Synergies-Based Teleoperation of a Robot Hand by a Hand Tracking System

Supervisor: Dr C. Della Santina

4777522 Chia-Fu, Lee

Master of Bio-Robotics Thesis

TU Delft Delft University of Technology

Delft Department of Cognitive Robotics

Synergies-Based Teleoperation of a Robot Hand by a Hand Tracking System

For the Master degree of Mechanical Engineering track
Bio-Robotics at Delft University of Technology

Supervisor: Dr C. Della Santina

4777522 Chia-Fu, Lee

November 26, 2021

Faculty of Mechanical, Maritime and Materials Engineering (3mE) Department of Cognitive Robotics ·
Delft University of Technology

The cover image is an illustration of HaptX's telerobotics system [1].

Cognitive
Robotics

Abstract

Hand pose tracking in 3D configuration is one of the popular components of human-computer interaction research, which can be seen applying in virtual reality (VR) and augmented reality (AR). Although the research field has been thoroughly studied in the past few decades, most methods emphasized recognizing hand gestures or joint locations. However, computer interaction these days requires numerous manual object interactions. To achieve the accurate hand tracking and control, the paper first reviews state-of-the-art hand pose estimation and tracking methods. Then we describe the general procedure for recovery 3D hand model. In the second place, the results of tracking whole hand kinematics using the extended Kalman filter (EKF) based on data recorded from the surface markers are presented. In the part of teleoperation, the paper presents a novel method for applying synergy-based mapping strategy for SoftHand 2 control. And we demonstrate the simulation results of telemanipulation with hand tracking through Simulink.

Table of Contents

List of Figures

List of Tables

Acknowledgements

Firstly, I appreciate my supervisor Dr. Cosimo Della Santina, much for his patience with guiding students and his assistance during the writing of this thesis. And of course, thank the considerable support from my family. Especially, I want to say thank you to Shu-Yu, Chen. He helped me not only think critically with questions but also see through the whole idea of tracking methods. Without any of the above, I would not have had the chance to complete my master thesis.

Delft, University of Technology
November 26, 2021

Chia-Fu, Lee

Chapter 1

Introduction

A background study for this thesis was first conducted, forming the foundation for the following thesis. In this chapter, the study will be divided into few sections, and each of them will briefly describe related topics in the research. It starts with Section 1-1, here we point out the motivation with the study. In section 1-2, it introduces state-of-the-art hand motion capture techniques. Section 1-3 introduces different methodologies for performing real-time hand tracking and highlights the challenges. In addition, the potential improvement in hand tracking lays out the motivation behind this project. Section 1-4 reviews the application of teleoperation of a robotic hand. Another focal point in sections 1-4 is the summary of mapping methods from a human hand to a robotic hand. The last section, 1-5 provides an overview of the framework of the thesis project.

1-1 Motivation

Robotics has risen rapidly in recent years. However, compared with artificial intelligence based on algorithms, application technologies that track and repeat human commands with vision are gradually increasing in demand in different industries, such as surgical robots, remote control technology and other fields. Mainly, this kind of academic research is gradually combined with Human-Computer Interaction (HCI) to form a new research direction in the robotics industry. Our study's ultimate goal is to combine a human hand movement tracking result and apply it to operate a robotic hand, which we consider has similar kinematics.

1-2 Hand Motion Capture and Applying Devices

With the growth of technology, Human-Computer Interaction (HCI) has drawn much attention. One of the typical applications in this field would be capturing the kinematics of a human hand. However, because of the human hand's complexity and the limitation of the current devices, the difficulty in motion capture is still great. In this section, the related research problems of capturing moving hand kinematics will be discussed first, and then existing methods with employed devices.

Performing hand motion capture is more challenging than a human body capture; due to fingers and palms' association, the degree of freedom is significant and leads to high dimensional space [18]. Moreover, self-occlusion, the variance of different person's hands, and the rapid hand movement speed are difficulties in hand motion capture and tracking. To cope up with occlusion, data gloves and other wearable sensors are used to make up for the deficiency of missing data [19], and these approaches can be classified as contact-based motion capture. These devices are wearing on the hand, sensors on the glove will provide data of each finger joint and the palm's angle at any time and any position. One example implemented the whole hand motion capture using CyberGlove has revealed the difference of hand size from different persons and concluded that it does not affect the data glove from acquiring data [20]. Apart from that, surface marker-based devices have been utilized for hand motion capture as well [21]. Nowadays, related technology is cheaper and more facile. As a result, they are more flexible than traditional data gloves, and bring benefits such as bending motion can be measured directly.

Although data glove provides a promising way to avoid a self-occlusion problem, analyzing the noisy raw data and dealing with the complex calibration are still topics that remain more investigated [18]. On the contract of contact-based devices, visual-based motion capture devices or so-called optical systems collect data through only cameras. Visual tracking can be traced back to the 1990s; JM Rehg [22] investigated the hand state model by projecting the spatial hand geometry into the image plane; thus, the model's output became lines and points. They applied a residual model in the algorithm to correct the prediction and perform the hand state estimation. The idea then lays the foundation of visual-based motion capturing. With more researchers immersed in the study, color and edge information are now widely applied for increasing the accuracy of hand matching in this field. Another study using the Kinect sensor to capture the image information, and successfully minimized the difference between the 3D hand model and the camera observation [23]. In this work, an objective function represents the discrepancy between the human hand and the 3D hand model. The authors solved the function by applying Particle Swarm Optimization (PSO) method. Nowadays, the latest works focus on tracking real-time motion with less information computation and enhancing the cooperation with AR/VR devices.

Both contact-based and video-based motion capture devices have their advantages as well as drawbacks. Some researchers conducted the fusion of the two techniques and fulfilled

for high accuracy information. The camera can directly provide a hand state while the data glove can handle the data gap problem such as occlusions. By using the fusion of the two techniques, the estimation of hand position becomes more reliable. A research [24] captured finger movement by twenty-four markers attached to subjects' hands and six calibrated cameras. Employing a hierarchical chain, they assembled the hand's rigid body segments into a twenty-two DoF kinematical hand model. At this work, they provided less than one percent occlusion in real-time hand motion tracking. Research applying 5DT Data Glove and Nimble VR Kinect has presented a robust hand motion [25]. The authors applied the Kalman filter to fuse vision-based with contact-based tracking data, successfully improving pose estimation precision. As the section has pointed out, the method for capturing hand motion can be classified into contact-based and visual-based, and both have pros and cons. To take their advantages and eliminate disadvantages, in our research, we applied a surface marker-based device to acquire hand joints data in advance, then in our plan is to applying the OptiTrack camera to implement hand tracking.

1-3 Vision-Based Hand Tracking

Hand pose tracking can be considered a recursive hand detection process with time and prior knowledge of the environment [26]. A hand model with kinematic information is required to perform an accurate hand pose estimation in the tracking process, while computation results from the model-based approach can be seen as reconstructing a hand structure. The structure follows the movement of the real hand, with the recognized appearance or joints. By implementing motion capture devices, a series of hand movements can be collected. The sorted data, such as joint locations or depth image information, need to be parsed by different methods to complete pose estimation or gesture recognition.In contact-based devices, data can be collected directly from sensors. However, the main challenge will be how to perform data analysis with numerous pieces of information. For vision-based devices, plenty of research with different approaches has been implemented due to the cheaper and easier access of cameras. In this section, various 3D hand models will be discussed first. Next to the definition of the hand model, the challenges of hand reconstruction and tracking will be described in detail. Last but not the least, the approaches for vision-based hand tracking, which is the system we decided to apply in work, will be discussed.

1-3-1 3D Hand Model

A model of the hand can provide the kinematic information of a human hand. Therefore, an illustration of the hand in accurate hand tracking approach system is needed. Even for some methods that consider hand model might be redundant, e.g., discriminative methods, the shape-oriented model can still provide a clear idea with the result of contour and appearance information of the hand. The anatomy of the human hand will first be discussed in this

section, and then some examples of the kinematic model. At the end of this section, some hand re-construction with 3D information will also be addressed.

As the picture that is shown in Figure 1-1, the human hand consists of 27 bones, and 19 of them form the palms and fingers, while the other eight bones are located at the wrist. These bones, without deformable tissues, with joints that have one or more DoF, form the rigid body system of the hand [3]. Joints between the bones are named according to their location, such as metacarpophalangeal (MCP) locates from fingers to the palm, interphalangeal (IP) is joining finger segments, and carpometacarpal (CMC) connects the metacarpal bones to the wrist. For interphalangeal (IP),they can be classified into distal interphalangeal (DIP) and proximal interphalangeal (PIP). Understanding hand anatomy helps to comprehend the characteristics of the hand, and the kinematic model can therefore be built. In the skeleton, IP joints have only flexion-extension ability (1 DoF), and the CMC joints are assumed to be fixed. Regarding the MCP located at the thumb, which is also called trapeziometacarpal (TM), it can be considered having 2 DoF or extending to 3 DoF. From the study of Erol el at. [18], the consideration of seeing TM as a two DoF saddle joint will form a restrictive model, but it has been used in many studies. Based on the different ways for analyzing the degree of freedom of the hand, it is common to see the kinematic model has either 26 DoF or 27 DoF.

Figure 1-1: The anatomical skeleton of the human hand [2]

As the example from Figure 1-2, the applied kinematic model, however, due to the complexity in result evaluation and the efficiency of computation, will be modified before processing in the tracking system. Researchers try to avoid complicated hand model or motion constraints to reduce the search space for pose estimation. Work from Santello et al. (1998) deployed a 15 DoF kinematic hand model for postural hand synergies [21]. To extend the accuracy in the description of a human hand, Della Santina et al. (2017)

proposed a hand structure with 20 DoF, which includes the 15 DoF hand model used in Santello et al. (1998) [16]. Aside from simple illustration with linkages and joints, some studies performed hand models that are supported with shape information to fulfill appearances of the hand in the arbitrary configuration. One study designed a system that can generate a subject-specific realistic hand model by a monocular image [27]. This system used a high-vertex count hand model with blend shapes learned from scanning, and it can produce realistic hand shapes in real-time (see Figure 1-3).

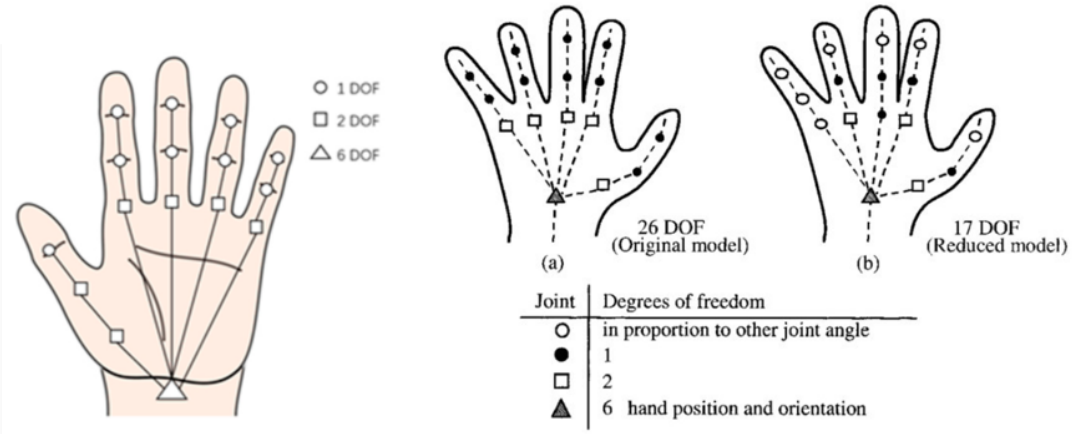


Figure 1-2: Left: A kinematic hand model with 27 DoF [3]. Right: An example of the hand model with 26 DoF and the reduced model [4]

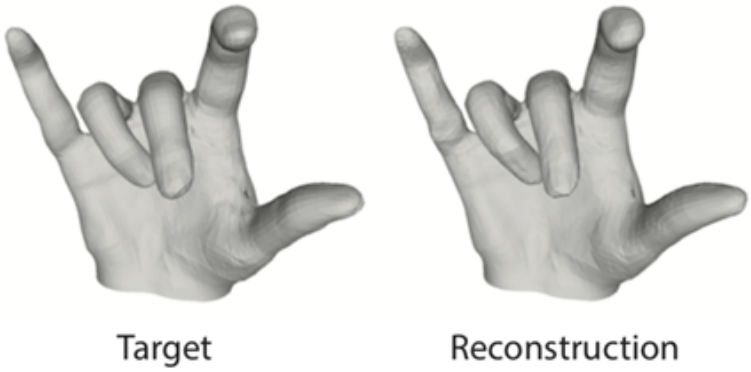


Figure 1-3: The target hand image and the result of the generating 3D hand shape [5]

Although the same motion model can be assumed for all users, for the shape model, the same assumption cannot hold true. Also, if precision is required for the application, these specific hand shape-related models then need calibration procedures to estimate user-specific measurements. To choose between precision and the computation effort, it is understandable that the real-time hand tracking with kinematic information tends to apply more simple illustration.

1-3-2 Challenges in Vision-Based Hand Tracking

One of the main challenges of visual hand tracking is the high-dimensional hand configuration space. Moreover, Human hand has high appearance changes in each individuals. When the hand motion is capturing, the potential interference from the environments can cause instability to the results. These challenges will be described separately below.

`The high-dimensional configuration space:` To estimates the full-degree-of-freedom hand posture problem, the dimensionality that needs to be considered is very high. Each finger has 4 DoF, and thus they create 20 local degrees of freedom for the hand posture. Although some studies claimed the MCP of the middle finger could be eliminated, the total number is 19 DoF still high. Other than that, the additional degree of freedom for the axial rotation that modeled the thumb also needs to count. And the 6 DoF for illustrating global position and direction. Thus, in total, the problem space has 26 dimensions [6].

`Hand variety of different individuals:` Both hand appearance and the way of performing movements can change significantly from person to person. The geometry of different hands causes one main effect in the 3D hand model. Thickness, length of finger bones, and the width of the hand are influential factors. In short, to detecting the hand in the input image is challenging because its appearance and position are not known in advance.

`Interference from the environment:` To detect the hand from an input image, it is necessary to apply image segmentation. The related algorithm can be skin-color segmentation or background subtraction. These methods can part the hand beside other objects in the image. Features such as contours, edges, and silhouettes are needed to identify a hand from the image. Above all, the more complex the background, the harder feature extraction can distinguish the hand. Neither to say tracking a hand from a moving real-time video.

1-3-3 Different Tracking Approaches

Generally, hand tracking methods for vison-based motion capture can be categorized into three main approaches: appearance-based, model-based, and the hybrid of appearance-based and model-based method [28]. Appearance-based approaches are also known as the discriminative, data-driven methods, which will learn from the training images and address the mapping between the observed feature space and the target hand space [26], [29], [30], [31]. Compared to other methods, appearance-based methods require less computational cost and less complicated feature recognition because feature learning can be done offline. Feature detection methods like image differencing, skin color detection, cross-correlation, and point distribution models are commonly seen in work related to appearance-based approaches. Since discriminative methods rely on feature recognition, the study of learning procedures with different algorithms, such as deep learning and the principal component analysis (PCA), also draws much attention in this field. In Chapter 4, the mathematical

theory of principal component analysis will be discussed. The quality of the result from the appearance-based approaches highly depends on the training data set since the methods learn from those employed pictures. Thus, they are more useful for recognizing discrete or continuous hand gestures than general hand movement tracking. A clear diagram of the discriminative workflow is provided from [6] (see Figure 1-4).

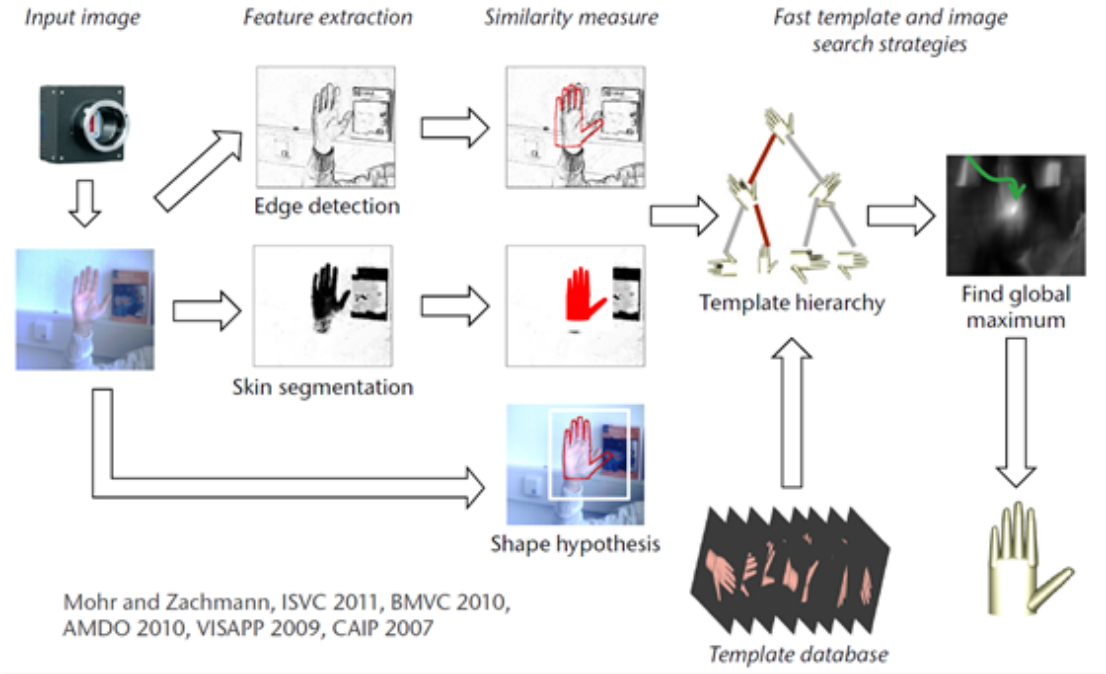


Figure 1-4: An example diagram of how appearance-based approach works [6].

In contrast with discriminative methods, the model-based approach does not require any data training to learn the hand model's parameters. Model-based methods are also known as generative or model-driven methods, which will require a 3D hand model of the hand based on prior knowledge. The parameters of the hand model must be initialized first, and the most seen initialization method would be estimating the pose from the previous frame and set it as the initialization value of the current frame [3]. During updating estimation, the 3D hand structure will be optimized continuously to find suitable parameters. In this approach, how to evaluate the similarity between the hypothesis and the observed image is the main difference with each model-based method. Generally, fast calculation and less computational cost are preferred. To achieve the comparison between 3D hand structure and 2D input image in a model-based approach, projecting a 3D hand model to image space is a commonly applied method [32], [33]. Simultaneously, some researchers provided 2D or 2.5D image transfer methods to 3D models (see Figure 1-5). Moreover, the similarity functions or sometimes called loss functions, to measure the discrepancy between the hypothesis hand model (target object) and detected features (from input images) such as edges, shading, optical flow, and depth value are widely used as an optimization process in model-based approaches [23], [34], [35], [22].

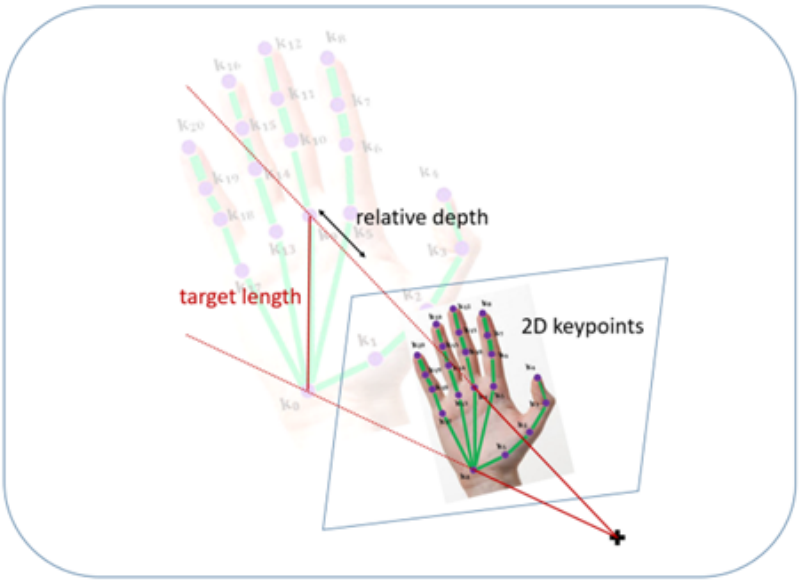


Figure 1-5: One example of the 3D image recovering from 2.5D representation [7]. The 2.5D approximate 3D method uses a specific bone length; for example, in the image, the length from the wrist to the middle finger, and the 2.5D pose is projected to a position that conforms to this length as an approximate 3D pose.

With the increasing popularity of motion capture research, some researchers combined the two mainstream hand tracking methods and sought better results. In the discussion above, it is not hard to see that generative methods require computation of the hand model parameters when updating each frame. The expensive computing cost drives the speed slow; thus, real-time hand tracking can barely perform in generative methods. For discriminant methods, the training of labeled data can be operated offline, which helps reduce the running time of tracking. However, the recognition of the hand poses is based on the training data set. If the scenes used for training are quite different from the observed video sequence, hand tracking will also be unreliable. To this stand, the hybrid of model-based and appearance-based approaches can be more practical. Jan et al. [28] extended the hand model's kinematic layer, which enables the hand parameters learnable. They modified the Spatial Transformer Networks to decrease the variance of the observed information. Another hybrid method applies a novel matrix to perform regression and estimate a 21 DoF hand model [8]. The authors applied deep convolutional neural networks (CNN) to train data. They repeated the same operation on both global and local pose parameters to get an activation feature in a depth map. An overview of this work is shown in Figure 1-6.

In short, an appearance-based method can effectively support a generative method in computing optimization. Furthermore, the initialization of the parameters of the hand in each frame can easily be calculated by the discriminative method. On the contrary, a model-based method can provide the kinematic model and physical constraints to make up

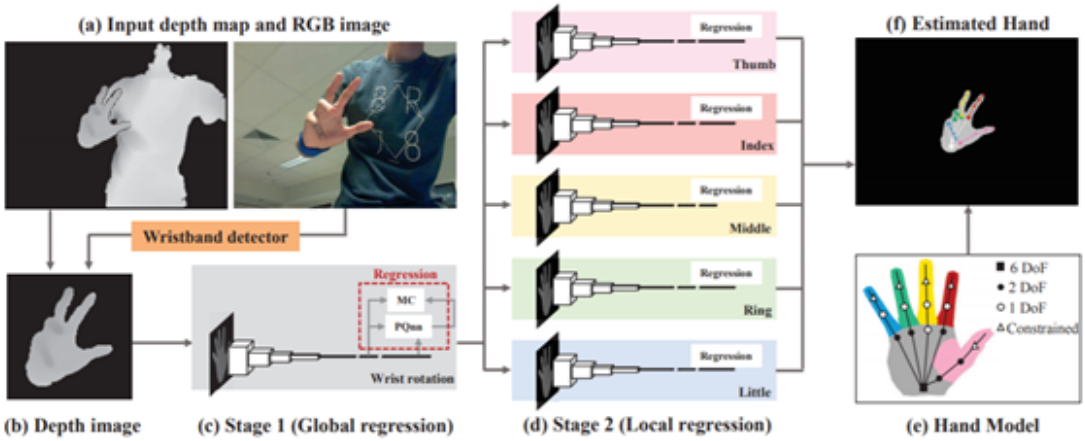


Figure 1-6: An overview of a hybrid-approach hand tracking work from [8], which estimates realtime hand pose with a hand model and discriminatively trained ConvNets. The work is performed with RGB images and depth information.

for the insufficiency in the appearance-based method and avoid self-occlusion. It is believed that the hybrid approach can bring a lot of advancement to hand tracking research.

1-4 Tele-operation of a Robotic Hand

As the need and interest in robots grow, the advancement of technology brings the robotics industry booming that even becomes significant in our lives. In the medical field, robot surgical instruments help doctors to provide safe and precise surgery. In the industrial field, robot arms and intelligent machining bring more efficient production lines. Moreover, humanoid robots company with the elder in our daily life while the smart housekeeper provides people a convenient lifestyle. In robotics, teleoperation is one of the main characteristics that affect how the whole mobile or a single part of the robot moves. In our current work, we aim to perform a robotic hand manipulation by applying the result from hand tracking. Previous sections have covered the topics related to hand tracking, and in this section, teleoperation with robotic hand will be detailed described.

1-4-1 Current Approaches

Teleoperation is different from autonomous control, while the human being is within the system and performing the remote control of robots. The main benefits are preventing humans from working in dangerous environments, allowing wider capacity and vision, and the overall task control can fully rely on human decision. Master-slave manipulators (MSMs) are the most commonly seen teleoperation system regarding human upper limbs. Exoskeleton is a new trend nowadays due to its convenience that the human operator can directly

wear. Exoskeleton has a kinematic structure similar to the operator's upper or lower limbs. It can simply replicate movements such as touching, grasping, and even delivering the human operator's contact forces [36]. However, these "on-hand" controllers have drawbacks include inaccuracy, heavy to carry, and interference with some of the motions that the operator would like to command [37]. To overcome the shortcomings mentioned above, while still retaining the advantages of master-slave manipulation, researchers are now conducting novel methods to perform teleoperation.

In implementing a novel method of teleoperation, motion capture devices play a vital role. In the 1990s, a complex anthropomorphic robotic hand's teleoperation was first published [37]. Through EMG signal collected from the electrodes attached to the operator's hand, the researchers accomplished teleoperation. Subsequently, the wearable motion capture device attached with magnetic measurement units (IMMUs) were released. These wearable motion capture devices can demonstrate robotic arm-hand teleoperation that simultaneously imitate movements as the motions of the operator's arm and hand. Unlike applying a contact-based motion capture device, F Gomez-Donoso et al. shows the possibility of employing only real-time image data to complete a robotic hand teleoperation [9]. They designed a scheme consisting of two convolutional neural networks; one is detecting the hand in images while the other infers the joints' 3D position. Furthermore, the result can be reconstructed to a 3D hand structure, and by mapping the hand model to a humanoid robotic hand, they perform the teleoperation of a robotic hand (see Figure 1-7, 1-8, and 1-9).

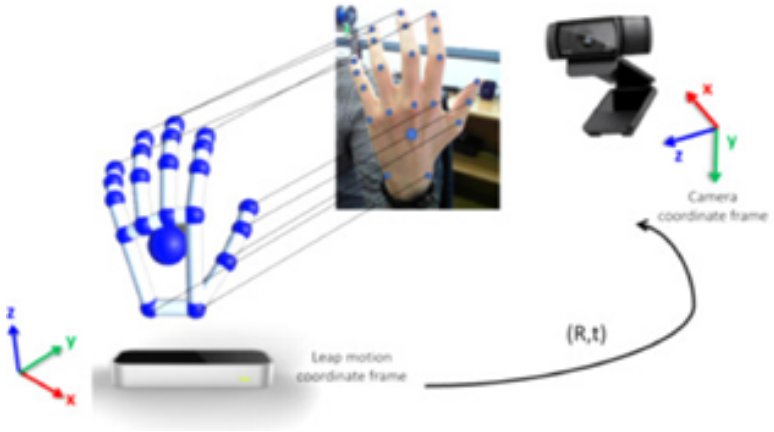


Figure 1-7: The image input from the RGB camera and the data will be detected and regress in real-time by two CNNs to form a 3D hand structure [9].

In the thesis project, we seek to implement a robotic hand control through motion capture camera input, which means the wearing device will only collect the initial hand model parameters. And the operation of the robotic hand will be driven by real-time input image data. To achieve novel methods for teleoperation requires motion mapping from the human hand to the robotic hand since the kinematic structure of robots is different from that of

Figure 1-8: The re-construction 3D hand model from the vision-input human hand information [9].

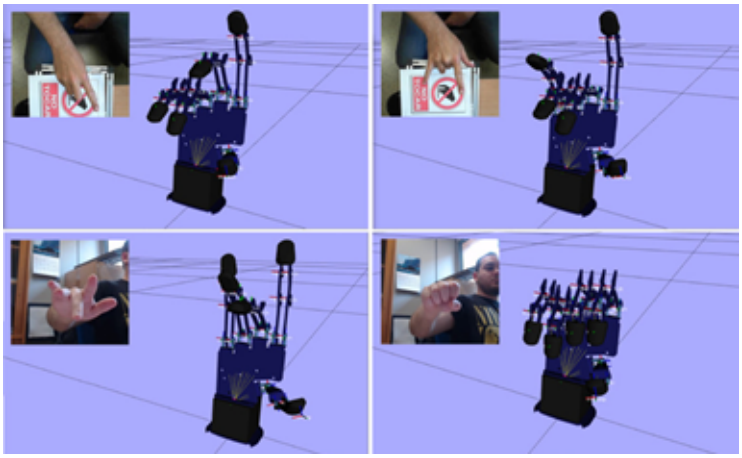


Figure 1-9: The teleoperation of the simulation humanoid robot hands [9].

the operator. Moreover, to realize user-friendly and avoid fatigue causing by long-term use, the motion configuration from human converting to robotic hand needs to be intuitive as much as possible.

1-4-2 Mapping from Human to a Robotic Hand

For the sake of intuitive teleoperation, motion mapping needs to be employed to keep the robotic hand doing the similar movement of the operator's hand in the whole course of telemanipulation. Different approaches have been proposed in the last few decades for feasible mapping (see Figure 1-10). They can be categorized into four main methodologies: fingertips mapping, joint-to-joint mapping, functional pose mapping, and object-specific mapping [38].

Fingertips mapping can also be called Cartesian space mapping, which is based on the computation of forward and inverse kinematics of the hand. The method gains positions of the human fingertips in 3D space by applying the forward kinematics for each finger

of the human hand. Next, using the inverse kinematics for the robotic hand to compute the sets of angles that drive the robot's fingertips to the same positions as the human hand's fingertips [39]. Joint-to-joint mapping is the simplest method, which is based on the assumption that the kinematics of the human and robot hands are quite similar. The mapping method imposes the human joint angles directly onto the robot joints with little or no transformation [40], [41]. Functional pose mapping has been proposed as a different approach for a human to robot hand motion mapping. It attempts to interpret the grasping function that establishes a correlation between human and robotic hand poses. In most cases, the input human pose needs to be identified before mapping between humans and the robot hand. Pao and Speeter [42] developed an algorithm that can convert the human hand posture to the corresponding manipulator position without losing functional information. The idea of object-specific mapping is that assumes the operator's thumb and index finger held a virtual circle. The virtual circle parameters have been defined, such as the size and position, and then non-linearly create a transformed virtual object in the robotic hand configuration. By doing so, the robotic fingertip locations can be computed [43].

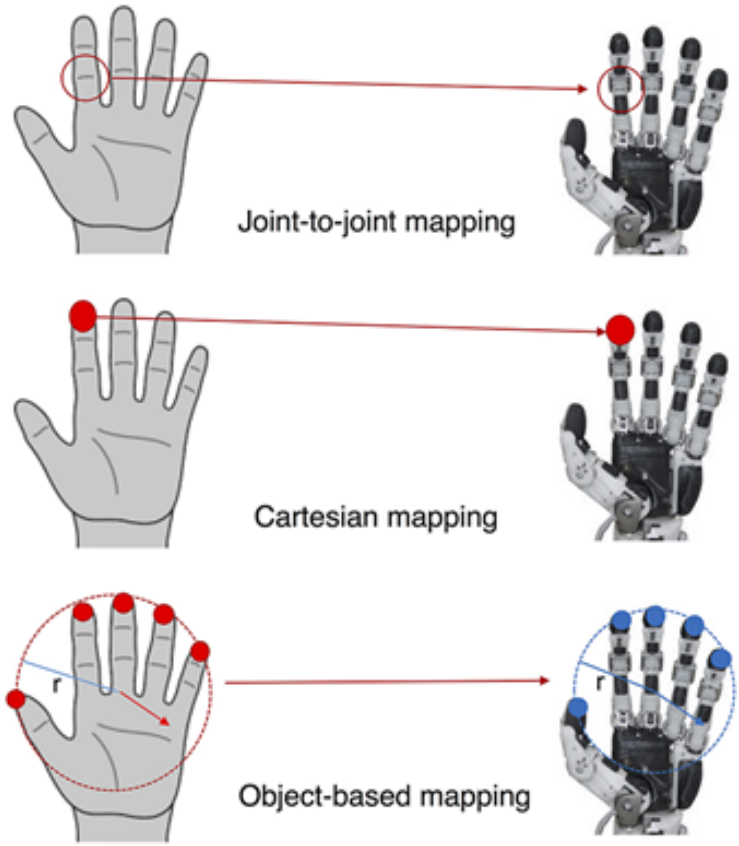


Figure 1-10: The illustration of different mapping approaches [10].

In this thesis project, the targeting robotic hand is the Pisa/IIT SoftHand 2 ($SH2$), an anthropomorphic robotic hand with 19 joints. It has five revolute joints to move each finger's adduction/abduction movement, while the other 14 are compliant rolling-contact element

(CORE) joints. The Pisa/IIT SoftHand 2 was constructed as an under-actuated manual and designed based on the neuroscience principle of movement coordination. Current neuroscience study explains that our brain organizes the control complexity of the hand movement into synergies or joints movement patterns. Therefore, the principal components reduced the involving dimensions of the hand movements and were named a robotic design concept called "Soft Synergies" [44]. The concept has been applied to robots in variant ways. For example, some works find robot hand synergies through robot poses are similar to human hand grasping poses [45], [46]. And they indicate that the muscle synergy represents a general neural strategy based on muscle coordination in hand postural tasks. Santello et al. [21] concluded in their work that there were only two principal components (PCs) needed to explain over 80% of the hand grasp posture variance. Based on the above researches, it is possible to reduce the high-dimensional data in the matrix representation and perform human-like grasping in low-dimensional space. Considering the design concept of SH2, in this project, a novel method of synergy-based mapping approach is performed to be the main application of the control.

1-5 Proposed Solution and Plan of the Thesis

Since we aim to realize reliable real-time hand movement tracking, applying a hybrid approach of the model-based and appearance-based methods to form a hand model is the best option. The reconstruction of the 3D hand model was based on the surface markers data, which has been collected in the work of [11].

Although the data-set offers enough upper limb poses through grasping taxonomies, hand kinematics is not considered in the actual experiments. As a result, we cannot directly apply the data set in the hand tracking process. We need to obtain a reliable 3D hand kinematic model; thus, the project starts with rebuilding a 3D hand structure. The progress of the hand model was accomplished through three parts as follows: Firstly, the hand kinematics model was defined in the previous study from [16]. So, we can then apply inverse kinematics to transfer markers' positions to joints' configuration and get joints' angles. In the second place, using the forward kinematics for calculating joint angles in the 3D hand kinematic model. Thirdly, apply the above result to the whole hand movement in recorded data. A flow chart can be seen below (Figure 1-11).

As the 3D hand model is completed, the tracking algorithm can then be designed based on it. From the data set, we noticed there were a lot of jerking and occlusion. In order to provide a reliable estimation of missing data, the Extended Kalman Filter (EKF) was then designed to be the tracking algorithm, and its scheme was implemented in MATLAB as post-processing. Tracking with the EKF provides a reasonable estimation with a moving hand's joint angles. In our work, we want to achieve real-time hand tracking with a pre-analysis hand kinematic model and then use this result to control a robotic hand. The control scheme is the last process of the project. As mentioned above, SH2 is designed with soft synergies concept. Therefore, applying a postural synergy-based approach is the

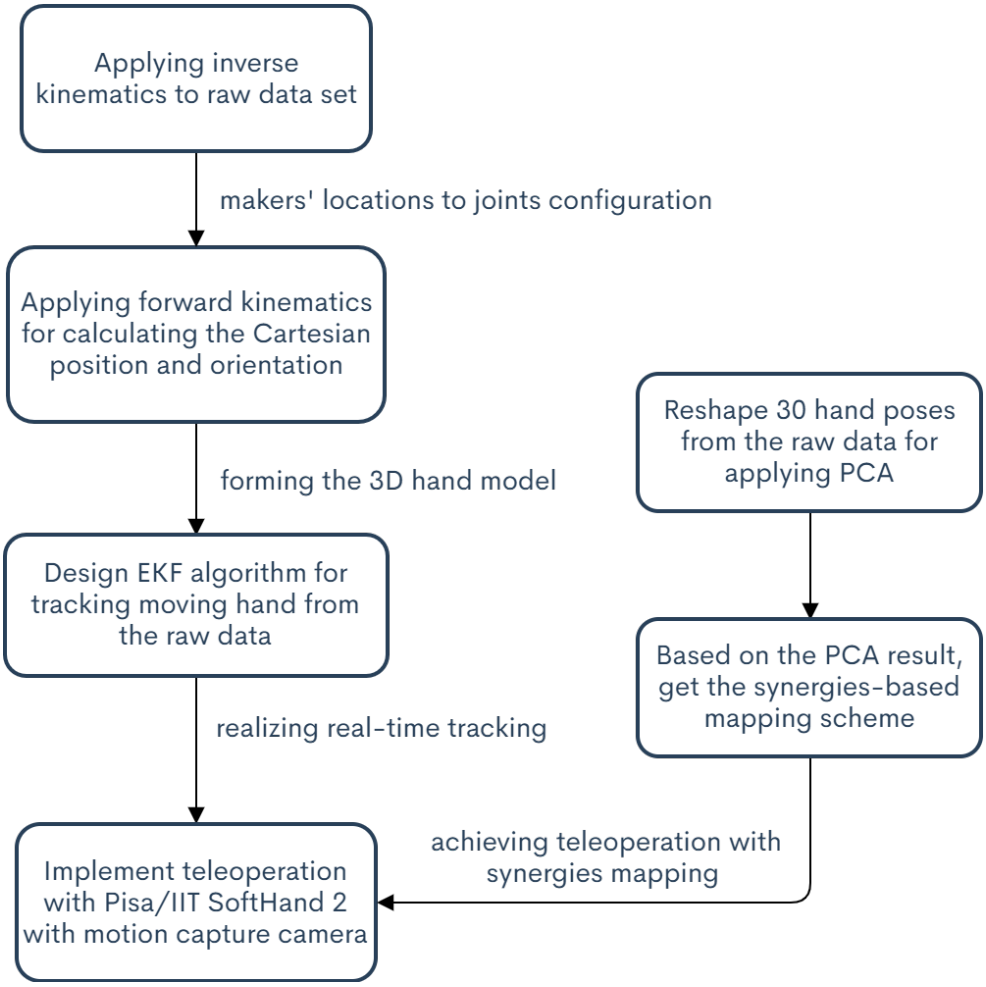


Figure 1-11: The flow chart of the thesis project.

most intuitive option to obtain a promising but straightforward control. The SH2 has two degrees of actuation (DoA) that drive this anthropomorphic hand to mimic a human being's hand pose. And that indicates the management of 19 degrees of freedom (DoF) relies only on the two motors' torsion. As a result, we introduce the dimension reduction method, principal component analysis (PCA), as the primary mapping method to operate motors' control. The PCA depicts a standard architecture for mapping many DoF to few. In this project, we will apply the PCA in the raw data set for finding two main components for representing human beings grasping poses. The details of mapping from the two components to the two DoA will be presented in chapter 5.

Chapter 2

3D Hand Model

In Figure 1-11, this section concerns the first two blocks that apply hand kinematics to reconstruct a 3D hand model. We applied the data-set collected within the recently ended H2020 EU-funded Project SoftPro [11], which was designed to combine neuroscientific outcomes and mechatronic devices for assistive and rehabilitation robotics and advanced human-machine interaction. The data set is based on the U-Limb, a large, multi-modal, multi-center dataset on human upper-limb movements. Which consists of data from 91 non-disabled and 65 post-stroke participants, is organized at three levels: (i) upper limb daily living activities, during which kinematic and physiological signals (electro-myography, electro-encephalography, and electro-cardiography) were recorded; (ii) force-kinematic behavior during precise manipulation tasks with a haptic device; (iii) neural hand control using functional magnetic resonance imaging. In our work, we only focus on the data of upper limb daily living activities, and in their experiments, a set of 30 different daily living tasks were done from 33 subjects. In which 20 active optical markers fastened to the upper limb and the chest of each individual (see Figure 2-1). Each surface marker set was placed on ABS-printed support, whose position was estimated during the participant-specific calibration procedure (See Appendix A-1 for more details).

2-1 Pre-Processing Raw Data and Joint Angle Calculation in Cartesian 3D Space

In this section, we will discuss how the raw data has been transferred to joint angles, and the method we applied for correcting the calculation. With the figure shown in the Appendix (Figure A-1), the identity of the markers has been demonstrated. By applying them and the information from Figure A-3, we first transferred the reference frame from the chest to the center of the hand support plane. It is worth mentioning that from the table

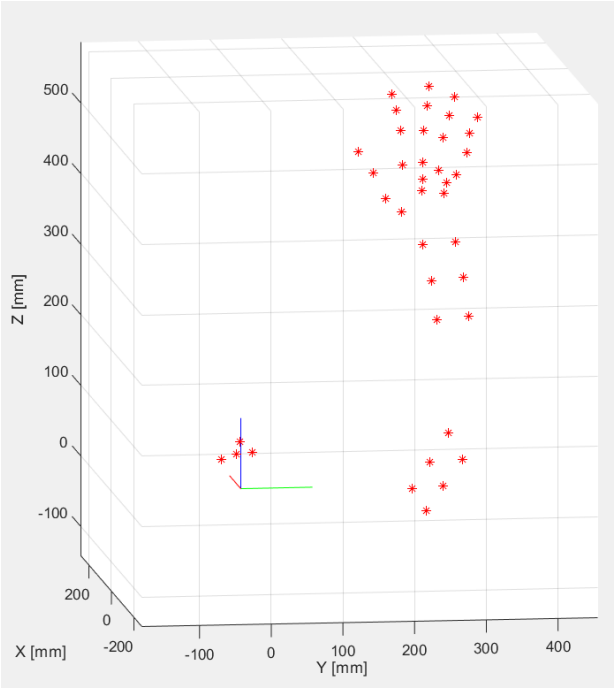


Figure 2-1: The illustration of upper limb electro-myography data from Harvard U-Limb original raw data [11]. There are a few points located around the world frame, which are actually allocated at the chest of the subject (see Figure A-2). The researchers attached plastic support on the chest to every subject for maintaining the same coordinate system.

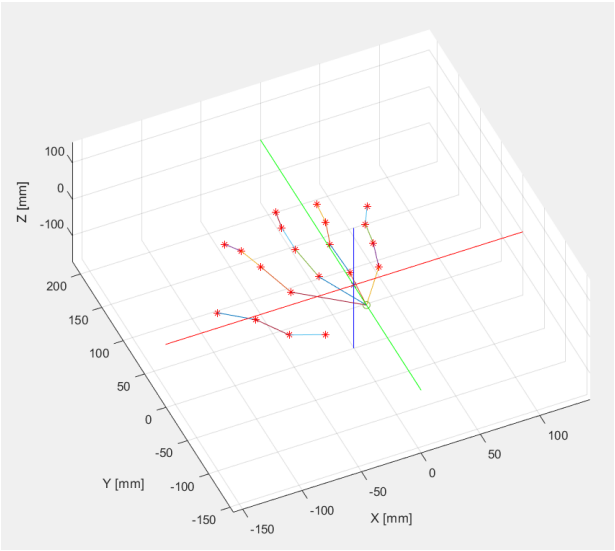


Figure 2-2: The modified hand raw data from the U-Limb. In this picture, the coordinate has moved to the center of the hand support plane, which will be our reference for joint angle computation. Aside from that, we portrayed the lines of each finger, and the results show the hand data does not follow kinematics.

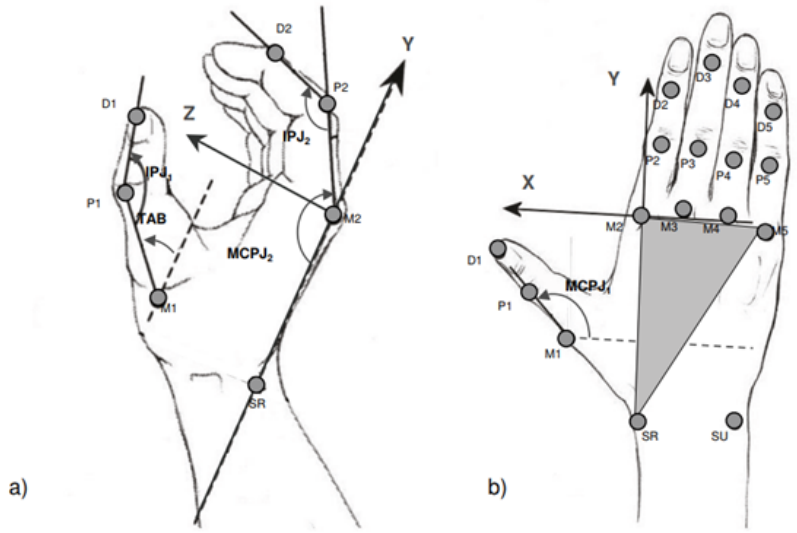


Figure 2-3: The illustration of sagittal plane (a) and metacarpal plane (b) of the hand [12].

in Figure A-1, we also obtained the bone length. With all the pre-processing preparation, the further computation can always stay at the world frame on the center of the support plane of the hand (see Figure 2-2). Directly computing angles in 3D space can cause errors with relative orientation and position of the coordinate system of the kinematic chain from 3D marker trajectories reconstructed in the motion sequence. As a result, we separated the 3D hand model into two 2D planes: the sagittal and the metacarpal planes (see Figure 2-3). This way, 3D joint angles can be simplified by calculating each finger joint on a 2D plane. However, estimating correct joint angles requires to revise in 3D space for each finger. The abduction/adduction angles can be calculated on the metacarpal plane, but they will get affected by extension/flexion angles if we directly apply the inner product of joint vectors. In the U-Limb data set [11], they attached plastic support on the subject's hand back, which is considered as a flat plane. Based on the assumption that the support platform is always flat and perfectly installed on the back of the subject's hand. We can see it as the metacarpal plane of the hand and compute its normal vector. Another vector that is perpendicular to the normal vector of metacarpal plane, the tangential vector, can also be found, and it is the Y-direction of the world frame. Figure 2-4 illustrates how the metacarpal plane was defined. To compute the abduction/adduction of each finger, we applied a hypothesis that the calculation of abduction angles only holds when the first phalanx of the hand is parallel to the line on the metacarpal plane. If the hypothesis is conceded, then we can calculate the marker-line (towards fingers) on the palm plane to find the angle between it and each 1st phalanx of fingers. In Figure 2-5, we illustrated how to create a reliable sagittal plane to evaluate the first phalanx's abduction/adduction angle in each finger. Firstly, the 1st joint of the index finger and the 1st joint of the thumb were needed for building a vector. Secondly, The vector will form the sagittal plane with the normal vector of the plastic support. Finally, the inner product of the 1st

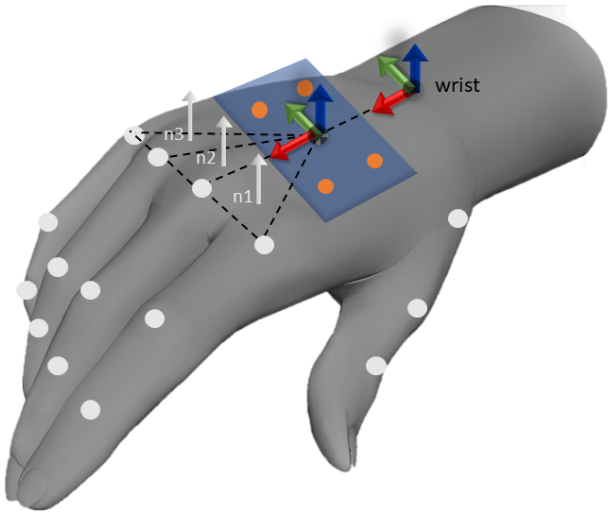


Figure 2-4: The world frame is located at the center of the plastic support, while the hand's wrist is set as 30 cm backward in Y-direction. The idea is to consider the 1st joint of the index and the middle with the center of the hand (hand O) as a plane and find its normal angle. Likewise, the 1st joint of the middle and ring finger and the ring with the little could form other planes.

phalanx of each finger and tangential vector of the plastic plane can then form the sagittal plane of each finger. This, of course, will involve the assumption that the "no abduction" hand pose is straight forward pointing directly to the Y-direction. It is not realistic in real situations, but the error can be tuned afterward. Luckily, the abduction/adduction angles only exist on the metacarpal phalanx. And in our case, the middle finger is seen as having no abduction/adduction. With the revise in the sagittal plane, the relative orientation on the metacarpal plane can be eliminated when computing each finger joint's abduction/adduction angle.

The method to compute the extension/flexion angle of each finger will not be affected by the relative orientation. We can see it as a cone in 3D space (see Figure 2-6). Therefore, to find the reliable extension/flexion angle of each finger joint, the vectors that form from the 1st phalanx and 2nd phalanx, and 2nd phalanx and 3rd phalanx need to perform the inner product. Above all, we have proposed a reliable method for joint angle calculation in 3D space, however, the data set is provided without kinematic information, so we cannot directly apply them to a hand tracking algorithm. The next section will introduce the forward and inverse kinematics of human hand to obtain the joints angles that follow hand kinematics.

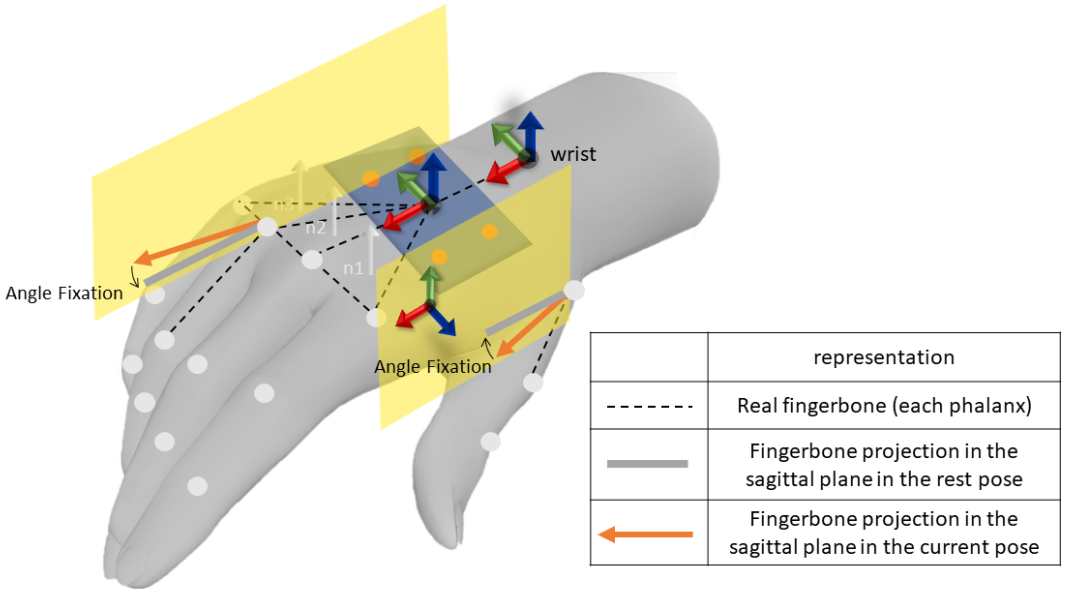


Figure 2-5: The gray line represents the finger bone in zero abduction angle and projected in Y-direction, while the orange arrow is the vector that project from current phalanx (finger pose). And by computing their difference, we can avoid the effect from the abduction angle. The 1st joint of the ring finger extension/flexion angle needs to calculate in the sagittal plane, and thus the abduction angle will not affect the result.

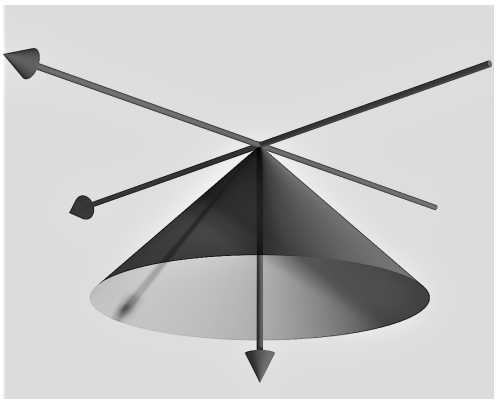


Figure 2-6: The illustration of extension/flexion angle in 3D space can be considered a cone and will not be affected by different orientations.

2-2 The Outliers Elimination

In this section, we focus on removing outliers from the recorded data-set and the recovering joint angles. These outliers may be caused by self-occlusion when the subject's hand was moving during the recording. And the kinematics of the hand was not considered during the experiment, as Figure 2-2 shows. Without kinematic constraints brings problems like

cannot calculating the correct joint angle and being hard to perform the hand movement in a real experiment. To ensure the joint angle we got from the previous section can be used in the hand model re-build, we start from the significant error elimination. Moreover, to avoid counting with impossibly operating angles, we considered works from [17] for listing the reasonable range of each finger's joint angle in table 2-1. Before the joints' angle is applied to the forward kinematics model, they will be filtered by the range limits to avoid invalid hand poses estimation. These filters, written in Matlab, have some simple thresholds for rejecting numbers that may cause serious errors when operating hand recovery.

Table 2-1: Finger joints normal range of motion [17].

Finger joint	Type of movement	Theoretical range
Thumb CMC	Hyperextension/Flexion	-10/55 deg
Thumb MCP	Hyperextension/Flexion	-10/55 deg
Thumb IP	Hyperextension/Flexion	-15/80 deg
Index MCP	Extension/Flexion	-45/90 deg
Index PIP	Extension/Flexion	0/100 deg
Index DIP	Extension/Flexion	0/80 deg
Middle MCP	Extension/Flexion	-45/90 deg
Middle PIP	Extension/Flexion	0/100 deg
Middle DIP	Extension/Flexion	0/80 deg
Ring MCP	Extension/Flexion	-45/90 deg
Ring PIP	Extension/Flexion	0/100 deg
Ring DIP	Extension/Flexion	0/80 deg
Little MCP	Extension/Flexion	-45/90 deg
Little PIP	Extension/Flexion	0/100 deg
Little DIP	Extension/Flexion	0/80 deg

To obtain the required geometric information, we followed the idea from [16] that performing kinematic chains, with the Denavit-Hartenburg (DH) parameters for the 3D hand model (see Appendix Figure A-5). The DH description of each link is augmented with an additional transform from the link frame to a shaped frame, which describes the position of the visible link geometry in space [47]. In the next section, we will explain the idea of hand kinematics and demonstrate the rebuild results.

2-3 The Application of Hand Kinematics

When re-building a hand structure, we applied both the forward and inverse kinematics of the hand, which has been defined in previous work from [16], [48], and [49] for reconstruct-

ing the posture of the hand in space and completing a 3D kinematic model (see Appendix A-2 for more details). The configuration of a rigid body is represented as an element T $\in$ SE(3). An element T $\in$ SE(3) may also be viewed as a mapping T : $\mathbb{R}^3 \to \mathbb{R}^3$ which preserves distances and angles between points. According to the D-H method, the kinematics in homogeneous coordinates of the hand then can be described by matrices with adjacent bars sequentially established:

$$\begin{aligned} T_{0,4} &= T_{0,1}T_{1,2}T_{2,3}T_{3,4} \\ &= \begin{bmatrix} R_{0,4} & P_{0,4} \\ 0_{1\times 3} & 1 \end{bmatrix} \\ &= \begin{bmatrix} c_1c_{234} & -c_1s_{234} & s_1 & c_1(l_2c_2 + l_3c_{23} + l_4c_{234}) \\ s_1c_{234} & -s_1s_{234} & -c_1 & s_1(l_2c_2 + l_3c_{23} + l_4c_{234}) \\ s_{234} & c_{234} & 0 & l_2s_2 + l_3s_{23} + l_4s_{234} \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned}$$

Due to the layout limit, here we don't specify each T matrix that forming the fingertip location $T_{0,4}$ (details can be found in the Appendix A). In the formulation, R $\in \mathbb{R}^{3\times 3}$ represents the rotation matrix, l_i represents the length of the phalanx, and $p \in P$ is the translation vector. Also, among them:

$$\begin{aligned} c_1 &= \cos\theta_1 & s_1 &= \sin\theta_1 \\ c_{23} &= \cos(\theta_2 + \theta_3) & s_{23} &= \sin(\theta_2 + \theta_3) \\ c_{234} &= \cos(\theta_2 + \theta_3 + \theta_4) & s_{234} &= \sin(\theta_2 + \theta_3 + \theta_4) \end{aligned}$$

The inverse kinematics of the finger can then be derived from the following operation:

$$\begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} c_1c_{234} & -c_1s_{234} & s_1 & c_1(l_2c_2 + l_3c_{23} + l_4c_{234}) \\ s_1c_{234} & -s_1s_{234} & -c_1 & s_1(l_2c_2 + l_3c_{23} + l_4c_{234}) \\ s_{234} & c_{234} & 0 & l_2s_2 + l_3s_{23} + l_4s_{234} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

From the above matrix, (p_x, p_y, p_z) represents the current position of a fingertip, and the $(n_{x,y,z}, o_{x,y,z}, a_{x,y,z})$ is the Euler–Rodrigues formula in 3D rotation. By solving geometric formulation in each backward state of finger joints, we can transfer every joint location to angles. It is worth mentioning that the geometric calculation of long fingers and the thumb is different. In the following algorithm pseudo-code, we briefly list the methods of both.

With kinematic adjusting and hand-tuning some errors, the new joint data will then form. The new hand joints data, however, cannot be applied directly to the hand tracking process. There are a couple of factors that would affect the accuracy of joint angles estimation. The raw data cause one; since the recording procedure did not consider hand kinematics, it frequently shows irrational movements. Another circumstance is due to the kinematics formulation of the hand, the idea of DH convention is transforming coordinates along a

Algorithm 1 Solving Long Finger Kinematics

procedure GEOMETRIC PROCESS($p_{x,y,z}, n_{x,y,z}, o_{x,y,z}, a_{x,y,z}, l_{2,3,4}$)
▷ l_2, l_3, l_4 is the corresponding phalanx length
    $\theta_1 \leftarrow \arctan(p_y, p_x)$
    if $\theta_1 \geq 0$ **then** ▷ if θ_1 is smaller than 0 we set it as π
        $\theta_3 \leftarrow p_{x,y,z}, n_z, o_z, \theta_1, l_{2,3,4}$
    end if
    $\theta_2 \leftarrow p_{x,y,z}, n_z, o_z, \theta_1, \theta_3, l_{2,3,4}$
    if $\theta_2 \neq empty$ **then** ▷ if θ_2 is empty we set it as 0
        $\tan\theta_4 \leftarrow \theta_2 + \theta_3 + \theta_4$
    end if
    if $\theta_4 \neq empty$ **then** ▷ if θ_4 is empty we set it as 0
        $\theta_1 = joint_1$,
        $\theta_2 = joint_2$,
        $\theta_3 = joint_3$,
        $\theta_4 = joint_4$
    end if
end procedure

Algorithm 2 Solving Thumb Kinematics

procedure GEOMETRIC PROCESS($p_{x,y,z}, n_{x,y,z}, o_{x,y,z}, a_{x,y,z}, l_{2,3,4}$)
▷ l_2, l_3, l_4 is the corresponding phalanx length
    $\theta_2 \leftarrow \arctan(-a_z)$
    $\theta_1 \leftarrow \arccos(-a_x/\sin\theta_2)$
    for m, n **do**
        $m = p_x\cos\theta_1\cos\theta_2 - p_z\sin\theta_2 - l_2 + p_y\cos\theta_2\sin\theta_1$;
        $n = p_x\sin\theta_1 - p_y\cos\theta_1$;
        $\theta_4 \leftarrow \arccos((m^2 + n^2 - l_3^2 - l_4^2)/(2 * l_3 * l_4))$;
        $\theta_3 \leftarrow \arcsin(-l_4 * (-n_y\cos\theta_1)) + n_x\sin\theta_1) + p_y\cos\theta_1 - (p_x\sin\theta_1)/l_3$;
    end for
    $\theta_1 = thumb_{joint_1}$,
    $\theta_2 = thumb_{joint_2}$,
    $\theta_3 = thumb_{joint_3}$,
    $\theta_4 = thumb_{joint_4}$
end procedure

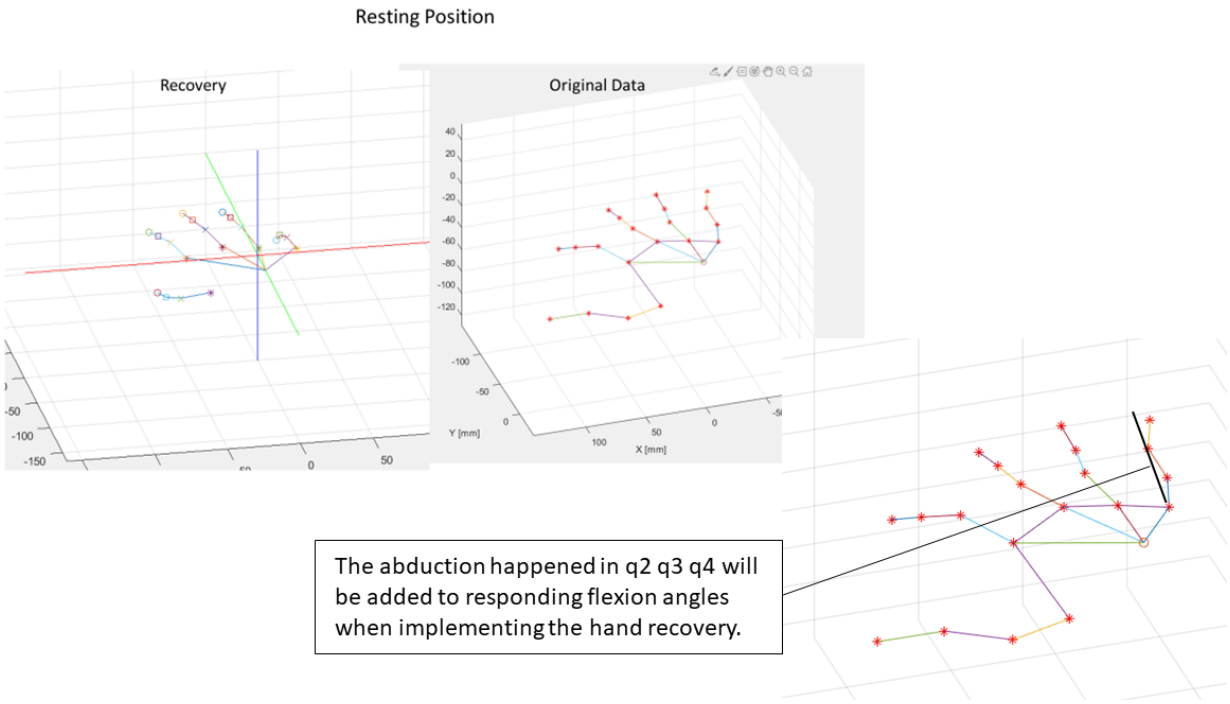


Figure 2-7: The comparison of original data and recovery hand model in resting position. While q2, q3, q4 represent metacarpal, proximal, and distal joints respectfully.

serial robot consisting of each link from the kinematics equations of the linkage, and this will result in accumulated errors in each coordinate. The last joint on each finger will reveal the largest estimation error. Aside from that, the kinematic model constrained all fingers to have abduction/adduction only in their first joint, which means if the recorded data has a situation like the following picture (Figure 2-7), then the abduction/adduction angles in PIP and DIP will be added to the flexion part of them. In Figure 2-7, the grasping recovery results are illustrated. It shows that even we have conducted the method for calculation joint angle in 3D space and applied the reasonable range of each finger joint is not enough for a hand model re-build.

The problems mentioned above and some irrational poses happen even seriously when the record is in a moving condition (see Figure 2-8). It is clear to see occlusions affected the data recording, and jerking occurred frequently. These situations require filters that can at the same time predict the correct hand state to revise. The extended Kalman filter is then introduced to the process. In the next section, we will describe the implementation with the extended Kalman filter in detail.

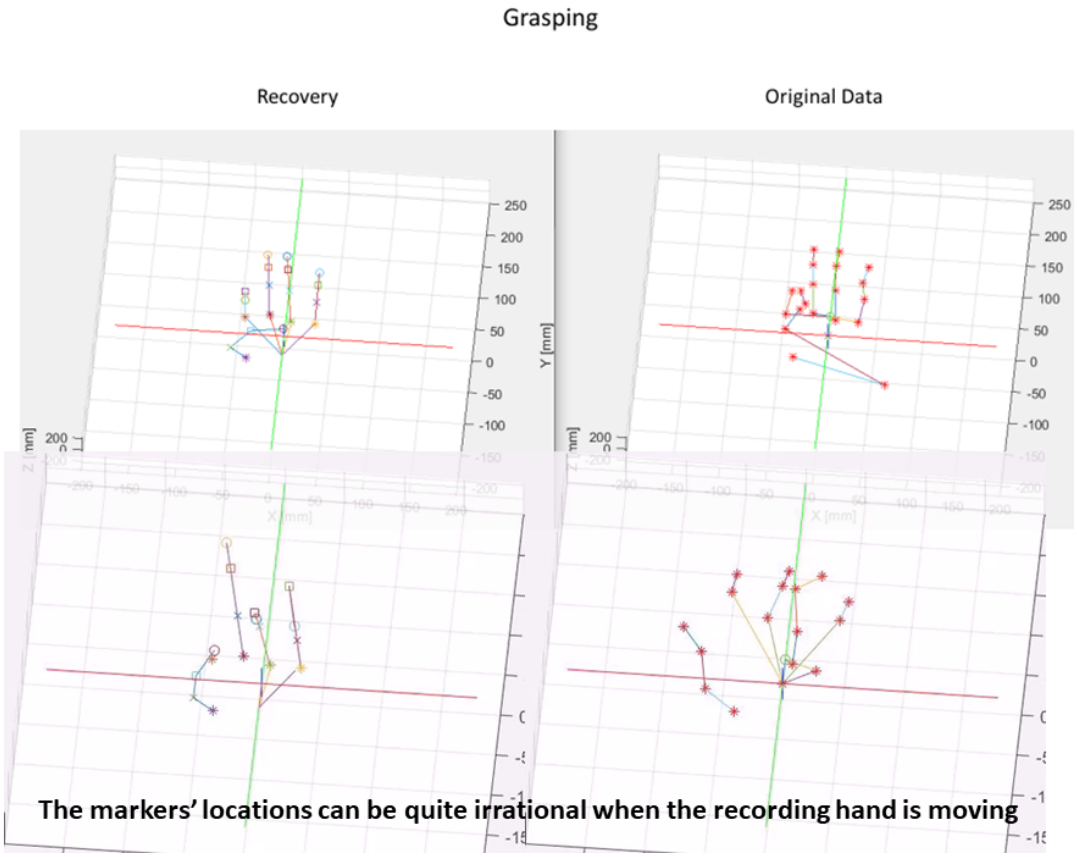


Figure 2-8: The comparison of original data and recovery hand model when the hand is grasping. It is obvious to see jerking and occlusion as the recording hand moving.

Chapter 3

Tracking with Extended Kalman Filter

The motivation of this chapter is to apply the EKF algorithm for improving the camera-based hand tracking results. We decided to use the extended Kalman Filter, though it is common to see hand tracking with hidden Markov models, deep learning-based algorithms, and the Particle Swarm Optimization (PSO). The EKF can provide fast computation, also, the data type we have is not suitable with a CV-based algorithm. Moreover, the sensors we may apply with (motion capture device) are relying on the kinematic constraints which the EKF can easily obtain. With the estimation, the tracking result will be more precise and implemented on the robotic hand control. However, the EKF also has its limits. At the end of this chapter, we will go through the difficulties in hand tracking with the extended Kalman filter.

3-1 Mathematical Background

Kalman filter is a filtering method that uses the dynamic model of the system, the known input of the system, and multiple continuous measurements (such as data from sensors) to form an estimate of the amount of change in the system (its state), which is better than an estimate obtained using only one measurement alone. Therefore, it is a common sensor fusion and data fusion algorithm [50]. The extended Kalman filter is an extended version of the Kalman filter as it is named. The main difference is the state transition and observation models can be nonlinear functions in extended Kalman filter.

State transition model:

$$x_k = f(x_{k-1}, u_k) + w_k \tag{3-1}$$

Observation model:

$$z_k = h(x_k) + v_k \tag{3-2}$$

Here w_k and v_k are the process and observation noises which are both assumed to be zero mean multivariate Gaussian noises with covariance Q_k and R_k respectively. Apart from them, u_k is the control vector, which can be neglected in our case.

The function f is used to compute the predicted state from the previous estimate. In our case, the joint angles of the hand are the state, and we divided the hand into 5 fingers and got 5 different functions. Thus, each function f can be written as the forward kinematics matrix of each finger, which contains MCP, PIP, and DIP angles forming joint positions in Cartesian coordinate. Nevertheless, the 3D marker positions are represented as the observation function h. It can be used to compute the predicted measurement from the predicted state. To obtain the covariance of function f and h, the Jacobian matrix (partial derivatives) is needed. At each time step, the Jacobian is evaluated with current predicted states. These matrices represent linearization of the non-linear function around the current estimate. Meanwhile, EKF also minimizes the posterior estimation error covariance. The following structure shows how EKF works.

`Predict:`

$$x_{k|k-1} = f(x_{k-1|k-1}, u_k)$$
$$P_{k|k-1} = F_k P_{k-1|k-1} F_k^T + Q_k$$

`Update:`

$$y_k = z_k - h(x_{k|k-1})$$
$$S_k = H_k P_{k|k-1} H_k^T + R_k$$
$$K_k = P_{k|k-1} H_k^T S_k^{-1}$$
$$x_{k|k} = x_{k|k-1} + K_k y_k$$
$$P_{k|k} = (I - K_k H_k) P_{k|k-1}$$

where P is the estimation error covariance, K_k is the Kalman gain, F and H are Jacobians of function f and h respectfully. They are defined as following:

$$F_k = \left.\frac{\partial f}{\partial x}\right|_{x_{k-1,k-1}, u_k} \qquad H_k = \left.\frac{\partial h}{\partial x}\right|_{x_{k,k-1}}$$

The Matlab code and explanation of the EKF application in our hand tracking are available assess from GitHub. Here we only discuss the results and the validation of the algorithm.

3-2 Validation and Tracking Results

In order to evaluate the algorithm, we first applied the filter onto a sure correct hand finger movement. The finger was designed from the synthesis of a particular type of planar

six-bar mechanism, which is called the Watt I mechanism [51]. In work from [51], they demonstrate the synthesis of a finger mechanism and achieve mimicking a human finger grasping pose. Figure 3-1 illustrates the finger mechanism with its movement provided by the software Sam61 [52]. By collecting each frame of the finger movement, we can form a new set of data that represents reliable finger joints positions in a 2D plane. Thus, we can apply the EKF on the tracking of the new data to validate the design of algorithm.

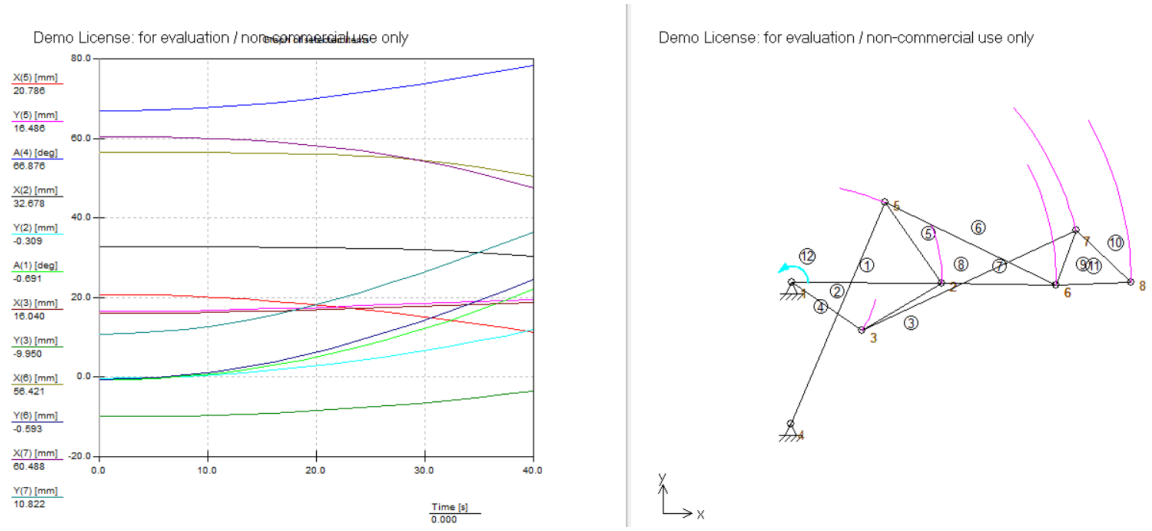


Figure 3-1: The illustration of how Watt I mechanism synthesis a finger and its movement.

Meanwhile, since v_k is a zero-mean multivariate Gaussian noise in measurement, we also need to track the result by tuning the standard deviation parameter σ_R in measurement noise to obtain the best parameter. In the definition of extended Kalman filter, Q is the covariance of w (w is the process noise), and R is the covariance of v (where v is the measurement noise). In which, R is determined by σ_R, the standard deviation of observation noises. And normally it is hand-tuned according to the estimated speed of hand movement during tracking experiment. However, considering the measurement error cannot be determined from the raw data set, we need to perform another test to find a proper σ_R for raw data tracking. Thus, 10 times repeating for different values of σ_R were done for verifying the convergence and divergence. The sample test with three different values of σ_R and results are shown in Figure 3-2 and 3-3. It is worth noticing that the first MCP angle (the abduction/adduction angle) does not exist in a 2D plane, so the first column of the testing result can be neglected.

As the results show in table 3-1, we decided to set the value of σ_R as 0.1 mm. We believe that the error in estimation results when $\sigma_R = 0.1$ mm are closer to the real situation. After the validation of the EKF algorithm, we applied the filter to five fingers. Following the same idea, the raw data need to be parsed with both forward and inverse kinematic models for getting joint angles. Moreover, the joint angles of each finger will then be saved as different files, and we applied the EKF tracking with these joints angles data. It's worth noting that, the results only focus on a subject (S7) from the experiment. In the following

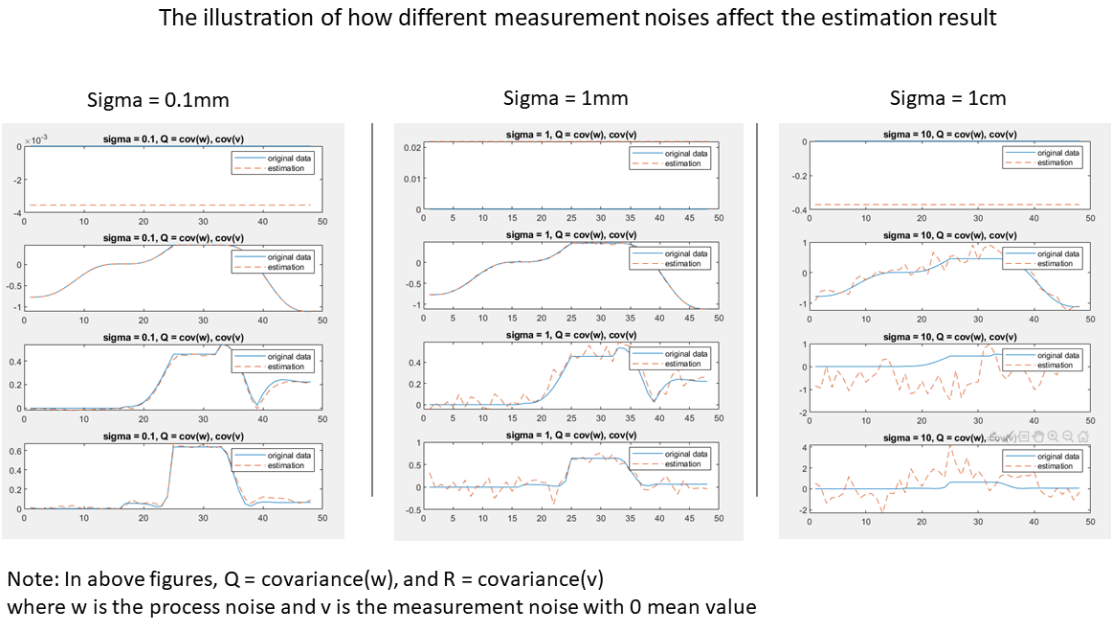


Figure 3-2: The tracking results with different standard deviation σ_R in measurement noise.

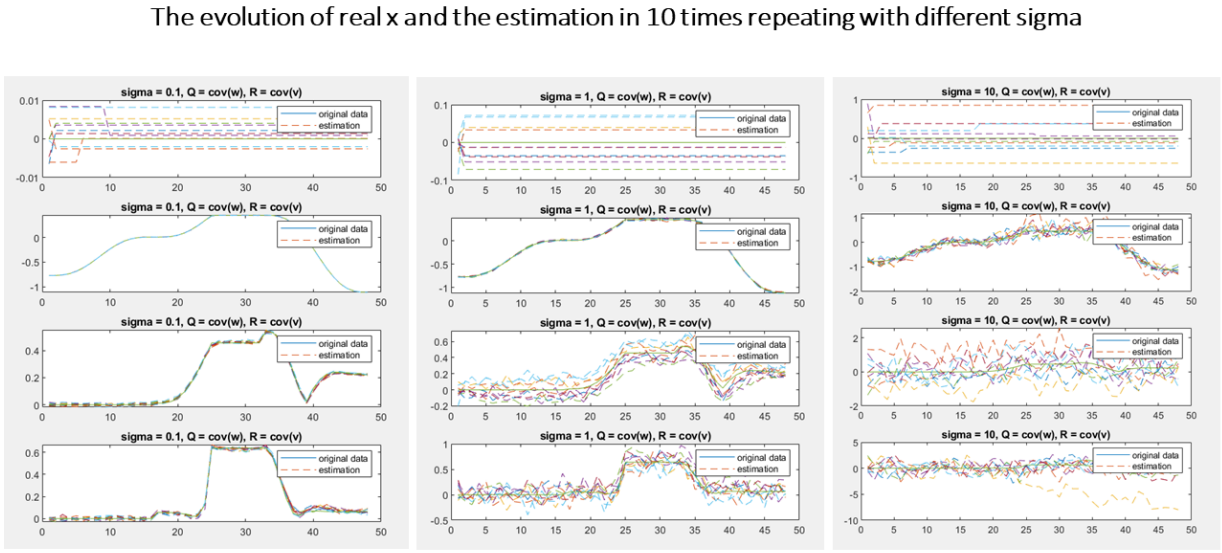


Figure 3-3: The 10 times results with σ_R equals to 0.1, 1, 10 mm in measurement noise. As the figure has shown, the value of σ_R directly affects the robustness of the estimation.

Table 3-1: Estimation Error in different σ_R during 10 times repeating test.

Turn	σ_R =0.01mm	σ_R =0.1mm	σ_R =1mm	σ_R =10mm
1	0.629128 mm	1.217492 mm	8.806767 mm	72.82048 mm
2	0.691331 mm	0.955043 mm	11.13055 mm	85.06250 mm
3	0.803799 mm	1.172389 mm	8.883702 mm	71.55041 mm
4	0.807322 mm	0.946418 mm	8.094886 mm	326.33463 mm
5	0.631474 mm	1.151795 mm	5.420166 mm	88.49551 mm
6	0.624739 mm	1.093376 mm	7.973390 mm	76.13414 mm
7	0.603146 mm	1.052899 mm	7.341866 mm	92.27715 mm
8	0.728434 mm	1.110531 mm	13.38066 mm	65.16027 mm
9	0.665615 mm	1.176667 mm	6.515464 mm	76.10397 mm
10	0.626242 mm	1.244694 mm	9.105936 mm	85.09066 mm
average	0.681123 mm	1.112130 mm	8.665339 mm	103.90297 mm

figures, various hand grasping results are shown in both actual data and re-building models. In the last part of this section, we will also analyze the error between static estimation and recovery joint angle.

3-3 Recovery and Estimation Error

From Figures 3-4 and 3-5, the recovery of the hand model and the tracking result of the EKFare shown in different colors. Due to the space limitations, we only demonstrated four different hand poses tracking results. In most cases, the error commonly occurred in abduction/adduction joints. The estimation error in every second can be pretty varied as capturing a moving hand's absolute position in 3D space is challenging. An illustration(see Figure 3-6 and Figure 3-7) is shown in the following tracking results that the "moving" period of the hand can cause major errors. Actually, there are only a few seconds of doing the particular hand poses (which can be seen as a few seconds static) that we want to estimate during the whole tracking period. In other words, when considering the errors in the whole hand record period, the significant error happened as the hand changed its moving orientation. We analyzed the total errors in the entire tracking period of each joint of each finger and calculated the average deviation of them (see table 3-2 and 3-3). However, these errors actually look inconspicuous when we record the whole tracking result into videos.

In Figures 3-8 and 3-9, we zoom in on the instant tracking error at every frame when the hand was performing specific movements. For example, in Figure 3-8, the hand is grabbing a suitcase while in Figure 3-9 the hand is doing the gesture OK. These results have shown

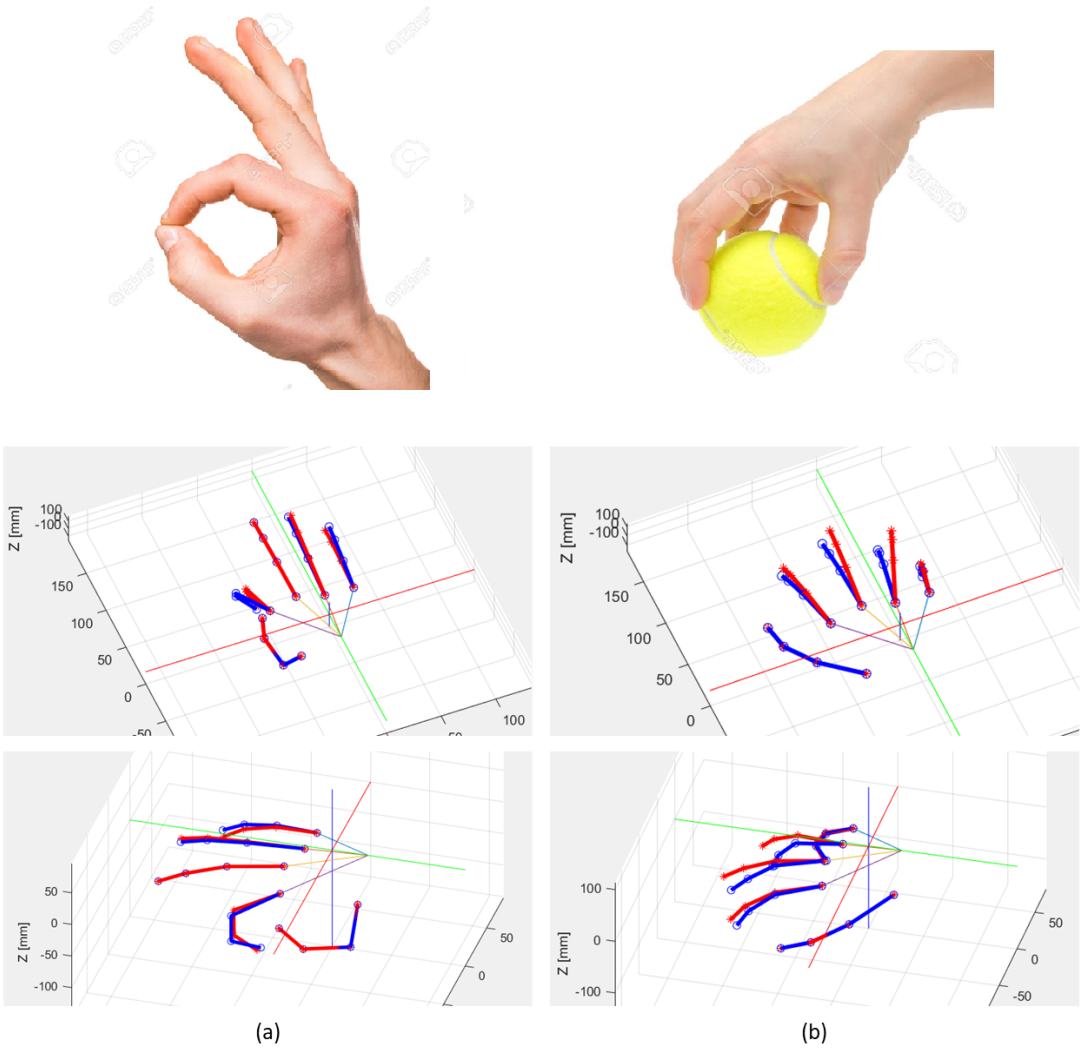


Figure 3-4: The column of (a) shows both the front and side view of a hand pose an OK posture. And on top of them is the natural human hand doing "OK" as a comparison with the estimation. The blue line represents recovered data, while the red line illustrates the estimation from the EKF. Column (b) shows a human hand grabbing a tennis ball. As the layout of (a), the blue line is the recovery data and the red line is the hand tracking result.

that the EKF hand tracking is promising and quick enough for performing real-time hand tracking. The error of the EKF results has been accumulated from the beginning of the hand recovery process. However, the error of the EKF results has been accumulated from the beginning of the hand recovery process. The first factor has been made to remedy missing or mislocated data points in the actual data from markers positions. The kinematics rebuilding progress causes another deviation; to form a hand pose close to reality, we turned some parts of the joints with apparent distortions. With all these adjustments,

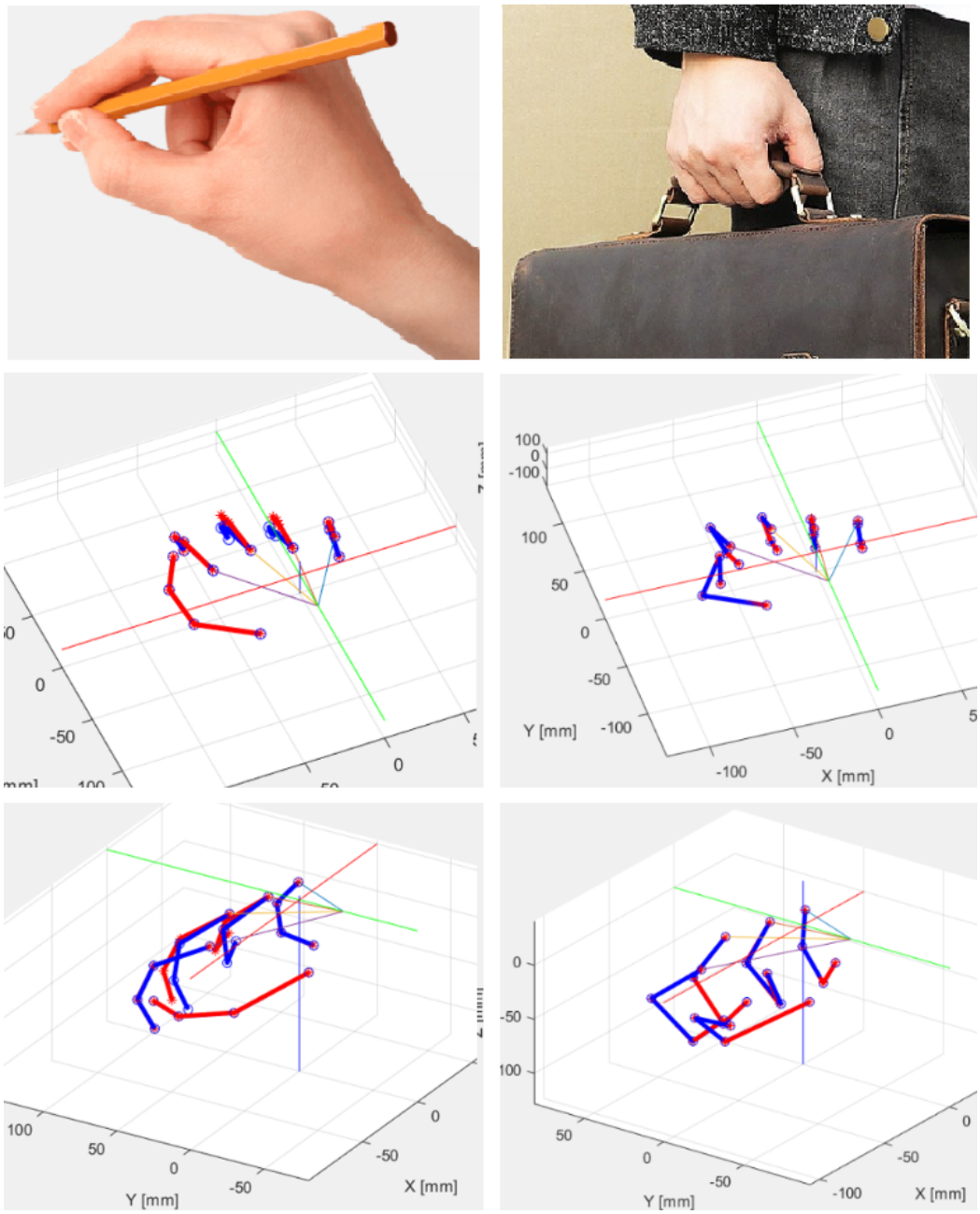


Figure 3-5: In the left column, it shows a human hand grabbing a pen and doing the writing. The recovered result is covered in blue lines, while the red lines show the EKF tracking results. Pictures in the right column have the same overall arrangement and it indicates the hand structure when carrying a suitcase.

the new forming data then be applied as the "real hand" for the EKF tracking. Although the data we applied has been slightly tuned, the tracking results show that it still has high reliability with tracking natural hand movements.

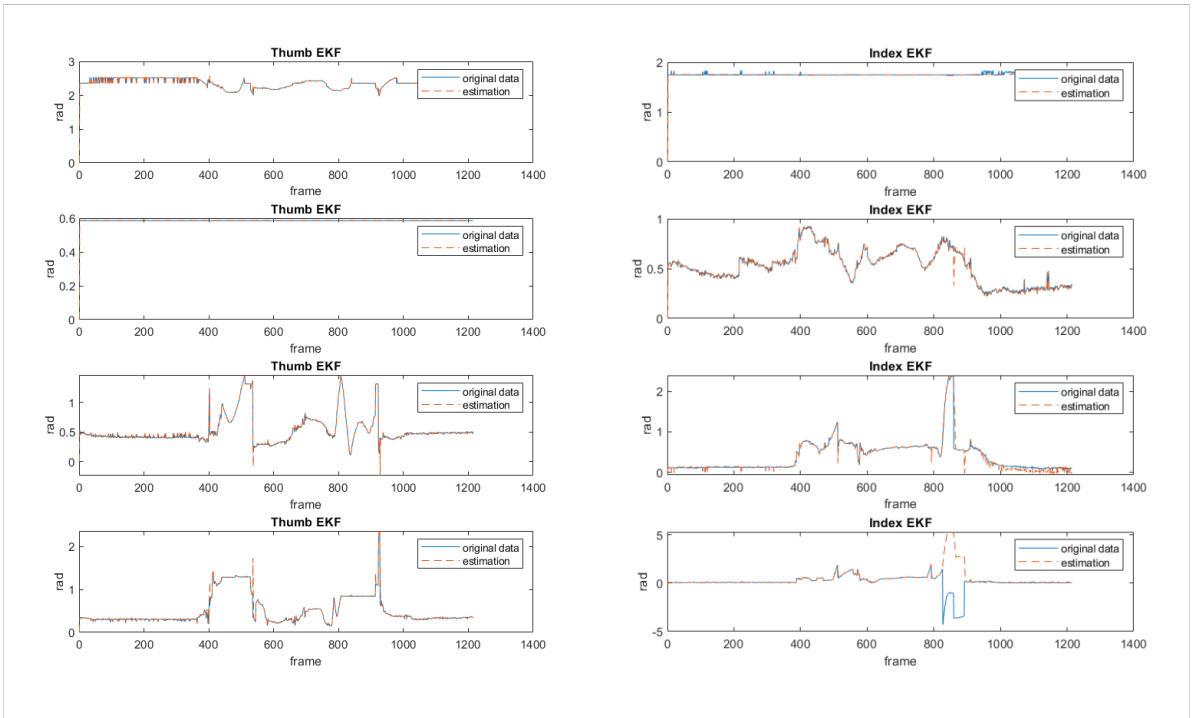


Figure 3-6: In the Thumb EKF tracking result, the sequence is organized as MCP abduction/adduction (q1), MCP flexion/extension (q2), PIP (q3), and DIP (q4) joints. And for the other pictures, they also followed the order.

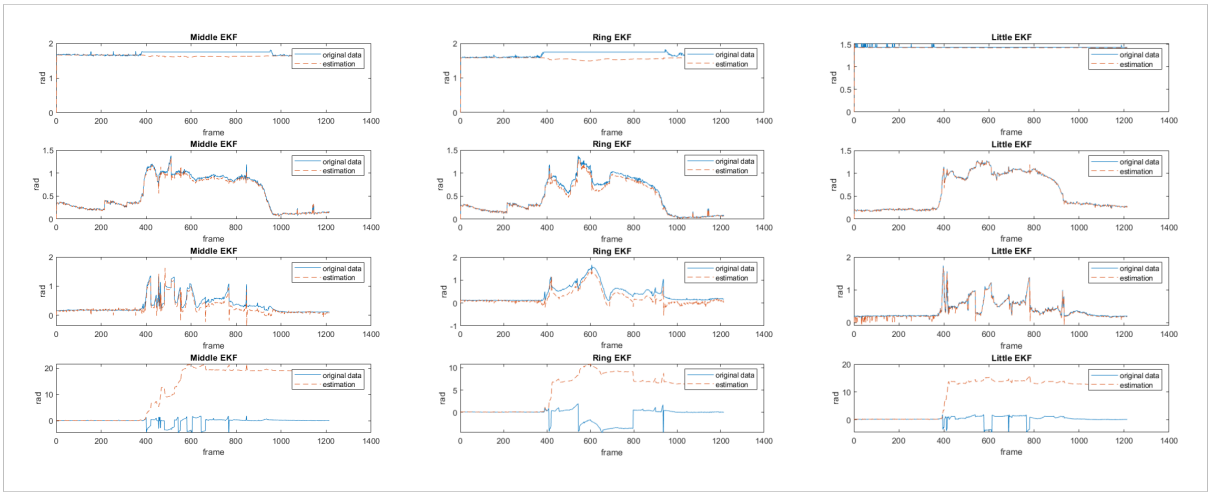


Figure 3-7: This Figure shows the tracking results of the middle, ring, and little fingers. One thing that is obvious to see from all three fingers is that the tracking results of the last joints (DIP) show significant errors.

Table 3-2: Estimation Error in tracking a hand is writing.

joint	Thumb	Index	Middle	Ring	Little	Unit
q_{1total}	5.014	17.687	N	147.961	6.677	deg
q_{2total}	2.550	5.632	26.357	46.222	3.714	deg
q_{3total}	8.235	36.518	104.100	196.624	19.265	deg
q_{4total}	14.152	439.945	14390.795	6760.705	10540.931	deg
q_{1avg}	0.0041	0.0146	N	0.1219	0.0055	deg/per frame
q_{2avg}	0.0021	0.0046	0.0217	0.0381	0.0031	deg/per frame
q_{3avg}	0.0068	0.0301	0.0857	0.1620	0.0159	deg/per frame
q_{4avg}	0.0117	0.3624	11.8540	5.5689	8.6828	deg/per frame

Table 3-3: Estimation Error in tracking a hand doing OK pose.

joint	Thumb	Index	Middle	Ring	Little	Unit
q_{1total}	6.252	128.290	N	25.655	27.343	deg
q_{2total}	1.774	63.970	1.679	5.720	9.918	deg
q_{3total}	6.558	195.052	8.397	50.825	62.023	deg
q_{4total}	10.987	306.414	1893.966	2725.141	4388.208	deg
q_{1avg}	0.0064	0.1306	N	0.0261	0.0278	deg/per frame
q_{2avg}	0.0018	0.0651	0.0017	0.0058	0.0101	deg/per frame
q_{3avg}	0.0067	0.1986	0.0086	0.0518	0.0632	deg/per frame
q_{4avg}	0.0112	0.3120	1.9287	2.7751	4.4686	deg/per frame

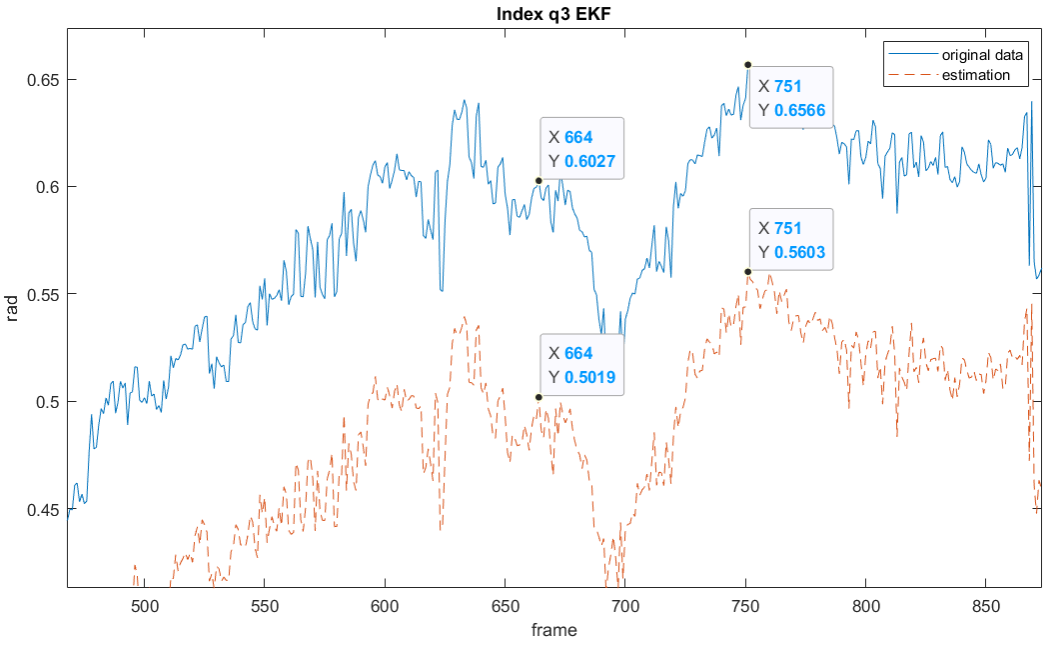


Figure 3-8: This figure indicates the natural hand joint configuration when it was carrying a suitcase. And the dotted line is the estimation from the EKF, in which we can see the error between them is only about 0.1 degrees.

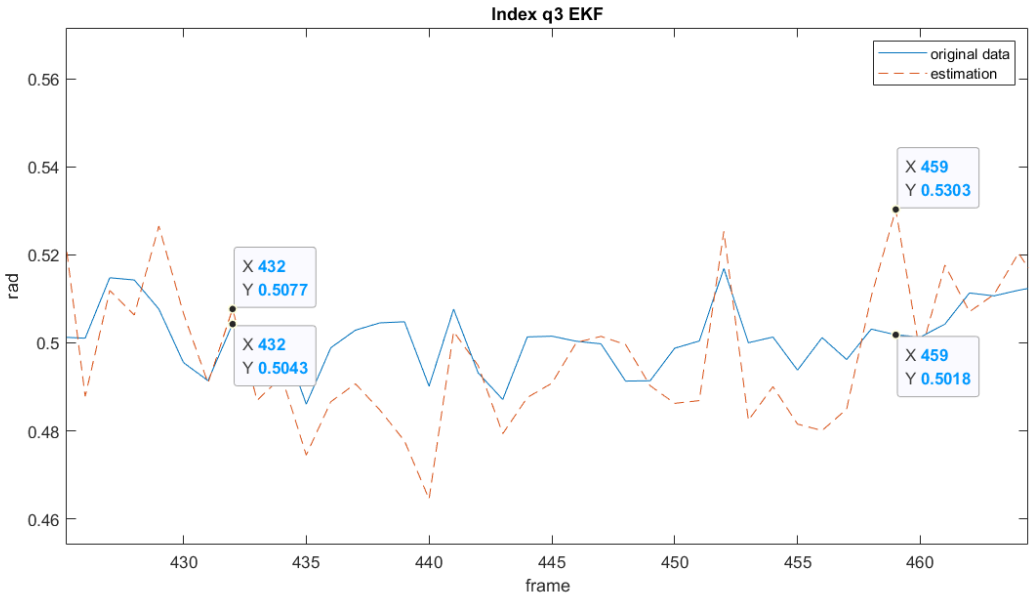


Figure 3-9: Here we can see even more minor errors in the EKF estimation. It is the period that a human hand is doing the gesture of OK. It is worth mentioning that the range of this frame has avoided the part that the subject lifts up his hand.

Chapter 4

Postural Synergy Analysis

In the late 19th century, neuroscientists indicated that the high complexity of the human hand sensorimotor system could be divided into layered sets of motion primitives or "synergies." This discovery shows that most human hand movements can be represented by combining a minimal number of linearly independent elements. In this Chapter, we look for the presence of a synergistic behavior in the hand motions from the discussed dataset, and we compare these results from the ones from literature.

4-1 Concept of Principle Component Analysis

Postural synergy provides dimensionality reduction for the human hand. It can reproduce a similar coordinated and orderly whole of human hand motions. According to the abundance, synergies may be defined as co-varying changes in the output of individual elements of a multi-element system that stabilize a value or a time profile of a vital performance variable produced by the system [53]. Santello et al. [21] recorded 285 hand poses that involve grasping operations. Furthermore, they measured static hand postures by recording the joint angles of all fingers. The result indicates that though the subjects changed different hand poses while grabbing other objects, the joint angles did not vary independently. In our data set, we have only 30 daily activities employed by principal components analysis. Moreover, as the original data-set was focusing on human upper limb movements, partial of them are not grasp poses. Thus, when conducting PCA, they became the outliers that seriously affected the important variables' performance. By eliminating them one by one, we concluded that the chosen 21 poses will lead to the best encoding result. The details will be illustrated in the following sections.

4-2 Result Validation

As already mentioned above, postural synergies represent a set of linear dependencies between the joint variables of the hand kinematics. Therefore, using a much lower-dimensionality approximation of the DoF configuration can clearly define the characterization of the recorded hand movements. In our case, the first two principal components account for more than 75% of the variance (see Figure 4-1). As the bar chart shows, if

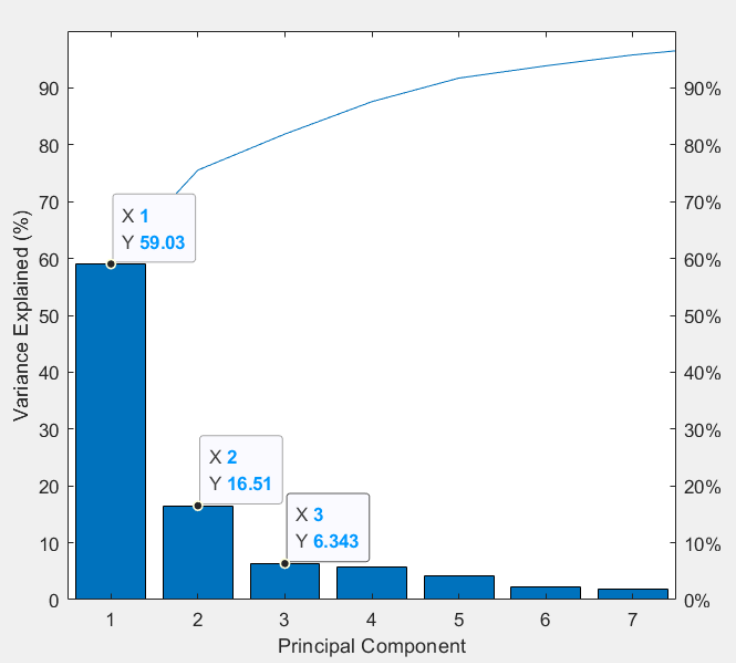
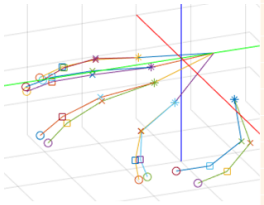


Figure 4-1: In our analysis, the percentage of the first two principal components has reached over 75 percent. The rest of the PCs have small values, and thus we only list the first seven of all.

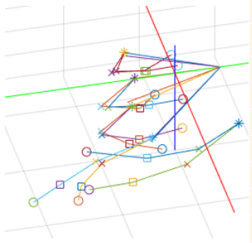
we only considered the first two PCs, there would be about 25% distortion in all synergy-based hand poses. In the dimensionality reduction system, distortion is sure to happen, however, we must verify that the distorted part will not affect the main gripping action. To do so, we re-plot some original poses which has been recovery from hand kinematics, and compared them with the synergy-based results. The quality of the estimated angular values at each time frame was evaluated through the Mean Squared Error (MSE).

$$E_{MSE} = \sqrt{\frac{1}{N_{DoF}} \sum_{i=1}^{N_{DoF}} (q_{real} - q_{pca})^2} \tag{4-1}$$

Where N_{Dof} is the number of the Degree of Freedom, q_{real} represents each joint that estimated from markers position via Inverse Kinematics and recovered through Forward



in rad	real	PCA
q1	-3.2474	-3.13413
q2	1.242478	0.95885
q3	0.872665	0.834534
q4	0.62644	0.478701
q5	1.896505	1.884579
q6	0.537099	0.531151
q7	0.995327	0.866083
q8	0.301085	0.18341
q9	0.334719	0.32214
q10	1.342227	1.354501
q11	0.150387	0.026789
q12	0.067906	0.163317
q13	1.20189	1.229006
q14	0.126183	0.063158
q15	0.276673	0.42475

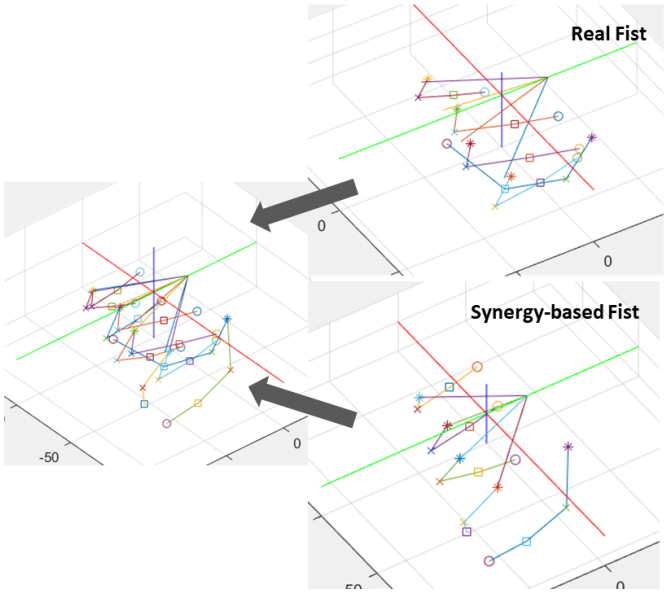


in rad	real	PCA
q1	-4.32527	-4.24011
q2	0.36075	0.427054
q3	0.444195	0.359245
q4	0.362206	0.18503
q5	1.701863	1.710886
q6	0.490676	0.665262
q7	0.403245	0.52766
q8	1.137866	1.144396
q9	1.147517	0.921069
q10	1.668411	1.740251
q11	1.092509	1.188888
q12	1.33557	1.345958
q13	1.782348	1.793858
q14	1.305448	1.405465
q15	1.178502	0.99118

Figure 4-2: We randomly chose two poses and compared their joint recovery angles from raw data with synergy-based joint angles. The 4th joint of each long finger has a slight difference because we only applied 15 joints PCA result, but the reconstruction of the whole hand has 20 joints. To mimic the pose, we set them roughly as 0.5*pi. Moreover, some researches suggest assuming that the distal interphalangeal rotation angle was 2/3 of the proximal one.

Kinematics. And q_{pca} is the synergy joint angle rebuilds via PCA. In our experiments we had E_{MSE} typically around 0.05 rad. However, with one exception, the fist pose, shown in Figure 4-3, the MSE has reached about 1.5 rad. From this observation, it can be expected that the result we got may be hard to compromise when the desired pose involved a lot of thumb movements.

In Figure 4-4, we compare PCA results from Santello et al. and our data-set. In both PC1 (horizontal) axis, the fingers are extended at the MCP joint and abducted (PC1 min). And we can see that our result shows more flatten than Santello et al.. It can be explained by the considered data-set having no grasping pose that demonstrates a hand grabs a large object (such as an ashtray); instead, we have poses like showing a "stop" sign by hand and which cause it tends to extend more. At the other extreme, they both are flexed at the MCP joint and adducted (PC1 max). The excursion at the PIP joints remains approximately constant. Along the PC2 (vertical) axis, we can see the differences with the changes in the angular excursion. In work from Santello et al., the PIP joints flex, whereas the MCP joints extend with a smaller amplitude moving toward PC2 max. However, the index and the thumb in our PC2 show opposite directions of significant movement when moving from one extreme to the other.



in rad	real	PCA
q1	-4.231	-3.8487
q2	0.45617	0.589474
q3	0.76539	0.587909
q4	1.12574	0.183255
q5	1.79398	1.77997
q6	1.28539	0.837175
q7	1.42188	0.784882
q8	1.51961	1.260808
q9	1.25977	1.113346
q10	1.35507	1.749669
q11	1.53667	1.279701
q12	1.37772	1.621581
q13	1.50535	1.772082
q14	1.39626	1.519526
q15	0.99042	1.355789

Figure 4-3: The comparison of real fist and synergy-based fist shows that the main effect from the distortion happens mainly on the thumb. In this figure, we can see the major difference is on the PIP joint.

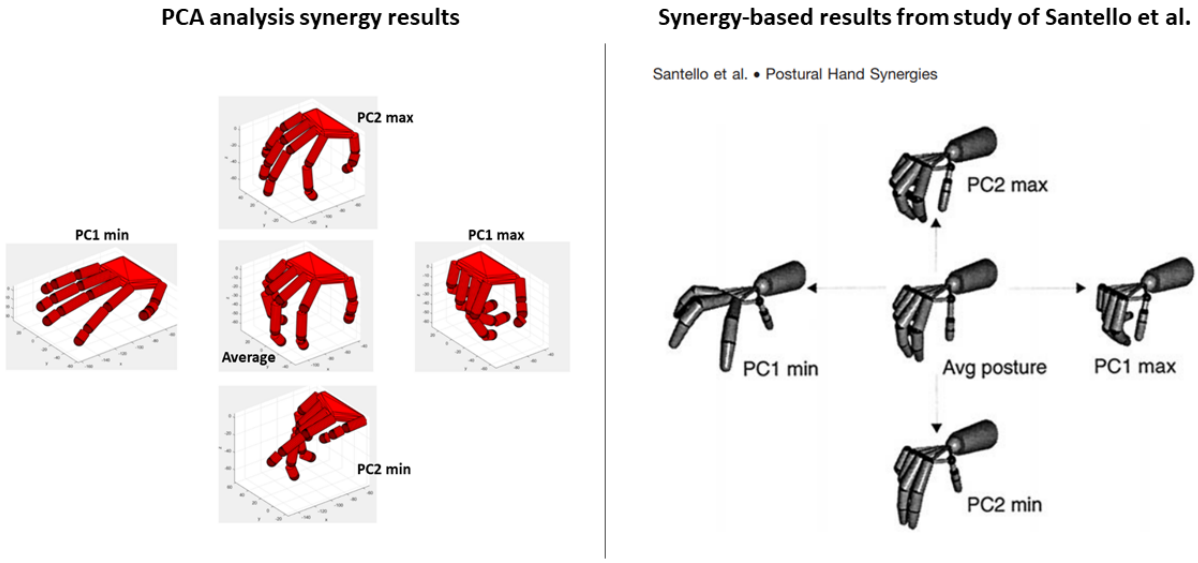


Figure 4-4: The Left picture shows our analysis result, and the percentage of the first two principal components has reached over 75 percent. The two dominant PCs in Santello's work are on the right side, which covers over 80 percent of the maximum possible amount of information about objects. We can see in the figure both PC1 has shown the same trend of movement.

Chapter 5

Postural Synergy and Hand Mapping

In this chapter, we will propose a novel method with simulation results of synergies-based control. The goal is to show that natural and artificial synergies can be used to enable a linear mapping strategy from an human hand and a robotic hand (SoftHand 2 is considered in this work). Researchers such as [21], [54], and [53] proved that human hand motions could be explained in a reduced set of eigenvectors, i.e., covariation patterns of joints, also called Principal Components (PC) or Postural Synergies [55], that the dimension is much lower than human hand kinematic structure, which we have discussed in the previous chapter. SH2 has applied the synergy concept with a slight difference in its design, and it has only two (Degree of Actuation) DoAs. Engineers provide a novel model which can control the internal forces needed to hold an object while still operating through synergies [56], [46]. The novel concept is called "soft synergies," and in the subsequent studies, it has been upgraded to "adaptive synergies [57], [58]."

5-1 Mathematics of Postural Synergy

Postural synergies define the hand posture through a lower number of degrees of freedom. To achieve so, the PCA is dedicated to extracting reduced dimensional data from the whole hand joint variables and represents them in a set of linear dependence. The hand pose can therefore be described in the reduced dimensional synergy space by neglecting higher principal components without significant loss in quality of data [59]. Using this low dimensional grasping space, the hand model was developed the simplified kinematic structure of real human hand. The linear relation can be explained by the following formulation:

$$q_m(t) = S \cdot z(t) + q_{avg}$$

The q_{avg} represents the average the average of data set hand postures (the illustration is in Figure 5-1), and the $z(t)$ is the vector of synergy variables at instant time t. $q_m(t)$ is the the joint space vector representing the hand posture at time t, which will be the tracking result from the EKF. S is the matrix of coefficients stemming from the principal component analysis on the grasping posture data set.

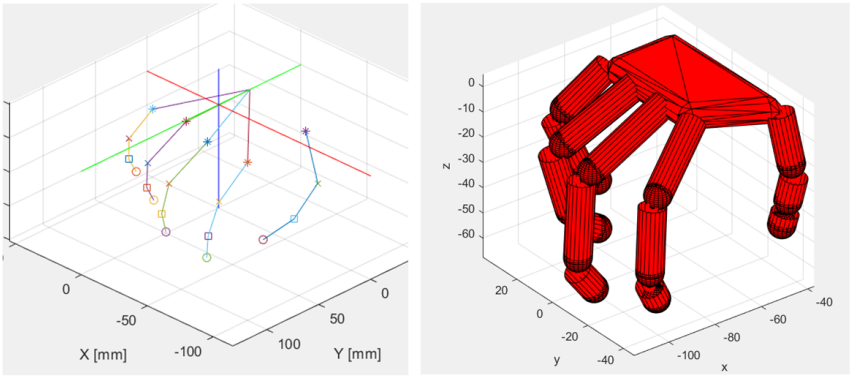


Figure 5-1: Left figure shows the average of 21 hand postures for one subject (S7) in our data set. We only considered 21 poses because, in the original 30 tasks, some of them are not directly related to hand grasping (see appendix A for more details). The right figure is the average of pose and simulated by a Matlab Tool SynGrasp [13].

5-2 Synergies-Based Mapping with Pisa/IIT SoftHand 2

The above results have successfully reduced a high-dimensionality human hand kinematics model into two principal components (PC1 and PC2). To complete the mapping between the human hand and the SH2, we need to assume another vector, $q_s(t)$, representing the robotic hand joints angle. And to map the results from human hand tracking $q_m(t)$ to a robotic hand, we surmise that the slave hand and master hand kinematics were the same. The following equations will explain the idea we applied.

$$q_s(t) = \begin{bmatrix} R \\ R_f \end{bmatrix} \cdot \begin{bmatrix} \sigma \\ \sigma_f \end{bmatrix} + \epsilon_s$$

$$q_m(t) = S \cdot \begin{bmatrix} PC_1 \\ PC_2 \end{bmatrix} + \epsilon_m$$

Since there is no object contact, thus ϵ_s is close to zero can be neglected. And as a result, we can write the mapping directly as:

$$q_m(t) = q_s(t)$$

To achieve the synergy-based mapping control, the last part of the mapping strategy is to find the relationship between PC1, PC2, and DoA1, DoA2. In Figures 5-3, we find out the

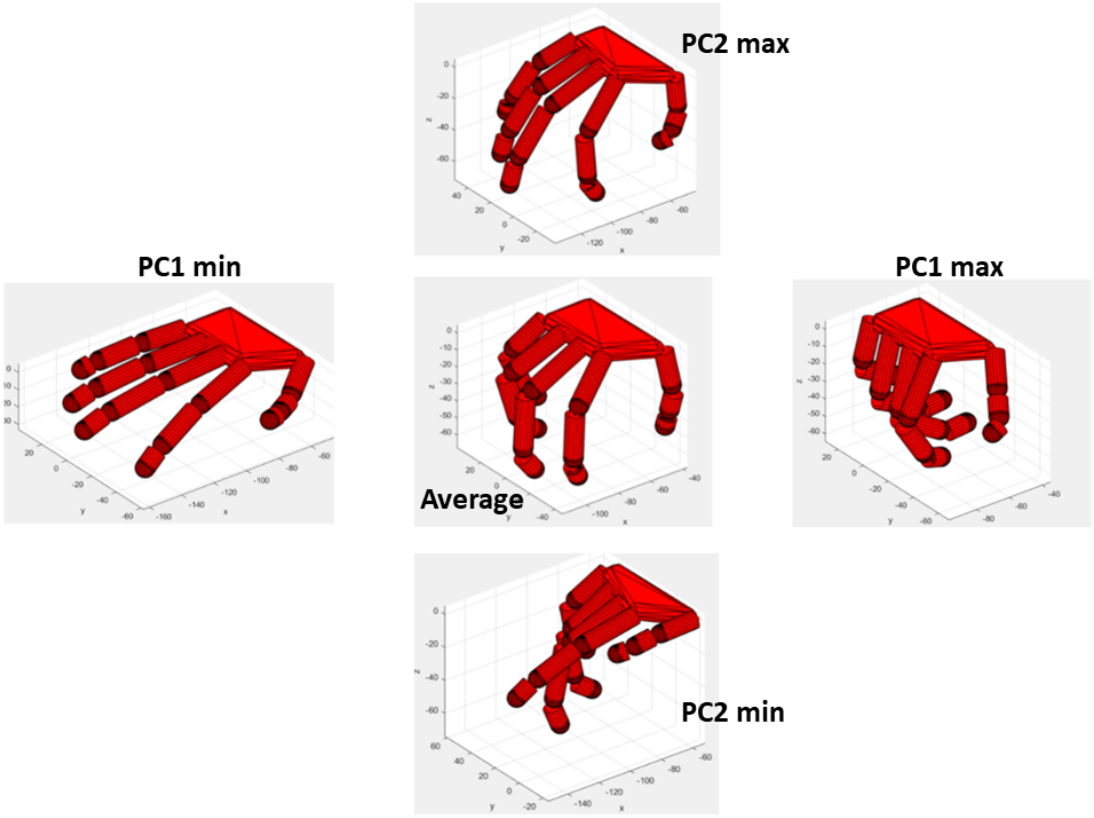


Figure 5-2: Postural synergies defined by the first two principal components in our data set of one subject (S7). To show them more clearly, here we also applied SynGrasp to illustrate these postures. The hand posture at the center of the PC axes is the average of 21 hand postures for one subject (S7). The postures to the right and left are for the postures for the maximum (max) and minimum (min) values of the first principal component (PC1), and coefficients for the other principal components having been set to zero. The postures at the top and bottom are for the maximum and minimum values of the second principal component (PC2).

correlation between each motor to each corresponding PC. The comparison of PC1 and motor σ (DoA1) shows positive linear regression, while PC2 and motor s (DoA2) shows a negative correlation. Though only 21 poses from the data-set have been concerned, the mapping results show a possible intuitive control scheme for future work to conduct in real experiments.

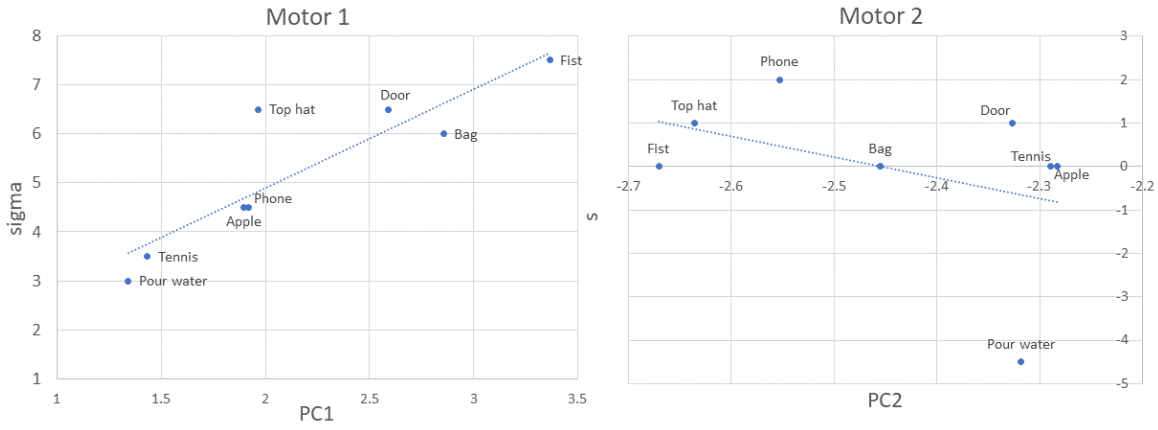


Figure 5-3: Left figure shows the average of 21 hand postures for one subject (S7) in our data set. We only considered 21 poses because, in the original 30 tasks, some of them are not directly related to hand grasping (see appendix A for more details). The right figure is the average of pose and simulated by a Matlab Tool SynGrasp [13].

Table 5-1: Synergy-based result (PC_1, PC_2) and the corresponding motor σ and s on SH2

Hand Movement	σ	s	PC_1	PC_2
Tennis	3.5π	0 mm	1.4343 rad	-2.2889 rad
Phone	4.5π	2 mm	1.9196 rad	-2.5532 rad
Door	6.5π	1 mm	2.5912 rad	-2.3263 rad
Pour water	3π	-4.5 mm	1.3418 rad	-2.3178 rad
Top hat	6.5π	1 mm	1.9667 rad	-2.6359 rad
Bag	6π	0 mm	2.8589 rad	-2.4549 rad
Fist	7.5π	0 mm	3.3688 rad	-2.6707 rad
Apple	4.5π	0 mm	1.8979 rad	-2.2824 rad

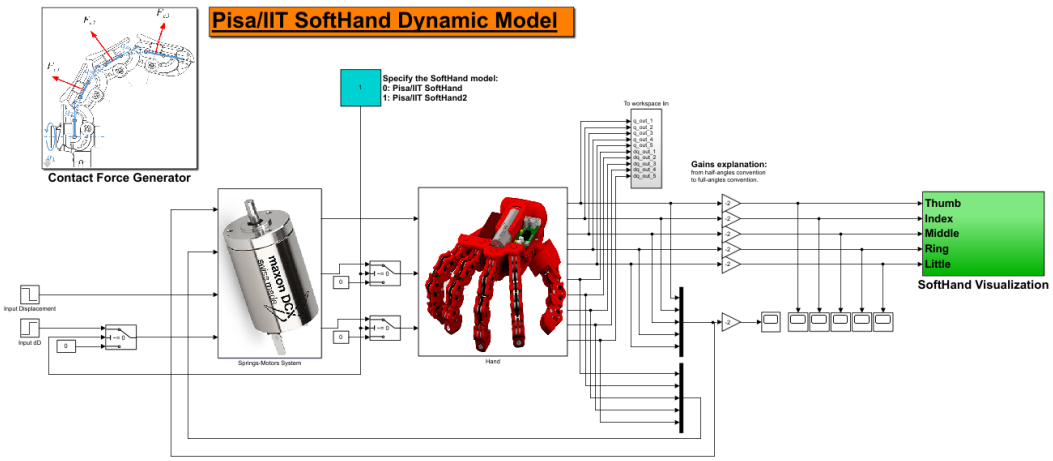


Figure 5-4: The Matlab and Simulink model of Pisa/IIT SoftHand and SoftHand 2 to simulate behavior during its free closure or during the application of external forces [14]

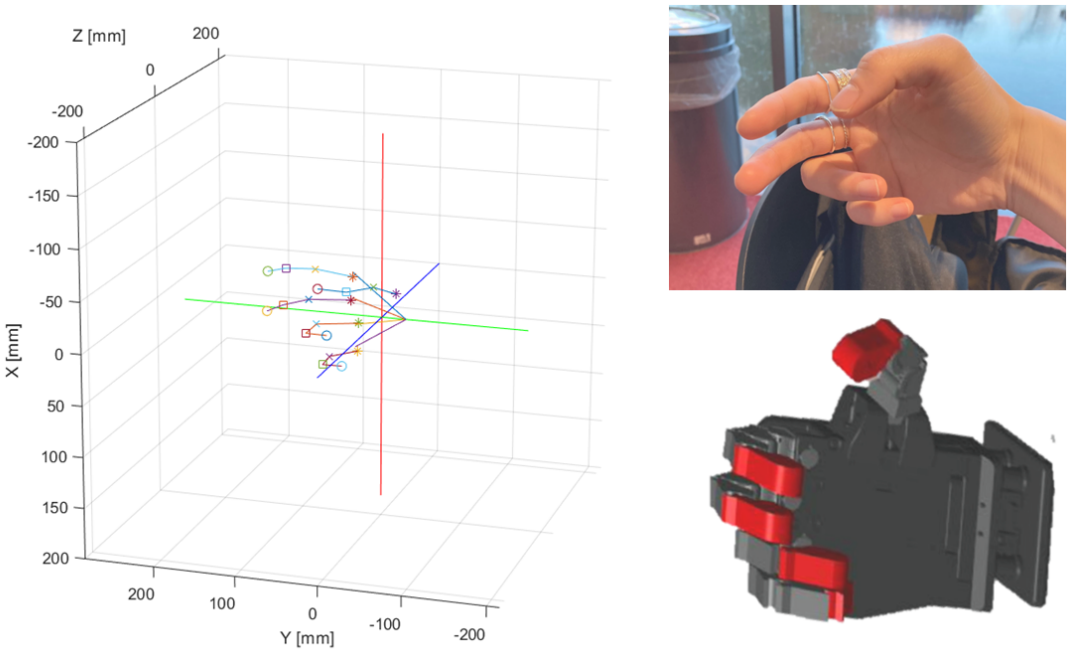


Figure 5-5: Simulation of the SoftHand2 humanoid robot hands being teleoperated using Simulink system (Human hand pose of "three finger grabbing" replication).

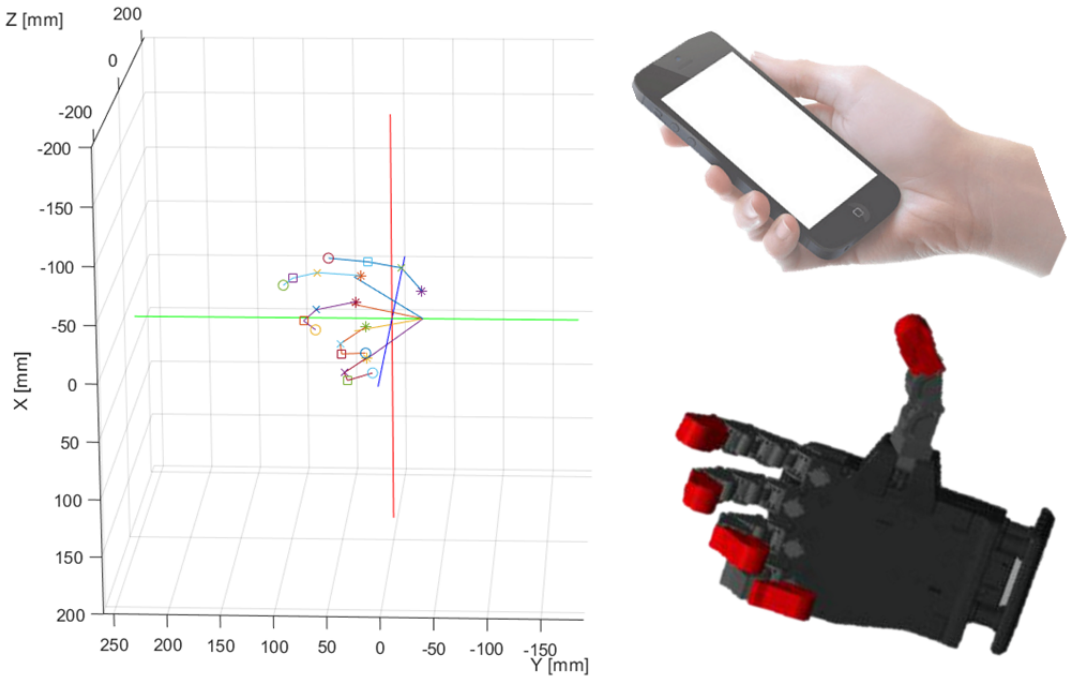


Figure 5-6: Simulation of the SoftHand2 humanoid robot hands being teleoperated using Simulink system (Human hand pose of "grabbing a phone" replication).

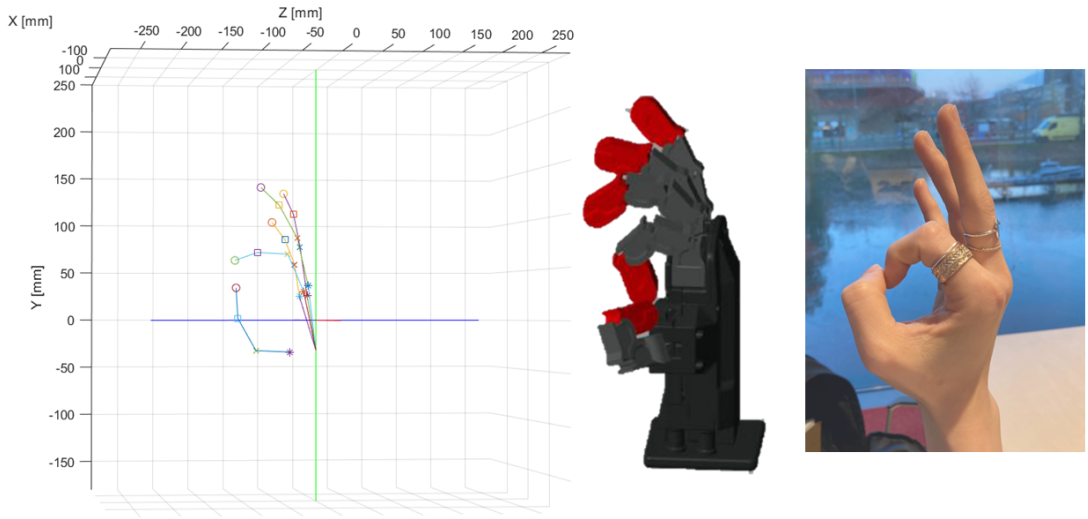


Figure 5-7: Simulation of the SoftHand2 humanoid robot hands being teleoperated using Simulink system (Human hand pose of "OK" replication).

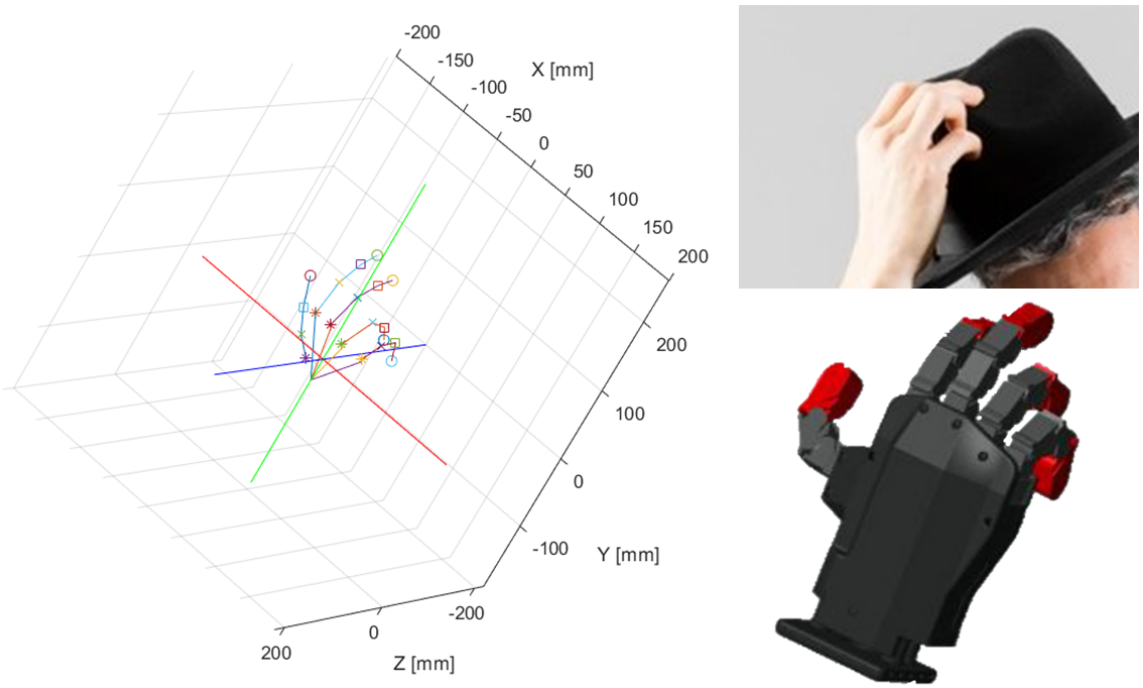


Figure 5-8: Simulation of the SoftHand2 humanoid robot hands being teleoperated using Simulink system (Human hand pose of "greeting with a top hat" replication).

Chapter 6

Conclusion and Future Works

In this work, we have presented a reliable approach for 3D hand pose estimation from the raw data of the motion capture camera. The Cartesian position input of each joint in hand can be accurately predicted by the approach to the 3D position of each joint in hand, thus, retrieving a 3D hand pose relative to the operator viewpoint. Moreover, we have proposed a novel method to teleoperate the Pisa/IIT SoftHand 2 based on the tracking result. The Synergies-based mapping control strategy allows the user to implement the robotic hand intuitively. In comparing two motors and the two dominant principal components (Figure 5-3), we find linear correlations between them, which shows a possibility of precise control. Finally, we simulated the proposed method on a robotics application: a robot teleoperation system where the SH2 replicates human hand fine movements.

6-1 Discussion

Although hand tracking with the extended Kalman filter performs a pretty promising result from the simulation, we still need to apply it with actual experiments for parameter tuning. Moreover, the 3D hand model is rebuilt with only one subject, and thus a further calibration with different subjects has to be done for obtaining better performance. Another noticed point is that the two DoAs of SH2 still have certain dissimilarities with human postural synergy from our results and Santello et al.s [21], which can be seen in Figure 4-4. This leads to another work evaluating whether engaging more than two synergies will get better results in hand posture evaluation. However, some researchers have concluded that adding higher-order synergies does not improve performances [60], [61]. This conclusion yields another possible control strategy of SH2 that, instead of direct mapping, we can design a subspace for correlating to certain SH2 motions. For example, in work from Meeker et al. [62], they provided a low-dimensional teleoperation subspace that can be used as

an intermediary for mapping between hand pose spaces. Then some poses related to a lot of thumb movements in our PCA results are hard to operate, can be organized into particular intermediate spaces, and be assigned into correct hand movements. Last but not least, the motion caption hand poses data also needs more varying collection. One example shows clearly how the data-set affects PCA result in our work. There is no hand grasp tremendous object pose in our data-set, so the PCA result shows a slight difference in PC1 min.

6-2 Future Works

As mentioned above, due to the severe pandemic, we do not have a chance to operate the synergies-based mapping in real SH2 for implementing experiments, which can be the first coming work to test and modify the simulation results. One thing worth mentioning here is that the SH2 is a soft robotic hand, which brings more flexibility than what we simulated in the "rigid body" condition. As a result, some poses may not be able to match in our computation, they have a high possibility to perform when conducting on the real SH2.

One primary research in hand teleoperation we did not conduct in the work is considering the haptics or force feedback, which has been shown in some researches [63], [64], [65]. The force feedback from the robotic hand can help the operator aware of the situation and improve performance. Furthermore, haptics feedback plays a vital work in providing an immersive experience in telemanipulation tasks. To obtain more humanoid hand grasping, considering haptics or force feedback based on synergy-based control is a feasible future work.

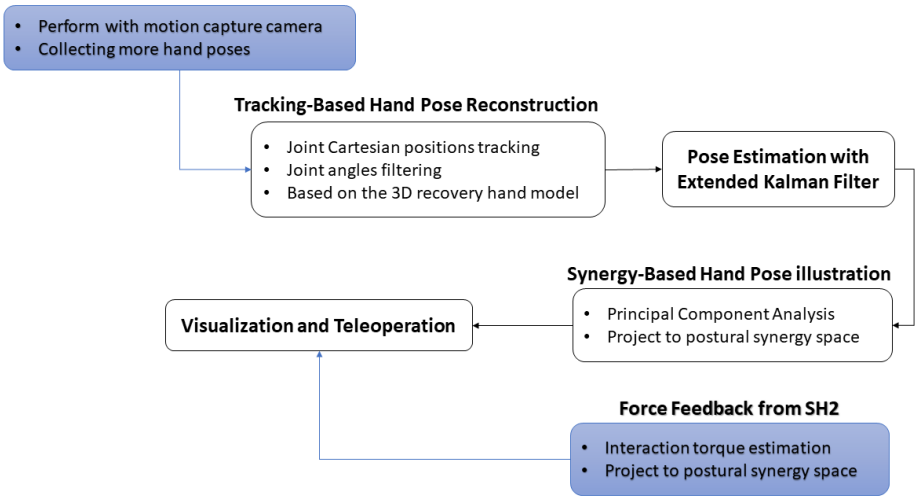
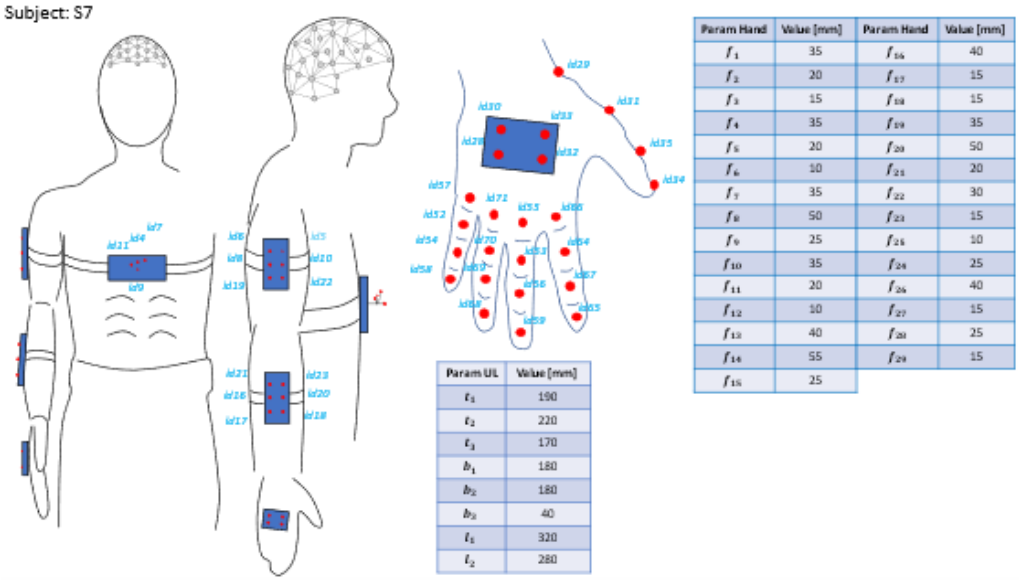


Figure 6-1: The integration of our work is illustrated in Black block and the future work is showed in blue block.

Appendix A

Appendix A

A-1 Previous Experiment



Param UL	Value [mm]
ℓ_1	190
ℓ_2	220
ℓ_3	170
b_1	180
b_2	180
b_3	40
l_1	320
l_2	280

Param Hand	Value [mm]	Param Hand	Value [mm]
f_1	35	f_{16}	40
f_2	20	f_{17}	15
f_3	15	f_{18}	15
f_4	35	f_{19}	35
f_5	20	f_{20}	50
f_6	10	f_{21}	20
f_7	35	f_{22}	30
f_8	50	f_{23}	15
f_9	25	f_{25}	10
f_{10}	35	f_{26}	25
f_{11}	20	f_{26}	40
f_{12}	10	f_{27}	15
f_{13}	40	f_{28}	25
f_{14}	55	f_{29}	15
f_{15}	25		

Figure A-1: The illustration of surface markers and their locations on the subject [15].

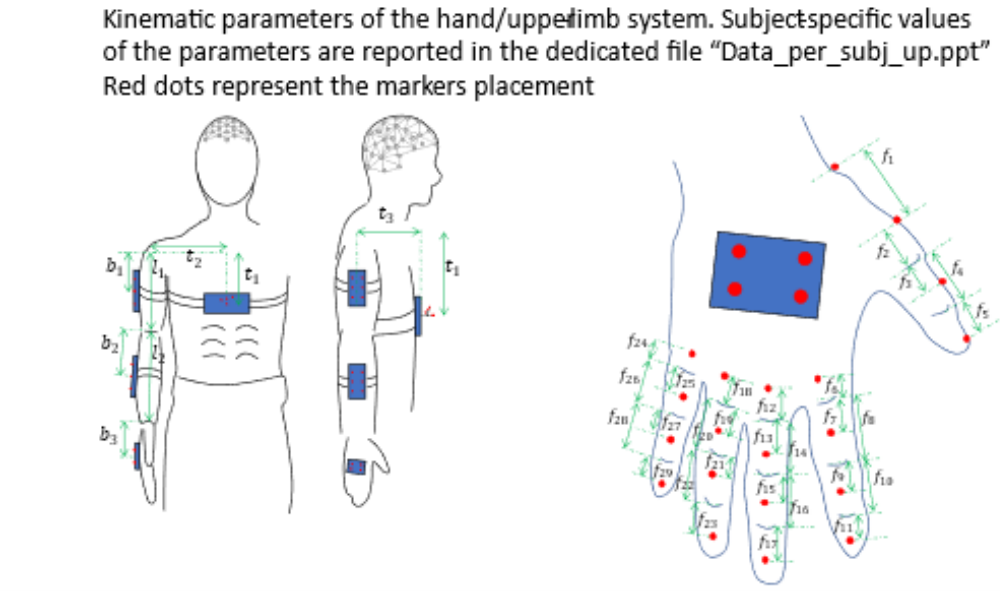


Figure A-2: The markers placement and the scale of hand surface markers [15].

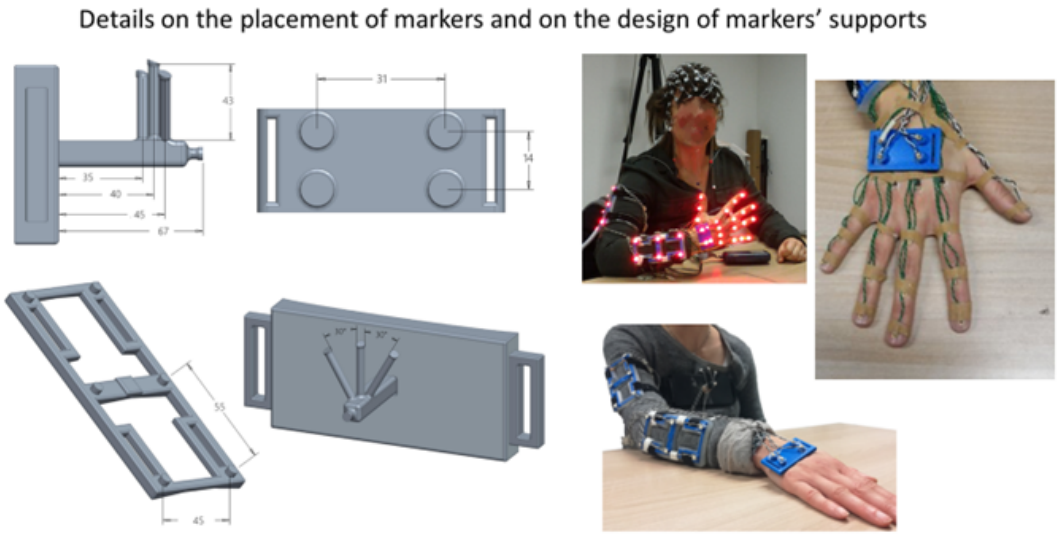


Figure A-3: The design of plastic supports that were attaching the subject in the experiment [15]. These supports play an important role in calibration and stabilization.

LIST OF MOVEMENTS

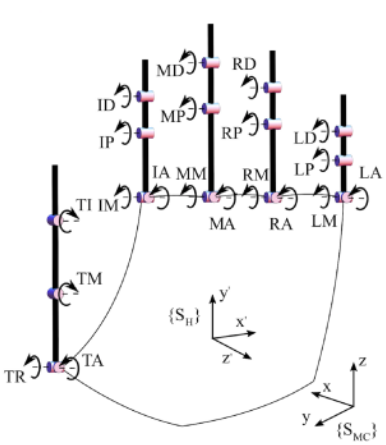
#	# Cut	Class	Description
1		Int	Ok gesture (lifting hand from the table)
2		Int	Thumb down (lifting hand from the table)
3		Int	Exultation (extending the arm up in the air and keeping it in with closed fist)
4		Int	Hitchhiking (extending the arm along the frontal plane, laterally, parallel to the floor, with extended elbow, closed fist, extended thumb)
5		Int	Block out sun from own face (with open hand, touching the face with the palm and covering the eyes)
6		Int	Greet (with open hand, moving wrist) (3 times)
7		Int	Military salute (with lifted elbow)
8		Int	Stop gesture (extending the arm along the sagittal plane, parallel to the floor, with extended elbow, open palm)
9		Int	Pointing (with index finger) of something straight ahead (with outstretched arm)
10		Int	Silence gesture (bringing the index finger, with the remainder of the hand closed, on the lips)
11	2	Tr	Reach and grasp a small suitcase (placed along own frontal plane) from the handle, lift it and place it on the floor (close to own chair, along own sagittal plane)
12	3	Tr	Reach and grasp a glass, drink for 3 seconds and place it in the initial position
13	4	Tr	Reach and grasp a phone receiver (placed along own sagittal plane), carry it to own ear for 3 seconds and place it in the initial position
14	6	Tr	Reach and grasp a book (placed overhead on a shelf), put in on the table and open it (from right side to left side)
15	8	Tr	Reach and grasp a small cup from the handle (2 fingers + thumb), drink for 3 seconds and place it in the initial position
16	11	Tr	Reach and grasp an apple, mimic biting and put it in the initial position
17	12,13	Tr	Reach and grasp a hat (placed on the right side of the table) from its top and place it on own head
18	12	Tr	Reach and grasp a cup from its top, lift it and put it on the left side of the table
19	15	Tr	Receive a tray from someone (straight ahead, with open hand) and put it in the middle of the table
20	16	Tr	Reach and grasp a key in a lock (vertical axis), extract it from the lock and put it on the left side of the table
21	1	T-M	Reach and grasp a bottle, pour water into a glass and put the bottle in the initial position
22	2,3,4	T-M	Reach and grasp a tennis racket (placed along own frontal plane) and play a forehand (the subject is still seated)
23	5	T-M	Reach and grasp a toothbrush, brush teeth (horizontal axis, one time left-right) and put it inside a holder (on the right side of the table)
24	6	T-M	Reach and grasp a laptop, open it (without changing its position) (4 fingers + thumb)
25	7,8,9	T-M	Reach and grasp a pen (placed on the right side of the table) and draw a vertical line on the table (from the top to the bottom)
26	7	T-M	Reach and grasp a pencil (placed along own frontal plane) (3 fingers + thumb) and put it inside a squared pencil holder (placed on the left side of the table)
27	9	T-M	Reach and grasp a tea bag in a cup (1 finger + thumb), remove it from the cup and place it on the table on the right side of the table
28	10	T-M	Reach and grasp a doorknob, turn it clockwise and counterclockwise and open the door
29	13	T-M	Reach and grasp a tennis ball (with fingertips) and place it in a basket on the floor (right)
30	14	T-M	Reach and grasp a cap (2 fingers + thumb) of a bottle (held by left hand), unscrew it and place it overhead on a shelf

Figure A-4: The list of actions defining the SoftPro protocol [15].

A-2 Hand Kinematics

$$T_{0,1} = \begin{bmatrix} \cos\theta_1 & 0 & -\sin\theta_1 & 0 \\ \sin\theta_1 & 0 & \cos\theta_1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, T_{1,2} = \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 & l_2\cos\theta_2 \\ \sin\theta_2 & \cos\theta_2 & 0 & l_2\sin\theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_{2,3} = \begin{bmatrix} \cos\theta_3 & -\sin\theta_3 & 0 & l_3\cos\theta_3 \\ \sin\theta_3 & \cos\theta_3 & 0 & l_3\sin\theta_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, T_{3,4} = \begin{bmatrix} \cos\theta_4 & -\sin\theta_4 & 0 & l_4\cos\theta_4 \\ \sin\theta_4 & -\cos\theta_1 & 0 & l_4\sin\theta_4 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



DoFs	Description
TA	Thumb Abduction
TR	Thumb Rotation
TM	Thumb Metacarpal
TI	Thumb Interphalangeal
IA	Index Abduction
IM	Index Metacarpal
IP	Index Proximal
ID	Index Distal
MM	Middle Metacarpal
MA	Middle Abduction
MP	Middle Proximal
MD	Middle Distal
RA	Ring Abduction
RM	Ring Metacarpal
RP	Ring Proximal
RD	Ring Distal
LA	Little Abduction
LM	Little Metacarpal
LP	Little Proximal
LD	Little Distal

(a) Kinematic model

Thumb (4 axis, 4R)

j	θ	d	a	α	offset
1	q_1	0	0	$-\frac{\pi}{2}$	0
2	q_2	0	a_2	$-\frac{\pi}{2}$	0
3	q_3	0	a_3	0	0
4	q_4	0	a_4	0	0

Long Finger (4 axis, 4R)

j	θ	d	a	α	offset
1	q_1	0	0	$-\frac{\pi}{2}$	0
2	q_2	0	a_2	0	0
3	q_3	0	a_3	0	0
4	q_4	0	a_4	0	0

(b) DH parametrization

Figure A-5: The illustration of different mapping approaches [16].

Bibliography

[1] B. Carlton, "Haptx vr gloves." https://vrscout.com/news/haptx-telerobotics-across-the-globe/, March 219.

[2] F. Chen, S. Appendino, A. Battezzato, A. Favetto, M. Mousavi, and F. Pescarmona, "Constraint study for a hand exoskeleton: Human hand kinematics and dynamics," *Journal of Robotics*, vol. 2013, 01 2013.

[3] W. Chen, C. Yu, C. Tu, Z. Lyu, J. Tang, S. Ou, Y. Fu, and Z. Xue, "A survey on hand pose estimation with wearable sensors and computer-vision-based methods," *Sensors*, vol. 20, no. 4, 2020.

[4] S. Walairacht, K. Yamada, S. Hasegawa, Y. Koike, and M. Sato, "4 + 4 fingers manipulating virtual objects in mixed-reality environment," *Presence: Teleoper. Virtual Environ.*, vol. 11, p. 134–143, apr 2002.

[5] D. Kulon, H. Wang, R. A. Güler, M. M. Bronstein, and S. Zafeiriou, "Single image 3d hand reconstruction with mesh convolutions.," *CoRR*, vol. abs/1905.01326, 2019.

[6] D. Mohr and G. Zachmann, "Model-based high-dimensional pose estimation with application to hand tracking," 2012.

[7] F. Zhang, V. Bazarevsky, A. Vakunov, A. Tkachenka, G. Sung, C.-L. Chang, and M. Grundmann, "Mediapipe hands: On-device real-time hand tracking," *ArXiv*, vol. abs/2006.10214, 2020.

[8] A. Sinha, C. Choi, and K. Ramani, "Deephand: Robust hand pose estimation by completing a matrix imputed with deep features," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4150–4158, 2016.

[9] F. Gomez-Donoso, S. Orts, and M. Cazorla, "Accurate and efficient 3d hand pose regression for robot hand teleoperation using a monocular rgb camera," *Expert Systems with Applications*, vol. 136, 06 2019.

[10] G. Salvietti, "Replicating human hand synergies onto robotic hands: A review on software and hardware strategies," *Frontiers in Neurorobotics*, vol. 12, p. 27, 2018.

[11] G. Averta, F. Barontini, V. Catrambone, S. Haddadin, G. Handjaras, J. P. O. Held, T. Hu, E. Jakubowitz, C. M. Kanzler, J. Kühn, O. Lambercy, A. Leo, A. Obermeier, E. Ricciardi, A. Schwarz, G. Valenza, A. Bicchi, and M. Bianchi, "U-Limb: A multi-modal, multi-center database on arm motion control in healthy and post-stroke conditions," *GigaScience*, vol. 10, 06 2021.

[12] I. Carpinella, J. Jonsdottir, and M. Ferrarin, "Multi-finger coordination in healthy subjects and stroke patients: A mathematical modelling approach," *Journal of neuroengineering and rehabilitation*, vol. 8, p. 19, 04 2011.

[13] M. Malvezzi, G. Gioioso, G. Salvietti, and D. Prattichizzo, "Syngrasp: A matlab toolbox for underactuated and compliant hands," *Robotics Automation Magazine, IEEE*, vol. 22, pp. 52–68, Dec 2015.

[14] "Natural machine motion initiative." https://www.naturalmachinemotioninitiative.com/blank-10. Accessed: 2014.

[15] G. Averta, F. Barontini, V. Catrambone, S. Haddadin, G. Handjaras, J. Held, T. Hu, E. Jakubowitz, C. Kanzler, J. Kuehn, O. Lambercy, A. Leo, A. Obermeier, E. Ricciardi, A. Schwarz, G. Valenza, A. Bicchi, and M. Bianchi, "U-limb: A multi-modal, multi-center database on arm motion control in healthy and post-stroke conditions," *GigaScience*, vol. 10, 06 2021.

[16] C. Della Santina, M. Bianchi, G. Averta, S. Ciotti, V. Arapi, S. Fani, E. Battaglia, M. G. Catalano, M. Santello, and A. Bicchi, "Postural hand synergies during environmental constraint exploitation," *Frontiers in Neurorobotics*, vol. 11, p. 41, 2017.

[17] M. Veber, T. Bajd, and M. Munih, "Assessing joint angles in human hand via optical tracking device and calibrating instrumented glove," vol. 42, no. 5, pp. 451–463, 2007-10-04.

[18] A. Erol, G. Bebis, M. Nicolescu, R. Boyle, and X. Twombly, "Vision-based hand pose estimation: A review," *Computer Vision and Image Understanding*, vol. 108, pp. 52–73, 10 2007.

[19] J. Condell, G. Moore, and J. Moore, "Software and methods for motion capture and tracking in animation.," pp. 3–9, 01 2006.

[20] G. D. Kessler, L. F. Hodges, and N. Walker, "Evaluation of the cyberglove as a whole-hand input device," *ACM Trans. Comput.-Hum. Interact.*, vol. 2, p. 263–283, dec 1995.

[21] M. Santello, M. Flanders, and J. F. Soechting, "Postural hand synergies for tool use," vol. 18, no. 23, pp. 10105–10115, 1998-12-01.

[22] J. Rehg and T. Kanade, "Model-based tracking of self-occluding articulated objects," in *Proceedings of IEEE International Conference on Computer Vision*, pp. 612–617, 1995.

[23] N. K. Iason Oikonomidis and A. Argyros, "Efficient model-based 3d tracking of hand articulations using kinect," in *Proceedings of the British Machine Vision Conference*, pp. 101.1–101.11, BMVA Press, 2011. http://dx.doi.org/10.5244/C.25.101.

[24] P. Cerveri, E. De Momi, N. F. Lopomo, G. Baud-Bovy, R. Barros, and G. Ferrigno, "Finger kinematic modeling and real-time hand motion estimation," *Annals of biomedical engineering*, vol. 35, pp. 1989–2002, 12 2007.

[25] E. A. Arkenbout, J. de Winter, and P. Breedveld, "Robust hand motion tracking through data fusion of 5dt data glove and nimble vr kinect camera measurements," *Sensors (Basel, Switzerland)*, vol. 15, pp. 31644 – 31671, 2015.

[26] J. Crowley and J. Martin, "Visual processes for tracking and recognition of hand gestures," 11 1998.

[27] X. Zheng, B. Jiang, and J. Zhang, "Deformation representation based convolutional mesh autoencoder for 3d hand generation," *Neurocomputing*, vol. 444, 11 2020.

[28] J. Wöhlke, S. Li, and D. Lee, "Model-based hand pose estimation for generalized hand shape with appearance normalization," 07 2018.

[29] H. Francke, J. Ruiz-del Solar, and R. Verschae, "Real-time hand gesture detection and recognition using boosted classifiers and active learning," in *Advances in Image and Video Technology* (D. Mery and L. Rueda, eds.), (Berlin, Heidelberg), pp. 533–547, Springer Berlin Heidelberg, 2007.

[30] P. Dreuw, D. Keysers, T. Deselaers, and H. Ney, "Gesture recognition using image comparison methods," vol. 3881, pp. 124–128, 05 2005.

[31] J. Tompson, M. Stein, Y. Lecun, and K. Perlin, "Real-time continuous pose recovery of human hands using convolutional networks," *ACM Trans. Graph.*, vol. 33, sep 2014.

[32] B. Stenger, P. Mendonca, and R. Cipolla, "Model-based hand tracking using an unscented kalman filter," 01 2001.

[33] F. Chen, S. Appendino, A. Battezzato, A. Favetto, M. Mousavi, and F. Pescarmona, "Constraint study for a hand exoskeleton: Human hand kinematics and dynamics," *Journal of Robotics*, vol. 2013, 01 2013.

[34] S. Bouaziz, A. Tagliasacchi, and M. Pauly, "Sparse iterative closest point," in *Proceedings of the Eleventh Eurographics/ACMSIGGRAPH Symposium on Geometry Processing*, SGP '13, (Goslar, DEU), p. 113–123, Eurographics Association, 2013.

[35] V. John, S. Ivekovic, and E. Trucco, "Articulated human motion tracking with hpso.," vol. 1, pp. 531–538, 01 2009.

[36] M. Bergamasco, A. Frisoli, and C. Avizzano, *Exoskeletons as Man-Machine Interface Systems for Teleoperation and Interaction in Virtual Environments*, vol. 31, pp. 61–76. 08 2007.

[37] K. Farry, I. Walker, and R. Baraniuk, "Myoelectric teleoperation of a complex robotic hand," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 5, pp. 775–788, 1996.

[38] M. Liarokapis, P. Artemiadis, C. Bechlioulis, and K. Kyriakopoulos, "Directions, methods and metrics for mapping human to robot motion with functional anthropomorphism: A review," 09 2013.

[39] M. V. Liarokapis, P. K. Artemiadis, and K. J. Kyriakopoulos, "Functional anthropomorphism for human to robot motion mapping," in *2012 IEEE RO-MAN: The 21st IEEE International Symposium on Robot and Human Interactive Communication*, pp. 31–36, 2012.

[40] C. Meeker, M. Haas-Heger, and M. Ciocarlie, "A continuous teleoperation subspace with empirical and algorithmic mapping algorithms for non-anthropomorphic hands," 11 2019.

[41] M. Ciocarlie, C. Goldfeder, and P. Allen, "Dimensionality reduction for hand-independent dexterous robotic grasping," in *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3270–3275, 2007.

[42] L. Pao and T. Speeter, "Transformation of human hand positions for robotic hand control," in *Proceedings, 1989 International Conference on Robotics and Automation*, pp. 1758–1763 vol.3, 1989.

[43] G. Gioioso, G. Salvietti, M. Malvezzi, and D. Prattichizzo, "Mapping synergies from human to robotic hands with dissimilar kinematics: An approach in the object domain," *IEEE Transactions on Robotics*, vol. 29, no. 4, pp. 825–837, 2013.

[44] A. Bicchi, M. Gabiccini, and M. Santello, "Modelling natural and artificial hands with synergies," *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, vol. 366, pp. 3153–61, 11 2011.

[45] G. Torres-Oviedo and L. Ting, "Muscle synergies characterizing human postural responses," *Journal of neurophysiology*, vol. 98, pp. 2144–56, 11 2007.

[46] M. Bianchi, P. Salaris, and A. Bicchi, "Synergy-based hand pose sensing: Reconstruction enhancement," *The International Journal of Robotics Research*, vol. 32, 06 2012.

[47] J. M. Rehg and T. Kanade, "Visual tracking of high DOF articulated structures: An application to human hand tracking," in *Computer Vision — ECCV '94* (J.-O. Eklundh, ed.), vol. 801, pp. 35–46, Springer Berlin Heidelberg, 1994. Series Title: Lecture Notes in Computer Science.

[48] G. Lu, N. An, and Z. Liu, "Design and kinematics analysis of a bionic finger hand rehabilitation robot mechanism," in *2019 34rd Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, pp. 715–718, 2019.

[49] R. M. Murray, Z. Li, and S. S. Sastry, *A mathematical introduction to robotic manipulation*. CRC press, 2017.

[50] Wikipedia contributors, "Extended kalman filter — Wikipedia, the free encyclopedia." https://en.wikipedia.org/w/index.php?title=Extended_Kalman_filter&oldid=1004599218, 2021. [Online; accessed 9-November-2021].

[51] S. Qiong, W.-T. Lee, K. Russell, and R. Sodhi, "On motion generation of watt i mechanisms for mechanical finger design," *Transactions of the Canadian Society for Mechanical Engineering*, vol. 32, pp. 411–421, 09 2008.

[52] A. Rankers and H. Schrama, "Sam: Simulation and analysis of mechanisms," 01 2002.

[53] M. Latash, V. Krishnamoorthy, J. Scholz, and V. Zatsiorsky, "Postural synergies and their development," *Neural plasticity*, vol. 12, pp. 119–30; discussion 263, 02 2005.

[54] C. Brown and H. Asada, "Inter-finger coordination and postural synergies in robot hands via mechanical implementation of principal components analysis," pp. 2877 – 2882, 12 2007.

[55] G. Averta, G. C. Bettelani, C. Della Santina, and M. Bianchi, *Kineto-Dynamic Modeling of Human Upper Limb for Robotic Manipulators and Assistive Applications*, pp. 23–51. 04 2020.

[56] M. Gabiccini, A. Bicchi, D. Prattichizzo, and M. Malvezzi, "On the role of hand synergies in the optimal choice of grasping forces," *Auton. Robots*, vol. 31, pp. 235–252, 10 2011.

[57] M. Catalano, G. Grioli, A. Serio, E. Farnioli, C. Piazza, and A. Bicchi, "Adaptive synergies for a humanoid robot hand," pp. 7–14, 11 2012.

[58] G. Grioli, M. Catalano, E. Silvestro, S. Tono, and A. Bicchi, "Adaptive synergies: An approach to the design of under-actuated robotic hands," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1251–1256, 2012.

[59] A. Portnova, F. Rizzoglio, I. Nisky, M. Casadio, F. Mussa-Ivaldi, and E. Rombokas, "Linear and non-linear dimensionality-reduction techniques on full hand kinematics," *Frontiers in Bioengineering and Biotechnology*, vol. 8, p. 429, 05 2020.

[60] M. Santello and J. F. Soechting, "Gradual molding of the hand to object contours," *Journal of Neurophysiology*, vol. 79, no. 3, pp. 1307–1320, 1998. PMID: 9497412.

[61] M. Santello and J. Soechting, "Force synergies for multifingered grasping," *Experimental brain research. Experimentelle Hirnforschung. Expérimentation cérébrale*, vol. 133, pp. 457–67, 09 2000.

[62] C. Meeker, M. Haas-Heger, and M. Ciocarlie, "A continuous teleoperation subspace with empirical and algorithmic mapping algorithms for non-anthropomorphic hands," 11 2019.

[63] N. B. Nadgere, "Haptic rendering/control for underactuated qb softhand using haption hglove," November 2020.

[64] S. Mulatto, A. Formaglio, M. Malvezzi, and D. Prattichizzo, "Using postural synergies to animate a low-dimensional hand avatar in haptic simulation," *IEEE Transactions on Haptics*, vol. 6, no. 1, pp. 106–116, 2013.

[65] A. Brygo, I. Sarakoglou, G. Grioli, and N. Tsagarakis, "Synergy-based bilateral port: A universal control module for tele-manipulation frameworks using asymmetric master–slave systems," *Frontiers in Bioengineering and Biotechnology*, vol. 5, 04 2017.