T}, {S} and {P} represent the coordinate frames of Sternum, Scapula and Humerus respectively. $\mathbf{b}$ is the location of GH relative to and expressed in {T}, $\mathbf{g}$ and $\mathbf{r}$ represent the same location but expressed in {S} and {P} respectively. The goal of the algorithm is to find $\mathbf{r}$ representing the relative transformation between the {P} and humerus head which is a constant rigid body relation which differs for each subject. Vectors $\mathbf{t}$, $\mathbf{s}$ and $\mathbf{p}$ show the origin of frames {T}, {S} and {P} respectively expressed in the world frame. Index y points to the locations of all the markers on the L-frame; the maximum number of markers is w and index $y = 1$ represents the origin of the frame; the one placed on the corner. Index i shows the current frame out of m total captured data.

2.6.1 Scapulohumeral Rhythm

The movement between the humerus and the scapula throughout shoulder flexion and abduction defines scapulohumeral rhythm (SR). For a healthy individual, this rhythm consists of approximately 120 degrees of glenohumeral movement and 60 degrees of scapula movement, totally about 180 degree elevation of humerus with respect to the torso. There is nearly a 2:1 ratio in the glenohumeral joint to that of the scapula movement which occur concurrently. Nevertheless, this rhythm can be analyzed based on the elevation of the shoulder before and after 30 degrees, (Figure 2.29).

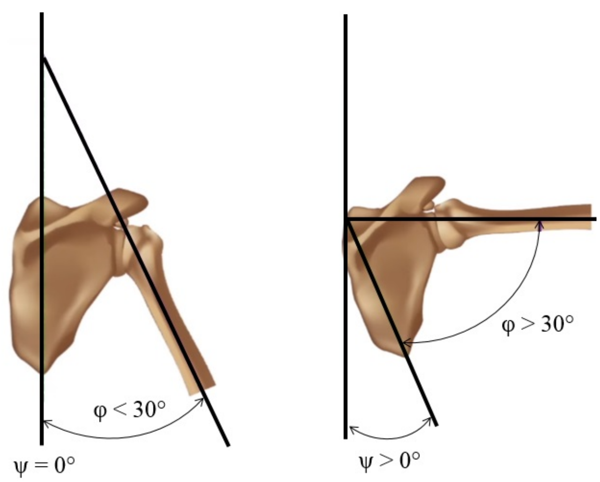


Figure 2.29: Scapulothoracic Rhythm.

The first 30 degrees involves a phase called *setting phase*, the scapula is relatively still and GH joint is largely contributing to the humerus movement. This part of the motions is where we can consider the GH joint as a spherical pair with the smallest error as there is quite no relative motion between

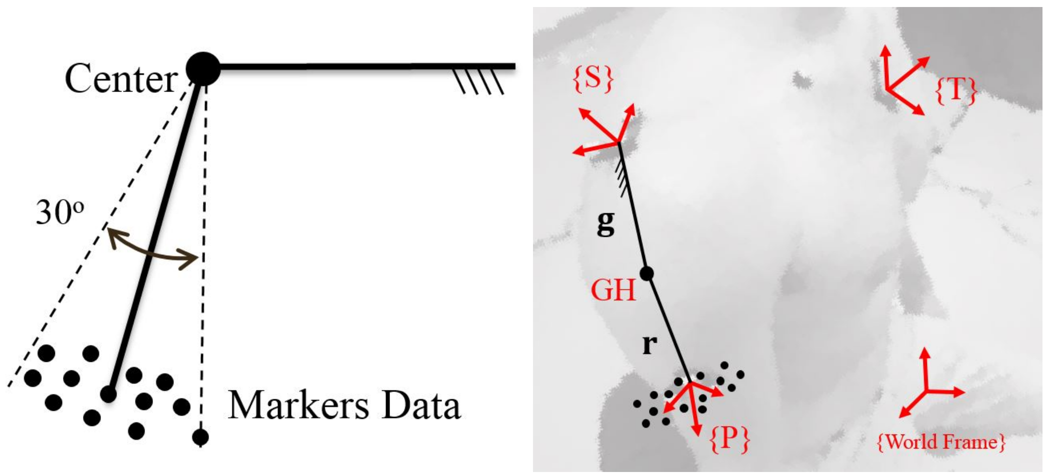


Figure 2.30: Left: two links connected to each other by a spherical joint. One link is fixed and the other can rotate around any axis. Right: corresponding system on human body, $\mathbf{g}$ is the fixed link and $\mathbf{r}$ is the moving link.

the humeral head and glenoid. We will focus on this part of the motion to calculate the center of GH. After the first 30 degrees of shoulder elevation, scapula starts moving simultaneously with the gleno-humeral movement to provide the total motion to elevate the humerus. The fact that the scapula has the least contribution in the humerus elevation in the first 30 degree applies not only on coronal plane but also on saggital plane and all other planes in between which are sometimes termed oblique planes. This is the key point to find the GH location for the elevation less than 30 degrees.

2.6.2 Spherical Fitting Methods

Figure 2.30, on the left, shows two bars; one is fixed and the other one can rotate around any axis as it is attached by a spherical joint to a stationary base. The markers are attached to the moving bars and the the goal is to find one center for as many spheres as the number of markers on the moving frame with corresponding radii; this center is called center of sphere (CoS) in this dissertation. The right picture in the same figure shows the equivelant system on the subject's body. All the data should be expressed in {S} frame as it is the closest to GH with the lease realtive motion for shoulder elevation less than 30 degrees. Vector $\mathbf{g}$ is a fixed translation which plays the fixed bar role and $\mathbf{r}$ is another translation which represents the moving bar. A survey of sphecrica methods to calculate CoS are given in [ETDH06b]. These methodologies appear in other themes as well ([GGS94], [CL05] and [Nie04]);

The methods mentioned in [ETDH06b] are used to find the GH joint and compare the results:

a) Geometric Sphere Fit Method (geom). This method tries to minimize the sum of the squared distances between the center and the marker positions ([ETDH06b]):

$$f_{geom} = \sum_{y=1}^{w} \sum_{i=1}^{m} (|\mathbf{p}_{yi} - \mathbf{g}| - \mathbf{r}_y)^2 \tag{2.14}$$

where $\mathbf{p}_{yi}$ is the captured point at frame i for marker y which is the given values to the equation, $\mathbf{g}$ is the vector pointing to the CoS and $\mathbf{r}_y$ is the radius of the sphere related to marker y whose center is at $\mathbf{g}$. Both $\mathbf{g}$ and $\mathbf{r}$ are the unknowns of the problem which are expressed in {S} frame.

b) Algebraic Sphere Fit Method (alg). This method is used to get rid of the initial guess in the previous method which is called modified least-square as well ([Kas76]).

$$f_{alg} = \sum_{y=1}^{w} \sum_{i=1}^{m} (|\mathbf{p}_{yi} - \mathbf{g}|^2 - \mathbf{r}_y^2)^2 \tag{2.15}$$

c) Pratt Sphere Fit Method (Pratt). In this method, none of the bias pertaining to alg method is present ([Pra87]) and is as follows:

$$f_{Pratt} = \sum_{y=1}^{w} \frac{1}{\mathbf{r}_y^2} \sum_{i=1}^{m} (|\mathbf{p}_{yi} - \mathbf{g}|^2 - \mathbf{r}_y^2)^2 \tag{2.16}$$

d) Centre Transformation Technique (CTT). To implement this method, we need three or more markers on the moving bar as the first step is to find the transformations of moving frame to world frame for all the captured frames. For the present experiment, the moving frame is {P} which transforms any vector from humerus to the world frame through $[P_i]$ matrices. Moreover, the transformation between the humerus and his head, where the GH is, is a rigid body translation which can be expressed in {P} as $\mathbf{r}$. Therefore, the position of GH can be written as $[P_i]\mathbf{r} + \mathbf{p}_i$. Then, the CTT technique is to determine $\mathbf{g}$ and $\mathbf{r}$ in the following minimization problem:

$$f_{CTT} = \sum_{i=1}^{m} |[P_i]\mathbf{r} + \mathbf{p}_i - \mathbf{g}|^2 \tag{2.17}$$

where $\mathbf{r}$ and $\mathbf{g}$ are constant-value vectors.

e) Minimal Amplitude Point Method (MAM). This is another approach introduced by [MMCD03] where $\mathbf{g}$ is defined as the point that moves the least under the transformations $[P_i]\mathbf{r} + \mathbf{p}_i$. Thus, the objective function changes into:

$$f_{MAM} = \sum_{k=x,y,z} [max([P_i]\mathbf{r} + \mathbf{p}_i)_k) - min([P_i]\mathbf{r} + \mathbf{p}_i)_k)] \tag{2.18}$$

in which i goes from 0 to the number of captured frames m.

2.6.3 Best Standard Deviation Method (BSTD)

This method is based on the standard deviation (std) of the distances of captured points of a set of markers to a set of CoS candidates. Whichever candidate that has the least STD with respect to all the markers is the best one. As it was stated before, scapula is the safest bone as the base for motion less than 30 degrees. The following steps describe the process of this algorithm for $\mathbf{g}$ (fixed bar) and $\mathbf{r}$ (moving bar), see Figure 2.28:

- Consider the origin of frame {S} as the center of an sphere with the radius equaling mid-diaphysis for each subject so we can cover all the possible CoS candidates. This sphere full of points is our search space where we try to find the best candidate for CoS, $\mathbf{g}_q$, $q = 1...u$ in which u is the number of candidates. The more markers on the humerus the less error, but in this work we used 3 markers, $w = 3$.
- Calculate the distances between each candidate and the points on the humerus (do it for each marker on the L-frame separately). d_{yiq} is the distance between candidate q and point i for marker y.
- Calculate the standard deviation of these distances (do it for each marker on the L-frame separately). std_{yq} is the standard deviation of candidate q for marker y.

- For each candidate, we have three *std* pertaining to each marker. So finally we have u groups, each group having three numbers. Now we are looking for a candidate which has the minimum standard deviations for all three marker data. To this end, each group is considered as a vector. The vector with the smallest norm (N_s) is the best candidate for CoS, $\mathbf{g}$:

$$N_s = min(\sqrt{std_{1q}^2 + std_{2q}^2 + std_{3q}^2}) \quad q = 1...u \tag{2.19}$$

2.6.4 SCoRE and IHA

Symmetrical Centre of Rotation (SCoRE) has beem introduced to locate joint centre

position ([ETDH06a]). Also IHA is another method to determine the centre of rotation as the point closest to all the instantaneous helical axis ([VY96], [SNR00]). None of these two methods need to assume that the CoS should be stationary. In [MDB$^+$07], an investigation and comparison of these two methods have been done.

SCoRE method is based on the fact that $\mathbf{r}$ and $\mathbf{g}$ are constant value vectors for all the frames captured if they are expressed in {P} and {S} respectively. We can write the vector pointing to GH in the world frame in two ways as follows:

$$\mathbf{p}_{1i} + [P_i]\mathbf{r} = \mathbf{s}_{1i} + [S_i]\mathbf{g} \quad => \quad [[P_i] \quad -[S_i]][\mathbf{r} \quad \mathbf{g}]^T = \mathbf{s}_{1i} - \mathbf{p}_{1i} \tag{2.20}$$

where $\mathbf{s}_{1i}$ is the location of the origin of the L-frame ($y = 1$) on the scapula for frame i.

Two configurations would be enough for this method to calculate $\mathbf{r}$ and $\mathbf{g}$ but the more configurations, the less the noise would have impact on the outcome. $\mathbf{r}$ and $\mathbf{g}$ can be calculated in

$$[\mathbf{r} \quad \mathbf{g}]^T = ([A]^T[A])^{-1}[A]^T\mathbf{b} \tag{2.21}$$

where $[A] = [[P_i] \quad -[S_i]]$ and $\mathbf{c} = \mathbf{s}_{1i} - \mathbf{p}_{1i}$. The solution has 6 elements from which the first three are the translation vector from the origin of {P} to the GH location and the second three element are the translation vector from the origin of {S} to the GH location.

In IHA, the goal is to find the CoS corresponding to the point closest to all the position vector of the instantaneous helical axes in a least squared sense. Consider the transformation of frame {P} to the world frame with $[P]$ as the rotation matrix and $\mathbf{p}$ as the translation, then the angular velocity matrix and linear velocity vector of frame {P} with respect to the world frame at index i is as follows:

$$\dot{\mathbf{p}}_i = \frac{\mathbf{p}_{i+1} - \mathbf{p}_i}{\delta t} \qquad \Omega_i = [P_{i+1}][P_i]^T - [I] \tag{2.22}$$

where t is time of capturing frame i, $[I]$ is a 3-by-3 identity matrix. Ω_i is a skew-symmetric matrix containing the elements of angular velocity vector of frame {P}, ω_i. Having the position, linear velocity and angular velocity, CoS ,in the world frame and for each captured data point, can be calculated with the following equation:

$$\mathbf{s}_i = \mathbf{p}_i + \omega_i \times \frac{\dot{\mathbf{p}}_i}{\sqrt{{\omega_i}^T \omega_i}} \quad i = 1...m \tag{2.23}$$

The closet point to all of these positions is the average vector:

$$\mathbf{s} = [\frac{\sum_{i=1}^{m-1} \mathbf{s}_{ix}}{m-1}, \frac{\sum_{i=1}^{m-1} \mathbf{s}_{iy}}{m-1}, \frac{\sum_{i=1}^{m-1} \mathbf{s}_{iz}}{m-1}]^T \tag{2.24}$$

This method is extremely sensitive to angular velocity; the slow frames should be removed.

2.6.5 Experimental Design

Six healthy males with an average age of about 28.8 gave informed consent to partake in the data collection. Table 2.4 shows the demographic data of the subjects. The experimental procedures involving human subjects described in this work were approved under expedited review of the Idaho State University Human Subjects Committee. Data was recorded after three motion sensors were applied to the shoulder girdle at specific locations while subjects were in the seated position, Figure 2.28.

Each subject was asked to perform 12 motions in the cardinal planes and various oblique motions outside of the cardinal planes as noted in Table 2.5. All motions were completed for five cycles with each motion returning to the setting phase at 0 degree of shoulder abduction for 1-2 seconds to insure

Table 2.4: Subjects Demographics

Age	Sex	Dominant Arm	AC to LatEpi(cm)	weight (kg)	height (cm)
30	male	right	34.5	86	165
29	male	left	34.5	69	175
26	male	right	35	95	178
29	male	right	33.5	84	185
29	male	right	33	93	180
30	male	right	34	82	182

Table 2.5: Targeted Movements

	Name	Abbreviation
M1	extension 30	Ext1
M2	extension 45	Ext2
M3	abduction–adduction	AbAd
M4	oblique 30	Ob1
M5	oblique 45	Ob2
M6	oblique 60	Ob3
M7	flexion–extension	FE
M8	horizontal adduction	HAd
M9	HAd/Elevation 30	HAd/E1
M10	HAd/Elevation 60	HAd/E2
M11	HAd/Depression 30	HAd/D1
M12	HAd/Depression 60	HAd/D2

muscle relaxation and completeness of data recording from start to finish of each motion.

2.6.6 Glenohumeral Joint Location Algorithm

Spherical methods are not reliable when CoS is not fixed. Moreover, despite the fact that both SCoRE and IHA are popular, the former method assumes that the GH position (humeral head) is in a constant relationship with respect to scapula, ignoring the relative motion and the latter method is significantly sensitive to slow angular velocities. Whereas the proposed method in this work neither suffers from slow movements nor ignores the relative motion between humeral head and scapula. This algorithm is based on scapulohumeral rhythm in which the scapula stays still in the first 30 degrees of shoulder elevation. In this specific part of the shoulder movement, humeral head acts like a socket lying on the glenoid which is the center of the three spheres created by the three markers on the humerus. If we have the coordinates of some points (more than 4) lying on a sphere we can compute the center of that sphere. A step-by-step schematic of the algorithm is illustrated in 2.31.

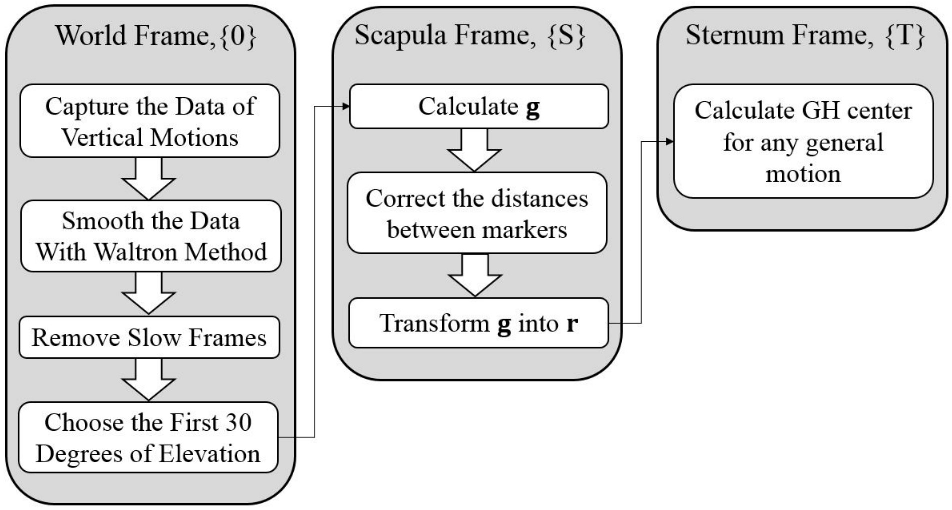


Figure 2.31: The Algorithm Process Steps.

The L-frame placed on the humerus gives the data of three spheres centered at the GH joint during the specified period of time which provides enough data to estimate the center. After capturing the data and filtering it, the correct part of the motion that represents the first 30 degrees of shoulder elevation should be selected. All these steps can be done on the data in the world frame. Afterwards, all the data should be expressed in scapula frame as it is the closest bone to CoS and supposedly has no relative motion with GH joint. Therefore, for motions less than 30 degrees elevation, GH joint can be considered as a ball-and-socket pair with a fairly small error. After we calculate the GH center we can find the transformation between the humerus frame and the humeral head which is a constant translation if expressed in the humerus frame.

The center of humeral head is a point on a rigid body whose frame is {P}. So if we can find the transformation between the humeral head and frame {P}, the expression of GH in {P} which is denoted by **r**, then we can use this transformation for all other general motions to find where GH is located. As a matter of fact, the input of the algorithm is the marker data on the subject body and its output is the relative rigid body relation between humerus and the humerus head expressed in frame {P}. This relation remains constant for each subject and differs based on the structure and anatomy of the bones for other subjects. Each one of the algorithm steps are explained in details in the following subsections.

a) Capture the data of the markers explained in 2.28 for vertical motions M1 to M7 from Table 2.5. Scapulohumeral rhythm takes place by doing the abduction-adduction motions in vertical planes including coronal and sagittal planes.

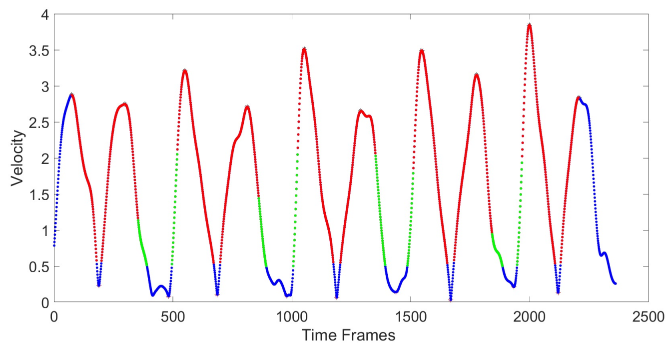


Figure 2.32: Velocity versus Time Frames. Green: Less than 30 degrees of elevation, Red: Velocity of the whole motion, Blue: Ignored frames

The subjects should not be asked to do a controlled 30 degree motion because it results in bad data as the subject tries to manage the motion as compared to having the subject doing the whole motion and take the first 30 degree elevation of the motion with a method. Another key point is that the first 30 degree part of all the vertical motions together should be considered as we need the data to be on more than one plane. Considering one motion only (for example Etao) will give insufficient input data as there are infinite number of spheres that can fit the data lying in one single plane.

b) Choose the right frames. The data captured by the cameras are already filtered in the software with Woltring method [Wol95] which minimizes the marker trajectory noise. However this filtering is not enough, the beginning and end of each cycle, where the velocity is very slow, the data is more chaotic and unreliable. So to be able to calculate the CoS more accurately we ignore such frames. In our experiments we ignore frames with velocities less than averagely 50 mm/s (is a slightly different for each subject). These parts of a motion is highlighted by blue in Fig 2.32.

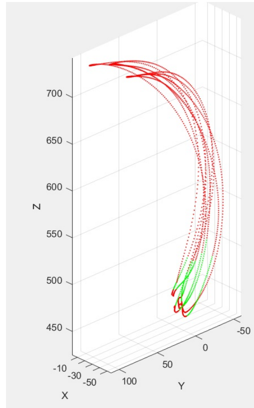


Figure 2.33: The selected data based on velocity for roughly about first 30 degrees of elevation

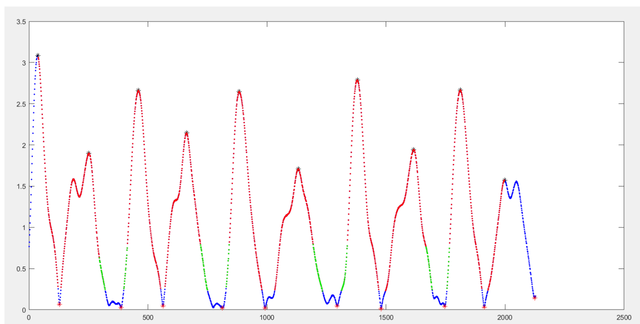


Figure 2.34: Velocity versus time frames for the second method: equality of velocity of {P} with respect to {T} and {S}

.

The velocity of the origin of frame {P} versus the number of time frames are shown in this figure.

To have a safe zone, the first half of the first cycle and the second half of the last cycle are also omitted. The next step is to choose the indexes of the first 30 degrees of elevation. The author introduce two methods to correctly choose the right frames. In the first method, the criterion is velocity-based where the algorithm finds the local min and max of the velocity in each cycle and choose about 30% of these indexes starting from the minimum index. Every other min is where the arm goes to the rest position. The chosen frames are shown by green in Figure 2.32. The corresponding dataset to the chosen indexes are shown in Figure 2.33.

The second method is based on the fact that the velocity of {P} for the first 30 degree of elevation

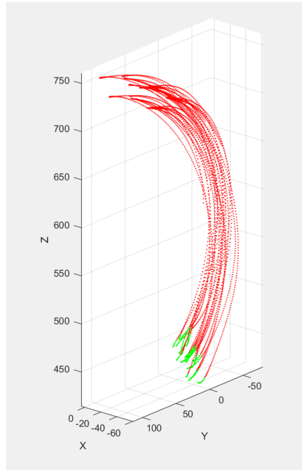


Figure 2.35: The selected data based on second method, equal velocity.

is the same if it is expressed in either {T} or {S}. Supposedly, sternum and scapula have no contribution in arm elevation for the first 30 degrees and this means that the velocity calculated relative to and expressed in {T} and {S} are equal. Figure 2.34 and 2.35 show the selected indexes and the corresponding position data of the humerus both calculated by the aforementioned method respectively. This method is more reliable mathematically but less robust experimentally as there are always some small relative movements between bones regardless of how precise the data are captured.

c) Express the data in scapula frame {S}. Scapula is the closest bone to humerus and has the least relative movements with respect to humerus for motions less than 30 degree. That is why we are considering this bone as a reference to find the GH location. As it was stated before, $\mathbf{p}_{yi}, i = 1...m, y = 1...w$ represents the captured data points of marker y out of w markers on the humerus L-frame at frame i of m captured data points that is expressed in the world frame and $[S_i]$ is the transformation between world frame and {S}. Then, with the following equation we can have captured pints on the humerus expressed in {S}:

$$\mathbf{p}_{yi\{S\}} = [S_i]^{-1}\mathbf{p}_{yi} \tag{2.25}$$

We consider the first 30 degrees of elevation of all 7 vertical motions so that we have the data in different planes from coronal to saggital plane to fit a sphere. Considering one motion only will cause a wrong result for CoS.

d) Find CoS in {S} frame. After having all the points expressed in {S} frame, the next step is to calculate the center of GH location. This can be done by any of the methods described in previous sections i.e. spherical methods, BSTD, SCoRE or IHA. In theory, SCoRE and IHA do not need to have a fixed link in one side of the spherical joint. However, in this work we have considered both of these methods to calculate the GH for less than 30 degree elevation. Implementing all of these methods for motion less than 30 degrees, Appendix B shows the tables for 6 subjects and the comparison among the methods to find GH.

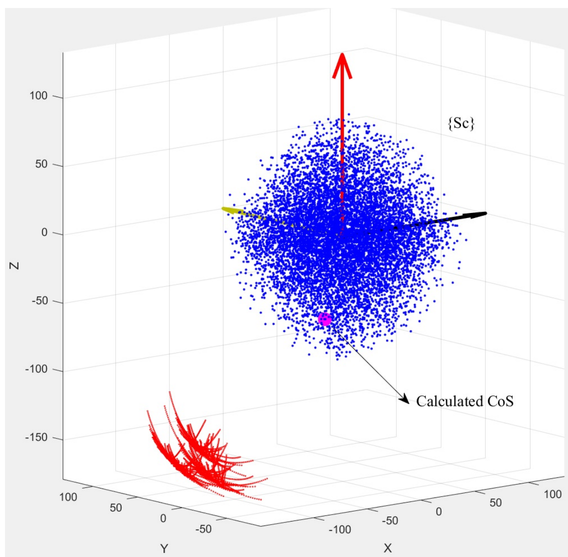


Figure 2.36: CoS calculated in Scapula frame based on std among all the candidates.

For example, Figure 2.36 shows the selected data points in red, CoS candidates in blue and the best candidate chosen by BSTD method in pink. This procedure was applied for all the subjects, Table 2.6 shows the location of CoS which is the humeral head (GH joint).

The z direction in all of them is negative and about 6 cm confirming that the GH location is under the acromion.

Table 2.6: The coordinates of GH location with respect to frame {S}, g

	S1	S2	S3	S4	S5	S6
x	-0.4591	-12.0198	-8.8396	-14.4994	-6.8193	-4.8432
y	-9.4755	-3.1101	-4.9282	-12.1834	-13.0300	-11.7775
z	-58.9022	-55.5081	-58.0653	-57.0527	-54.6694	-62.8978

e) Correct markers distances. The goal of this step is to correct the distances between the markers. Due to the skin motions and camera errors, the distances between the markers do not stay constant while they are placed on a rigid body, L-frame. This will result in having a non-unique vector pointing to CoS expressed in frame {P}.

We need to know which marker is more reliable meaning it has the least std error so we can use that as the reference for the other two markers when we want to correct the distances between them. To do this, the distances between each marker data and the calculated CoS is computed (for each marker separately). Then the std of each of these three groups are calculated and sorted ascending. The first one is the most reliable marker and the last one is the least reliable marker with respect to the center of sphere. Finally, we need the radius of each sphere which can be computed by the mean values of each group.

Having the distances between markers, we can correct the marker data based on the following algorithm.

- For the first reliable marker, project the points on the sphere centering at CoS along the first reliable radius, see Figure 2.37.
- For the second reliable marker, find intersection of the second reliable sphere centering at CoS and another sphere centering at the first reliable marker(this assures the constant distance between first marker and the second marker). The result of intersection of two spheres with different radii of which one is centering on the surface of the other is a circle; find the closest point on this circle to the corresponding captured data (Figure 2.38).
- To correct the third reliable marker, we consider three spheres. One is centered at the calculated CoS, the other one is created from the sphere centering at the first reliable marker

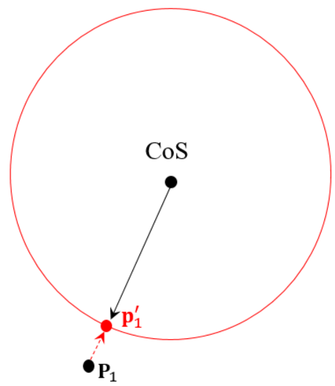


Figure 2.37: Projecting the captured points on the reliable sphere. $\mathbf{p}_1$ is the captured data for marker one and $\mathbf{p}_1'$ is the corrected one.

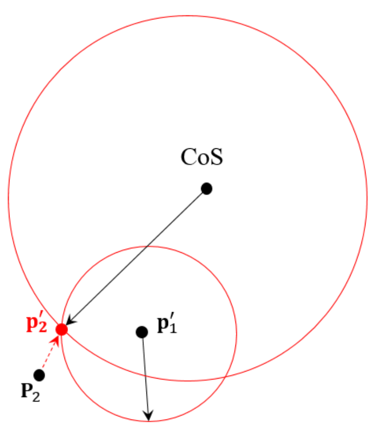


Figure 2.38: Calculating projection of the second reliable point with two spheres, one centering at CoS and the other at the first reliable point.

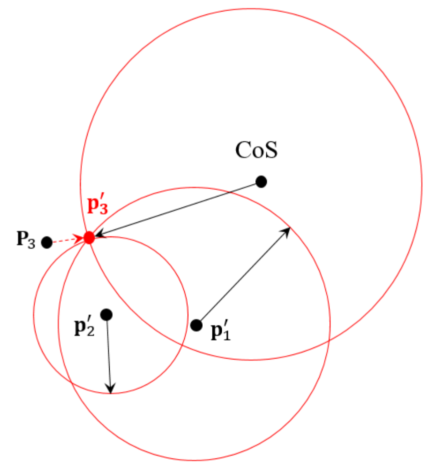


Figure 2.39: Calculating projection of the third reliable point with three spheres, one centering at CoS, another centering at the first reliable point and the third one centering at the second reliable point.

(to assure the constant distance between first and third marker), and the last one is from a sphere centering at the second reliable marker (to assure the constant distance between the second and third marker), see Figure 2.39. The intersection of three invading spheres with different radii are two points. Any of these two points which is closer to the captured data is chosen.

f) Calculate CoS in the humerus frame. So far, we calculated the location of GH joint for motions less than 30 degrees of elevation expressed in frame {S}, $\mathbf{g}$. Now in this section, we will transform this point into frame {P}, $\mathbf{r}$, see Fig 2.28. With the same setup of markers and cameras in the same session, we capture all the data of one subject. So if we have the CoS expressed in humerus frame, {P}, then we can use it for any other general motion that takes place in the same session.

$$\mathbf{r} = [S_i]^T [P_i] \mathbf{g} \tag{2.26}$$

Regardless of the fact that matrices $[S_i]$ and $[P_i]$ are different for each data point i, the result i.e. $\mathbf{r}$ is a unique vector. Figure 2.40 shows this vector pointing to GH.

As illustrated in this figure, the length of this vector and its component expressed in frame {P}

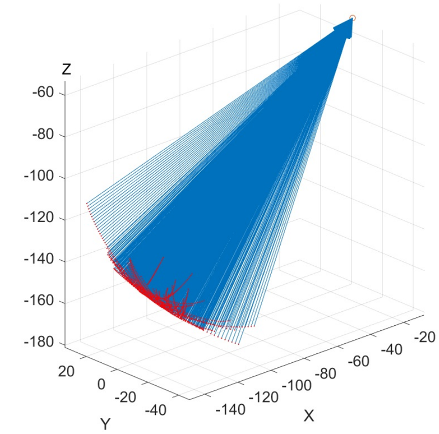


Figure 2.40: Vector $\boldsymbol{r}$

are constant. This is the vector we need to find the location of humeral head for any other general motion.

g) CoS for any General Motion. The final step is to calculate the GH for any other motion. We consider frame {T} as the base for all general motions as eventually the goal is to find a mechanism which can mimic the shoulder complex motion with all the joints in it. The mechanism with a base at sternum would includes the motions of sternoclavicular, acromioclaviacle and relative motion between humeral head and glenoid. For a given $\mathbf{r}$, we an compute the transformed coordinates in frame {T} by Eq 2.27.

$$\mathbf{b}_i = [T_i]^T[P_i]\mathbf{r} \tag{2.27}$$

Fig 2.41 shows motion Ob1 of subject two. As mentioned before the base frame is {T} and the calculated locations of humeral head is shown by blue dots.

The data of scapula and humerus are also added to the figure so that we make sure the calculated GH locations is in the reasonable position comparing to the other captured data.

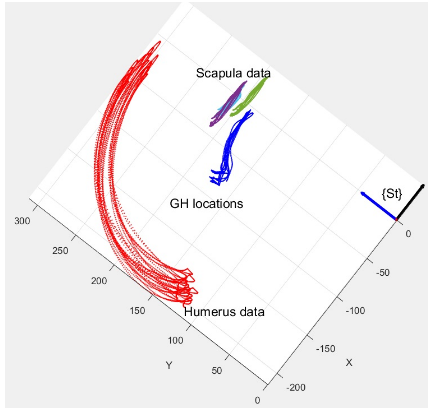


Figure 2.41: Calculated GH location with respect to {T} frame, the blue ones in figure.

2.6.7 Results Discussion

Fig 2.42 illustrates all the calculated GH for 7 motions expressed in frame {T}. x and y directions are in coronal plane and z is in the saggital plane. the maximum range is taking place in x-y plane as expected and differs for different subjects but averagely is less than 9 cm.

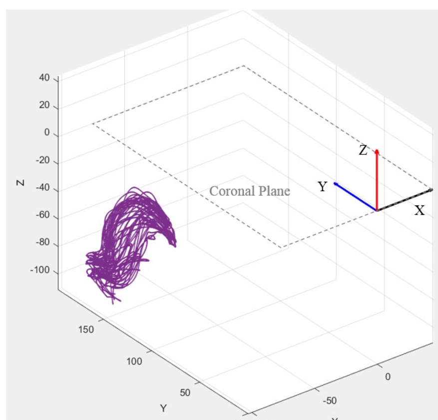


Figure 2.42: Calculated GH location with respect to {T} frame, for all 7 motions of subject

Moreover, we can analyze the shoulder workspace by PCA for which we need joint variables. So first, we need to choose a mechanism which has 3DOF to cover all the DOF of the workspace. Then after, we solve the inverse kinematics (IK) problem for the calculated GH locations to find the joint variables. In this work, we choose an RRP serial robot placed at the world frame, frame {T}, which is basically equal to the spherical coordinates of the points. Therefor, the IK of the robot can be computed as:

$$\vartheta_1 = \tan^{-1}\frac{y}{x} \qquad \vartheta_2 = \tan^{-1}\frac{z}{\sqrt{x^2+y^2}} \qquad r = \sqrt{x^2+y^2+z^2} \tag{2.28}$$

It was found that for most of the motions, the first two degrees of freedom is enough to explain about more than 90% of the whole motion. Fig 2.43 show the original data (in the standardized form) and the reconstructed ones by the reduced dimension.

Table 2.7 shows the first eigenvector of motions number 1 to 5. The interesting point here is that all of these vectors lye in the same plane which means that two variables are dependent suggesting that the degrees of freedom of the shoulder to provide vertical motions is 2.

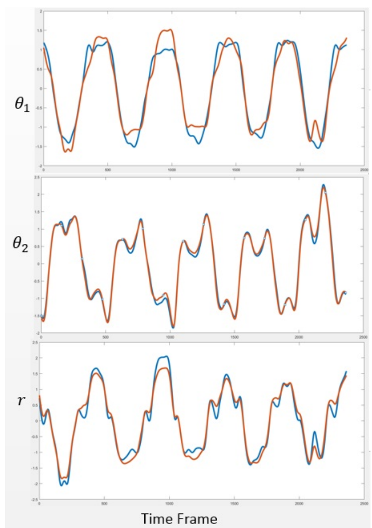


Figure 2.43: The reconstructed data by the reduced dimension, first two DOF

This relation turned out to apply to all other subjects as well which confirms that to move the arm in one plane and do a 1 DOF motion, the shoulder is using 2 DOF of its capacity. Mechanism RRP was chosen for the point synthesis here to investigate the dimensions constraints of the workspace of the shoulder using PCA. However, the ultimate aim is to do a point synthesis for a parallel mechanisms with less than 2 DOF to investigate if there is any constraint in the human should workspace. This research is to described separated done in the future where we will determine what linkage can be the best candidate to represent the shoulder complex.

Table 2.7: The first eigenvector of motion number 1 to 5 for subject #5

M1	M2	M3	M4	M5
-0.5988	-0.5824	-0.5882	0.6510	0.7031
-0.5071	-0.5479	-0.5437	0.5577	0.6544
0.6199	0.6005	0.5987	-0.5150	-0.2782

Chapter 3: A 2-DOF Parallel Manipulator for Shoulder Complex

In this section, a parallel manipulator with 2 degrees of freedom is synthesized to investigate if the shoulder complex degrees of freedom can be estimated by less than 3 independent variables. Looking at the conditions of the workspace of the human shoulder, it can be deduced that a parallel mechanism would be a better candidate comparing to a serial one for creating the similar motions. The reasons of using parallel mechanisms include a higher payload, smaller workspace and even being more similar anatomically to the shape of the shoulder complex. At first we calculate GH locations for all the motions as was described in the previous sections. Then, we choose 20 points among the data set of each subject and do the point synthesis for the parallel mechanism. If a 2-DOF mechanism can describe the human complex motion with a fairly reasonable error, then we can come to the conclusion that GH location is lying on a surface and we introduce a 2-DOF parallel mechanism that would create the best estimation for shoulder worksapce.

We are looking for a parallel mechanism whose end-effector has 2 degrees of freedom and all the joints are of revolute type. The procedure of finding one mechanism among all the possible ones relays on the constraint space created by each leg of the mechanism. One can come up with different combinations of legs in terms of number of joints in each leg and number of legs themselves which can produce 4 constraints in space for EE resulting in a desired 2-DOF mechanism but for the purpose of simplicity of the synthesis equations, we select a parallel manipulator with two legs and 4 revolute joints in each leg (2-4R). No special relationship between joints are assumed and it is left to the design process to find any if they exist. Moreover, this type is closer to the anatomy of the shoulder too as there are two branches from sternum and scapula reaching each other at GH joint.

On the other side, to ensure that the degree of freedom of the EE remains at the constatn value of 2, we use an alternative concept of DOF introduced in [ZZF04] which involves the configuration of the joints rather than the traditional approaches which only depend on the number of joints and bodies involved. This should be implemented in the design process to make sure the EE does not loose or gain any DOF at different given pints which results in a parallel robot with different DOF than the desired one. The following section explains how this approach works.

3.1 DOF based on Reciprocal Screw System

The mobility of the moving base of a parallel mechanism is basically determined by how many of its degrees of freedom are restricted by the constraint space created by all the branches connecting it to the fixed base, [ZZF04].

$$F = 6 - d + \sum_{j=1}^{n} r_j \tag{3.1}$$

where F is the degree of freedom of the manipulator's end-effector, d is the rank of the total reciprocal screw system of all the branches together, n is the number of branches connecting the fixed base to moving base, and r_j is the rank deficiency of branch j. Rank deficiency of a serial chain is actually the rank deficiency of the screw system consisting of all the joints kinematic screws. Each screw is defined by the direction of the joint line and its moment which is also known as *plucker coordinate*:

$$\$^i = [\mathbf{u}^i, \mathbf{m}^i] \tag{3.2}$$

where $\mathbf{u}$ is the direction of the joint, $\mathbf{m}$ is the moment of the joint line and i is the index of the joint in the branch.

Now, the kinematic screws of chain j is the screws of all the joints in that chain:

$$\$_j = \begin{matrix} [\$^1 \\ \$^2 \\ . \\ . \\ \$^n] \end{matrix} \tag{3.3}$$

The rank of this matrix is equal to the DOF of the serial chain. On the other hand, the rank deficiency of this matrix adds to the DOF of the EE of the parallel robot. Moreover, the reciprocal screw system of this matrix gives the constraints this branch creates:

$$\$_j \bullet \$_j^r = 0 \tag{3.4}$$

where $\bullet$ is the reciprocal product of two screws and r represent the reciprocal concept. The next step is to calculate d which is gained by computing the DOF of a matrix containing all the reciprocal screws (null spaces) of all the branches:

$$\$^r = \begin{matrix} [\$_1^r \\ \$_2^r \\ . \\ . \\ \$_b^r] \end{matrix} \tag{3.5}$$

where b is the number of branches. If this matrix has rank deficiency, it means that some branches are sharing the constraint space they create; restricting an already restricted dimension would not have an effect.

When synthesizing a parallel robot, we should check the DOF of the EE to makes sure that the different configuration of the joints in each leg neither adds nor reduce the degrees of freedom of the EE. We do not want any rank deficiency in the legs ($\sum_{j=1}^{n} r_j = 0$) and desire to design a 2-DOF mechanism ($F = 2$), then Equation 3.1 could be written as the following one for a part of the synthesis equations system to ensure that the legs create 4D constraints.

$$d - 4 = 0 \tag{3.6}$$

3.2 Point Synthesis Formulation

As mentioned before, the duty of the shoulder complex is to locate the GH location. Thus, the mechanism that we are synthesizing should reach the positions of the GH regardless of the orientation. GH itself is a 3-DOF spherical pair that creates the orientations for the humerus. Let T_i be the given pose at each step i representing a transformation matrix that holds the orientation information, Q_i, and translational information, r_i. Also, Q_i^l and r_i^l are the orientation and translation of the left branch EE respectively. The pose of the right branch EE is represented by Q_i^r and r_i^r as well. We can write the following equations to ensure that both branches of our 2-4R parallel mechanism can reach the desired points:

$$\begin{aligned} r_i^l &= r_i \\ r_i^r &= r_i \end{aligned} \tag{3.7}$$

And because we want the two branches connect to each other so that they create a closed chain we add the following equations to the system:

$$\begin{aligned} Q_i^l &= Q_i \\ Q_i^r &= Q_i \end{aligned} \tag{3.8}$$

where i goes to 20, the number of selected points for synthesis problem. The kinematic chains of each branch is built by a sequence of Denavit-Hartenberg matrices. Also, to provide enough design parameters, the left leg has an extra transformation along the local z axis and the right leg has one constant transformation for locating the first joint and one constant transformation at the end of the chain along the local z axis. To make the base of the branches stay close to sternum, the left leg has its first joint coincident with the global z axis and the translation of the first constant transformation for the right leg is equal to 0.

Solving the aforementioned system of equation with numerical methods does not guarantee that the EE of the right and left leg meet each other precisely. Changing the problem to an optimization format with having Eq. 3.7 as the cost and equality of the poses of both legs at their ends as the constraint of the problem was not successful based on many trials with different initial states. However, it was observed that if we use the solution to the system of 3.7 and 3.8 as an initial state for a new system of equations that is created by removing the target points, we can reach to a better solution. The new system of equation is as follows:

$$\begin{aligned} r_i^l &= r_i^r \\ Q_i^l &= Q_i^r \end{aligned} \tag{3.9}$$

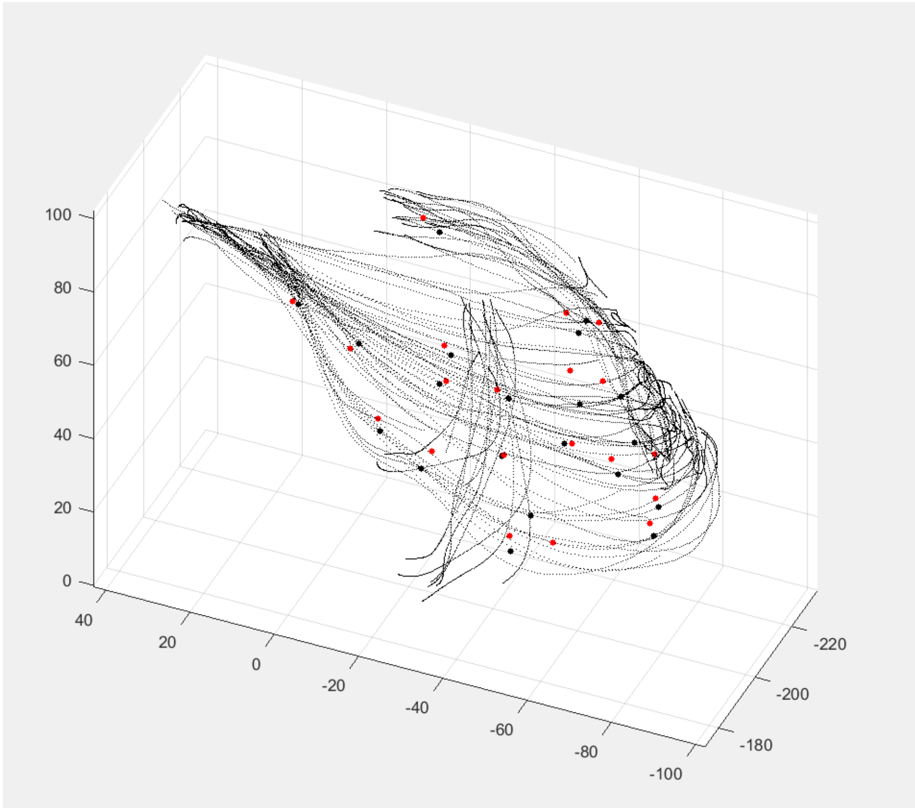


Figure 3.1: The workspace of subject number 2 with black colors, the selected 20 points with bigger size and black color and the reachable points for the designed robot with blue color

This system does not enforce the solution to be close to the targets but as the initial state is already close to the targets, the final designed robot has a bigger chance to have a workspace close to the targets. Most numerical methods are only able to find a local minimum which can result in a good solution if the initial state is close enough to the global minimum. It was observed that solving this system by optimization methods without any constraint and having only cost functions can result in perfect parallel robots; two serial chains that perfectly match at their EEs in terms of both position and orientation.

Another important point is that the 20 selected points are representing a workspace made up of a large set of number of points, a couple of thousands discrete data points, see Figure 3.1. This means that even if the designed robot can not reach to the 20 selected points, it might still be in the workspace. This is checked for each designed mechanism to make sure that the EE of the synthesized parallel manipulator stays close to the workspace with a fairly small error. The last point is that this works is inves-

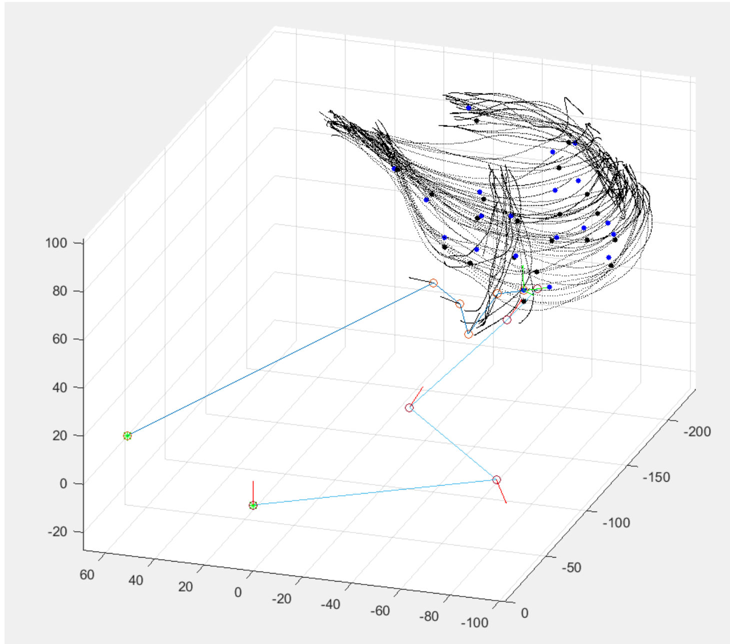


Figure 3.2: The designed robot reaching the first data point approximately.

tigating if a 2-DOF mechanism can describe human shoulder workspace with supposedly 3-DOF. If the result of the synthesis fits the target points, we conclude that the shoulder works in a 2-DOF workspace but if not, the result would represent the best estimation of a 2-DOF parallel mechanism to mimic the shoulder complex motion.

3.3 Result

The point synthesis mentioned in the previous section consists of two steps, finding an initial state and solve the closure equation. This process was repeated for all the 6 subjects to find 2-4R parallel manipulator for each human shoulder complex. The result of the subject number 2 is given here. Figure 3.2 shows the designed robot reaching to the first design point approximately. The small black points are the workspace of the shoulder complex, the bigger black points are the selected ones and the blue ones are the points that the designed robot can reach. In the same figure, red lines show the direction of each joint. There has not been found any special arrangement among the joints such as being parallel, coincident or intersecting. For this reason, we can call the designed robot a general 2-4R parallel manipulator. The depicted links are following the DH parameters convention where the common normal of

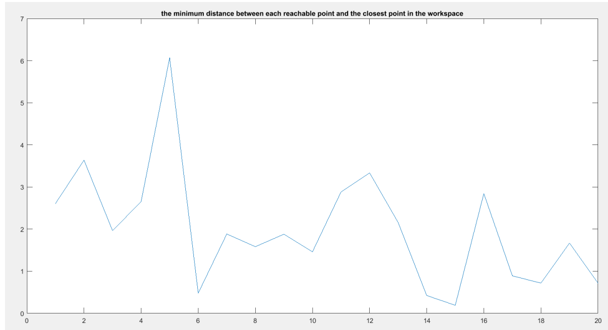
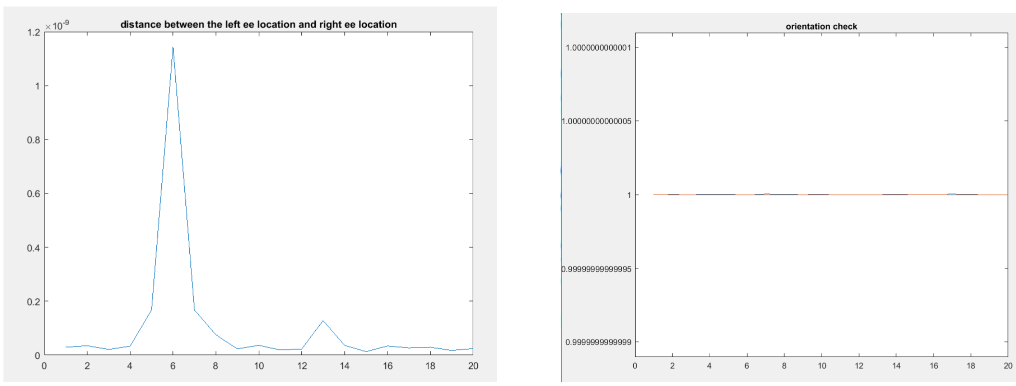


Figure 3.3: The minimum distance between each reachable point and the closest point in the workspace



(a)

(b)

Figure 3.4: (left) Position check. (right) Orientation Check

each two consecutive joint lines are assumed to provide the structure of the robot.

The designed robot is a parallel manipulator with two legs; each leg is a 4R serial manipulator. In the first step of the synthesis, the number of equations for one data point is 16 which includes Eq. 3.7 and 3.8 considering all three axes of rotations. Including Eq. 3.6, we have total of 16 for one point. Also unknowns include 32 structural parameters plus eight joint variables for one data point. This leads to a highly nonlinear system of 320 equations with 192 unknowns. Step 2 includes 12 equations for each data point which leads to a system of 240 equations with 192 unknowns. Levenberg-Marquardt which is a Least-square method for nonlinear problem is used as the numerical method.

Figure 3.3 shows the minimum distance between each reachable point and the closest one in the worksapce. The mean of the error is about 2 mm showing that the robot can reach to a fairly good distance to the points in the workspace.

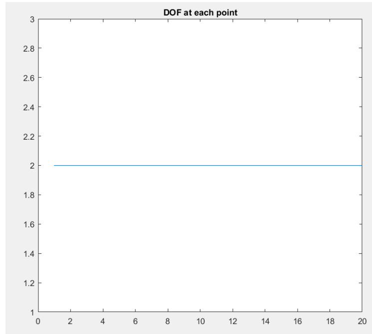


Figure 3.5: The designed robot reaching the first data point approximately.

To make sure that the left and right leg reach each others' ends precisely, the distance between the EE locations from left and right is calculated and plotted in Figure 3.4 on the left. In the same figure, on the right, the orientation of the EE examined by calculating the dot product of the x-axis from left and right as well as the z-axis from left and right. As it can be seen, the EE from left and right leg meet each other perfectly for all the data point meaning the EE of the parallel robot stays rigid along the motion. This is the key point when dealing with a closed chain as two separate serial chains meeting each others' end-effector. The advantage of formulating the problem this way is that the nonlinearity of the equations decreases. On the other hand, the disadvantage is that the constraints ensuring that the end-effectors of the serial chain meet should be added to the system. These constraints should be met fully otherwise the mechanism will not be a closed linkage which contradicts the goal of the project.

The next parameter that needs to be checked is the DOF of the robot EE at all data points which should stay at 2. Figure 3.5 shows the value of the degrees of freedom of the parallel robot's EE at each data point. This ensures that different arrangements of the joints axes would not add or remove DOF to the EE of the parallel robot.

The condition number of the matrices that whose rank matter in the DOF calculation are also calculated to make sure that these matrices do not become ill-conditioned. Eq. 3.3 and 3.5 are the the matrices whose condition number should be checked.

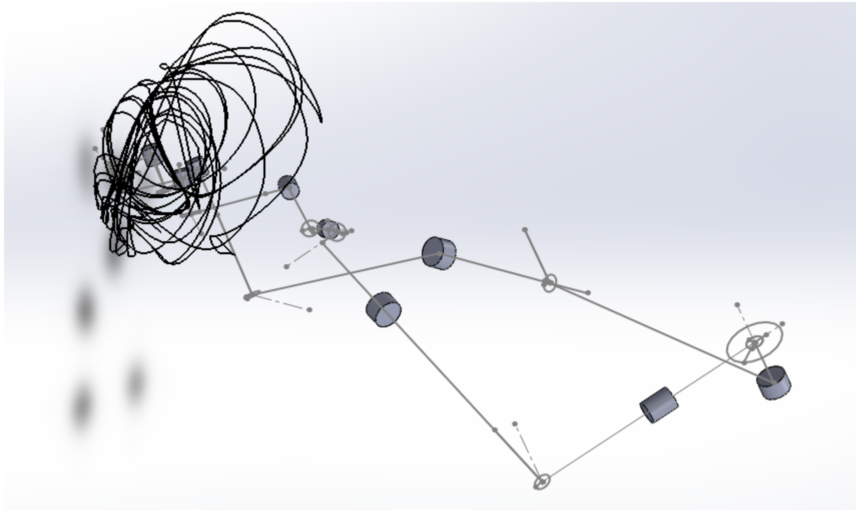


Figure 3.6: Position workspace of the point at the origin of the end-effector

3.4 Workspace Analysis

The most important factor for a solid conclusion is the comparison between the workspace of the designed robot and that of the human shoulder complex. Calculating the workspace of the robot analytically is cumbersome if not impossible. More investigation and work is needed to find the workspace numerically as well. The current analysis is based on the 3D model simulation of the robot in SolidWorks where all the constraints among the joints and links based on the calculated DH parameters are applied. Firure 3.6 shows the workspace of the designed 2-4R manipulator by moving joint number 2 from the left leg only.

Two designed serial chains are created in the 3D environment at the first data position by the calculated DH parameters and joint values for each step. Then end-effectors of the serial chain must meet each other precisely or it would not represent a closed chain. As it was expected, the EE of the both legs match and create the EE of the parallel robot. The last link of the left leg has the same pose of the last link of the right leg so that they create a rigid body which serves as the end-effector of the parallel robot.

Figure 3.7 depicts the workspace of the robot on top of the shoulder workspace. The number of selected points that the synthesis has been done based on are 20 while there are more than couple of thousands points in the workspace for each subject. As the number of equations and variables increase dras-

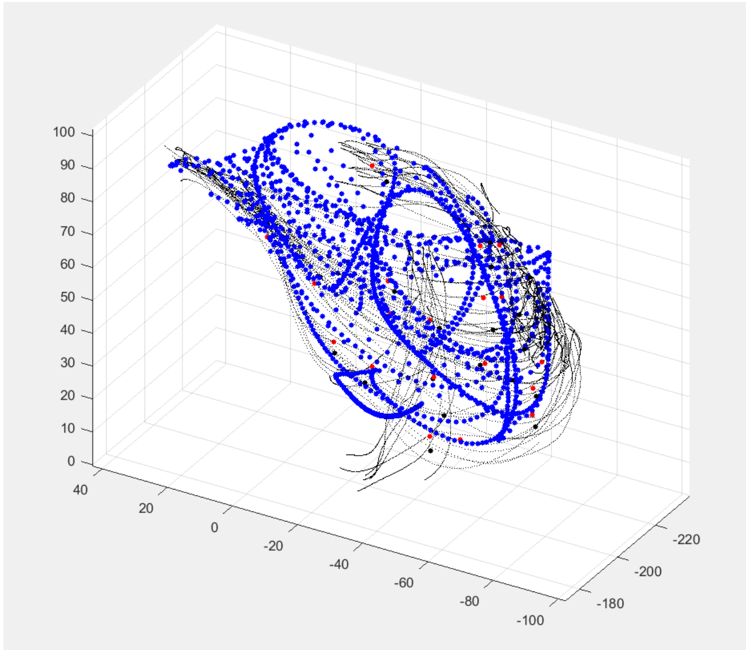


Figure 3.7: 2-4R parallel workspace (blue) on top of the shoulder workspace(black) for subject 2

tically by increasing the number of design points, we are confined to select a limited number of points. On the other side, as the goal is to do an approximate synthesis we can compromise the low number of design points in the first place with having a designed workspace that is consisting of thousands points itself. As a matter of fact, choosing 20 points out of thousands is not a big problem, from design point of view, if the selected points are covering all parts of the target workspace marginally. Besides the robot may be able to pass through points between the design points.

To see how close the workspace of the robot to that of the shoulder complex is, the minimum distance between each point from robot workspace and the shoulder workspace is calculated and plotted in Figure 3.8. The mean value of the distances in this figure is 5.6 mm which is a reasonable values considering the error sources in the data in the first place. Camera error and skin motion error are some of sources of uncertainties in capturing data points. Therefore, we can conclude that the designed parallel robot can mimic the motion created by shoulder complex fairly reasonably. This means that we can estimate the shoulder complex mechanism with a 2-DOF parallel manipulator with a surface workspace in a 6-dimensional space. The independent variables of this surface might be of type translation, orientation or a combination of both. The effort of the author to fit a surface in the position space was

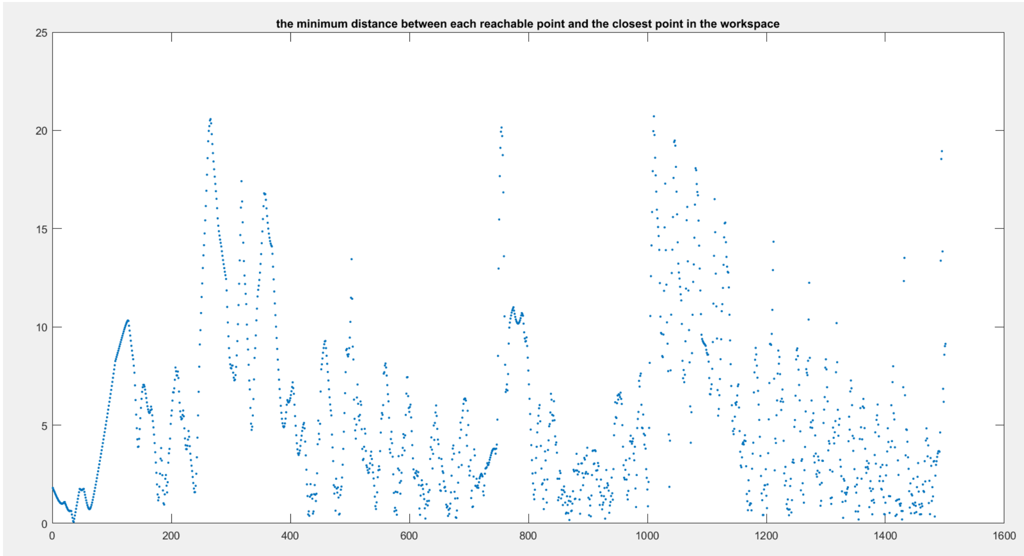
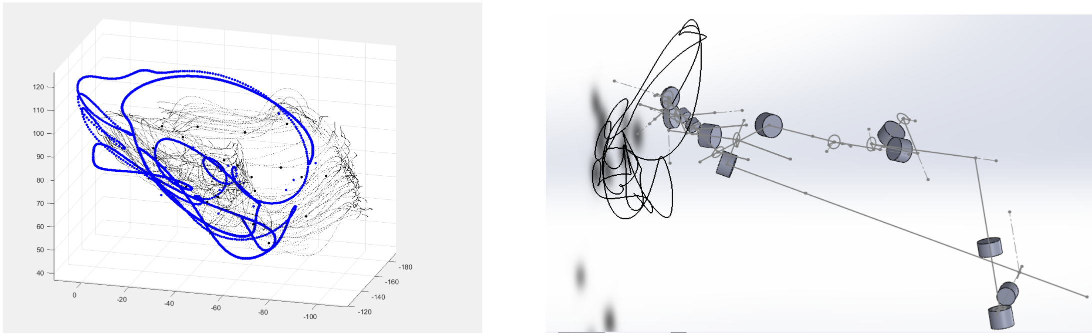


Figure 3.8: Distance between robot and shoulder workspace

unsuccessful which means the independent variables are a combination of all 6 dimensions.

Similar results have been observed in the workspace of the robots designed for all other subjects. It seems that because of the small workspace of the shoulder complex as well as its shape, 2-4R parallel manipulator is a good candidate for estimating a 3-DOF workspace with a lower degrees of freedom mechanism. For all other subjects, it was possible to find a parallel mechanism of the suggested type, 2-4R. As a result we can conclude that a general 2-4R manipulator can mimic human shoulder complex motion. The structural parameters of this robot differs subject to subject based on the dimension of the anatomical parameters but the similarity between the workspace of the robot and the subject's human shoulder remains. For instance, Figure 3.9 shows part of the workspace of the robot on top of the shoulder complex workspace of subject 1. In the same figure, the prototype of the designed robot in 3D simulation environment is shown. In this case, the average value of the distances between the workspace of the designed robot and the shoulder workspace is 8 mm.



(a) Robot workspace on top of that of the shoulder.

(b) Robot worksapce in SolideWorks.

Figure 3.9: 2-4R parallel workspace for subject 1

Chapter 4: General Task-Based Methodology Design

To determine the best mechanism which is able to mimic shoulder motion, the concept of black box was introduced. With a slightly more work, we can generalize that methodology for any limb of the human body which can be called task-based design. In this research, a systematic design procedure is proposed where it is shown that the designed exoskeletons can follow the 3D motions of a human limbs. Geometry of the limb is not the point of focus, but rather the description of its motion is to be investigated.

4.1 Design Process

The design process introduced in this work is a task-based and body-adapted methodology. The input for this method is the human limb motion data captured by any method such as optical-based and some performance constraints. The output of motion capture system is the target motion expressed in discrete data sets which can be used as the input for kinematics synthesis. In this stage, the topology of the system also should be determined where the type and number of joints are selected. Parallel manipulators are reasonable candidates due to their higher robustness, high payload, and lower degrees of freedom, which leads to a lower number of actuated joints. At this point, several solutions may be obtained, and a manual selection of the candidate or an automatic ranking of the candidates according to their kinematic fitness is required.

A link-based hybrid optimization approach is needed to satisfy the performance constraints for the

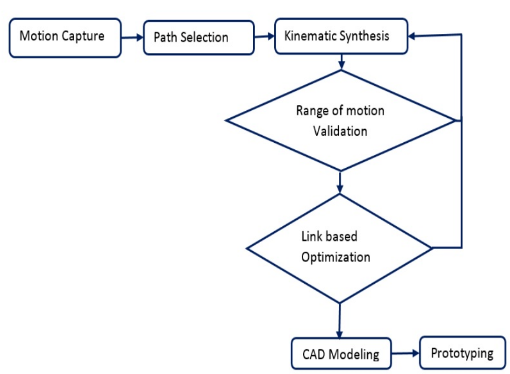


Figure 4.1: The general design methodology.

Figure 4.2: Markers placed on the thumb (left) and data capture setup (middle), and the thumb's proximal phalanx path (right).

designed mechanism. As the designed linkage is to be used as exoskeleton, more optimization in terms of obstacle avoidance, smoothness and limb physical constraints need to be done. The result is a mechanism with determined link lengths and shapes on which the joint axes are known as well. To make the mechanism ready for prototype, a computer-aided design program can be used to apply further details. The general design process scheme is shown in Figure 4.1.

4.2 Human Motion Capture

Discrete finite poses describes the path that the end-effector of a kinematics linkage should pass through. This approach has been one of the popular method to the kinematics synthesis ([Ang82], [CYG12]). These finire poses for anthropomorphic tasks are captured by the cameras that detect the markers on human limb [Ahs08] or infrared technology, as demonstrated in [SB06]. As it was stated in the previous section, Vicon motion capture system with Nexus software has been used for the present work.

The set up for the infrared cameras were set around the room, primarily for larger applications; however it worked very well for the hand motions too. The markers that are used in the system are small white balls that reflect the infrared light. For instance, for one of our thumb exoskeleton designs we used arrays with the markers placed 1.25 inches apart, making it easy to collect data in the three dimensions. In order to assess the exact location of the fixed link with respect to the hand, additional sets of sensors are placed on the arm, see Figure 4.2.

The captured poses are the location of white balls that reflect the infrared light placed on human limbs. Figure 4.3 shows several paths repeating the same task expressed in reference frame. The path

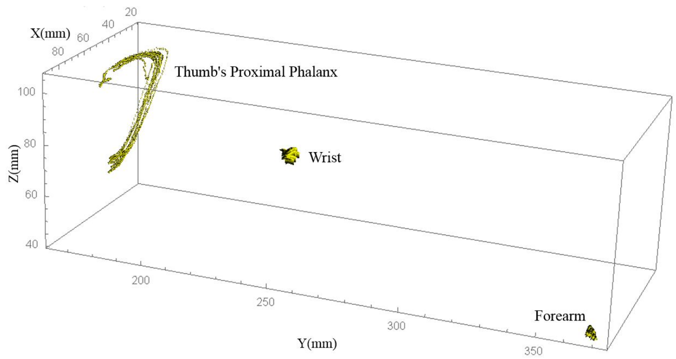


Figure 4.3: The paths of the captured frames including thumb's proximal phalanx, wrist and forearm, all expressed in camera frame.

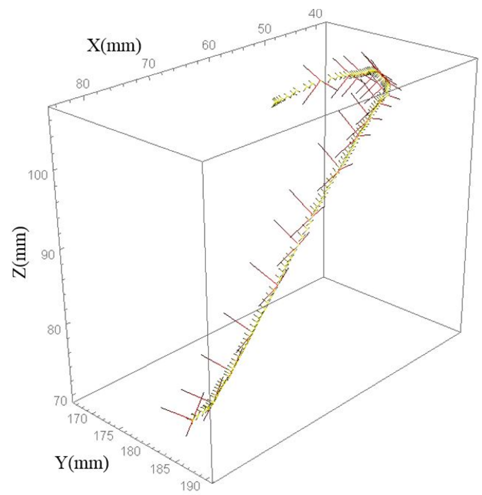


Figure 4.4: Selected path for the thumb's proximal phalanx. The red bigger frames in the figure shows the selected ones along the path.

that has less noise and covers the sufficient range of desired motion is selected, frames with bigger size and in red color in Figure 4.4. The pose including the position and orientation of the limb, obtained from this stage will be used as an input to the kinematic synthesis, in which the design equations are formulated for the selected linkage to fulfill the desired workspace of the human motion.

4.3 Kinematics Synthesis

The number of solutions for kinematics synthesis can increase rapidly by increasing the number of joints in a serial chain ([PGM06]). This provides more flexibility for designers in terms of having more options to choose from. As it was stated in previous sections, the process of kinematic synthesis can be divided into two sections i.e. type synthesis and dimensional synthesis [HD64]. In the first step, the number of joints and their types as well as the connectivity between them are determined. The second step can be divided itself into two categories: exact synthesis, where the end-effector can reach to all the desired poses exactly and approximate synthesis where the robot performs the task with some acceptable errors.

4.3.1 Type Synthesis for Exoskeletons

There is no fixed procedure to do the kinematic topology for all different cases. Based on the nature of the problem, shape and dimension of the desired workspace ([PG10]) or fitting to a particular one-dimensional workspace [LZGP13], the type and number of joints are selected. However, because human joints are just approximations of the subgroups of motion and human variability for a simple selected task makes on motion identification the problem becomes even more challenging.

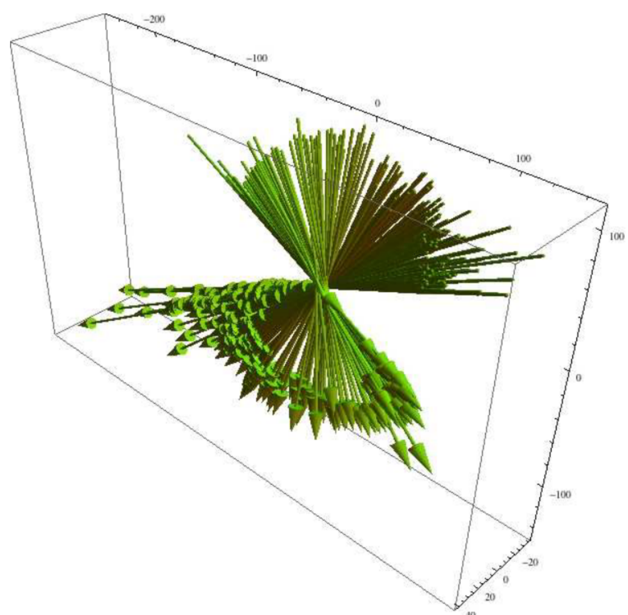


Figure 4.5: Thumb's proximal phalanx path: screw surface of relative screw axes

We need to take two components into consideration for integrating the type synthesis process in exoskeleton design. First one is a collection of mechanism topologies as an atlas and the second one is an optimization to fit the captured motion to the algebraic workspaces. Lacking these components, we have to rely on a selection mechanism which is based on DOF and shape of the workspace. Analyzing the workspace of a limb can help designers narrow down the vast number of available mechanisms.

In particular, Figure 4.5 shows the screw axes of displacements of the thumb's proximal phalanx path. These screw axes have pitches that are proportional to the lengths of the screw lines. The screw axes of the displacements with their pitches generate a *screw hypersurface*; each screw is a transformation from a reference configuration. In this representation, the rotation values are not appearing but we can calculate them separately. Similarly, Figure 4.6 shows the same information for a Bricard mechanism, which is a closed chain with one degree of freedom. Comparing these two sets of screw axes, reveals that Bricard mechanism can be a good replacement for the thumb's proximal phalanx as they create similar surfaces.

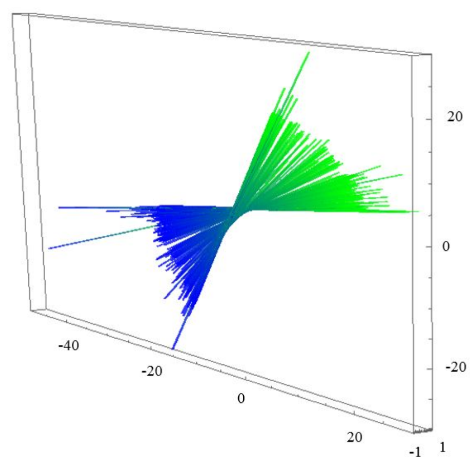


Figure 4.6: Screw workspace for a Bricard mechanism.

4.3.2 Dimensional Synthesis for Exoskeleton

Exact dimensional synthesis is not usually applicable for designing exoskeleton as the number of desired poses is high. Consequently we use approximate dimensional synthesis where the goal is to design the robot so that its end-effector follows the task trajectory as closely as possible. This means that the robot will not fit the workspace of the limb but the error can be reduced by optimization methods that aim at minimizing the distance between the desired path and the exoskeleton trajectory.

For serial chain, the kinematics equations are formed by applying the transformations of all joints from the base to the end-effector. On the other hand, the design equations for parallel robots include constraints to impose the loop restriction. Moreover, another approach for parallel linkages is to consider each loop as a combination of two serial chains that have the same end-effector.

The kinematics equations are written in terms of the joint parameters and structural dimensions of the robot. Denavit-Hartenberg is popular method which assigns the local joint coordinate frames to define the kinematic equations [Cra89],[Tsa99] or matrix exponentials to define the 4×4 homogeneous transformations.

The joint axes are expressed as lines using the Plucker coordinates. For every joint i, the plucker coordinates are defined by $S_i = \mathbf{s}_i + \varepsilon \mathbf{s}_i^\circ$ where $\varepsilon^2 = 0$, the first three-dimensional vector, $\mathbf{s}$, is a unit vector defining the direction of the axis and $\mathbf{s}^\circ$ is the moment and is obtained as the cross product of a point on the axis, $\mathbf{c}$, and the direction $\mathbf{s}$. The unit screw of this line is $J = \mathbf{s} + \varepsilon(\mathbf{s}^\circ + p\mathbf{s})$ which by adding the pitch $p = t/\varphi$ relates the translation along and the rotation about the line together. Then the motion can be expressed as the representation-agnostic exponential, see Equation 2.1 where it represents a revolute joint if $t = 0$ and a prismatic joint if $\varphi = 0$.

The synthesis equations for a serial robot are obtained from the kinematic equations. For n number of joints, $S_i, i = 1, \ldots, n$ and m desired absolute positions $P^j, j = 1 \ldots m$, Equation (4.1) defines the workspace of the robot for $j \geq 2$.

$$\begin{aligned}
S_1(\theta_1^1)\ldots S_i(\theta_i^1)\ldots S_n(\theta_n^1)P^1 &= P^2 \\
\vdots \qquad\quad \vdots \qquad\quad \vdots \qquad\quad & \ \ \vdots \\
S_1(\theta_1^j)\ldots S_i(\theta_i^j)\ldots S_n(\theta_n^j)P^1 &= P^j \\
\vdots \qquad\quad \vdots \qquad\quad \vdots \qquad\quad & \ \ \vdots \\
S_1(\theta_1^m)\ldots S_i(\theta_i^m)\ldots S_n(\theta_n^m)P^1 &= P^m
\end{aligned} \tag{4.1}$$

For a given set of task positions, the goal is to find the dimensions of the robot that can position the end-effector at the given set of task positions. In other words, for each position P^j, there is at least one

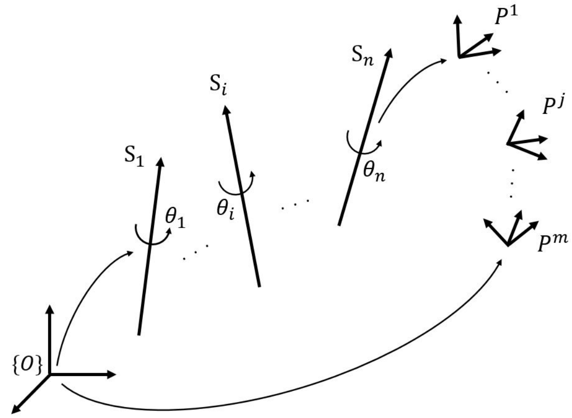


Figure 4.7: A serial chain, its joints axes and desired poses.

joint parameter vector $[\vartheta_1^j, \vartheta_2^j, \ldots \vartheta_n^j]$ and a set of structural parameters such that the robot chain can reach to the desired task .

An exoskeleton's mechanism can reduce the performance by adding weight, inertia and friction to the system. According to [RM05], the natural frequency of the swinging of a body is affected by the mass and inertia of the exoskeleton, which can have important consequences on the metabolic cost and the speed of the body [BMKG07]. Compensating inertia through control is particularly difficult due to instability issues [BH07], [New]. In order to overcome these problems, and to fulfill additional performance requirements, such as total length, force transmission, obstacle avoidance or geometry at a given configuration, a post-synthesis optimization method can be used [YYPG13]. This method is based on considering the links as anchored to sliding points on the set of joint axes, and making the additional requirements be a function of the location of the link relative to the two joints that it connects.

The combination of the kinematic synthesis together with the link based optimization allows to interactively monitor, control and adjust objectives and constraints, to yield practical solutions to realistic exoskeleton mechanisms design.

When dealing with the synthesis of parallel robots or closed kinematic chains, the fact that several serial chains need to follow the same path can be dealt with in several ways. Optimization techniques based on minimizing the distance to the desired trajectory can be accompanied with assembly constraints consisting on inequalities on the loop equations of the mechanism, see for instance [JH06], in order to approximate the closure of the mechanism. Another techniques include designing for entire workspaces of closed chains, such as [BBPG14], however this approach is only available for a few mechanism topologies currently. A third way is to *reduce* the parallel mechanism to the most constrained serial chain, see [SSPG14b], synthesize for that serial chain and let the rest of the serial chains adapt to the motion of this one. When all serial chains joining the base to the end-effector present the same level of constraint to the motion, then synthesizing for one of them allows the exact reach of a finite set of positions. In all of these methods except the exact workspace synthesis, the kinematic behavior of the system is defined at discrete poses, and no information about the motion from position to position will be available. An extreme case will happen when a singularity is found between the discrete positions being targeted.

A different approach is the one taken for instance in [SR13], in which the human body is considered a kinematic chain and additional chains are attached to it in order to restrain the motion, forming a closed system. This technique is effective for human limbs in which the joints' motion faithfully approximate the motion of a mechanical joint such as a revolute joint. Its results on more complex joints have not been assessed yet.

4.4 An Example of Exoskeleton Design for Thumb Motions

The human thumb presents a complex 3D motion that can be modeled, depending on the needed accuracy, with three to four degrees of freedom, and using variable joint axes. We postulate that it is still possible to use simplified, low-dof linkages for assisting in particular thumb motions. As candidate mechanisms we focus on a set of closed, spatial overconstrained and non-overconstrained four-bar to six-bar linkages with low mobility that present the desired characteristics for this application, see [Wal73] and [Tsa00]. The spatial mechanism is to be attached to the proximal phalanx of the thumb. In addition, the designed mechanism is confined to the back of the hand, so as to minimize sensory feedback interference, and to allow the mechanism to be manufactured with minimal size. This, combined with the intended location of the actuators, will allow the device to be constructed with low apparent inertia.

For this example, we synthesize the selected linkage to follow as closely as possible experimental paths of the human thumb. The overall outline or design approach of the mechanism is shown in Figure 4.1, however in this case the post-synthesis optimization step has been omitted.

The thumb data were acquired using a Vicon motion tracking system as shown in Figure 4.2. The captured data for several paths doing the sane task and the selected positions are shown in Figure 4.3, 4.4.

For the design of spatial motion, it is sometimes advantageous to work with relative displacements. Each relative displacement expresses a motion of the thumb from a reference configuration, taken as the thumb position at the first frame. Each displacement can be modeled as an axis, plus a rotation about and a translation along the axis. This information is encoded as a *screw*, where the screw axis is the axis of the displacement and the pitch is the ratio of translation to rotation for that displacement, see Figure 4.5.

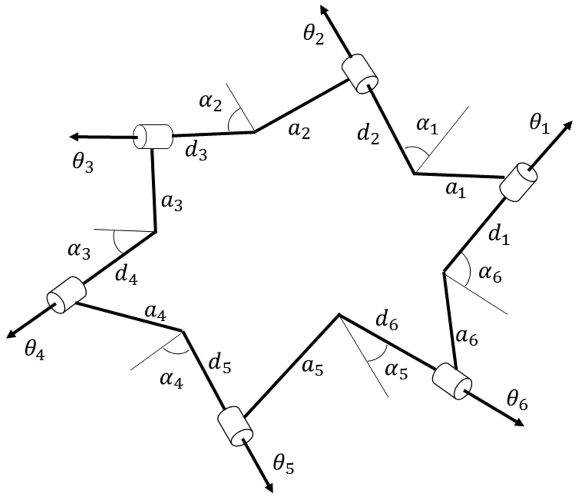
4.4.1 Mechanism Selection

In order to accomplish simplicity together with spatial motion under a one-degree-of-freedom system, an initial set of closed spatial linkages with four to six links and standard revolute (R), prismatic (P) and cylindrical (C) joints have been selected. Some of these linkages are overconstrained, while others are trivial; all of them with mobility equal to one [Wal73], [Tsa00].

In particular, the following four-bar linkages: RC-CC, RP-RP, RR-RR, and the following six-bar linkages: CRR-RRR, RRC-CRR and Bricard [Bak80], which is an RRR-RRR mechanism, were selected as candidates. Here, the dash separating joints indicates where the end-effector, or attachment to the thumb, is being placed.

Among the properties of these linkages that are useful for our application we can cite the 1-dof motion, requiring only one actuator, and topological simplicity while creating a complex motion. In addition, overconstrained linkages have other advantages, such as inherent structural rigidity.

The workspace of relative displacements of the candidate linkages was plotted and the Bricard mechanism showed the highest match with the set of displacements of the thumb, see Figures 4.5 and 4.6.



	α	a	d
1	α_1	a_1	d_1
2	α_2	a_2	d_2
3	α_3	a_3	d_3
4	$\alpha_4 = \alpha_1$	$a_4 = a_1$	$d_4 = d_1$
5	$\alpha_5 = \alpha_2$	$a_5 = a_2$	$d_5 = d_2$
6	$\alpha_6 = \alpha_3$	$a_6 = a_3$	$d_6 = d_3$

Figure 4.8: The Bricard mechanism with its DH parameters and dimensions. The table shows the relations among DH parameters of a Bricard mechanism.

The Bricard was selected for dimensional synthesis.

4.4.2 Bricard Kinematics Synthesis

In this section, the design equations corresponding to the Bricard mechanism are presented. Consider the closed RRR-RRR linkage as two serial RRR chains, joined at their end-effectors. The axes are labeled as shown in Figure 4.8, starting at the fixed link and going around in two ways to reach each other at their end-effectors. The table in Figure 4.8 shows the constraints among DH parameters in a Bricard mechanism, in which a_i is the length of link $i+1$, α_i is the twist angle between the axes of joints i and $i+1$, d_i is the offset distance at joint i and ϑ_i is the joint rotation angle for $i = 1, \ldots 6$. It is known that the offset distances, twist angles and link lengths in the opposite side of this mechanism must be equal [WC10] resulting in $\vartheta_4 = \pm\vartheta_1, \vartheta_5 = \pm\vartheta_2, \vartheta_6 = \pm\vartheta_3$.

In order to create the design equations, the distance between the displacements captured and the displacements of the candidate chain is minimized, with the goal of finding the location and dimensions of the mechanism that approximately performs the task.

The design equations are created by equating the forward kinematics of the mechanism (Both left and right chain) to each of the discrete positions obtained from the motion capture. If each finite displacement of the thumb is denoted by P^j, then the design equations are

$$\begin{aligned}
&Left:\\
&S_1(\theta_1^j)S_2(\theta_2^j)S_3(\theta_3^j)P^1 = P^j\\
&Right:\\
&(S_6(\theta_3^j))^{-1}(S_5(\theta_2^j))^{-1}(S_4(\theta_1^j))^{-1}P^1 = P^j, j = 1, \ldots, m.
\end{aligned} \tag{4.2}$$

In these equations, the variables we are interested in are what we call the *structural variables*, which are the Plucker coordinates of the joint axes at the reference configuration. In addition, the optimization process outputs the angles of the chains in order to reach the thumb displacements.

Moreover, the Bricard constraints in terms of the joint axes are:

$$\begin{aligned}
&\mathsf{S}_1 \cdot \mathsf{S}_2 = \mathsf{S}_4 \cdot \mathsf{S}_5\\
&\mathsf{S}_2 \cdot \mathsf{S}_3 = \mathsf{S}_5 \cdot \mathsf{S}_6\\
&\mathsf{S}_3 \cdot \mathsf{S}_4 = \mathsf{S}_6 \cdot \mathsf{S}_1\\
&(\mathsf{S}_6 \times \mathsf{S}_1) \cdot (\mathsf{S}_1 \times \mathsf{S}_2) = (\mathsf{S}_3 \times \mathsf{S}_4) \cdot (\mathsf{S}_4 \times \mathsf{S}_5)\\
&(\mathsf{S}_1 \times \mathsf{S}_2) \cdot (\mathsf{S}_2 \times \mathsf{S}_3) = (\mathsf{S}_4 \times \mathsf{S}_5) \cdot (\mathsf{S}_5 \times \mathsf{S}_6)\\
&(\mathsf{S}_2 \times \mathsf{S}_3) \cdot (\mathsf{S}_3 \times \mathsf{S}_4) = (\mathsf{S}_5 \times \mathsf{S}_6) \cdot (\mathsf{S}_6 \times \mathsf{S}_1)
\end{aligned} \tag{4.3}$$

where $\times$ is the dual cross product of dual lines.

Eighteen positions were selected from the thumb path, and the first frame was taken as the reference configuration. If the forward kinematics is written by dual quaternion, each equality is composed of 8 equations for each of two serial chains composing the mechanism. This gives a total of 272 nonlinear equations. In addition, we have the constraints Equations (4.3) which add 12 to the system of equations. In sum, we have 284 equations.

The variables to solve for are the Plucker coordinates of the axes, that is, six parameters per axis per chain ($2 * 3 * 6$), and the joint variables to reach each thumb position ($17 * 3$). Then the total number of unknowns is 87. This over-constrained system of nonlinear equations were solved using Levenberg-Marquardt method.

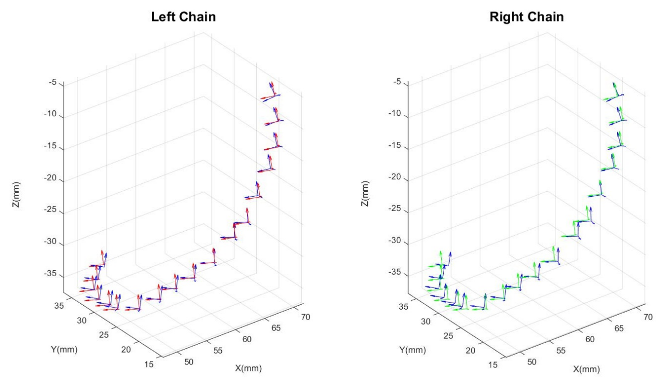


Figure 4.9: Comparison between desired positions (blue frames) and linkage positions (red for the left chain and green for the right chain)

4.4.3 Prototype

The set of selected positions and the ones reachable by the synthesized robot are shown in Figure 4.9. As it is seen, both chains are reaching to the targets fairly well. The equations were run 14 times for three different sets of positions chosen from the thumb frames. The distance to the desired path has been optimized by minimizing the distance at each step. It took a variable amount of time, from a few minutes to a few hours to find solutions. For the 14 runs, 14 considerably different solutions were found.

	α (degree)	a (mm)	d (mm)
1	165.607	36.487	94.384
2	279.412	160.897	20
3	264.546	166.603	40.763
4	165.607	36.482	94.384
5	279.412	160.897	20
6	264.546	166.603	40.763

Figure 4.10: The DH parameters of the resulting mechanism. Angles are in degrees and distancses are in milimeters.

Out of these 14 solutions, 1 linkage was selected based on its overall dimensions and placement on the hand. For this solution, the translational error which is the distance between the desired and computed end-effector positions, varies from 1 mm to 3 mm while the orientation error, using Euler angles as a criterion, varies up to about 20° for α, 18° for β, and 30° for γ. Figure 4.10 shows the DH parameters of the designed robot. Due to the potentially very large number of solutions for this problem, not all the solution space has been searched and hence we cannot assume that the selected candidate is the optimal one, but rather an acceptable one. The SolidWorks model at the initial position is illustrated in Figure 4.11. The cylinder shows the forearm position on the mechanism. There are some adjustable bars to fix the forearm on the base to make sure that it is positioned in the right place.

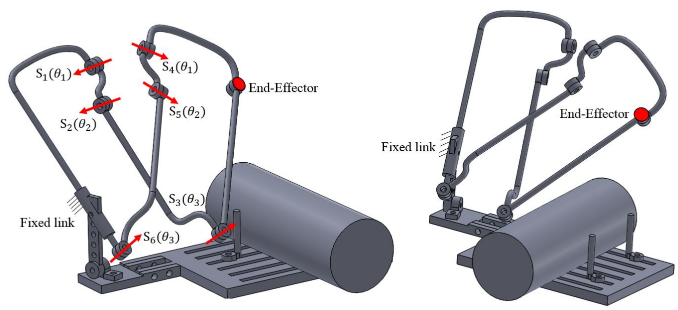


Figure 4.11: Solidworks Model of the designed robot in reference configuration.

Figure4.12 shows the prototype of the Bricard mechanism which was designed based on the proposed methodology to follow the thumb's path.

Figure 4.13 shows the same mechanism with the hand on it. The mechanism is designed so that the height and orientation of the base is adjustable to provide a good tolerance for different sizes and shapes of forearms and hands.

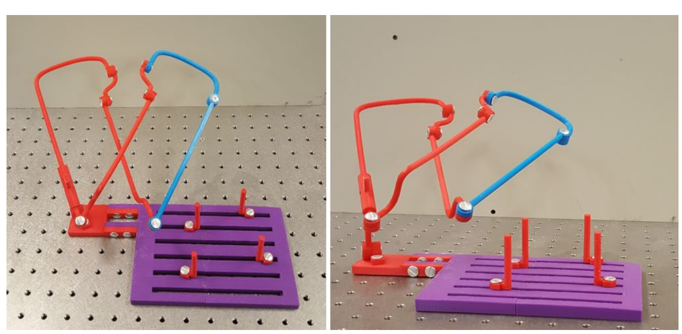


Figure 4.12: The prototype of the designed Bricard mechanism with the fixture of forearm attaching to the base.

This work presents a method to design lower-mobility exoskeletons for specific sets of human motion. The lower-mobility exoskeletons require less actuation, which in turn results in a lighter exoskeleton. In this method, only the path of the target limb is required in order to calculate the dimensions and position of the exoskeleton. This path can be obtained using motion capture technology. The resulting design is non-anthropomorphic, which allows us to locate the joints away from the limb if required.

As an example, a one-degree-of-freedom mechanism is designed to be used as exoskeleton to guide the motion of the proximal phalanx of the thumb finger for applications such as rehabilitation. The resulting design matches the motion of the finger within its range of motion.

Further work will be devoted to the post-synthesis optimization in order to obtain a more compact solution.

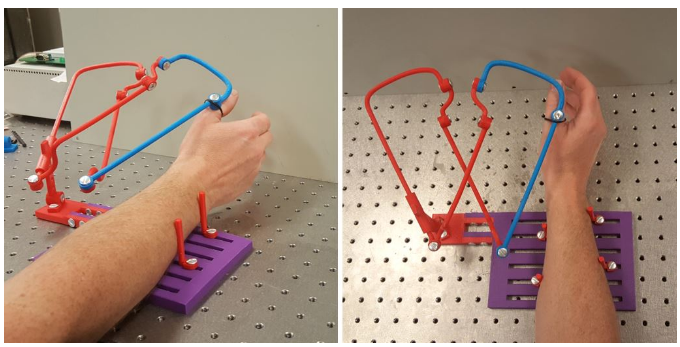


Figure 4.13: The prototype of the designed Bricard mechanism with the subject's hand on it.

Chapter 5: Virtual Reality and Its Applications in Rehabilitation and Robotics

Based on *Webster's New Universal Unabridged Dictionary* Virtual means something that is in essense or effect but not in fact while reality means a place that exist and we can experience [20119].

The key point of Virtual Reality is immersion. This is basically a physical immersion in comparison to mental immersion where the brain is in a state of being deeply involved. Physical immersion means bodily entering to a medium by stimulating the human senses [20119]. VR, in the current level of technology, mostly stimulate the user's eyes and creates an illusion of the reality. Because of the advanced technology used in VR, the illusion is highly convincing and can be used for many different purposes. On the other hand the interaction with computer programs such as games and engineering software is beyond mouse and keyboard. The user can uses their own hands to manipulate the virtual object and have a much more natural and intuitive communication. VR environment does not have real world limitation and can expand as human imagination. Among many applications of VR such as telepresence and collaboration environment, the focus of this chapter is on the rehabilitation and robotics. The immersing VR environment can help therapist to improve the rehabilitation process by creating an illusion of the motions of stroked limbs. On the other hand, robot designers can benefit from such interactive environment towards creating a perfect model before prototyping. Providing direct sensory feedback, the illusive VR gives the possibility of testing any device before it comes to the real world.

In this work, we use a VR system developed for upper-arm motion identification and immersing user experience with a kinematic synthesis process consisting of chain topology enumeration and selection and a numerical dimensional synthesis solver. VR with depth sensing allows the easy creation of 3D motions and, unlike infrared motion capture, the interactive selection of the important parts of the motion in real time. The input from the user's motion is utilized to match a kinematic chain with the required end-effectors. The information is sent in real time to the solver, which efficiently computes a solution within a few seconds. The output solution can be displayed and animated within the virtual system simultaneously to the user's motion. In this submission we present the results of task creation using the human hand, the communication among the different parts of the system and the display of the solution designs. These results will build into a complete VR-based system for the design of proprio and extero limbs, for applications in exoskeletons, rehabilitation, prostheses and supernumerary limbs.

On the other side, one of the difficulties in rehabilitation and therapies is training the motor system of patients who lost their ability to move their body parts due to strokes. In traditional ways, the session of retraining is long and sometimes as not effective and in many cases, limited success is accomplished. The goal is to improve this process by using Virtual Reality as a means to help overcome some of the proprioceptive loss that stroke patients experience. The strategy is to map the patient's healthy limb on to his/her affected limb to test the action-observation linked to the mirror-neuron system.

5.1 VR Concepts and Tools

The key point in a good Virtual Reality experience is overriding human sensors in order to send desired signal to the brain instead of what the brain receives from real world. We, our brains, experience the world through senses. Basically, the brain can not distinguish between reality and imagination or virtual world as long as the signals that reach the brain are correctly and in accordance manipulated. Among human senses, the eye sight is the most powerful one in terms of understanding the environment and that is why eyes are the target of VR hardware to manipulate. However eye is not the only sense that can get tricked, we can expand the concept of virtual to any other senses such as tactile or hearing. As far as the information received by one sense is not contradicting the others strongly, the brain would not care where the sources of the signals are and would react the same. Thus, the duty of VR hardware is to stimulate these senses in order to trick the brain. Stimulating one sense and ignoring the others sometimes create bad feelings in human such as nausea.

Another important concept in VR is rendering. A virtual world can be created by any type of software but the essence of the matter is how the projection of this world to display is being done so the human senses get stimulated correctly (Figure 5.1). If the projected virtual world is fixed to the display, then all the virtual objects move as the headset moves. However, to trick the brain and stimulate the eye sight in order to create a feeling of being inside another medium, we need to have stationary objects. These objects should not move or in a better word, they should move in a way that makes the user believe they do not move. And that takes place by couter-transformation. If the head moves to the right the rendered virtual object on display needs to move left in order to create the illusion of be-

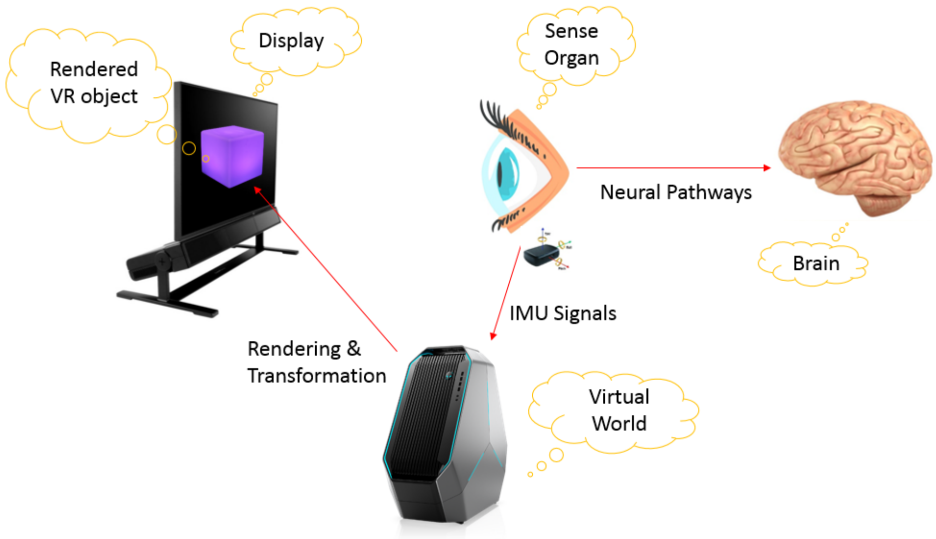


Figure 5.1: The main component in a Virtual Reality System.

ing stationary. This is the heart of creating a convincing virtual world; having stationary, moving and fixed-to-headset objects. Now if this counter-transformation is not done properly, it creates distorting stimulation which results in VR sickness. Nowadays, the technology is good enough to compute the correct transformation and displaying it without any delay while in the past this was a more challenging problem.

Figure 5.2: (Left) Microsoft Kinect, (Middle) 3D Scanning Structure Sensor, (Right) Intel Realsense Depth Camera.

Sources: www.engadget.com, www.techmed3d.com, in.rsdelivers.com

Tracking system is another essential part of any VR system. The pose of human organ which is the target of the VR hardware should be calculated for the localization purpose. The orientation of the pose is on inertial measurement unit or IMU which usually includes gyroscope, accelerometer and magnetometer. Cameras are also used for the tracking systems but the challenge is to find the anchors on the captured image to keep the track of the objects without using markers. Depth-sensing cameras, on the other hand, are used to help tracking systems by projecting light to the objects and observing the reflection. Some of these cameras are depicted in Figure 5.2. Leap Motion is another depth-sensing device consisting of two cameras and three infrared LEDs which can track human hand and fingers. The software of this technology creates a 3D representation of the nearby environment and extract the hand motions, see Figure 5.3.

Figure 5.3: Leap Motion Camera.

Source: www.hackmag.com

VR Headsets are the units to that carry the display and the sensors for localization purposes. Among the popular ones, Oculus Rift, HTC Vive, google Cardboard, google Daydream and Oculus Go and Sony Play Station VR are worth mentioning, see Figure 5.4.

Nevertheless, the software that process the data received from sensors and create the proper output for a good virtual reality experience is a crucial part of the system. Game engines adapted for VR are the most popular platform for developers. Unity3D is the most accepted game engine, at this point of time, due to its high-level operations. Some of the features of a proper VR software includes: creating high quality graphics that create the right illusion of the real counter part, maintaining a good correspondence between motion in the real wold and virtual world, geometric aspect, physics to create real behavior for virtual objects, collision algorithm, localization and mapping algorithms, lighting, occlusion and performance optimization. Explaining all these concepts are out of the scope of this dissertation, the reader is referred to book "Virtual Reality" by Steven M. LaValle [lav] for detailed description of different topics in virtual reality.

Figure 5.4: Popular VR headsets. From top left to the bottom right: Oculus Rift, HTC Vive, google Cardboard, google Daydream and Oculus Go and Sony Play Station VR

Sources: www.republiclab.com

5.2 Uses in Rehabilitation

Human action perception and action execution are directly related based on recent studies. Also visual feedback of performance can help the process of training and retraining the limbs. This section is devoted to how VR can enhance this process by allowing the user to be immersed in an environment where they can visualize the healthy motion of their arm. An important part of the AR system is the correct and user-tailored identification of the upper-limb kinematics, as well as the reduction of complexity of the model for faster rendering. The following sections describe the VR tools used and the VR environment that are made.

5.2.0.1 VR requirements

In this work case, human vision is targeted by VR hardware and the goal is to edit visual input of the objects inside the virtual world. Oculus Rift is used as the device to display the virtual world and im-

Figure 5.5: Leap motion camera mounted on Oculus Rift headset to synchronize with the tracking system and provide 3D hand models which are able to interact with virtual objects. Picture from

Source: www.primitivebuteffective.com

merse the user. This device has fast tracking and rendering so that the virtual limb approaches a realistic merging in the real environment. The technology used in Oculus Rift provides features such as fast realistic photo rendering, motion smoothness, shadow casting, estimation of appropriate transformation and scaling parameters of the virtual limb to smoothly align with the object(s) in the real environment.

A motion tracking system is needed to capture the subject's limbs motion in real time. Systems that are commercially available include magnetic, acoustic, inertial or optical systems. In general, optical systems that capture the markers placed on the body may be a good option but transferring the data of the markers to simulate the human limb needs a lot of work while it creates a lot of error due to skin motions, camera errors. In addition, synchronizing the captured data with the Oculus tracking system is another difficulty. Whereas Leap Motion camera is designed to detect the hands and forearms whose 3D model are available in real time. Additionally, it can be mounted on the Oculus Rift goggle and work in accordance with the headset tracking system. The software developed for Leap motion provided texture mapping for human hands that is one of the main requirement to convince the user to accept the 3D model as one of his/her body part. As a result, Leap Motion technology combined with Oculus provides perfect interactive virtual environment for the healthy and stroke patients to do the experiments in, see Figure 5.5.

The workspace of human upper-limb motion needs to be considered before recording the tasks in

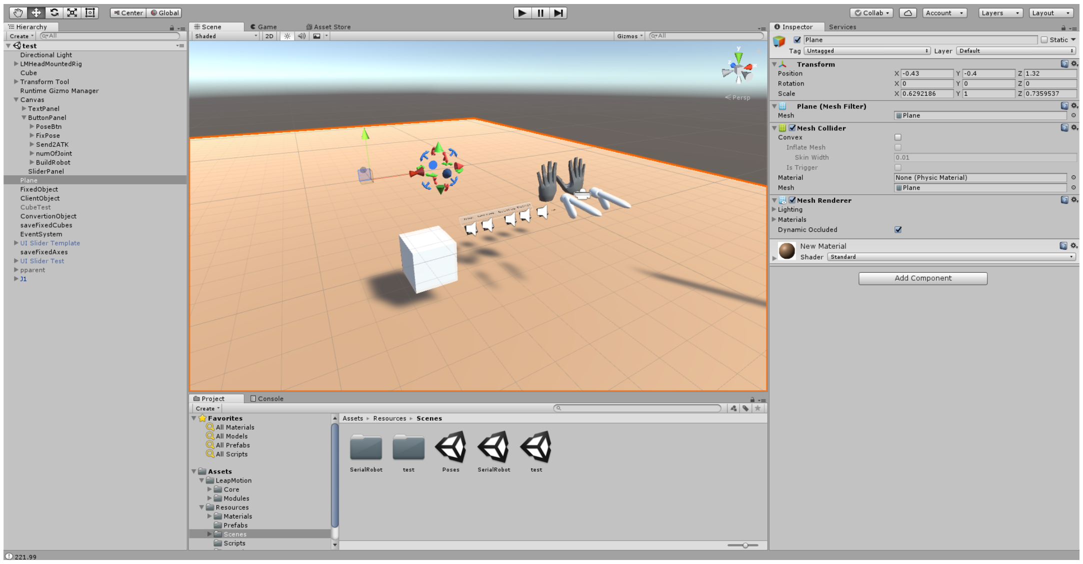


Figure 5.6: Unity3D game engine editor.

the 3D-space [LU94]. Once an upper-limb workspace is modeled, then tasks can be determined based on work that has examined action-observation priming with both human and robotic displays.

In addition to the capturing data of the hand and tracking systems, we need a platform where the 3D models and the physics are taking care of. In the last two decades, with emergence of computational hardware and 3D modeling software tools, the human anatomy has been studied extensively which has found many applications in different areas such as forensics, surgical training and simulation, and human animation in entertainment.

Unity3D is a game engine adapted for VR to which VR goggles and Leap motion can be connected (Figure 5.6). Some of the features of this platform:

- accessing the low-level graphics API which enables developers to take advantage of the latest GPU and hardware improvements.
- building shaders visually instead of pure coding.
- Scriptable Render Pipeline which enables the developers to modify the rendering process towards their needs.
- networking and multiplayer options

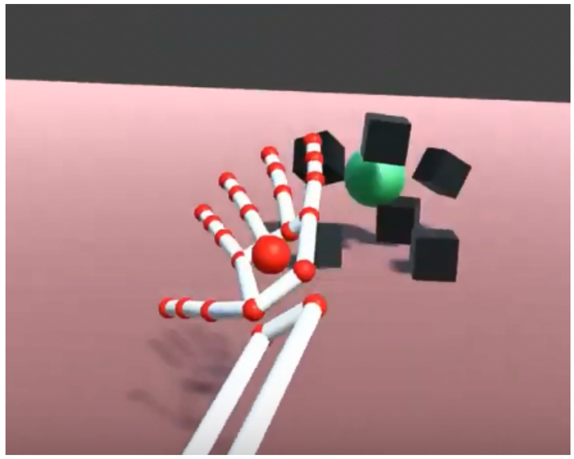


Figure 5.7: Ball Throw

- giving real-world behavior to game objects through its physics engine. It provides components like ”rigid body” or ”kinematics” that can be added to any game object to change their behavior.
- graphical user interface to build many different components.
- scripting which allows to define logic.

5.2.0.2 Games

The goal of the experiments is to stimulate the mirror neuron system to facilitate the training process. In the first place, healthy subjects are asked to perform some predetermined tasks including hand gestures and reach-grasp repeatedly using the system described above. These are the motions that are used in daily-basis of people lives. These experiments will be repeated for stroke patients who have difficulties in moving their upper arm.

In all the games, The cube has collision component so it can be interactive and picked up by virtual hands that are provided by Leap Motion. Also the ball needs the rigid body component to behave as it is in the real world. A unity package is provided by Leap Motion where all the needed components such as reading the raw data coming from the camera and interpreting it to detect hands and also making the virtual hand in the virtual world are prepared.

The first game built is called ”Ball Throw” (Figure 5.7) where the user is asked to pick up a virtual

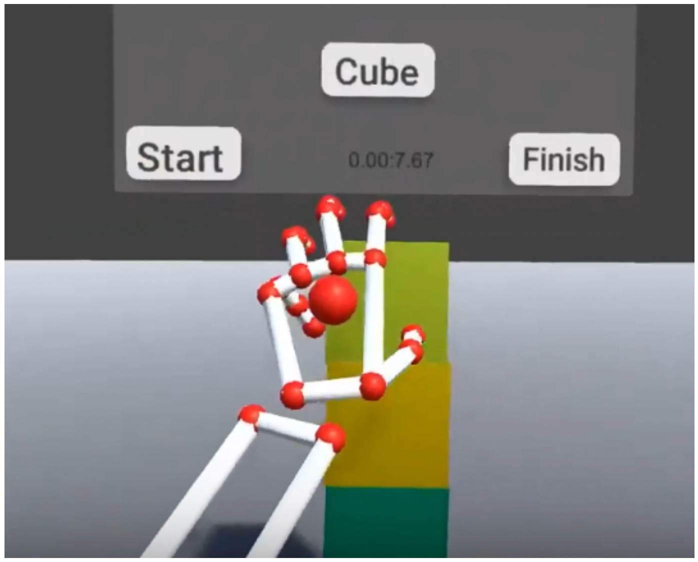


Figure 5.8: Block Stacking

ball and throw it to a stack of cubes. The task is self-paced and requires gross motor movements. Besides, it is a fast movement action where performance will not change by the feedback. The purposes of this game is:

- This is a task that most of people have done in the real world so they are familiar with the concept, but at the same time picking up the ball with the virtual hands need some level of skill and practice. As a result, the aim is skill acquisition and motor learning.
- By switching the virtual models of the hands, a virtual version of ”mirror therapy” is provided where movements of the intact arm maps to the opposite hand to yield action observation of tasks not possible when using a framed mirror.

The second game is ”Block Stacking” (Figure 5.8). This is a self-paced game and allows feedback during the activity to guide changes in movements as they are occurring. There is not a lot to learn in this game for the subject except the manipulation of the cubes with virtual hands which makes it an easy game. Therefore the purpose is acclimation to the virtual environment and creative play.

”Aperture” is the third game (Figure 5.9). In this game the subject is asked to reach through an aperture to a cube, pick it up and bring it back without touching the sides of the aperture. There are settings where the user can adjust the length and height of the aperture to the anatomy of the subject.

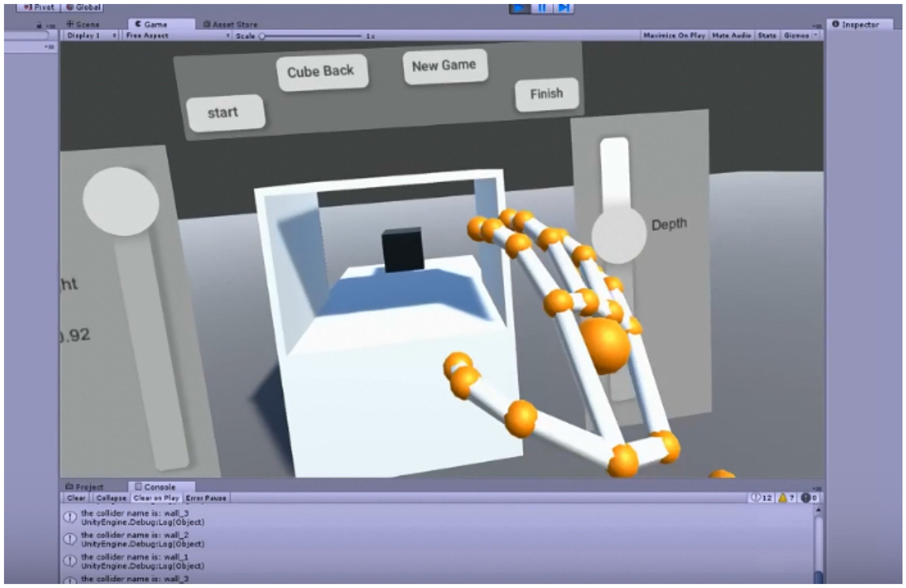


Figure 5.9: Aperture

This task requires fine motor control to reach through an opening and retrieve a cube without touching the sides. The purpose is to develop fine motor control and precision in a virtual environment.

”Cannon Catch” is the name of the fourth game (Figure 5.10). This is not a self-paced game and the subject needs to time the movements based on what happens in the virtual world. The subject is required to catch virtual cannon balls being shot from alternating sides of the environment. The speed and range of the cannon ball trajectories can be adjusted according to the participants abilities. The purpose is motor control with external timing imposed.

5.3 Uses in Robotics

Virtual Reality is a quite new technology that has found its way into many areas such as military, healthcare, fashion, business, sport, education, media and entertainment. Nevertheless, VR has some applications in industry where it is applied to solve or enhance real issues. What follows is the VR uses in industry:

- Simulation and visualization of environment which are hazardous, expensive or difficult to handle or where it is hard to present data and processing.

Figure 5.10: Cannon Catch

- Training. Multiple trainee from totally different locations can be trained by one trainer in an environment in which the work space is simulated.
- Collaboration with sharing a VR space. Developers or employees can share a VR environment to exchange ideas and cooperate on projects.
- Engineering design and prototype where a 3D model can be created, visualized, tested and modified based on the needs.
- Architecture and construction design in which the user can be place inside a house before it is built.

Also VR has been recently used in robotics where the user can interact with and program the robots. One good example of this is RobotStudio from ABB company. The VR option in this software can bring the programmer to a 3D model environment in which all the elements of the robotic system can be modified. This space also can be shared with the customer in case the company would like to present the progress or final product.

The author, however, has been trying to find other applications of VR in robotics. One of the interesting uses is designing a robot is a VR environment that kinematicians can set inputs, solve kinematics equations and see the resulting robot. The important steps of a non-VR design process are shown in

Fig. 2.11. In this work, we utilize the virtual reality capabilities to improve the interaction of the designer in some of these steps, or to replace them with a more user-friendly alternative.

The definition of the task is transformed into an immersive experience by capturing the motion of the hands using a depth-sensing camera, and transforming the user's hands into the VR world by using the SDK made for Unity. The hands can be used to position the task object, as seen in Fig. 5.13, or by directly recording the coordinates of the fingertips at the locations specified by the user.

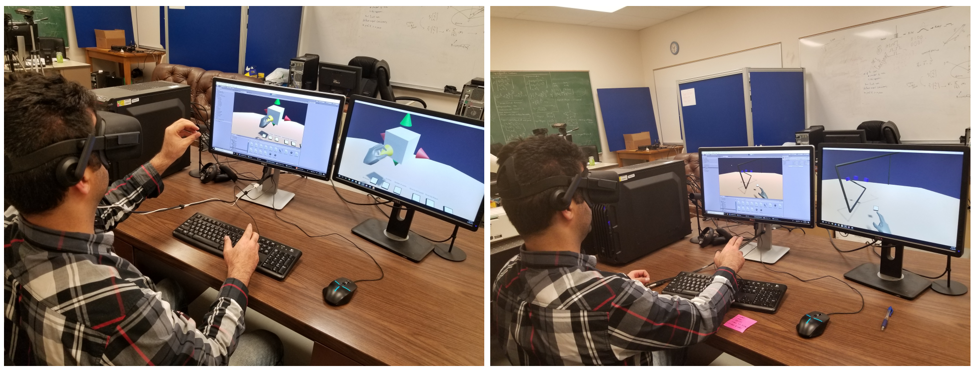


Figure 5.11: The designer creating the task in the VR environment.

Unlike regular motion capture systems, all trials and repetition of a desired motion do not need to be stored as the user and designer can visualize the captured path relative to the environment in real time and repeat the task till they are satisfied. Fig. 5.11 shows the interactive definition of poses in the VR design suite.

This step provides the input to the synthesis process, which implies heavy numerical calculations that need to be performed outisde of the virtual reality system. The captured poses inside the VR world are stored and sent to a numerical solver specifically designed for kinematic synthesis, where the corresponding dimensional synthesis equations are created and numerically solved using a combination of a genetic algorithm and a line search algorighm.

After a solution is obtained, the visualization step greatly benefits from is display in the VR suite. The solution is simulated and placed in the environment, so that the designer can make sure that the designed robot is able to pass through planned path and has the correct dimensions and placement for the task and the surrounding environment. In addition, using the physics engine of the VR space can help the developers in checking the performance and in comparing different prototypes and CAD models.

5.3.0.1 VR Requirements

The system consists of the Oculus Rift goggles, a Leap Motion™ Camera, and the solftware packages Unity® and ArtTreeKS. The environment uses the prepared Interaction Callbacks example from "interaction engine" of the Leap Motion [Winc], see Fig. 5.12.

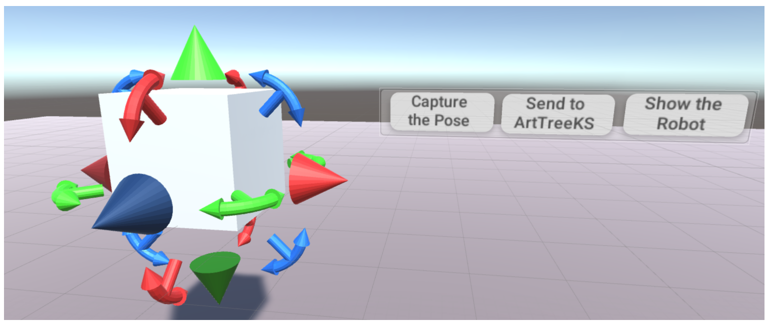


Figure 5.12: An interaction cube from Interaction Engine in the synthesis virtual environment

The motion of the hand is sensed and captured by Leap Motion camera, which is an infrared stereo camera, and the virtual model of the hand is provided in the environment. The poses of different parts of the hand are accessible through Leap Motion API. The cube (guide box) can be translated and rotated by the provided handles on it and the user can grab them and transform the guide box to a new pose. For the objects to work with the interaction engine, the rigid body component should be added as well as having the interaction behavior module added to the scene.

The proposed Virtual environment provides three steps to do the robot synthesis Fig. 5.13. The first step copies a smaller version of the guide box fixed in the space on each single pose that the user chooses by pushing the "capture the pose". These poses are stored as dual quaternions so they can be fed to the numerical solver ArtTreeKS. In the second step, these poses are sent to ArtTreeKS as input, when the designer pushes "Send to ArtTreeKS". This software finds a single numerical solution for the dimensional kinematic synthesis and saves it in a file that will be loaded whenever the user pushes "Show the Robot". There the designed robot is modeled and simulated in the VR environment and the user can

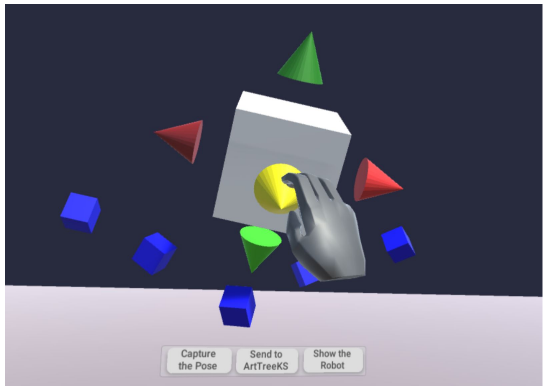


Figure 5.13: Transforming the guide box to get any desired pose

see its dimensions, location and its performance while executing the desired task.

The output of the solver ArtTreeKS contains the Plucker coordinates of the joints in a reference configuration plus the joint values relative to the reference configuration in order to move through the positions of the task. This information is enough to instantiate cylinders that represent the links and joints of the robot, to have a preliminary representation of the system. More sophisticated and user-controlled visualization tools could be added to the system in order to make it more realistic.

5.3.0.2 VR Synthesis of Robotics System

The whole process of the proposed VR synthesis is shown in Fig. 5.14, and contains a series of communications among the different modules of the suite. The first challenge is to make the connection between the numerical solver ArtTreeKS and the application running in Windows OS. ArtTreeKS is an open source software, compiled and used on a Linux platform, while the virtual reality hardware and software have their best performance on a Windows platform. In Windows 10, a subsystem is designed for Linux, WSL, which makes the bridge to allow running the Linux-based program in Windows.

ArtTreeks uses C-based libraries to solve the nonlinear equations, such as CMinpack, and Lua is the script language used to read the input and run the program. So to connect the Unity to ArtTreeKS, in the Windows side a TCP connection is created in Unity (with TcpClient class) and the captured poses are sent through this connection. On the WSL side, a Lua program creates a socket through which it can read the messages sent by Unity and run ArtTreeKS. The result is going to be a file containing the

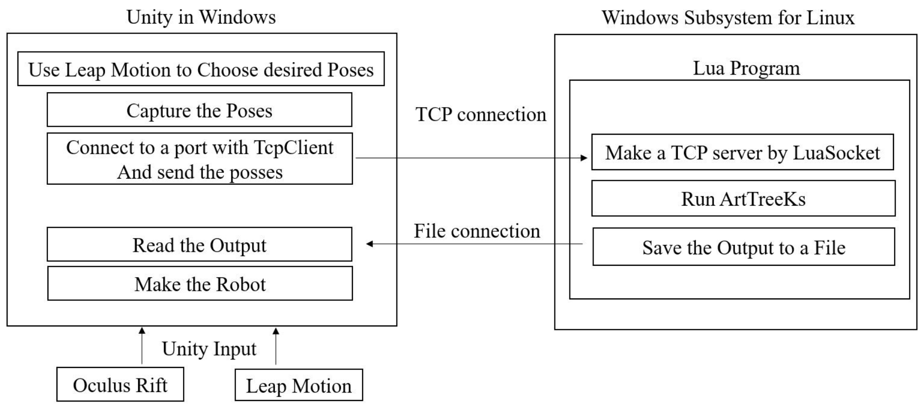


Figure 5.14: The flowchart of the whole proposed process

Plucker coordinates and joint angles of the synthesized robot. WSL is running in Windows so the files can be shared between Linux and Windows programs. Next step takes place back in Unity, where we read the file in C# script triggered by the "Show the Robot" button.

This work presents the first automatic and interactive system for the design of any robotic arm and robotic hand. The immersive virtual reality environment allows for an easy and intuitive definition of the task by the designer, including fingertip tasks for grasping and manipulation of objects. The communication among the different subsystems allow the task to be sent to a fast solver and the solution to be returned to the virtual reality system for display and animation, which allows assessing the result quickly and efficiently. The overall VR-based design suite for robotic systems will help designers to create a great variety of designs in a fast and easy way.

5.3.0.3 VR Synthesis Example

As an example, two robots have been designed to follow the motions of the human hand. The first one is a 3R serial robot, that is an arm manipulator with three revolute joints, and the second one is a robotic hand which could be used as an exoskeleton or to perform human hand tasks.

The 3R serial robot consists of 3 revolute joints assembled in series and with general location and orientation. A 3R serial chain is shown in Fig. 5.16. The task is generated by the motion of the user's hand. The user moves the guide box to whichever position and orientation is desired and store that pose by pushing the Capture the Pose button. Blue boxes in Fig. 5.15 show the desired task for this robot.

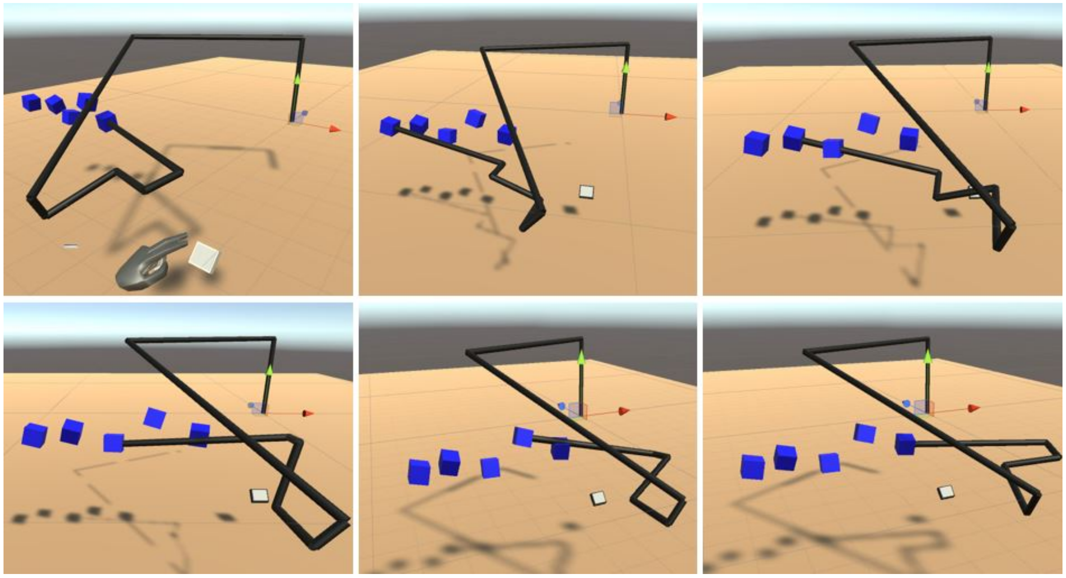


Figure 5.15: The synthesized robot

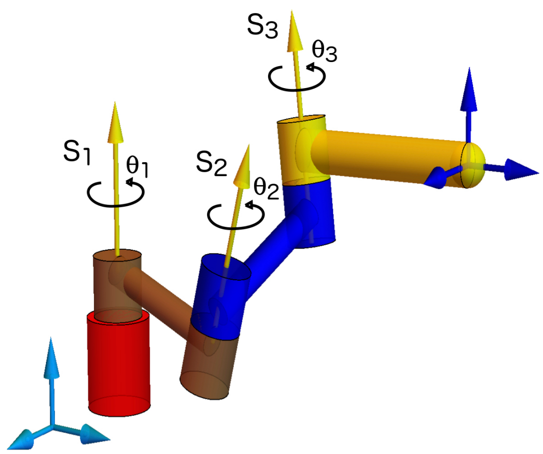


Figure 5.16: A serial 3R robot.

After sending the task to ArtTreeKS and receiving the solution, the robot can be shown in the VR environment. The first picture from the top left shows the virtual hand pushing a button "Pose" to change the current pose. And the rest of the pictures show the robot in other task positions.

For a hand example we use the $p = \{0, 1, 1, 1\}$ with $j = \{1, 3, 3, 3\}$, that is a hand with one wrist joint and three fingers, each of them consisting of three revolute joints and a fingertip. Fig. 5.17 shows the hand topology and sketch.

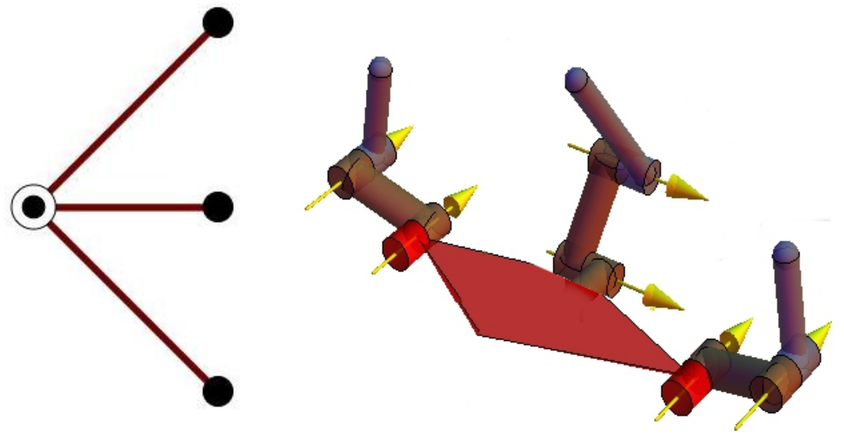


Figure 5.17: A three-fingered hand with one wrist R joint and three R joints per finger.

This hand can reach exactly $m = 6$ positions for simultaneous task of each of the three fingertips. These positions are captured from the human hand immersed in the virtual reality environment. Fig 5.18 shows the result where the three designed fingers can reach to the desired poses. The first picture from left on top shows the virtual hand as well. The user changes the configuration of the robot by pushing the button. Each time the button is pushed, the relative value of each joint with respect to the reference configuration is set and the end-effector of the robot will reach to the next pose. Also A linear interpolation between joints values correspond to desired configurations is performed so that transition between poses look smooth and natural.

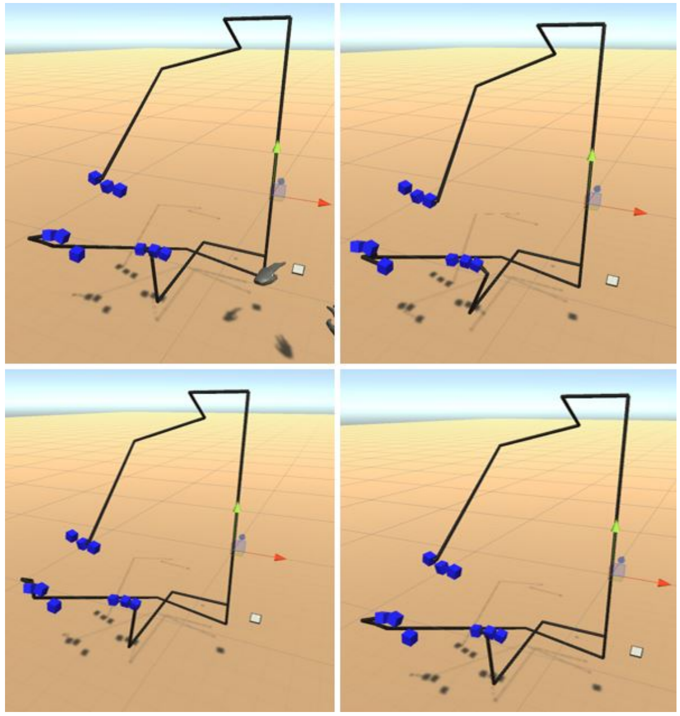


Figure 5.18: The synthesized Hand

Chapter 6: Augmented Reality and Its Application in Robotics

The definition of Augmented Reality (AR) can be very broad but what we consider as an AR experience in this dissertation is any digital information that is added to the physical world in order to augment the perception and interaction with the world. There are three factors that define AR ([Azu97] and [Cra13]):

- Combines real and virtual
- Interactive in real time
- Registered in 3D

Adding virtual objects in the user's surrounding will augment the reality in a natural way when the virtual objects are in consistent with the real objects. The location, orientation, lighting and material of the virtual object are some of the properties that can be considered to improve the AR experience. The second important characteristics is the interactivity. The rendered virtual objects should react to the changes in the environment or to the movement and actions of the user so that it provides the right impression. The augmented objects need to move in opposite transformation of user displacements so that it is believable for the user that they are stationery. In the case of moving augmented objects, the movements of the user should be taken into account as well. Moreover, changes in lighting and sound have big impacts on the user's experience.

6.1 Augmented Reality Concepts

Localization and mapping through IMU's, depth-sensing cameras and other sensors is an essential part of a good augmented reality experience. The first step is to understand the state of the real world in real-time. The second step is to localize the user's position (AR device position) with respect to this environment and the third step is to display the virtual objects in the right place considering the changes in the environment and the user.

Three major components that make the AR experience possible include [Cra13]):

- Sensor

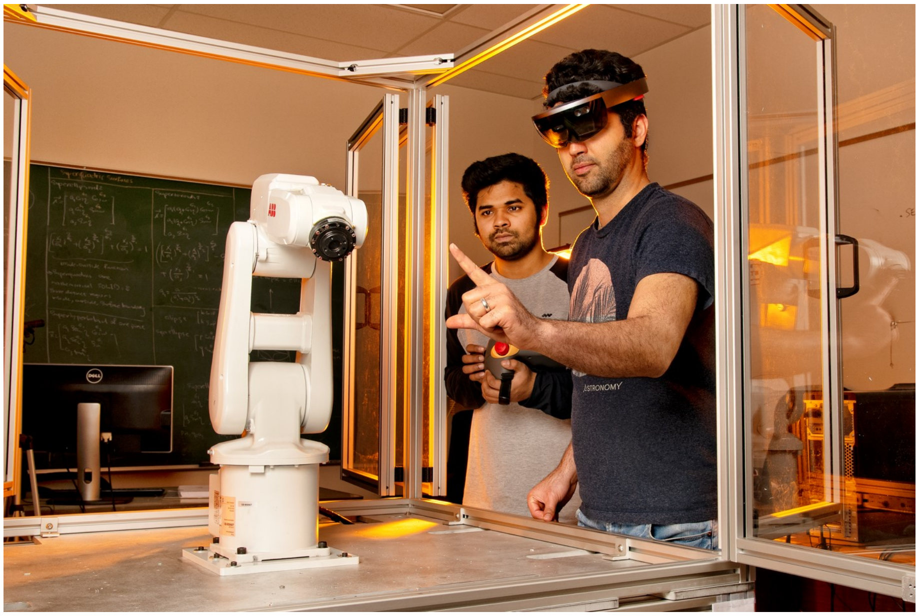


Figure 6.1: AR application in HoloLens to communicate with ABB robots

- Processor
- Display

The most popular sensor is camera where 2D and 3D images can be detected and processed for understating the environment; these camera are used to identify hand gestures too as the user input information. To localize in the environment, Global Positioning System (GPS), for example, could be used to find the location of the user. Gyroscopes and accelerometers are used to determine the amount of movement and rotation in different directions. Moreover, voice is another source of user input that is easily recorded by microphones and filtered to get rid of the noises.

Processor analyzes the sensors input, computes the coordinates and generated the proper output signal for the display. Processor can benefit from different powerful software applications which can help understanding the environment more precisely. For instance, Cortana, which is a virtual assistant created by Microsoft for Windows 10, is a popular platform to recognize the voice of the user. Hand gestures can be identified and interpreted by the powerful software recently created by Leap Motion and Microsoft. The processor processes all of these sensors information by the software provided in each platform to map the surroundings and localize the user in that map. The better the processor is in

Figure 6.2: Microsoft HoloLens

terms of hardware and software, the more precise AR experience is expected to be created. That is why the price of a legitimate AR device is still fairly high.

Display, in the general definition, is a component to manipulate the user's senses including vision, smell, haptics, hearing and taste. The most obvious display is the monitor of a computer or a screen of a mobile cell phone. The duty of this component is to receive the manipulated the signal from the processor and display it to each specific sense so that the user can not distinguish between the reality and *virtuality*.

6.2 Augmented Reality Tools

Augmented reality tools that have been utilized in this work including the two aspects of hardware and software are described in the following subsections.

6.2.1 Hardware

Microsoft HoloLens is the first fully self-contained augmented reality device that can create holographic objects and run in Windows 10.

It is totally wireless and constantly mapping the area to localize the user in the real world. The sensors used in this device are:

- 1 IMU (Accelerometer, gyroscope, and magnetometer)
- 4 environment sensors
- 1 energy-efficient depth camera with a 120 by 120 degree angle of view
- Four-microphone array
- 1 ambient light sensor

Moreover, this device has a powerful technology of human understanding capabilities. It includes:

- **Spatial Sound** which stimulates the hearing sense of the user by creating spatial effects; this allows the user to locate the virtual objects or have a better perception of the interactive objects.
- **Gaze Tracking** which brings the application focus to whatever the user is perceiving to navigate and explore, the technology can tell exactly what and where to show the images for each pupil to generate stereoscopic 3D illusions.
- **Gesture Input** which recognizes a few hand gestures including *bloom* gesture to pull up a UI navigation menu screen and *air tap* gesture to select menu commands similar to clicking by compute mouse.
- **Voice Support** which allows the user to use voice commands.

6.2.2 Software

Window Mixed Reality (WMR) and Windows 10 are the software that HoloLens needs to process and run on respectively. WMR is a the core software package that enables developers to use the HoloLens features for different applications and purposes.

On the other hand the SDK prepared for Unity called HoloToolkit provides a complete set of tools in Unity to work with HoloLens. This package consists of Unity scripts, prefabs and other combined

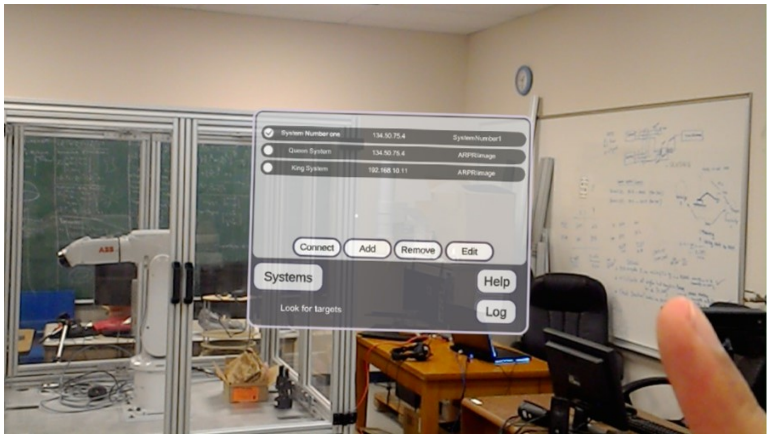


Figure 6.3: Main menu to configure the database

software like Vuforia to ease the process of making an augmented reality application. Using these tools, in the next section, an application of AR in robotics is introduced and investigated.

6.3 Augmented Reality Platform for Robot-Human Interaction

Augmented reality may have many applications in different industries but in the current work, we focus on its uses in robotics industry. Augmented reality is used to connect and communicate wit ABB robots. ABB controller IRC5 with RobotWare 6 and higher provides a web services which is a set of RESTful APIs that uses HTTP protocol and the messages that are made up of XML or JSON format. On the other hand, we can make applications in HoloLens that can be connected to these types of web services. This lets one connect to ABB controllers and have a two-way communication to get information from robots or send information to them.

6.3.1 Main Menu Front-End

The front-end of the application is composed of a set of user-interface (UI) elements that let the users find a system by different methods including voice recognition, 2D-image recognition or directly entering the IP address of the robot controller. The user is able to add, remove, edit or connect to systems that are stored in a database holding the information of robots that are available to connect to, 6.3. Systems information include a unique name for each system that can be detected by using *Cortona*, the

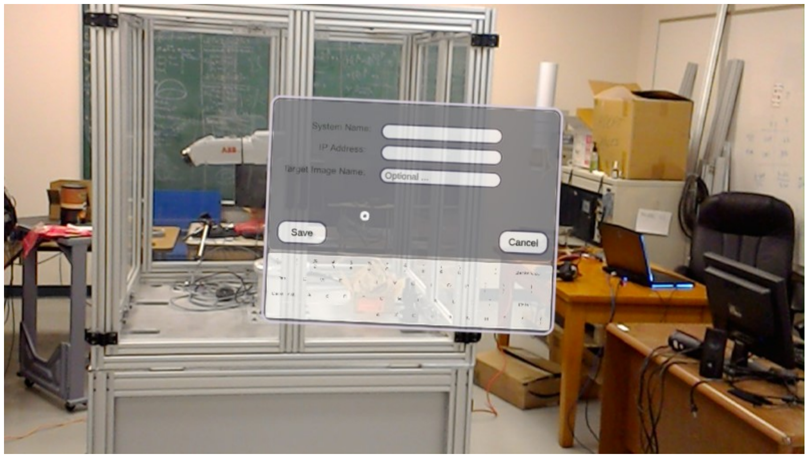


Figure 6.4: Using virtual keyboard

name of an image that is ready to detect by cameras using *Vuforia* platform and the IP address of the controller in the network which can be entered by the designed virtual keyboard 6.4.

6.3.2 Communication

The communication is through a set of web services that the controller provides. XHTML and JSON are the format for the messages sent or received. For this purpose, HTTP has a number of predefined methods that can be used:

- GET: Retrieve a resource
- PUT: Create or update a resource
- POST: Update a resource
- DELETE: Delete a resourc

GET does not change the resources while the rest will make a change in the state of the resources. Any attempt to connect to the controller is followed by an authentication step where a username and password must be provided to ensure that the user has permit to connect to the robot.

The Robot Web Services has many services available from which one can get additional services or one or more resources. A tree chart of available services and resources are depicted in 6.6.

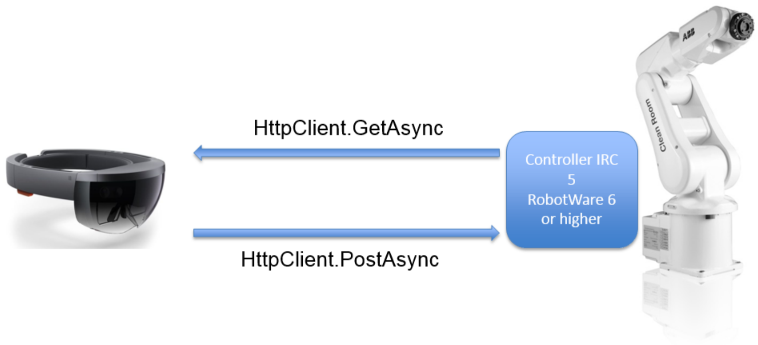


Figure 6.5: Robot Communication

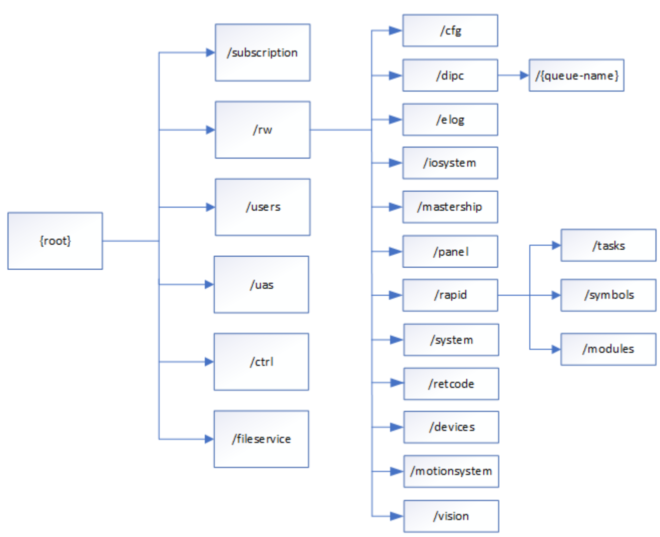


Figure 6.6: Services

Fileservice provides remote access to files and directories by which we can create, remove, rename and create files and directories. Subscription service is used for when we need some information to get updated when thy change. Ctrl service handles robot controller global functionality, such as access to the controller clock, controller identification, performs restart etc. Users service takes care of registration of connected clients. And finally, RW is for RobotWare services, such as IO, RAPID, E-log, CFG etc.

All of these services are available in RobotStudio or the flex pendant of the controller but the mission of this work is to replace those platforms with a more human-friendly platform where human hand gestures and voice could be a part of robot-communication. Furthermore, visualizing the simulated motion superimposed on top of the real hardware can create a productive and constructive environment for the users.

6.3.3 Asynchronous Operations

An important part of the written code quality is the use of asynchronous operations. This type of operation aims at improving the performance by running tasks in parallel. For example, file manipulation such as opening, reading and closing should be getting done in a parallel thread so that it does not create any delay or lag on the main thread. This improved the quality of the user experience drastically. In present work, this concept is implemented in the Unity main *update()* function. This function runs the main thread of the application and any disturbance or delay could cause a bad experience for the user.

Async/await is a concept in C# by which we can implement asynchronous operations. Basically, they are the code markers, which marks code positions from where the control should resume after a task completes. In Unity we make the main thread a *async* method in which an awaitable (an asynchronous operation) can be done by using *await* keyword.

6.3.4 Control Panel

However a system is detected, the next step is to have another scene (called control panel) consisting of UI elements that helps users communicate with the robots.

The functionalities available in the current version of the app includes following, 6.8:

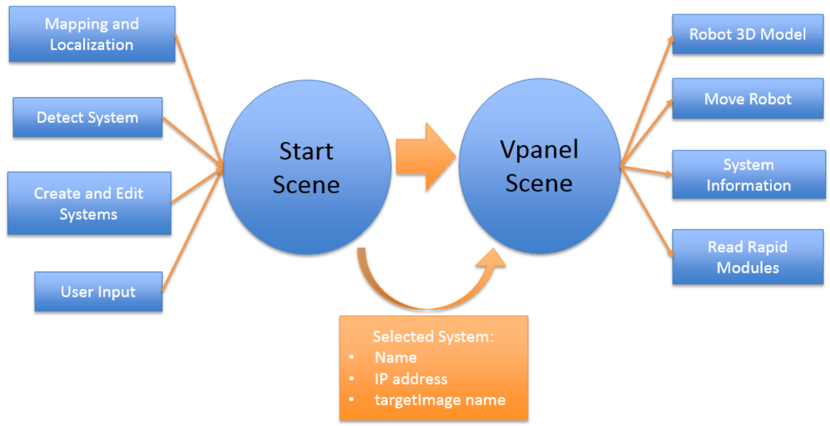


Figure 6.7: Two main scenes of the Application

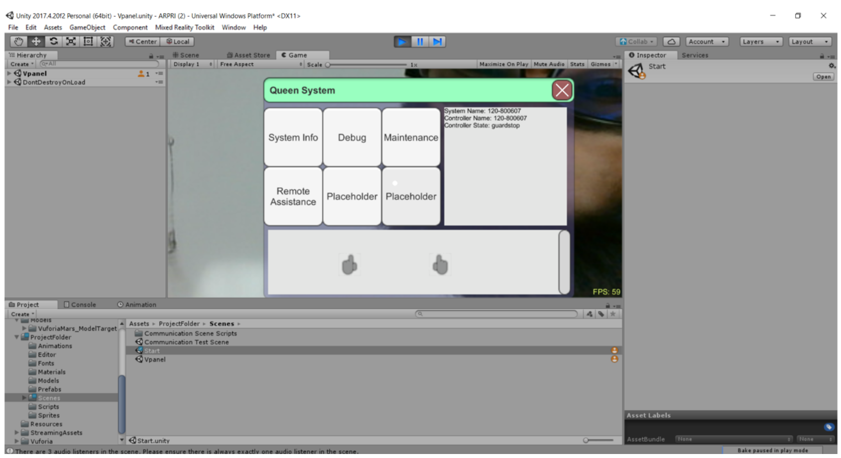


Figure 6.8: Control Panel

- System Information, this extracts some general information from the controller to display on the right side of the menu.

- RunModule, this feature runs the current module available in the controller. This could be triggered by voice or clicking on the button.

- 3D Model, shows the model of the robot which can be relocated and superimposed on top of the real one. This has different advantages such as comparing the simulated the motion planning with the real one and fast debugging process where the result of any change in the RAPID code could be seen in the real world.

Chapter 7: Conclusion

In this work, human upper limb is studied in two different mediums i.e. reality and virtuality. Primarily, two kinematic models for human upper limb are suggested. The first model considers the shoulder as 3DOF while the purpose of the second model is seeking a better understanding of the shoulder complex in lower dimensions. An effective method to calculate the GH locations is introduced. Based on the proposed idea, in an attempt to find the best mechanism to mimic the shoulder complex motions, the calculated GH locations is used as the input for the point synthesis. A 2-4R parallel manipulator whose workspace is similar to the human shoulder complex motion is suggested. This strategy is extended for exoskeletons which are designed to accomplish specific tasks in the human limb workspace. On the other side, the uses of Virtual Reality in rehabilitation and robotics are investigated. Several immersive virtual environments have been designed to study the effect of mirror therapy in healthy and stroke patients. Moreover, The kinematics synthesis process has been leveled up by utilizing VR capabilities in terms of easiness and interaction. At last, an augmented reality application in HoloLens device is made for robot-human interaction for industrial purposes.

7.1 Contribution

The highlights and potential contributions of the present work are described below:

- Kinematic model I is designed for human upper limb, including a 3 DOF shoulder mechanism, to investigate the synergy of the human motion. The results show that the shoulder model uses its most of the degrees of freedom for a faithful reconstruction.
- A new method to locate GH joint more accurately is introduced ans explained.
- Kinematics model II has been designed to focus on shoulder complex workspace. The result of this research suggests a parallel mechanism with constrained joint space which improve the accuracy and efficiency.
- Virtual Reality environments are built for assessing the mirror therapy for healthy and stroke patients. Leap motion technology has been used to create a natural and conniving model of human limb.

- The first VR application for robot dimensional synthesis has been developed. All the process of the synthesis can be performed in this application which will result in a successful prototype of the final product.

- Human-robot interaction may be more natural and intuitive. Connecting to the robots through AR goggle like HoloLens expand human perception of the robot states and tasks. Similarly, with the same platform, the robots can perceive human commands and presence more accurately. As a result, the interaction between human and robots is expected to be enhanced drastically.

7.2 Outlook

The algorithms and methods to analyze human shoulder introduced in this dissertation could be used for any other part or limb of human body. The focus is more on the task the limb does instead of its anatomy.

The synthesized 2-4R parallel mechanism is an estimation of the human shoulder workspace in the lower dimensions. The approach used in this point synthesis can be expanded for any other joint whose location is determined by other limb joints. Moreover, the suggested mechanism introduces a complicated 2-DOF space which can be more investigated and explored for other similar task with small worksapce and high payload.

Having the created platforms for the uses and applications of VR in Robotics, one can expand the features and add more capabilities for robot designers in the virtual world. Also the introduced augmented reality platform for human interaction with ABB robots can expanded to any other industrial and collaborative robots.

References

[2019] Dedication. In William R. Sherman and Alan B. Craig, editors, *Understanding Virtual Reality (Second Edition)*, The Morgan Kaufmann Series in Computer Graphics, page v. Morgan Kaufmann, Boston, second edition edition, 2019.

[AC07] J. Aleotti and S. Caselli. Robot grasp synthesis from virtual demonstration and topology-preserving environment reconstruction. In *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2692–2697, Oct 2007.

[Ahs08] H. Ahsan. *3D Computer Vision System for Hand Joint Motion Calculation*. PhD thesis, College of Engineering, Idaho State University, Pocatello, ID, USA, 2008.

[AM01] S.G. Ahlers and J.M. McCarthy. Ch. 12: Clifford algebra and the optimization in robot design. In *Clifford Algebra with Applications in Science and Engineering*, pages 235–251. Birkhauser, 2001.

[AMMB16] Rasmus S. Andersen, Ole Madsen, Thomas B. Moeslund, and Hani Ben Amor. Projecting robot intentions into human environments. In *25th IEEE International Symposium on Robot and Human Interactive Communication, RO-MAN 2016*, pages 294–301, United States, 11 2016. Institute of Electrical and Electronics Engineers Inc.

[Ang82] J. Angeles. *Spatial kinematic chains: analysis, synthesis, optimization*. Springer-Verlag, New York, first edition, 1982.

[Azu97] Ronald T. Azuma. A survey of augmented reality. *Presence: Teleoper. Virtual Environ.*, 6(4):355–385, August 1997.

[Bak80] J. Eddie Baker. An analysis of the bricard linkages. *Mechanism and Machine Theory*, 15:267–286, 1980.

[BBPG14] Wolper James S Batbold Batchimeg, Yihun Yimesker and Alba Pérez-Gracia. Exact workspace synthesis for rccr linkages. In *Computational Kinematics*, pages 349–357. Springer, 2014.

[BCM07] M. Baritz, D Cotoros, and O. Moraru. Virtual and augmented reality for mechanism motion modeling in technical applications. In *Proceedings of the 7th WSEAS Int. Conf. on Signal Processing, Computational Geometry and Artificial Vision*, Athens, Greece, August 24-26, 2007 2007.

[BD15] J. Borras and A.M. Dollar. Dimensional synthesis of three-fingered robot hands for maximal precision manipulation workspace. *The Int. Journal on Robotics Research*, 34(14), 2015.

[BGS11] A. Bicchi, M. Gabiccini, and M. Santello. Modelling natural and artificial hands with synergies. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 366:3153–3161, 2011.

[BH07] Stephen P Buerger and Neville Hogan. Complementary stability and loop shaping for improved human–robot interaction. *Robotics, IEEE Transactions on*, 23(2):232–244, 2007.

[BJ04] Herr H Blaya JA. Control of a variable-impedance ankle-foot orthosis to assist drop-foot gait. *IEEE Trans Neural System Rehabilitation Engineering*, 12:24–31, 2004.

[BK90] N. Benjuya and S. B. Kenney. Hybrid arm orthosis. *Prosthetics Orthotics*, 2:155–163, 1990.

[BKB10] S. Balasubramanian, J. Klein, and E. Burdet. Robot-assisted rehabilitation of hand function. *Curr. Opin. Neurol.*, 23(6):661–670, 2010.

[BMKG07] Raymond C Browning, JESSE R Modica, Rodger Kram, and Ambarish Goswami. The effects of adding mass to the legs on the energetics and biomechanics of walking. *Medicine and science in sports and exercise*, 39(3):515, 2007.

[Bog15] Robert Bogue. Robotic exoskeletons: a review of recent progress. *Industrial Robot: An International Journal*, 42(1):5–10, 2015.

[BRBFM00] E.V. Biryukova, A. Roby-Brami, A.A. Frolov, and M. Mokhtari. Kinematics of human arm reconstructed from spatial tracking system recordings. *Journal of Biomechanics*, 33(8):985 – 995, 2000.

[Bru66] S. Brunnstrom. Motor testing procedures in hemiplegia, based on recovery stages. *J Am Phys Ther Assoc*, 46:357, 1966.

[BRV16] K. Baraka, S. Rosenthal, and M. Veloso. Enhancing human understanding of a mobile robot's state and actions using expressive lights. In *2016 25th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pages 652–657, Aug 2016.

[BRZ+16] A. Batinica, M. Raković, M. Zarić, B. Borovac, and M. Nikolić. Motion planning of a robot in real-time based on the general model of humanoid robots. In *2016 IEEE 14th International Symposium on Intelligent Systems and Informatics (SISY)*, pages 31–38, Aug 2016.

[BV17] L. P. Berg and J. M. Vance. An industry case study: Investigating early design decision-making in virtual reality. *ASME Journal of Computing and Information Science in Engineering*, 17(1):011001–011001–7, 2017.

[BVZ+15] Magdo Bortole, Anusha Venkatakrishnan, Fangshi Zhu, Juan C Moreno, Gerard E Francisco, Jose L Pons, and Jose L Contreras-Vidal. The h2 robotic exoskeleton for gait rehabilitation after stroke: early findings from a clinical study. *Journal of neuroengineering and rehabilitation*, 12(1):1, 2015.

[CAKL15] Ravi Teja Chadalavada, Henrik Andreasson, Robert Krug, and Achim Lilienthal. That's on my mind! robot to human intention communication through on-board projection on shared floor space. 09 2015.

[CBHG16] Viet Anh Dung Cai, Philippe Bidaud, Vincent Hayward, and Florian Gosselin. Self-adjustment mechanisms and their application for orthosis design. *Meccanica*, pages 1–16, 2016.

[CCC14] M. Controzzi, C. Cipriani, and M.C. Carrozza. The Human Hand as an Inspiration for Robot Hand. In R. Balasubramian and V.J. Santos, editors, *Design of Artificial Hands: A Review*, volume 95 of *Springer Tracts in Advanced Robotics*, pages 219–244. Springer Switzerland, 2014.

[CCF11] Antonino Casile, Vittorio Caggiano, and Pier Francesco Ferrari. The mirror neuron system: a fresh view. *The Neuroscientist : a review journal bringing neurobiology, neurology and psychiatry*, 17 5:524–38, 2011.

[CCK+14] C. Charbonnier, S. Chagué, F.C. Kolo, J.C.K. Chow, and A. Lädermann. A patient-specific measurement technique to model shoulder joint kinematics. *Orthopaedics & Traumatology: Surgery & Research*, 100(7):715 – 719, 2014.

[CE14] Charalambos P. Charalambous and Sarah Eastwood. *Normal and Abnormal Motion of the Shoulder*, pages 331–333. Springer London, London, 2014.

[CGWH10] M. Chalon, M. Grebenstein, T. Wimbock, and G. Hirzinger. The thumb: Guidelines for a robotic design. In *Proc. of the 2010 Int. Conf. on Intelligent Robots and Systems*, Taipei, Taiwan, October 18-22, 2010.

[CHB07] J. Y. C. Chen, E. C. Haas, and M. J. Barnes. Human performance issues and user interface design for teleoperated robots. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 37(6):1231–1245, Nov 2007.

[CHJ+14] Matei Ciocarlie, Kaijen Hsiao, Edward Gil Jones, Sachin Chitta, Radu Bogdan Rusu, and Ioan A Şucan. Towards reliable grasping and manipulation in household environments. In *Experimental Robotics*, pages 241–252. Springer, 2014.

[CL05] N. Chernov and C. Lesort. Least squares fitting of circles. *Journal of Mathematical Imaging and Vision*, 23(3):239–252, Nov 2005.

[Clo65] W. Cloud. Man amplifiers: Machines that let you carry a ton. *Popular Science*, 187:70–73, 1965.

[CLZ+14] Yanyan Chen, Ge Li, Yanhe Zhu, Jie Zhao, and Hegao Cai. Design of a 6-dof upper limb rehabilitation exoskeleton with parallel actuated joints. *Bio-medical materials and engineering*, 24(6):2527–2535, 2014.

[CMGG+04] B. Calvo-Merino, D.E. Glaser, J. Grèzes, P. Haggard, and R.E. Passingham. Action Observation and Acquired Motor Skills: An fMRI Study with Expert Dancers. *Cerebral Cortex*, 15(8):1243–1249, 12 2004.

[CPG+15] P. Carlson, A. Peters, S. P. Gilbert, J. M. Vance, and A. Luse. Virtual training: Learning transfer of assembly tasks. *IEEE Transactions on Visualization and Computer Graphics*, 21(6):770–782, 2015.

[Cra89] J. J. Craig. *Introduction to Robotics: Mechanics and Control*. Prentice Hall, second edition, 1989.

[Cra13] Alan B. Craig. Chapter 2 - augmented reality concepts. In Alan B. Craig, editor, *Understanding Augmented Reality*, pages 39 – 67. Morgan Kaufmann, Boston, 2013.

[CYG12] H.-J. Su C. Yue and Q. Ge. A hybrid computer-aided linkage design system for tracing open and closed planar curves. *Computer-Aided Design*, 44:1141–1150, 2012.

[CZ15] M. Ceccarelli and M. Zottola. Design and simulation of an underactuated finger mecanism for larm hand. *Robotica*, 2015.

[DCMP03] Mike Daily, Youngkwan Cho, Kevin Martin, and Dave Payton. World embedded interfaces for human-robot interaction. In *Proceedings of the 36th Annual Hawaii International Conference on System Sciences (HICSS'03) - Track 5 - Volume 5*, HICSS '03, pages 125.2–, Washington, DC, USA, 2003. IEEE Computer Society.

[DLS13] Anca D. Dragan, Kenton C.T. Lee, and Siddhartha S. Srinivasa. Legibility and predictability of robot motion. In *Proceedings of the 8th ACM/IEEE International Conference on Human-robot Interaction*, HRI '13, pages 301–308, Piscataway, NJ, USA, 2013. IEEE Press.

[EGL16] Valerio Elia, Maria Grazia Gnoni, and Alessandra Lanzilotto. Evaluating the application of augmented reality devices in manufacturing from a process point of view: An ahp based model. *Expert Systems with Applications*, 63:187 – 197, 2016.

[ESP+07] Kynan Eng, Ewa Siekierka, Pawel Pyk, Edith Chevrier, Yves Hauser, Monica Cameirao, Lisa Holper, Karin Hägni, Lukas Zimmerli, Armin Duff, Corina Schuster, Claudio Bassetti, Paul Verschure, and Daniel Kiper. Interactive visuo-motor therapy system for stroke rehabilitation. *Medical & Biological Engineering & Computing*, 45(9):901–907, Sep 2007.

[ESS+07] Denis Ertelt, Steven Small, Ana Solodkin, Christian Dettmers, Adam Mcnamara, Ferdinand Binkofski, and Giovanni Buccino. Action observation has a positive impact on rehabilitation of motor deficits after stroke. *NeuroImage*, 36 Suppl 2:T164–73, 02 2007.

[ETDH06a] Rainald Ehrig, William Taylor, Georg Duda, and Markus Heller. A survey of formal methods for determining the centre of rotation of ball joints. *Journal of biomechanics*, 39:2798–809, 02 2006.

[ETDH06b] Rainald M. Ehrig, William R. Taylor, Georg N. Duda, and Markus O. Heller. A survey of formal methods for determining the centre of rotation of ball joints. *Journal of Biomechanics*, 39(15):2798 – 2809, 2006.

[FA06] E. M. Frick and J. L. Alberts. Combined use of repetitive task practice and an assistive robotic device in a patient with subacute stroke. *Physical Therapy*, 86(10):1378–1386, 2006.

[FFG+05] Leonardo Fogassi, Pier Francesco Ferrari, Benno Gesierich, Stefano Rozzi, Fabian Chersi, and Giacomo Rizzolatti. Parietal lobe: From action organization to intention understanding. *Science*, 308(5722):662–667, 2005.

[GBCC08] Scott A. Green, Mark Billinghurst, XiaoQi Chen, and J. Geoffrey Chase. Human-robot collaboration: A literature review and augmented reality approach in design. *International Journal of Advanced Robotic Systems*, 5(1):1, 2008.

[GGS94] Walter Gander, Gene H. Golub, and Rolf Strebel. Least-squares fitting of circles and ellipses. *BIT Numerical Mathematics*, 34(4):558–578, Dec 1994.

[GO06] A. Gupta and M. K. O'Malley. Design of a haptic arm exoskeleton for training and rehabilitation. *IEEE/ASME Trans. on Mechatronics*, 11:280–289, 2006.

[GSMP13] G. Gioioso, G. Salvietti, M. Malvezzi, and D. Prattichizzo. An object-based approach to map human hand synergies onto robotic hands with dissimilar kinematics. *Robotics*, page 97, 2013.

[HD64] Richard S Hartenberg and Jacques Denavit. *Kinematic synthesis of linkages*. McGraw-Hill New York, 1964.

[HDV14] O. Heidari, H. M. Daniali, and S. M. Varedi. Geometric design of 3r manipulators for three precision poses using dual quaternions. In *2014 Second RSI/ISM International Conference on Robotics and Mechatronics (ICRoM)*, pages 601–606, Oct 2014.

[HMS+15] W. Hönig, C. Milanes, L. Scaria, T. Phan, M. Bolas, and N. Ayanian. Mixed reality for robotics. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5382–5387, Sep. 2015.

[HPG16] N. Hassanzadeh and A. Perez-Gracia. Dimensional synthesis of wristed binary hands. *ASME Journal of Mechanisms and Robotics*, 8(2), 2016.

[HRPGK16] Omid Heidari, John O Roylance, Alba Perez-Gracia, and Eydie Kendall. Quantification of upper-body synergies: A case comparison for stroke and non-stroke victims. In *ASME 2016 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, pages V05AT07A032–V05AT07A032. American Society of Mechanical Engineers, 2016.

[HSC+90] D. T. Harryman, J. A. Sidles, J. M. Clark, K. J. McQuade, T. D. Gibb, and F. A. Matsen. Translation of the humeral head on the glenoid with passive glenohumeral motion. *Journal of Bone and Joint Surgery - Series A*, 72(9):1334–1343, 1990.

[HWS18] Hooman Hedayati, Michael Walker, and Daniel Szafir. Improving collocated robot teleoperation with augmented reality. In *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*, HRI '18, pages 78–86, New York, NY, USA, 2018. ACM.

[IEF+13] D. Ingram, C. Engelhardt, A. Farron, A. Terrier, and P. Mullhaupt. A minimal set of coordinates for describing humanoid shoulder motion. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 3-7 Nov., 2013.

[IKTC14] Jamshed Iqbal, Hamza Khan, Nikos G. Tsagarakis, and Darwin G. Caldwell. A novel exoskeleton robotic system for hand rehabilitation - conceptualization to prototyping. *Biocybernetics and Biomedical Engineering*, 34(2):79 – 89, 2014.

[IRKS17] Vladimir Iglovikov, Alexander Rakhlin, Alexandr Kalinin, and Alexey Shvets. Pediatric bone age assessment using deep convolutional neural networks. 12 2017.

[JB13] Y. Jung and J. Bae. Kinematic analysis of a 5 dof upper-limb exoskeleton with a tilted and vertically translating shoulder joint. In *2013 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, 9-12 July, 2013.

[JH06] OF Jensen and JM Hansen. Dimensional synthesis of spatial mechanisms and the problem of non-assembly. *Multibody System Dynamics*, 15(2):107–133, 2006.

[Kas76] I. Kasa. A circle fitting procedure and its error analysis. *IEEE Transactions on Instrumentation and Measurement*, IM-25:8–14, 1976.

[KFBW05] Gary Klein, Paul J. Feltovich, Jeffrey Bradshaw, and David Woods. *Common Ground and Coordination in Joint Activity*, pages 139 – 184. 06 2005.

[KH02] Sankai Y Kawamoto H. Power assist system hal-3 for gait disorder person. In *Proceedings of the International Conference on Computers Helping People with Special Needs (ICCHP), Berlin, Germany*, volume 2398, 2002.

[KHS14] Manon Kok, Jeroen D. Hol, and Thomas B. Schön. An optimization-based approach to human body motion capture using inertial sensors. *IFAC Proceedings Volumes*, 47(3):79 – 85, 2014. 19th IFAC World Congress.

[KKF03] M. Yasuda K. Watanabe K. Kiguchi, K. Iwami and T. Fukuda. An exoskeletal robot for human shoulder joint motion assist. *IEEE/ASME Trans. on Mechatronics*, 8:125–135, 2003.

[KM91] H. Kazerooni and S. L. Mahoney. Dynamics and control of robotic systems worn by humans. *ASME Journal of Dynamics Systems, Measurement and Control*, 113:379–387, 1991.

[KMB$^+$12] H. Kim, L. M. Miller, N. Byl, G. M. Abrams, and J. Rosen. Redundancy resolution of the human arm and an upper limb exoskeleton. *IEEE Transactions on Biomedical Engineering*, 59(6):1770–1779, June 2012.

[KTL07] N. Klopčar, M. Tomšič, and J. Lenarčič. A kinematic model of the shoulder complex to evaluate the arm-reachable workspace. *Journal of Biomechanics*, 40(1):86 – 91, 2007.

[KVL02] J. N. Kihonge, J. M. Vance, and P. M Larochelle. Spatial mechanism design in virtual reality with networking. *ASME Journal of Mechanical Design*, 124(3):435–440, 2002.

[KYH$^+$14] Abdul Manan Khan, Deok-won Yun, Jung-Soo Han, Kyoosik Shin, and Chang-Soo Han. Upper extremity assist exoskeleton robot. In *Robot and Human Interactive Communication, 2014 RO-MAN: The 23rd IEEE International Symposium on*, pages 892–898. IEEE, 2014.

[lav] Dedication. In Steven M. LaValle, editor, *Virtual Reality*.

[LM04] E. Lee and C. Mavroidis. Geometric design of 3r manipulators for reaching four end-effector spatial poses. *The International Journal of Robotics Research*, 23(3):247–254, 2004.

[LPM] Nicholas A Morton Lorin P Maletsky, Junyi Sun. Accuracy of an optical active-marker system to track the relative motion of rigid bodies. *Journal of biomechanics*, 40.

[LS03] J. Lenarcic and M. Stanisic. A humanoid shoulder complex and the humeral pointing kinematics. *IEEE Transactions on Robotics and Automation*, 19(3):499–506, June 2003.

[LU94] John Lenarcic and Andreja Umek. Simple model of human arm reachable workspace. *Systems, Man and Cybernetics, IEEE Transactions on*, 24:1239 – 1246, 09 1994.

[LVK02] P. M. Larochelle, J. M. Vance, and J. Kihonge. Interactive visualization of the line congruences for spatial mechanism design. *ASME Journal of Computing and Information Science in Engineering*, 2(3):208–215, 2002.

[LZGP13] Xiangyun Li, Ping Zhao, QJ Ge, and Anurag Purwar. A task driven approach to simultaneous type synthesis and dimensional optimization of planar parallel manipulator using algebraic fitting of a family of quadrics. In *ASME 2013 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, pages V06BT07A057–V06BT07A057. American Society of Mechanical Engineers, 2013.

[mata] MathWorks. mocap toolbox. `https://www.mathworks.com/matlabcentral/fileexchange/62513-mocap_toolbox`. Accessed: 2019-03-25.

[matb] YouTube.com. mocap toolbox tutorial. `https://youtu.be/W6C94F21_R0`. Accessed: 2019-03-25.

[MBP+12] Daniel F. Massimini, Patrick J. Boyer, Ramprasad Papannagari, Thomas J. Gill, Jon P. Warner, and Guoan Li. In-vivo glenohumeral translation and ligament elongation during abduction and abduction with internal and external rotation. *Journal of Orthopaedic Surgery and Research*, 7(1):29, Jun 2012.

[MD95] J.M.R. Martinez and J. Duffy. On the metrics of rigid body displacements for infinite and finite bodies. *ASME Journal of Mechanical Design*, 117:41–47, 1995.

[MDB+07] Tony Monnet, Eric Desailly, Mickael Begon, Claude Vallée, and Patrick Lacouture. Comparison of the score & ha methods for locating in vivo the glenohumeral joint centre. *Journal of biomechanics*, 40:3487–92, 02 2007.

[MGE01] C. R. Mason, J. E. Gomez, and T. J. Ebner. Hand synergies during reach-to-grasp. *Journal of Neurophysiology, Am Physiological Soc.*, 86 (6):2896–2910, 2001.

[MKKM16] Sotiris Makris, Panagiotis Karagiannis, Spyridon Koukas, and Aleksandros-Stereos Matthaiakis. Augmented reality system for operator support in human–robot collaborative assembly. *CIRP Annals*, 65(1):61 – 64, 2016.

[MKM+16] George Michalos, Panagiotis Karagiannis, Sotiris Makris, Önder Tokçalar, and George Chryssolouris. Augmented reality (ar) applications for supporting human-robot interactive cooperation. *Procedia CIRP*, 41:370 – 375, 2016. Research and Innovation in Manufacturing: Key Enabling Technologies for the Factories of the Future - Proceedings of the 48th CIRP Conference on Manufacturing Systems.

[MLMON07] Alejandra Menchaca, Andrew Liu, Charles M. Oman, and Alan Natapoff. Influence of perspective-taking and mental rotation abilities in space teleoperation. pages 271–278, 01 2007.

[MMCD03] F Marin, Henrich Mannel, Lutz Claes, and Lutz Dürselen. Accurate determination of a joint rotation center based on the minimal amplitude point method. *Computer aided surgery : official journal of the International Society for Computer Aided Surgery*, 8:30–4, 02 2003.

[Mos67] R. S. Mosher. Handyman to hardiman. *Society of Automotive Engineers*, MS670088, 1967.

[MPG14] A. Makhal and A. Perez-Gracia. Solvable multi-fingered hands for exact kinematic synthesis . In *Advances in Robot Kinematics*, Ljubljiana, Slovenia, June 2014.

[MRL+14] Fabien Dal Maso, Maxime Raison, Arne Lundberg, Anton Arndt, and Mickaël Begon. Coupling between 3d displacements and rotations at the glenohumeral joint during dynamic tasks in healthy participants. *Clinical Biomechanics*, 29(9):1048 – 1055, 2014.

[MS10] J. M. McCarthy and G.S. Soh. *Geometric Design of Linkages*. Springer-Verlag, New York, second edition, 2010.

[MSVR10] M.T. Mason, S.S. Srinivasa, A.S. Vazquez, and A. Rodriguez. Generality and simple hands. *International Journal of Robotics Research*, 2010.

[MZDG93] Paul Milgram, Shumin Zhai, David Drascic, and J Grodski. Applications of augmented reality for human-robot communication. pages 1467 – 1472 vol.3, 08 1993.

[New] Wyatt S Newman. Stability and performance limits of interaction controllers. *Journal of dynamic systems, measurement, and control*, 114.

[Nie04] Yves Nievergelt. Perturbation analysis for circles, spheres, and generalized hyperspheres fitted to data by geometric total least-squares. *Math. Comput.*, 73(245):169–180, January 2004.

[NMG04] S.S. Nudehi, Ranjan Mukherjee, and Moji Ghodoussi. A haptic interface design for minimally invasive telesurgical training and collaboration in the presence of time delay. pages 4563 – 4568 Vol.5, 01 2004.

[Oea14] L.U. Odhner and et al. A compliant, underactuated hand for robust manipulation. *The International Journal of Robotics Research*, 33(5):736–752, 2014.

[OMRP07] Lindsay M. Oberman, Joseph P. McCleery, Vilayanur S. Ramachandran, and Jaime A. Pineda. Eeg evidence for mirror neuron activity during the observation of human and robot actions: Toward an analysis of the human qualities of interactive robots. *Neurocomputing*, 70(13):2194 – 2203, 2007. Selected papers from the 3rd International Conference on Development and Learning (ICDL 2004) Time series prediction competition: the CATS benchmark.

[O'S14] S.B. O'Sullivan. Ch 15: Stroke. In *Physical Rehabilitation Assessment and Treatment*, Springer Tracts in Advanced Robotics. F.A.Davis Co, 2014.

[PBFH05] Clare Press, Geoffrey Bird, Rüdiger Flach, and Cecilia Heyes. Robotic movement elicits automatic imitation. *Cognitive Brain Research*, 25(3):632 – 640, 2005.

[PG10] A. Perez Gracia. Synthesis of spatial rprp closed linkages for a given screw system. *ASME Journal of Mechanisms and Robotics, 2010*, accepted for publication:xxx–xxx, 2010.

[PGM06] A. Perez-Gracia and J. M. McCarthy. Kinematic synthesis of spatial serial chains using clifford algebra exponentials. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 220(7):953–968, 2006.

[PHB12] Isaac Pastor, Heather Hayes, and Stacy Bamberg. A feasibility study of an upper limb rehabilitation system using kinect and computer games. *Conference proceedings : ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference*, 2012:1286–9, 08 2012.

[Pra87] Vaughan Pratt. Direct least-squares fitting of algebraic surfaces. *SIGGRAPH Comput. Graph.*, 21(4):145–152, August 1987.

[PRC+17] F. J. Perez-Grau, R. Ragel, F. Caballero, A. Viguria, and A. Ollero. Semi-autonomous teleoperation of uavs in search and rescue scenarios. In *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 1066–1074, June 2017.

[QSNS14] M. Quigley, C. Salisbury, A.Y. Ng, and J.K. Salisbury. Mechatronic design of an integrated robotic hand. *The International Journal of Robotics Research*, 33(5):706–720, 2014.

[RAS14] N. Robson, J. Allington, and G.S. Soh. Development of Underactuated Mechanical Fingers Based on Anthropometric Data and Anthropomorphic Tasks. Buffalo, USA, 2014. ASME.

[RBTB16] Emanuele Ruffaldi, Filippo Brizzi, Franco Tecchia, and Sandro Bacinelli. Third point of view augmented reality for robot intentions visualization. pages 471–478, 06 2016.

[RGM+07] D. J. Reinkensmeyer, J. A. Galvez, L. Marchal, E. T. Wolbrecht, and J. E. Bobrow. Some key problems for robot-assisted movement therapy research: A perspective from the university of california at irvine. In ASME, editor, *Proc. 10th IEEE Intl. Conf. on Rehabilitation Robotics*, pages 1009– 1015, Noordwijk, The Netherlands, 2007.

[RM05] Todd D Royer and Philip E Martin. Manipulations of leg mass and moment of inertia: effects on energy cost of walking. *Medicine and science in sports and exercise*, 37(4):649–656, 2005.

[Rus12] Alexandru Rusu. *Segmentation of bone structures in Magnetic Resonance Images (MRI) for human hand skeletal kinematics modelling*. PhD thesis, Erasmus Mundus in Vision and Robotics (VIBOT), 2012.

[SB06] L. Sigal and M. J. Black. Humaneva: Synchronized video and motion capture dataset for evaluation of articulated human motion. *Brown Univertsity TR*, 120, 2006.

[Sel04] J. M. Selig. *Geometric Fundamentals of Robotics (Monographs in Computer Science)*. SpringerVerlag, 2004.

[SFC+13] Giuseppina Sgandurra, Adriano Ferrari, Giuseppe Cossu, Andrea Guzzetta, Leonardo Fogassi, and Giovanni Cioni. Randomized trial of observation and execution of upper extremity actions versus action alone in children with unilateral cerebral palsy. *Journal of Neurologic Rehabilitation*, 27(9):808–815, 11 2013.

[SFS02] M. Santello, M. Flanders, and J. F. Soechting. Patterns of hand motion during grasping and the influence of sensory guidance. *The Journal of Neuroscience*, 22 (4):1426–1435, 2002.

[SG10] Zhibin Song and Shuxiang Guo. Force sensor-based platform for upper-limb motor function for stroke assessment. In *Robotics and Biomimetics (ROBIO), 2010 IEEE International Conference on*, pages 58–63, Dec 2010.

[SK08] B. Siciliano and O. Khatib. *Handbook of Robotics*. Springer, Berlin, 2008.

[SKS+14] JF Schorsch, AQL Keemink, AHA Stienen, FCT Van der Helm, and DA Abbink. A novel self-aligning mechanism to decouple force and torques for a planar exoskeleton joint. *Mechanical Sciences, 5 (2), 2014*, 2014.

[SMF14] Daniel Szafir, Bilge Mutlu, and Terrence Fong. Communication of intent in assistive free flyers. In *Proceedings of the 2014 ACM/IEEE International Conference on Human-robot Interaction*, HRI '14, pages 358–365, New York, NY, USA, 2014. ACM.

[SNR00] Mariëlle Stokdijk, Jochem Nagels, and Piet M. Rozing. The glenohumeral joint rotation centre in vivo. *Journal of biomechanics*, 33 12:1629–36, 2000.

[SPS+11] Takaaki Shiratori, Hyun Soo Park, Leonid Sigal, Yaser Sheikh, and Jessica K. Hodgins. Motion capture from body-mounted cameras. *ACM Trans. Graph.*, 30(4):31:1–31:10, July 2011.

[SR13] Gim Song Soh and Nina Robson. Kinematic synthesis of minimally actuated multi-loop planar linkages with second order motion constraints for object grasping. In *ASME 2013 Dynamic Systems and Control Conference*, pages V003T38A004–V003T38A004. American Society of Mechanical Engineers, 2013.

[SS73] G. Schmeisser and W. Seamone. An upper limb prosthesis-orthosis power and control system with multi-level potential. *Bone Joint Surg*, 55:1493–1501, 1973.

[SSPG14a] E. Simo-Serra and A. Perez-Gracia. Kinematic synthesis using tree topologies. *Mechanism and Machine Theory*, 72 C:94–113, 2014.

[SSPG14b] E. Simo-Serra and A. Perez-Gracia. Kinematic synthesis using tree topologies. *Mechanism and Machine Theory*, 72:94–113, February 2014.

[SvdS10] G. Stillfried and P. van der Smagt. Movement model of a human hand based on magnetic resonance imaging (mri). In *Proceedings of the 1st Int. Conf. on Applied Bionics and Biomechanics*, Venice, Italy, October 14-16,, 2010.

[TB08] Chia-Chin Tsai and Marcel Brass. Does the human motor system simulate pinocchio's actions? coacting with a human hand versus a wooden hand in a dyadic interaction. *Psychological science*, 18:1058–62, 01 2008.

[TMFD08] Russell H. Taylor, Arianna Menciassi, Gabor Fichtinger, and Paolo Dario. *Medical Robotics and Computer-Integrated Surgery*, pages 1199–1222. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.

[TMM05] B. Thomas, C. B. Moeslund, and E. G. Madsen. Modeling the 3d pose of a human arm and the shoulder complex utilizing only two parameters. *Integrated Computer-Aided Engineering*, 12(2):159–175, 2005.

[Ton05] B. Tondu. Modelling of the shoulder complex and application the design of upper extremities for humanoid robots. In *2005 5th IEEE-RAS International Conference on Humanoid Robots*, 5 Dec., 2005.

[TPGP16] A. Tamimi, A. Perez-Gracia, and M. Pucheta. Structural Synthesis of Hands for Grasping and Manipulation Tasks. In *Advances in Robot Kinematics*, June 2016.

[TPLC+16] Christine Tempelaere, Jérome Pierrart, Marie-Martine Lefèvre-Colau, Valérie Vuillemin, Charles-André Cuénod, Ulrich Hansen, Olivier Mir, Wafa Skalli, and Thomas Gregory. Dynamic three-dimensional shoulder mri during active motion for investigation of rotator cuff diseases. *PLOS ONE*, 11(7):1–12, 07 2016.

[TPPG18] A. Tamimi, M. Pucheta, and A. Perez-Gracia. Enumeration, structural and dimensional synthesis of robotic hands: theory and implementation. *IEEE Transactions on Robotics*, under submission, 2018.

[TRK+17] John Thomason, Photchara Ratsamee, Kiyoshi Kiyokawa, Pakpoom Kriangkomol, Jason Orlosky, Tomohiro Mashita, Yuki Uranishi, and Haruo Takemura. Adaptive view management for drone teleoperation in complex 3d structures. In *Proceedings of the 22Nd International Conference on Intelligent User Interfaces*, IUI '17, pages 419–426, New York, NY, USA, 2017. ACM.

[Tsa99] L.-W. Tsai. *Robot Analysis: The Mechanics of Serial and Parallel Manipulators*. Interscience, New York, 1999.

[Tsa00] L.-W. Tsai. *Enumeration of Kinematic Structures According to Function*. CRC Press, 2000.

[Tsa01] Lung W. Tsai. *Mechanism Design: Enumeration of Kinematic Structures According to Function*. CRC Press, Boca Raton, 2001.

[UMCA06] Cosimo Urgesi, Valentina Moro, Matteo Candidi, and Salvatore Maria Aglioti. Mapping implied body actions in the human motor system. *The Journal of neuroscience : the official journal of the Society for Neuroscience*, 26 30:7942–9, 2006.

[VBK+13] Michael Villiger, Dominik Bohli, Daniel Kiper, Pawel Pyk, Jeremy Spillmann, Bruno Meilick, Armin Curt, Marie-Claude Hepp-Reymond, Sabina Hotz-Boendermaker, and Kynan Eng. Virtual reality–augmented neurorehabilitation improves motor function and reduces neuropathic pain in patients with incomplete spinal cord injury. *Neurorehabilitation and Neural Repair*, 27(8):675–683, 2013. PMID: 23757298.

[Vee00] H.E.J. Veeger. The position of the rotation center of the glenohumeral joint. *Journal of Biomechanics*, 33(12):1711 – 1715, 2000.

[VUPGK06] M.C. Villa Uriol, A. Perez-Gracia, and F. Kuester. Humanoid synthesis using clifford algebra. In *Proc. of the 2006 IEEE Int. Con. on Robotics and Automation (ICRA)*, May 2006.

[VY96] Dirkjan Veeger and B Yu. Orientation of axes in the elbow and forearm for biomechanical modelling. pages 377 – 380, 04 1996.

[Wal73] K.J. Waldron. A study of overconstrained linkage geometry by solution of closure equations - part ii- four-bar linkages with lower pair joints other than screw joints. *Mechanism and Machine Theory*, 8:233–247, 1973.

[WC10] Y. Chen W.H. Chai. The line-symmetric octahedral bricard linkage and its structural closure. *Mechanism and Machine Theory*, 45:772–779, 2010.

[WF04] Erica J Weiss and Martha Flanders. Muscular and postural synergies of the human hand. *Journal of Neurophysiology*, 92(1):523–535, 2004.

[WHLS18] Michael Walker, Hooman Hedayati, Jennifer Lee, and Daniel Szafir. Communicating robot motion intent with augmented reality. In *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*, HRI '18, pages 316–324, New York, NY, USA, 2018. ACM.

[WIM+15] Atsushi Watanabe, Tetsushi Ikeda, Y Morales, Kazuhiko Shinozawa, Takahiro Miyashita, and Norihiro Hagita. Communicating robotic navigational intentions. 10 2015.

[Wina] innerbody. hand and wrist. `https://www.innerbody.com/image/skel13.html`. Accessed: 2019-03-25.

[Winb] innerbody. humerus. `https://www.innerbody.com/image_skelfov/skel19_new.html`. Accessed: 2019-03-25.

[Winc] Leap motion sdk for unity. `https://developer.leapmotion.com/unity`. Accessed: 2018-09-30.

[Wol86] Herman J. Woltring. A fortran package for generalized, cross-validatory spline smoothing and differentiation. *Advances in Engineering Software (1978)*, 8(2):104 – 113, 1986.

[Wol95] Herman J Woltring. Smoothing and differentiation techniques applied to 3-d data. *Three-dimensional analysis of human movement*, pages 79–100, 1995.

[YYPG13] Ken Bosworth Yimesker Yihun and Alba Perez-Gracia. Link-based performance optimization of spatial mechanisms. In *ASME 2013 International Design Engineering Technical Conferences and Computers and Information in Engineering ConferenceVolume 6A: 37th Mechanisms and Robotics ConferencePortland, Oregon, USA, August 4–7, 2013*, 2013.

[ZZF04] Jing-Shan Zhao, Kai Zhou, and Zhi-Jing Feng. A theory of degrees of freedom for mechanisms. *Mechanism and Machine Theory*, 39(6):621 – 643, 2004.

Appendix A

Mocap Toolbox

This tool box is specifically created for Robot Kinematics and Synthesis but can be used only for visualization purpose as well. First the motions of the human body parts are captured using markers on the skin and cameras like Bonita. These whole process can be done by Vicon Systems, for example. The output of the motion capture process can be an excel file (csv) giving the coordinates of the markers. But what is important in Kinematics is orientation too. So by making an L-frame and putting three markers on it we can find the orientation of any rigid body. This toolbox uses the data of each three markers on each L-frame to make the orientation of the desired rigid body and make 4x4 transformation homogeneous matrices. Then it plots all these matrices for all the markers having data in csv file. The user can choose the color and size of the frames. Also if the user does not want the toolbox to plot any of the markers, just its box should be unchecked. The range of the plotted data can be specified too. Accordingly the slider bar changes and the user can see the whole motion or part of it by sliding the slider.

There are three sliders in the toolbox which basically do the same thing. However, the second section is for selecting the specific range of frames. It enables the user to visualize and compare the selected frames with whole motion. The last section is for the solution after doing the synthesis. The user can compare the calculated frames with the selected ones and also with whole motion (See Figure A.1).

There are two test files (test1.csv and test2.csv). The input section of MCD.m file should be edited according to the csv file and the rest of the file should remained unchanged. The csv file must only contain numbers (no letters). The number of rows are equal to the number of frames captured by cameras and the columns contain x, y and z of each marker. Then the number of columns is equal to the number of markers multiplied by 3.

To run the toolbox, you need only to run MCD.m file. The toolbox is able to plot the frames in different coordinates meaning that all the homogeneous matrices can be expressed in whatever base the user wants. Running MCD.m gives the information about the number of the final markers (each single marker is considered as one final marker and each three markers placed on L-frame is considered as one final marker as well). There is prompt question. The user should define which marker is going to be

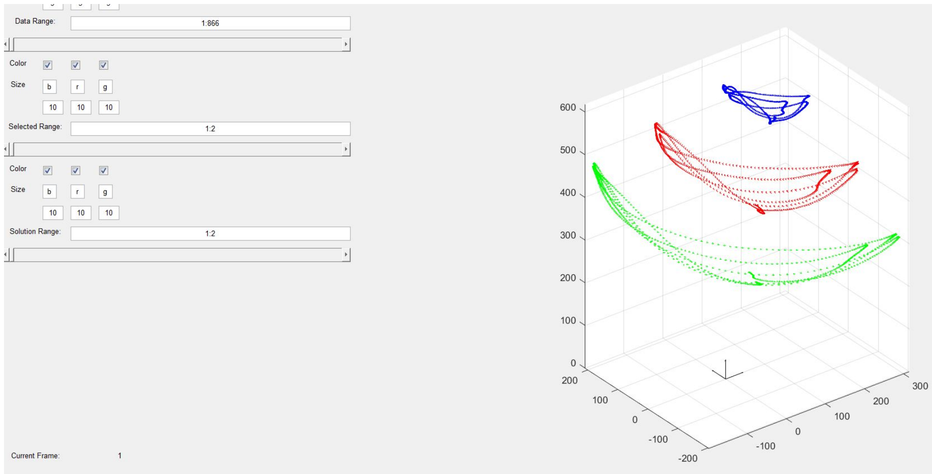


Figure A.1: Mocap _toolbox plotting data for three L-frame markers

the base (The frame that all other frames are going to be expressed in). The numbers of the final markers define the base. If the user want to see the frames relative and expressed in camera frame then 0 is the answer.

The first part of MCD.m is for inserting the inputs. This is the only section that the user needs to edit based on the data. - The name and path of the csv file. - The names of the markers - The order (number) of the markers The other parts of the file and all other file should be untouched.

[mata] refers to the tool box in MathWorks.com. And a demonstration of what this toolbox is capable of, as a tutorial video, is given in [matb].

Appendix B

Comparison For GH Location

Table B.1: Subject 1

	g_x	g_y	g_z	N_s
geom	-33.8115	-26.3174	-64.9564	2.2576
alg	39.0473	-29.2831	-65.7504	2.3135
Pratt	-35.3680	-27.1644	-65.0832	2.2619
CTT	-17.8986	-4.1869	-50.9539	4.5
MAM	-11.0381	-1.4041	-64.1404	3.2908
SCoRE	-17.8986	-4.1869	-50.9539	4.5
IHA	-70.6014	-52.2918	-74.0195	6.0864
BSTD	-30.6479	-19.5937	-68.0963	2.5138

Table B.2: Subject 2

	g_x	g_y	g_z	N_s
geom	-30.2021	-1.8489	-60.6682	1.7272
alg	-68.4819	11.4385	-101.5668	2.2459
Pratt	-57.8410	7.5254	-90.0157	1.9283
CTT	-30.2470	-10.8394	-50.6161	1.9492
MAM	-30.6159	-7.1325	-49.0957	2.0002
SCoRE	-30.2470	-10.8394	-50.6161	1.9492
IHA	-61.6237	12.5133	-79.2640	3.0864
BSTD	-22.1117	-3.6210	-54.1893	1.7419

Table B.3: Subject 3

	$\mathbf{g}_x$	$\mathbf{g}_y$	$\mathbf{g}_z$	N_s
geom	-53.0706	-6.5596	-71.4458	2.6032
alg	-57.8378	-7.1240	-72.0079	2.6471
Pratt	-54.9340	-6.7625	-71.4408	2.6095
CTT	-35.3363	9.4688	-44.5908	6.1351
MAM	-45.0950	28.2621	-53.6695	4.3602
SCoRE	-35.3363	9.4688	-44.5908	6.1351
IHA	-67.6808	-6.0287	-54.0050	3.1733
BSTD	-56.2992	-3.6698	-73.0683	2.7149

Table B.4: Subject 4

	$\mathbf{g}_x$	$\mathbf{g}_y$	$\mathbf{g}_z$	N_s
geom	-66.1433	2.1608	-78.4869	1.9254
alg	-83.9074	12.9775	-85.0035	2.1276
Pratt	-87.4049	15.0729	-85.9669	2.2285
CTT	-55.0998	7.9232	-61.8618	3.9332
MAM	-52.8328	9.4416	-60.7966	4.1043
SCoRE	-55.0998	7.9232	-61.8618	3.9332
IHA	-64.5759	23.2242	-48.4079	6.1359
BSTD	-63.6725	-2.3586	-74.4562	2.0579

Table B.5: Subject 5

	$\mathbf{g}_x$	$\mathbf{g}_y$	$\mathbf{g}_z$	N_s
geom	-72.2540	-2.4655	-77.3657	3.9638
alg	-122.5052	5.0235	-95.7990	5.9005
Pratt	-89.9308	-0.3247	-83.2262	4.0421
CTT	-20.8721	11.1056	-48.3451	4.9738
MAM	-35.9131	11.5771	-48.5103	5.2444
SCoRE	-20.8721	11.1056	-48.3451	4.9738
IHA	-88.6529	4.7724	-67.8833	5.4280
BSTD	-40.4352	-3.8152	-65.1963	4.0668

Table B.6: Subject 6

	$\mathbf{g}_x$	$\mathbf{g}_y$	$\mathbf{g}_z$	N_s
geom	-24.7635	-15.0274	-69.0734	1.7948
alg	-78.9999	-23.2430	-103.8320	2.3424
Pratt	-83.7926	-24.3570	-106.7721	2.5069
CTT	-18.0818	-16.7776	-57.3233	1.8983
MAM	-35.9529	-12.3133	-57.5117	2.4139
SCoRE	-18.0818	-16.7776	-57.3233	1.8983
IHA	-60.6155	-14.6963	-57.6213	3.7277
BSTD	-29.3477	-15.5948	-68.7418	1.8134

Appendix C

Publications

1. Omid Heidari, Alba Perez-Gracia, ”Virtual Reality Synthesis of Robotic Systems for Human Upper-limb and Hand Tasks”, IEEEVR Conference, Osaka, Japan, 2019.

2. Omid Heidari, Vahid Pourgharibshahi, Alex Urfer, Alba Perez-Gracia, ”A New Algorithm to Estimate Glenohumeral Joint Location Based on Scapula Rhythm”, 2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC).

3. Omid Heidari, Eric T Wolbrecht, Alba Perez-Gracia, Yimesker S Yihun, ”A task-based design methodology for robotic exoskeletons”, Journal of Rehabilitation and Assistive Technologies Engineering, 2018.

4. Omid Heidari, John O Roylance, Alba Perez-Gracia, Eydie Kendall, ”Quantification of upper-body synergies: a case comparison for stroke and non-stroke victims”, ASME 2016 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, 2016.