

**A KINEMATIC HUMAN UPPER BODY MODEL FOR EVALUATING
CLOTHING**

A Dissertation
Presented to
The Academic Faculty

By

Franziska Schlagenhauf

In Partial Fulfillment
of the Requirements for the Degree
Master of Science in the
George W. Woodruff School of Mechanical Engineering

Georgia Institute of Technology

August 2017

Copyright © Franziska Schlagenhauf 2017

A KINEMATIC HUMAN UPPER BODY MODEL FOR EVALUATING CLOTHING

Approved by:

Dr. William Singhose, Advisor
George W. Woodruff School of Mechanical
Engineering
Georgia Institute of Technology

Dr. Kok-Meng Lee
George W. Woodruff School of Mechanical
Engineering
Georgia Institute of Technology

Dr. Oliver Sawodny
Institut für Systemdynamik
Universität Stuttgart

Dr. Cristina Tarin
Institut für Systemdynamik
Universität Stuttgart

Date Approved: July 28, 2017

ACKNOWLEDGEMENTS

I want to thank my advisor, Dr. William Singhose, for his advice and guidance while writing my thesis, and Siddarth Sreeram, Prachi Sahoo, and Seth Burdette for help doing Human Motion Capture experiments. I also want to thank C.J. Adams, Arto Kivila, Andrew Leonard, Stefan Schneider, my roommates, friends, and family for your help and support.

TABLE OF CONTENTS

Acknowledgments	iii
List of Tables	viii
List of Figures	x
Summary	xiv
Chapter 1: Introduction and Background	1
1.1 Motivation and Objectives	1
1.1.1 Evaluating Clothing Fit	1
1.1.2 Motion Analysis and Injury Prevention	2
1.1.3 Low-Cost Human Tracking	3
1.2 Background	3
1.3 Thesis Contributions	5
1.4 Outline of Thesis	5
Chapter 2: Previous Human Body Modeling	7
2.1 Level of Detail of Human Body Models	7
2.1.1 Joint Type Classification	8
2.2 Lower Body Models	8

2.2.1	Walking	10
2.3	Upper Body Models	11
2.3.1	Torso Models	11
2.3.2	Human Arm Models	11
2.3.3	Upper Body Motions	12
Chapter 3: Kinematic Human Upper Body Model		14
3.1	Key Joints and Degrees of Freedom	14
3.2	Denavit-Hartenberg Parameters	15
3.3	Torso Model	16
3.4	Arm Model	17
3.5	Forward Kinematics	23
3.6	Inverse Kinematics	23
3.7	State Estimation Methods for Joint Tracking	24
3.7.1	State Space Models	24
3.7.2	Linear Kalman Filter	25
3.7.3	Extended Kalman Filter	27
Chapter 4: Motion Capture using a Single Kinect Sensor		28
4.1	Kinect Baseline Performance Evaluation	28
4.1.1	Tracking of a Robot Arm	28
4.1.2	Frequency Tests	35
4.2	Comparison of Kinect and Vicon Motion Capture Systems	37
4.2.1	Joints Tracked by Kinect	37

4.2.2	Vicon Marker Placement	38
4.3	Comparison of Kinect and Vicon Motion Capture for Upper Body Joint Angle Tracking	38
4.3.1	Instrumentation and Setup	40
4.3.2	Data Processing	41
4.3.3	Planar Motions	41
4.3.4	Non-planar Motion	44
4.3.5	Weighted Tests	46
4.3.6	Rotation Tests	47
4.3.7	Low Light Conditions	47
4.3.8	Summary	50
Chapter 5: Dual-Kinect Motion Capture		51
5.1	Dual-Kinect Motion Capture Process	51
5.2	Implementation Details	52
5.2.1	Hardware and Implementation Restrictions	53
5.2.2	Dual-Kinect Configuration	53
5.3	Sensor Calibration and Sensor Fusion	55
5.3.1	Calibration	55
5.3.2	Sensor Fusion	57
5.4	Linear Kalman Filter for Kinect Joint Tracking	58
5.4.1	Linear Kalman Filter Implementation	58
5.5	Extended Kalman Filter for Kinect Joint Tracking	59
5.5.1	Extended Kalman Filter Implementation	61

5.5.2	Handling Missing Data	62
5.6	Experimental Setup	62
5.6.1	Tracked Motions	62
5.6.2	Marker-based Tracking	63
5.6.3	Marker Trajectory Data Processing	63
5.7	Results and Comparison with Vicon Motion Capture	65
5.7.1	Linear Kalman Filter	65
5.7.2	Extended Kalman Filter	69
5.8	Tracking with Garments of Different Fit	72
5.9	Graphical User Interface for Real-Time Joint Tracking with Dual-Kinect	75
Chapter 6: Conclusion		78
6.1	Discussion	78
6.2	Future Work	80
Appendix A: Robot Tracking Trajectories		82
Appendix B: Plug-In-Gait Marker Placement		86
Appendix C: Plug-In-Gait Marker Identifiers and Location		91
Appendix D: Dual-Kinect Setup Test Trajectories		93
Appendix E: Tracking Accuracy of Motion Capture Experiments with Dual-Kinect		97
References		102

LIST OF TABLES

2.1	Synovial Joint Type Classification	9
3.1	DH Parameters for Torso Model	18
3.2	DH Parameters for the Left and Right Arm Model	20
3.3	Body Segment Lengths	21
3.4	Joint Motion Definitions	22
4.1	Mean absolute error (MAE), maximum absolute error, mean absolute deviation (MAD) and maximum absolute deviation for robot arm tracking experiments with Kinect. All values in mm.	35
4.2	Joints tracked by the Kinect v2 Sensor	38
4.3	Kinect Joints and corresponding Markers	40
4.4	RMSE (°) in Joint Angle Trajectories from Rotation Tests for two-handed Wave Motion and one-handed "Move Along" Signal for Left and Right (L, R) Elbow and Shoulder (E, S)	49
5.1	List of tested Dual-Kinect Configuration Variants	54
5.2	Mean Absolute Error (MAE) and Maximum Absolute Error (MAD) for LKF2 for Tracking the Position of the Left Wrist. All values in mm.	67
5.3	Mean Absolute Error (MAE) for all Filter Variants Averaged Over Ten Upper Body Joints	71
C.1	Marker Identifiers and Marker Placement for Plug-in-Gait model used for tracking with Vicon 3D Motion Capture System	92

E.1 Accuracy of the Different Filter Outputs for Setup 4 98

LIST OF FIGURES

1.1	Robot arm with elbow and shoulder angle	2
2.1	Example Lower Body Model used in Walking Simulations ¹	10
2.2	Example for Modeling of the Human Vertebral Column ²	12
2.3	Example of a Robotics Based Human Upper Body Model ³	13
3.1	DH Parameters and Link Frames	15
3.2	Human Torso Model - Joint Locations	17
3.3	Human Torso Model - Joint Angles and Reference Frames	18
3.4	Human Arm Model - Joint Locations	19
3.5	Human Left Arm Model - Joint Angles and Reference Frames	20
3.6	Human Upper Body Model - Body Segment Lengths	21
4.1	3-DoF Robot Arm.	29
4.2	Schematic Diagram of Robot Arm.	30
4.3	Labeled Experimental Setup.	30
4.4	Tracked Marker.	31
4.5	XYZ Coordinates of End Effector vs. Kinect Measurement for Trial 1.	32
4.6	X, Y, Z Coordinates of End Effector for Three Trials.	33
4.7	Error between Kinect Measurements and Robot Trajectory.	33

4.8	End Effector Trajectory in 3D Space.	34
4.9	Pendulum setup for varying suspension lengths	36
4.10	Theoretical and measured natural frequency for varying suspension lengths	36
4.11	Marker Placement for Measurements with a Single Kinect Sensor	39
4.12	Marker Placement on Test Subject	39
4.13	Experiment Setup: Vicon Cameras and Kinect v2 Sensor	40
4.14	Illustration of the planar motions performed in this study: (a)-(c) Two-handed wave, (d)-(f) Two-handed "slow down" signal.	42
4.15	Right wrist joint trajectory for two-handed wave motion.	43
4.16	Joint angle trajectories for two-handed wave motion.	43
4.17	Joint angle trajectories for two-handed "slow down" signal motion.	44
4.18	Illustration of the non-planar motion performed in this study: one-handed "move along" motion	45
4.19	Joint angle trajectories for one-handed "move along" signal motion.	46
4.20	Right shoulder joint angle trajectories weighted tests.	46
4.21	Joint angle trajectories for two-handed wave motion for different orientations towards Kinect: (a)-(d) Right Shoulder and (e)-(h) Right Elbow	48
4.22	Left Elbow joint angle during two-handed wave motion in low light conditions	50
5.1	The Dual-Kinect Motion Capture Process	52
5.2	Different Kinect Configurations Tested in this Study	54
5.3	Chosen Dual-Kinect Configuration	55
5.4	Marker Placement for Full Body Plug-in-Gait Model	64
5.5	Test Subject Standing in T-Pose and Dual-Kinect Test Setup	64
5.6	Marker Placement for Full Body Plug-in-Gait Model on Test Subject	65

5.7	Z Component of the Left Wrist Joint Trajectory with Sensor Fusion and Linear Kalman Filter with Constant Velocity Model ($Q = 0.005$)	66
5.8	Error Between Kinect 1 and 2 and LKF2 Outputs for the Z Component of the Left Wrist Joint Trajectory	67
5.9	Left Wrist Joint Trajectory with Sensor Fusion and LKF2 ($Q = 0.005$) - Comparison with Vicon Motion Capture Data	68
5.10	Error Between the LKF2 Output and the Vicon Data for the Left Wrist Position	68
5.11	Z Component of the Left Wrist Joint Trajectory with Sensor Fusion and Extended Kalman Filter ($Q = 0.0001$)	69
5.12	Left Wrist Joint Trajectory from EKF - Comparison with Vicon and LKF2 Outputs	70
5.13	Error Between Vicon and LKF2, and Vicon and EKF Outputs for the Left Wrist	70
5.14	Length of the Left Arm during Motion Capture Trial	72
5.15	Test Subject wearing a tight-fitting Motion Capture Suit with retroreflective Markers	73
5.16	Test Subject performing Test Motion with Tight-Fitting (left) and Loose-Fitting Clothes (right)	74
5.17	SpineBase Trajectory for Test Motions with Tight-Fitting and Loose-Fitting Clothes	75
5.18	GUI for Real-Time Tracking with the Dual-Kinect System	76
5.19	Real-time Tracking Results with Dual-Kinect and Linear Kalman Filter: (a)-(c) Torso Twist, (d)-(f) Wave Motion	77
A.1	XYZ Coordinates of End Effector vs. Kinect Measurement for Trial 2. . . .	83
A.2	XYZ Coordinates of End Effector vs. Kinect Measurement for Trial 3. . . .	84
A.3	XYZ Coordinates of End Effector vs. averaged Kinect Measurement (Trials 1-3).	85

D.1 Fused Left Wrist Trajectory for Dual-Kinect Setup 1-4	94
D.1 Fused Left Wrist Trajectory for Dual-Kinect Setup 5-6	95
D.2 Fused Left Wrist Trajectory for Dual-Kinect Setup 7-9	96

SUMMARY

In this thesis a kinematic human upper body model for evaluating clothing fit and appearance is developed and validated. Realistic and accurate human body models are required in many different application areas, including medicine, computer graphics, biomechanics, and sport science. A particular application of interest for a human body model is a virtual reality clothing model to evaluate fit and appearance of garments.

A robotics-based model for the human upper body skeleton is derived. To validate the model, upper body motion data is collected with a markerless motion capture system using Microsoft Kinect. A baseline evaluation of markerless motion capture with a single Kinect sensor presents results from tracking a robot arm trajectory, frequency tests, and human motion capture experiments. Because occlusion causes a single Kinect sensor to fail in accurately predicting the human pose, a second Kinect sensor is integrated into the system. Data from the two sensors is fused and filtered using an Extended Kalman filter. The results are compared to marker-based tracking with a Vicon Motion Capture system.

The Extended Kalman filter is shown to ensure constant body segment lengths, thus producing a more realistic estimation of the joint positions than obtained from the raw Kinect data. The proposed setup offers a low-cost, markerless, and portable alternative to marker based motion tracking.

CHAPTER 1

INTRODUCTION AND BACKGROUND

The objective of this master thesis is to develop and validate a kinematic human upper body model for evaluating clothing fit and appearance. Realistic and accurate human body models are required in many different application areas, including medicine, computer graphics, biomechanics, and sport science. A particular application of interest for a human body model is a virtual reality clothing model to evaluate fit and appearance of garments. In order to accurately evaluate clothing, a human body model that can produce realistic human motions is required.

The next section contains a brief motivation for the research and the objectives of this thesis. Section 1.2 provides a description of the background and scope of work, including an overview of human motion capture technologies, their capabilities and limitations. In Section 1.3 the contributions of this thesis are described. Finally, Section 1.4 outlines the remaining chapters of this thesis.

1.1 Motivation and Objectives

1.1.1 Evaluating Clothing Fit

Clothing fit is considered to be the most important criterion for customers to evaluate clothing appearance. There is no clear definition of the quality of clothing fit [1]. However, psychological comfort, appearance and physical dimensional fit do contribute to the customer's perceived satisfaction of fit. To assess dimensional fit of a garment, dress forms and 3D body scanning systems are currently used [1, 2]. These methods can reliably evaluate the fit in static poses, but they cannot be used to quickly and accurately assess the quality of fit or change of appearance of a wide range of garments during dynamic poses e.g. walking.

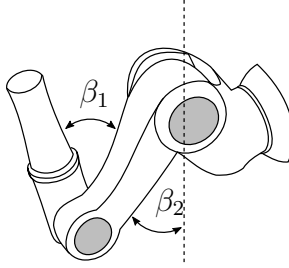


Figure 1.1: Robot arm with elbow and shoulder angle

In recent decades, human body and motion modeling has received increasing attention, with applications in computer vision, virtual reality and sports science. To date, synthesis of realistic human motions remains a challenge in biomechanics. While clothing simulation is usually accomplished using finite element analysis [3], evaluation of clothing fit on a real human body performing motions requires a kinematic model capable of predicting realistic human-like motion.

1.1.2 Motion Analysis and Injury Prevention

Work related musculoskeletal disorders (WRMSDs) are a major issue plaguing factory workers, traffic policemen, and others who routinely perform significant upper-body motions [4, 5]. Muscular fatigue is induced due to long working hours, as well as incorrect or sub-optimal motion techniques [6]. Assessment of the range of motion (ROM) of a human joint can yield information about the use, injury, disease, extendability of tendons, ligaments and muscles [7].

An additional area of interest is the derivation of joint angle trajectories from motion capture data collected from humans in an experimental setting. Such trajectories can, for example, be used to drive a robot through motions that mimic human arm movements. An example for such a robot is shown in Figure 1.1, where the shoulder and elbow angles β_1 and β_2 are used to drive the robot.

1.1.3 Low-Cost Human Tracking

While many established optical motion capture systems involve multiple high definition cameras and have been proven to be accurate, they are often expensive and infeasible to use outside the confined space in which they are installed. On the other hand, the Microsoft Kinect sensor is a non-invasive, low-cost camera primarily used in the video-gaming industry which can be used to track 25 joints of a human skeleton. The sensor provides RGB, depth, and infrared data.

Numerous studies have been presented evaluating the accuracy of skeleton and joint tracking using the first version of the Kinect sensor [8, 9, 10]. Motion capture of upper-body movements using the Kinect compared to a marker-based system has been studied and compared to established optical motion capture methods with respect to applications in ergonomics, rehabilitation, and postural control [11, 12, 13]. Overall, these studies found that the Kinect's precision is less than the optical motion capture system, yet the Kinect has various advantages such as portability, markerless motion capture, and price. To improve the Kinect's motion capture precision, some approaches used additional wearable inertial sensors [14]. With this approach, more accurate joint angle measurements were obtained.

1.2 Background

To further understand the foundation of this thesis, the background and scope of the work must be considered. This includes a thorough analysis of the available human motion capture tools to assess their capabilities and limitations. The most common approach is to model the human body as a serial multibody system, in which the rigid or flexible bodies (limbs) are connected via joints.

In order to produce realistic and natural human-like motions, one needs to understand the basic concept of the human structural system and the major movable joints in the real human body. The human musculoskeletal system consists of the bones of the skeleton,

cartilage, muscles, ligaments, and tendons. The human skeleton consists of more than 200 bones [7] driven by over 250 muscles, which introduces a great number of degrees of freedom (DoF) into human body models. Different techniques such as physics-based simulation [15], finite element analysis [16], and robotic-based methods [17] have been employed with the goal of modeling realistic human motion.

The suitability of an existing model and the derived human-like motions can be evaluated by comparing with human motion capture systems. The most commonly used motion capture systems are vision-based. These system can be divided into marker-based and markerless systems. While marker-based systems such as OptiTrack or Vicon use multiple cameras to track the positions of reflective markers attached to a human test subject, markerless systems such as the Microsoft Kinect sensor estimate a human pose and joint position based on a depth map acquired with infrared or time-of-flight sensors.

Marker-based systems are widely used and have been established to be fairly accurate [18]. In contrast, markerless systems use position estimation algorithms that introduce error into the measurements. Because most markerless systems have a single camera, only one point of view is available. Occlusion of limbs or movement out of the camera view can cause the pose estimation to fail. While marker-based systems are costly and confined to a certain volumetric workspace, markerless systems are more affordable and can easily be used in many different settings. To reduce occlusion problems in Kinect measurements, motion capture setups involving multiple Kinect sensors have been developed [19, 20].

Vicon 3D Motion Capture systems involve multiple high definition cameras which are accurate, but expensive, and infeasible to use in shopping malls, airports, aircraft carriers, road settings, etc.. On the other hand, the Kinect can be used for human-body motion analysis in a wide variety of settings. The primary differentiating factor between the Kinect and Vicon system is the necessity of retro-reflective markers in the Vicon system. Light from the Vicon cameras is emitted and is reflected back from markers in the field of view. This yields the 3D position of each marker. However, the Kinect does not require markers

for human-body tracking because a proprietary Microsoft software possesses the ability to track human body joints.

1.3 Thesis Contributions

This thesis addresses the development of a new kinematic model for the human body in three dimensions with the goal of evaluating the fit and appearance of clothing. A robotics-based model for the human upper body skeleton is derived. To mimic realistic human movements, motion data is collected using a markerless motion capture system (Microsoft Kinect) that tracks 25 joints of the human skeleton. Because occlusion causes a single Kinect sensor to fail in accurately estimating the human pose, a second Kinect sensor is integrated into the system, creating a dual-Kinect system. Different sensor orientations are tested to find the best relative orientation between the two Kinect sensors and a human test subject. Data acquired from both sensors is calibrated and fused using a weighted average calculation. Joint tracking performance is subsequently enhanced using two different approaches: in the first approach, the tracked positions are filtered using a linear Kalman filter. In the second approach, the positions are fed into an extended Kalman filter based on the novel kinematic human upper body model. This approach ensures constant limb lengths during the tracked motions. To investigate how the fit of clothing affects tracking, experiments with a human test subject wearing garments of different fit are conducted. The accuracy of the dual-Kinect system is evaluated by comparison with a marker-based Vicon motion capture system.

1.4 Outline of Thesis

This thesis consists of six chapters. The first chapter covered the background, objectives, and motivation to develop a kinematic human upper body model to evaluate clothing fit. Chapter Two presents an overview of previous work done on human motion modeling and is divided into lower body, torso, and upper body models. The proposed kinematic hu-

man body model and the framework used for improved joint tracking is laid out in Chapter Three. Chapter Four covers an evaluation of the baseline performance of depth measurements obtained with the Kinect, and human motion capture with a single Kinect sensor. Chapter Five describes the improved human motion capture with a dual-Kinect system. It presents a method for fusing the sensor data and demonstrates improvements in accuracy with the employed method by comparing with a Vicon motion capture system. Finally, Chapter Six discusses the results of the motion capture experiments, tracking performance of the different filters with fused data, other significant findings, and future work.

CHAPTER 2

PREVIOUS HUMAN BODY MODELING

Vision based studies of human motion can be classified into 'model-based' methods that use *a priori* shape models of the human, and 'appearance-based' or 'view-based' methods [21]. Appearance-based methods first detect features in an image, and subsequently build a human body representation, whereas model-based approaches fit image data to a predefined human body model. While appearance-based approaches do not employ an object model, and thus can be used in more diverse situations, they are, in general, sensitive to noise e.g. clothes or other objects in the frame. In contrast, model-based methods can integrate knowledge about the shape into visual inputs, and can be used for high-level classification of motions. However, additional processing steps, e.g. parameter estimation, are usually required.

This chapter presents a review of model-based methods used in human motion analysis. The reviewed models are grouped into lower body models and upper body models. While lower body models include part of the trunk, the pelvis, the hips and the legs, upper body models are often a combination of torso and arm models. The structure of the models, as well as their application in human motion motion tracking and prediction are reviewed. Lower body models are mainly used in gait analysis, while torso and upper body models have been used in investigation of broader applications.

2.1 Level of Detail of Human Body Models

Depending on the complexity required in an application, the human body can be represented at different levels of detail, from bounding boxes and stick figures to 3D volumetric models that include muscle and soft tissue dynamics [15, 21]. Comprehensive human body models can be very detailed and often contain multiple layers e.g. a skeleton layer account-

ing for the articulated motion of the bones caused by muscle contraction, and a layer for soft tissue deformation.

Because human motion is essentially caused by movement of the supporting bones of the musculoskeletal system, more general representations treat the human body as a combination of rigid body segments (links) and the joints connecting them. Many human body models further decompose the anatomical human joints into combinations of single-DoF prismatic or revolute joints commonly used for the description of serial robot manipulators [17]. To describe and analyze the kinematics and dynamics of such models, a robotics based approach can be used.

2.1.1 Joint Type Classification

Anatomical human joints can be defined as the union of two or more bones, and as the region in which motion occurs [22, 23]. There are many joint classifications available. Because the focus of this thesis lies on modeling joint motion, a classification based on joint mobility is presented here [22, 23, 24]: fibrous, cartilaginous, and synovial. While fibrous and cartilaginous joints provide stability or a limited amount of mobility (e.g. the sutures of the skull or the intervertebral joints of the spine), synovial joints allow for a large degree of motion (e.g. the shoulder and hip joints) [7, 23]. Commonly, six different types of synovial joints can be distinguished. Table 2.1 lists these joint types with a description of the allowed motion and examples [25].

2.2 Lower Body Models

The lower body is generally comprised of trunk and pelvis segments, and two legs. Motion of the lower limbs is enabled through the hip, knee and ankle joints. A common approach is to consider the lower body as a mechanical linkage or as a multibody system with defined links or segments interconnected by joints [26]. The motion of the links is generally driven by joint rotations.

Table 2.1: Synovial Joint Type Classification

Name	Allowed motion
Plane joint	Sliding or gliding movement of two bones e.g. Acromioclavicular joint
Hinge joint	Flexion and extension in one plane e.g. Elbow (humero-ulnar) joint
Pivot joint	Rotation of one bone about the other e.g. Atlanto-axial joint
Condylar joint	Movement around two perpendicular axes e.g. Wrist joint
Saddle joint	Permit the same movements as condylar joints; the articular surfaces are saddle shaped e.g. Carpometacarpal joint of thumb
Ball-and-socket joint	All movements except gliding e.g. Shoulder (glenohumeral) joint, hip joints

In a review of physics-based modeling of human walking, Xiang et al. [27] present a classification of lower body models based on geometry of planar and spatial models. The mechanical models for the human body can be further divided into skeletal models, where a muscle group driving a joint is lumped and represented as joint torque, and musculoskeletal models, which consider the motions and forces at the muscular level. Some approaches further include modeling of muscle activation patterns, and the interaction between the nervous system and muscles used to produce a coordinated motion [28]. Figure 2.1 shows an example of a lower body model with 23 DoF and 10 segments. The effect of the upper body mass on the lower body model is accounted for by lumping the head, arms, and torso into a single segment.

The most common use of lower body models is to investigate walking, one of the most fundamental human motions, yet complex in terms of neural control and stability [27]. Other previously investigated motions include jumping and pedaling [28].

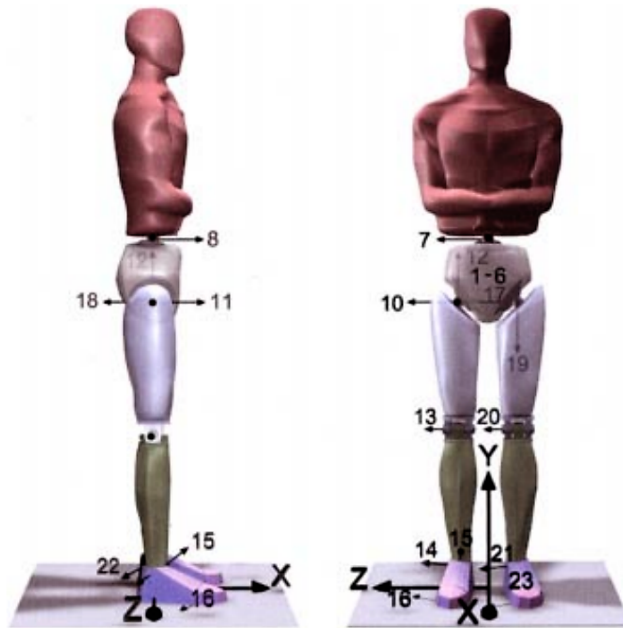


Figure 2.1: Example Lower Body Model used in Walking Simulations¹

2.2.1 Walking

Modeling and simulation of human walking is an important field of study in both robotics and biomechanics. Knowledge about gait mechanisms can be applied to real-time control of biped robots, clinical biomechanics, or pathological gait analysis [27]. A variety of methods for modeling and simulating of human walking motion have been developed. In this section, a simplified model (inverted pendulum model), and a more complicated method (optimization method) are discussed.

Inverted Pendulum Model

Because walking involves a periodic exchange of kinetic and potential energy, a common simplified approach in gait modeling is based on a planar inverted pendulum, treating the body mass as concentrated mass located at the center of gravity (COG). With this approach, trajectories for the COG can be generated in closed form. However, due to the simplicity

¹Image from [26]

of the model, natural and realistic human walking motion is difficult to recreate [29].

Optimization Method

Optimization-based motion prediction offers a more complex approach to analyze and simulate human motions. This method uses large-DoF models and optimizes joint angle and torque trajectories according to a predefined human performance measure. Commonly used performance measures include mechanical energy, metabolic energy, and stability [27].

2.3 Upper Body Models

The upper body consists of the torso, head, and arms. Models of the upper body usually have some degree of redundancy and use optimization techniques to define the pose. Lura [17] proposed a robotics based upper body model for predictive simulation of upper body motions, which was used to evaluate prosthetics performance. Lee et al. [15] introduced a comprehensive biomechanical model that takes into account, more or less, all the relevant articular bones of the human upper body. The model used by Lee includes physics-based simulation of soft-tissue deformations.

2.3.1 Torso Models

Motion of the torso is mainly attributed to the spine, which consists of vertebrae separated by vertebral discs. Often, parts of the spine are grouped and treated as rigid bodies interconnected by joints. This approach simplifies the motion within the corresponding area [7]. Figure 2.2 shows an example of this approach in which torso motion is modeled using 9 joints.

2.3.2 Human Arm Models

Using a robotics based approach, the human arm can be modeled as a series of rigid links connected by multiple joints. These joints represent the anatomical joints (shoulder joint,

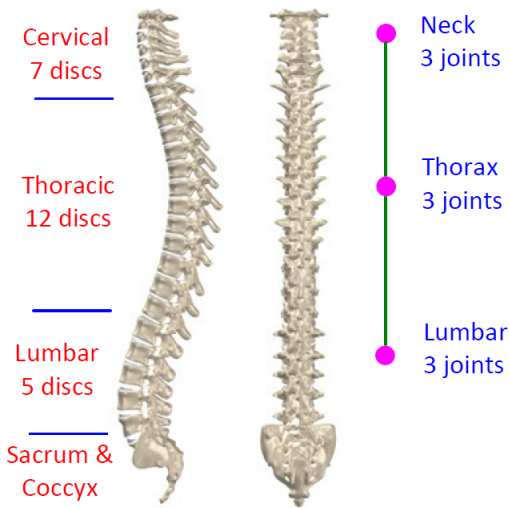


Figure 2.2: Example for Modeling of the Human Vertebral Column²

elbow joint, and wrist joint) [30]. Models of the human arm commonly have up to 7 DoF, assuming the origin of the base frame is fixed and located at the shoulder joint. The anatomical shoulder joint itself is then modeled as three revolute joints with intersecting orthogonal axes. The elbow joint is modeled as either a single or two serial revolute joints, and the wrist joint usually has two or three DoF [17].

To resolve the redundancy in the inverse kinematics of the 7 DoF model, a common solution is to reduce the model to a 6 DoF model, allowing for a purely analytical solution. This has been achieved by optimizing the “swivel angle” of the elbow [30] or by minimizing the upper arm elevation. Figure 2.3 shows an example of a robotics based model for the human upper body that uses rigid links and joints to represent body segments.

2.3.3 Upper Body Motions

Previously studied motions of the upper body include hand tool use in ergonomic work site analysis [4], ball throwing [7], dumbbell curls, and respiratory torso movement [15]. Lura [17] used a robotics-based upper body model to investigate several range of motion (RoM) tasks, as well as activities of daily living (ADL), e.g. drinking from a cup, or

²Image from [7]

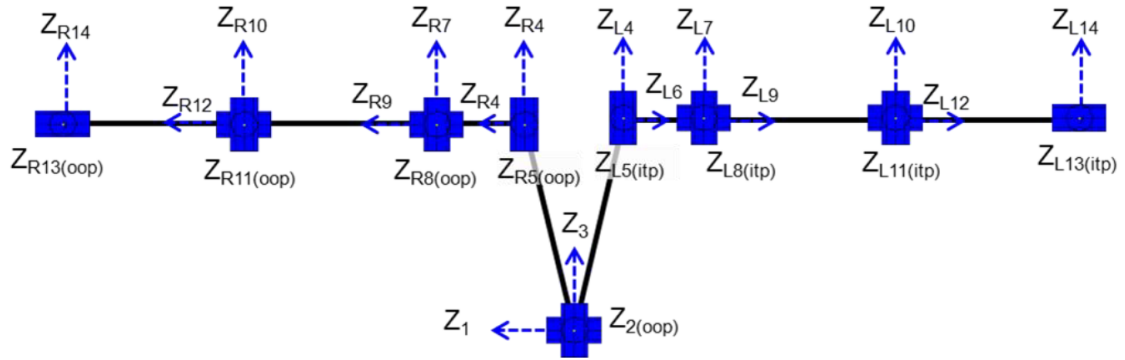


Figure 2.3: Example of a Robotics Based Human Upper Body Model³

opening a door.

³Image from [17]

CHAPTER 3

KINEMATIC HUMAN UPPER BODY MODEL

In this chapter, a novel kinematic human upper body model is presented. The human upper body is modeled as a series of links that are connected by joints. In order to employ a robotics-based framework, the anatomical joints are decomposed into a series of revolute, single DoF joints.

This chapter is structured as follows. First, the key joints and degrees of freedom necessary to adequately model the human upper body are discussed. Then, a robotics-based framework using Denavit-Hartenberg (DH) parameters is provided and the kinematic models for the torso and arms are explained in detail. Forward kinematics and inverse kinematics for general serial link manipulators are discussed. Finally, a linear Kalman filter and an extended Kalman filter incorporating the described upper body model for joint tracking with a Kinect sensor are developed.

3.1 Key Joints and Degrees of Freedom

In order to develop a kinematic model, one needs to understand the major movable joints of the real human body. The upper body can be divided into a torso segment, a head segment including the neck, and the arms. Because the intended application of the model is to evaluate clothing fit not including hats, the head segment is neglected in the modeling process.

Motion of the torso segment arises mainly from the vertebral column or spine, which consists of multiple discs. To sufficiently model the mobility of the spine, but at the same time limit the degrees of freedom, the spine can be divided into three regions: a lower region (sacrum and coccyx), a middle region (chest or thoracic region), and an upper region (located approximately at the sternum). The movable parts in each of these regions will be

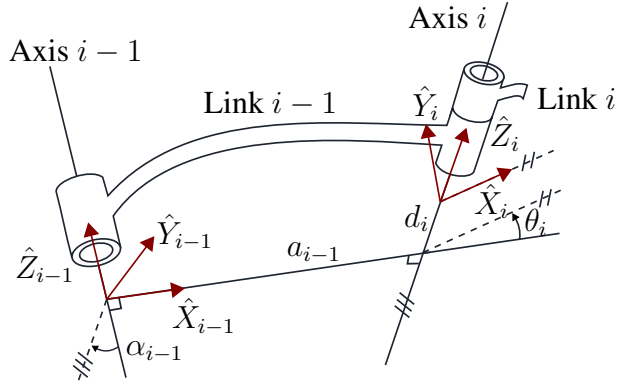


Figure 3.1: DH Parameters and Link Frames

modeled as a 3-DoF universal joint, enabling 3-axis motion.

The major joints of the human arm are located in the shoulder, elbow, and wrist. Shoulder motion is achieved through the shoulder complex, which consists of 20 muscles, three functional joints and three bony articulations. However, the term “shoulder joint” usually refers to only one particular joint, the glenohumeral joint, which is a ball-and-socket-type joint [23]. Usually only the shoulder joint is considered in models of anthropometric arms [30]. It is commonly modeled as a 3-DoF universal joint, which is sufficient to enable 3-axis motion of the upper arm. The elbow and wrist joints are each modeled with two DoF.

Using a robotics-based approach to modeling the human upper body, the rotation of each body segment is defined by joint angles θ_i , $i = 1 \dots n$, where n is the number of single-DoF joints in the complete model. The orientation and position of the links in the kinematic chain can then be expressed using Denavit-Hartenberg parameters.

3.2 Denavit-Hartenberg Parameters

In order to describe the spatial configuration of a serial robot, Denavit-Hartenberg (DH) parameters [31] are commonly used. Each joint i is assigned a frame O with location p . Figure 3.1 shows the relation between DH parameters and frames $i - 1$ and i for a segment of a general manipulator. d_i is the distance from O_{i-1} to O_i , measured along Z_{i-1} . a_i is the

distance from Z_i to Z_{i+1} , measured along X_i . θ_i is the joint angle between X_{i-1} and X_i , measured about Z_i . α_i is the angle between Z_i and Z_{i+1} , measured about X_i . A 4×4 homogeneous transformation matrix can be used to transform frame i to $i + 1$:

$$T_i^{i+1} = \begin{bmatrix} \cos \theta_i & -\sin \theta_i \cos \alpha_i & \sin \theta_i \sin \alpha_i & a_i \cos \theta_i \\ \sin \theta_i & \cos \theta_i \cos \alpha_i & -\cos \theta_i \sin \alpha_i & a_i \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.1)$$

with joint angle θ_i , link twist α_i , link length a_i and link offset d_i .

It should be noted that multiple options for the placement of the coordinate frames generally exist. In the following sections, the major anatomical joints of the upper body are decomposed into single-DoF revolute joints and the DH parameters for the torso and arm model are derived.

3.3 Torso Model

The torso is modeled as a tree-structured chain composed of four rigid links: one link from the base of the spine to the spine midpoint, one link from the spine midpoint to the spine at the shoulder, approximately located at the sternum, and two links connecting spine at the shoulder to the left and right shoulder. The corresponding joints in the torso model will be referenced to as "SpineBase", "SpineMid", and "SpineShoulder", with the SpineShoulder connecting to the "ShoulderLeft" and "ShoulderRight". Figure 3.2 shows the locations of these joints in the human body.

Because we are only considering movement in the upper body, the base of the spine is assumed to be fixed in space. The lower spine region is considered as a universal joint that can be modeled as three independent, single-DoF revolute joints with intersecting orthogonal axes. The corresponding joint angles are θ_1 , θ_2 , and θ_3 . The same approach is taken to model motion in the mid region of the spine. The SpineMid enables the torso to rotate and

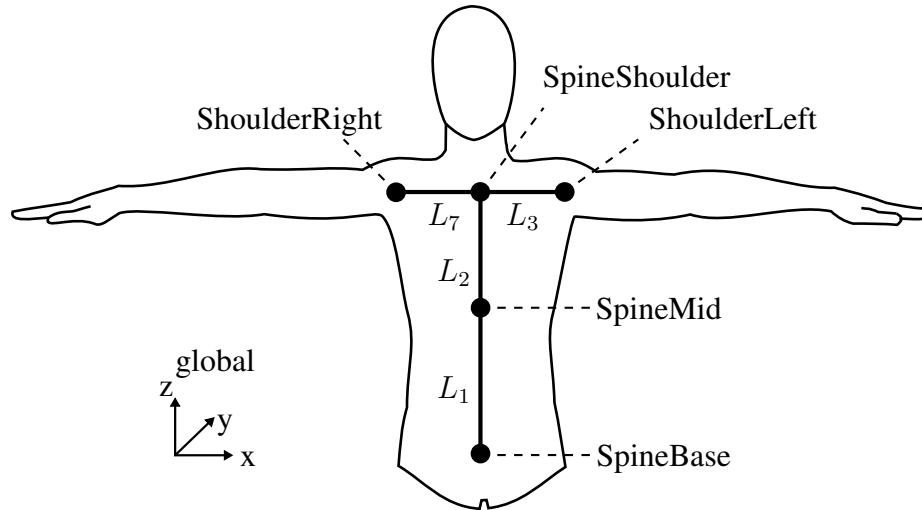


Figure 3.2: Human Torso Model - Joint Locations

bend about three axes with joint angles θ_4 , θ_5 , and θ_6 . At the SpineShoulder, the kinematic chain is split into two branches, allowing for independent motion of both shoulder joints relative to the sternum. For each branch, the shoulder joint is modeled as three independent, single-DoF revolute joints. The link connecting the SpineShoulder with the ShoulderLeft can be moved with joint angles θ_7 , θ_8 , and θ_9 , while the right link can be moved with θ_{10} , θ_{11} , and θ_{12} , respectively.

In summary, the complete torso model consists of four rigid links, interconnected by 12 single-DoF revolute joints. Using the DH conventions, coordinate systems and corresponding DH parameters are assigned to each joint. Figure 3.3 shows the coordinate frames assigned to the joints and the joint angles. The corresponding DH parameters for the torso model are listed in Table 3.1. Provided the link lengths L_1, L_2, L_3 and L_7 , and the 12 joint angles $\theta_1, \theta_2, \dots, \theta_{12}$, the spatial configuration of the torso model is completely defined.

3.4 Arm Model

Each arm is modeled as a serial kinematic chain consisting of three links: one link from the shoulder joint to the elbow joint, one from elbow to the wrist and one link from the wrist to the tip of the hand. The corresponding link lengths are L_4 , L_5 , and L_6 for the left

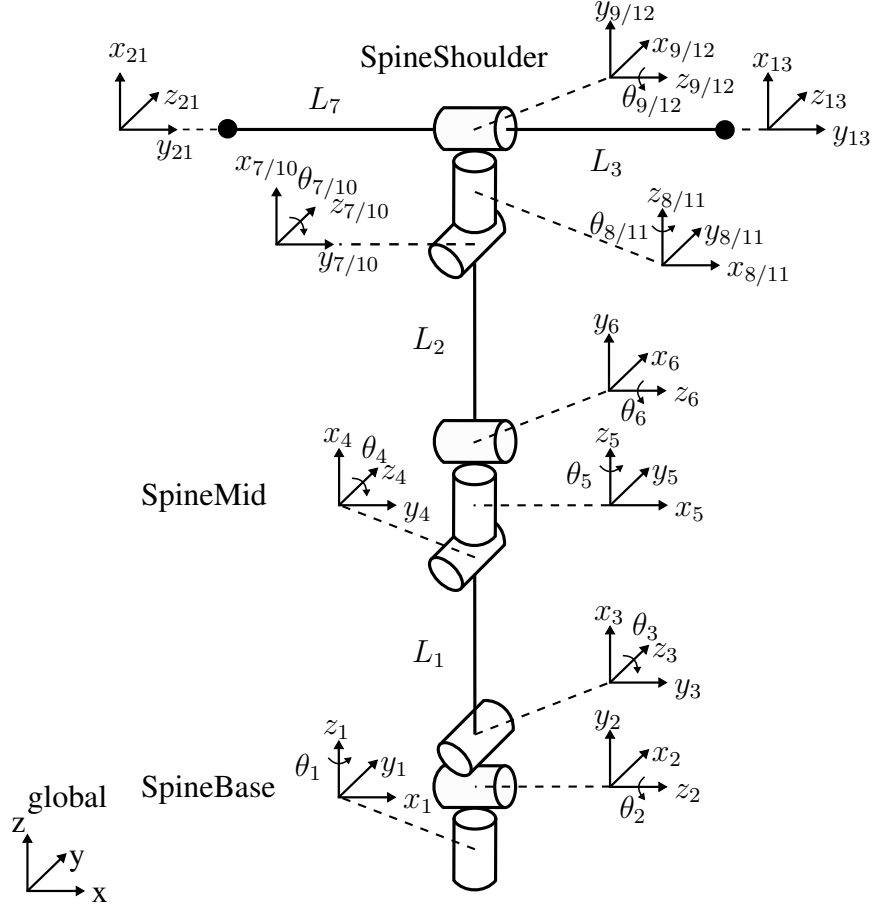


Figure 3.3: Human Torso Model - Joint Angles and Reference Frames

Table 3.1: DH Parameters for Torso Model

i	Joint	θ_i	d_i	a_i	α_i
1	SpineBase Z	$\theta_1 + \pi/2$	0	0	$\pi/2$
2	SpineBase X	$\theta_2 + \pi/2$	0	0	$\pi/2$
3	SpineBase Y	θ_3	0	L_1	0
4	SpineMid Y	$\theta_4 + \pi/2$	0	0	$\pi/2$
5	SpineMid Z	$\theta_5 + \pi/2$	0	0	$\pi/2$
6	SpineMid X	$\theta_6 + \pi/2$	0	L_2	$\pi/2$
7	SpineShoulder Y (left)	$\theta_7 + \pi/2$	0	0	$\pi/2$
8	SpineShoulder Z (left)	$\theta_8 + \pi/2$	0	0	$\pi/2$
9	SpineShoulder X (left)	$\theta_9 + \pi/2$	L_3	0	$\pi/2$
10	SpineShoulder Y (right)	$\theta_{10} + \pi/2$	0	0	$\pi/2$
11	SpineShoulder Z (right)	$\theta_{11} + \pi/2$	0	0	$\pi/2$
12	SpineShoulder X (right)	$\theta_{12} + \pi/2$	$-L_7$	0	$\pi/2$

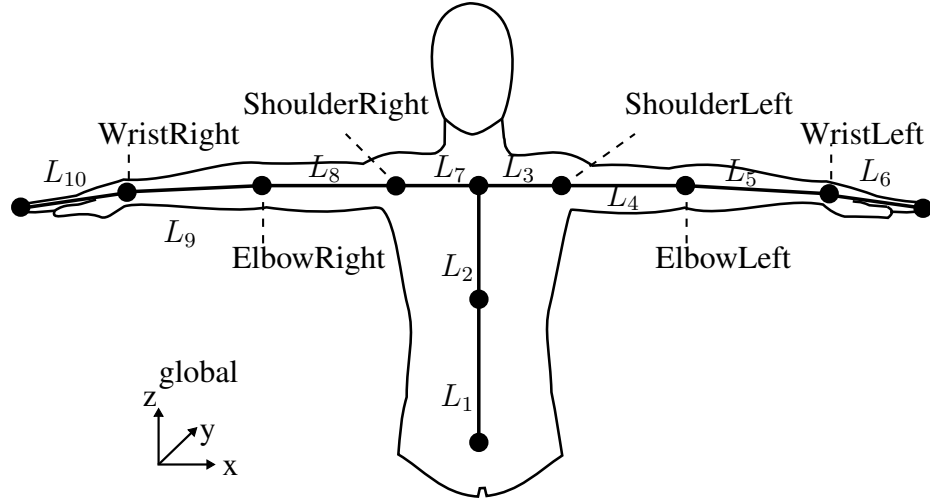


Figure 3.4: Human Arm Model - Joint Locations

arm, and L_8 , L_9 , and L_{10} for the right arm. The joints considered here will be referenced to as “ShoulderLeft”, “ElbowLeft”, and “WristLeft”, and “ShoulderRight”, “ElbowRight”, and “WristRight”, respectively. Figure 3.4 shows the location of these joints in the body. The anatomical shoulder joint is modeled as a universal joint, providing three DoFs for the rotation of the upper arm. The left (right) shoulder joint can therefore be modeled as three independent, single-DoF revolute joints with intersecting orthogonal axes with joint angles θ_{13} , θ_{14} , and θ_{15} (right: θ_{20} , θ_{21} , and θ_{22}). The elbow is modeled as two single-DoF revolute joints with joint angles θ_{16} and θ_{17} (right: θ_{23} and θ_{24}). The wrist is modeled as two single-DoF revolute joints with joint angles θ_{18} and θ_{19} (right: θ_{25} and θ_{26}).

Figure 3.5 shows the coordinate frames and joint angles for the left arm model. The corresponding DH parameters for the left and right arm model are listed in Table 3.2. Adding up the DoF for the shoulder, elbow, and wrist, each arm model has 7 DoFs.

Because only 6 DoF are required to define the position and orientation of the end-effector (tip of the hand), it follows that the human arm model is redundant. Redundancy is defined as the number of joints exceeding the output degrees of freedom [7]. For the human arm, this redundancy can be observed by, first, fixing the positions of the shoulder and wrist in space. Then allow the elbow to move without moving the shoulder or wrist

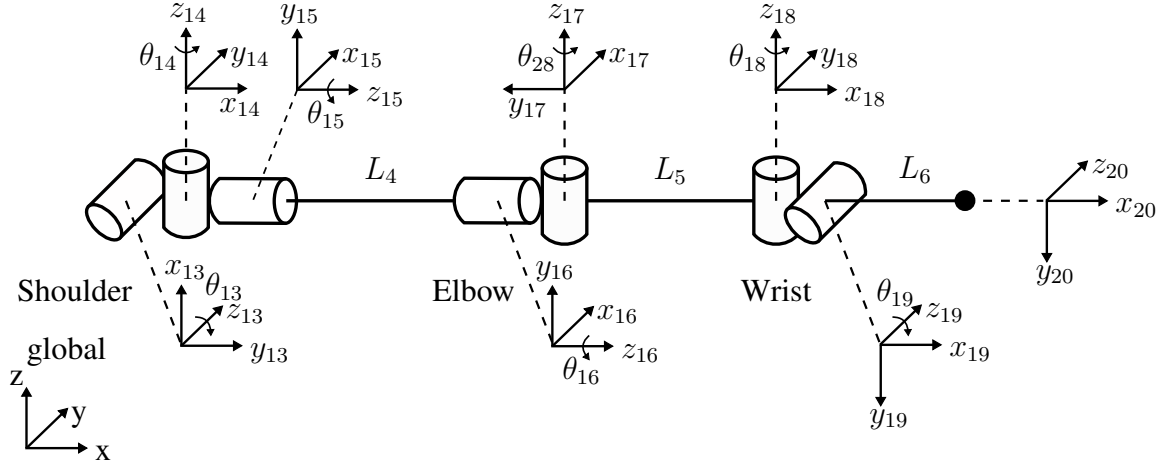


Figure 3.5: Human Left Arm Model - Joint Angles and Reference Frames

Table 3.2: DH Parameters for the Left and Right Arm Model

i	Joint	θ_i	d_i	a_i	α_i
13	ShoulderLeft Y	$\theta_{13} + \pi/2$	0	0	$\pi/2$
14	ShoulderLeft Z	$\theta_{14} + \pi/2$	0	0	$\pi/2$
15	ShoulderLeft X	θ_{15}	L_4	0	0
16	ElbowLeft X	θ_{16}	0	0	$-\pi/2$
17	ElbowLeft Z	$\theta_{17} - \pi/2$	0	L_5	0
18	WristLeft Z	θ_{18}	0	0	$-\pi/2$
19	WristLeft Y	θ_{19}	0	L_6	0
20	ShoulderRight Y	$\theta_{20} + \pi/2$	0	0	$\pi/2$
21	ShoulderRight Z	$\theta_{21} + \pi/2$	0	0	$\pi/2$
22	ShoulderRight X	θ_{22}	$-L_8$	0	0
23	ElbowRight X	θ_{23}	0	0	$-\pi/2$
24	ElbowRight Z	$\theta_{24} - \pi/2$	0	$-L_9$	0
25	WristRight Z	θ_{25}	0	0	$-\pi/2$
26	WristRight Y	θ_{26}	0	$-L_{10}$	0

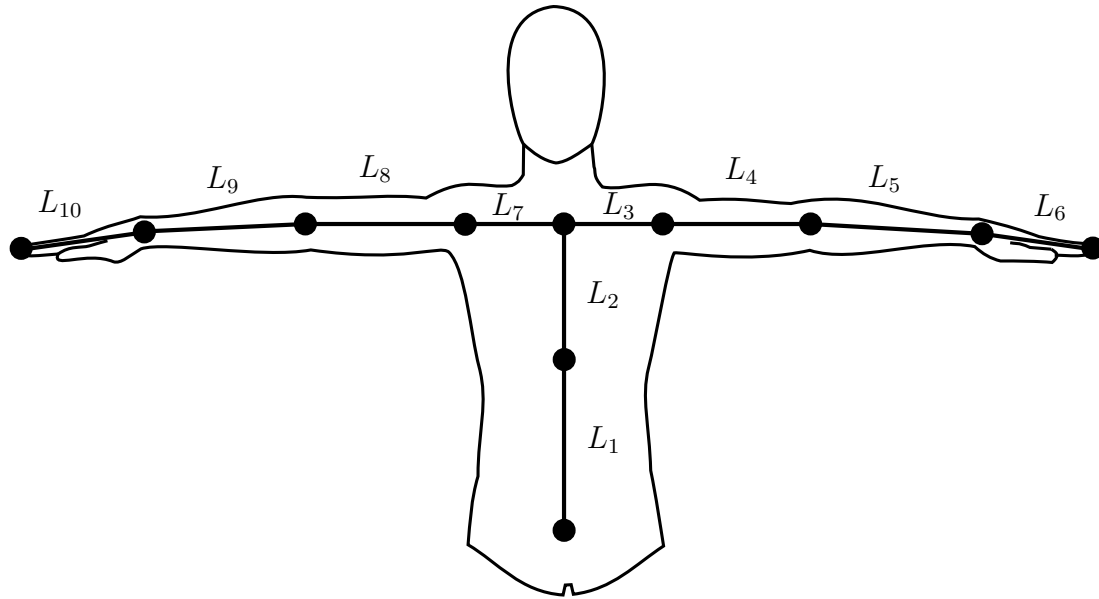


Figure 3.6: Human Upper Body Model - Body Segment Lengths

Table 3.3: Body Segment Lengths

L_i	Body Segment	L_i	Body Segment
L_1	Lower Torso	L_6	Left Palm
L_2	Upper Torso	L_7	Right Clavicle
L_3	Left Clavicle	L_8	Right Upper Arm
L_4	Left Upper Arm	L_9	Right Forearm
L_5	Left Forearm	L_{10}	Right Palm

position. Combining the torso and arm model further increases redundancy, making the upper body model a highly redundant system.

Offsets in the joint angles θ_i are introduced to place the upper body model in the rest position with both arms fully extended to the sides (=T-Pose), shown in Figures 3.2 and 3.4, when $\theta_i = 0$ for $i = 1, \dots, 26$. The body segment lengths for the upper body model are shown in Figure 3.6. Table 3.3 lists the names of the corresponding segments. Table 3.4 gives an overview of the biomechanical motions provided by each joint angle.

Table 3.4: Joint Motion Definitions

i	Joint Name	Joint Angle	Motion
1	SpineBase Z	θ_1	Lumbar Flexion/Extension
2	SpineBase Y	θ_2	Lumbar Lateral
3	SpineBase X	θ_3	Lumbar Rotation
4	SpineMid Y	θ_4	Thoracic Lateral
5	SpineMid Z	θ_5	Thoracic Flexion/Extension
6	SpineMid X	θ_6	Thorax Rotation
7	SpineShoulder (left) Y	θ_7	Left Clavicle Elevation/Depression
8	SpineShoulder (left) Z	θ_8	Left Clavicle Flexion/Extension
9	SpineShoulder (left) X	θ_9	Left Clavicle Rotation
10	SpineShoulder (right) Y	θ_{10}	Right Clavicle Elevation/Depression
11	SpineShoulder (right) Z	θ_{11}	Right Clavicle Flexion/Extension
12	SpineShoulder (right) X	θ_{12}	Right Clavicle Rotation
13	ShoulderLeft Y	θ_{13}	Left Shoulder Abduction/Adduction
14	ShoulderLeft Z	θ_{14}	Left Shoulder Flexion/Extension
15	ShoulderLeft X	θ_{15}	Left Shoulder Medial/Lateral Rotation
16	ElbowLeft X	θ_{16}	Left Elbow Pronation/Supination
17	ElbowLeft Z	θ_{17}	Left Elbow Flexion/Extension
18	WristLeft Z	θ_{18}	Left Wrist Radial/Ulnar Deviation
19	WristLeft Y	θ_{19}	Left Wrist Flexion/Extension
20	ShoulderRight Y	θ_{20}	Right Shoulder Adduction/Abduction
21	ShoulderRight Z	θ_{21}	Right Shoulder Flexion/Extension
22	ShoulderRight X	θ_{22}	Right Shoulder Medial/Lateral Rotation
23	ElbowRight X	θ_{23}	Right Elbow Pronation/Supination
24	ElbowRight Z	θ_{24}	Right Elbow Flexion/Extension
25	WristRight Z	θ_{25}	Right Wrist Radial/Ulnar Deviation
26	WristRight Y	θ_{26}	Right Wrist Flexion/Extension

3.5 Forward Kinematics

Given the values for all link lengths and joint angles, the position and orientation of the joints up to the end-effector (tip of the hand) can be expressed in the base frame. It can be calculated using the transformation matrices with the DH-Parameters of the kinematic model listed in Tables 3.1 and 3.2. These kinematic equations state the forward kinematics of the upper body model. Using the joint angles as generalized coordinates in the joint vector $q = [\theta_1 \dots \theta_{26}]^T$, the pose of the serial manipulator can be calculated as a function of the joint angles:

$$x = f(q). \quad (3.2)$$

The position p and orientation $[n \ s \ o]$ of the i^{th} joint, expressed in the base frame, can be calculated by multiplication of the transformation matrices:

$$\begin{bmatrix} n & s & o & p \\ 0 & 0 & 0 & 1 \end{bmatrix} = T_0^1 T_1^2 \dots T_{i-1}^i. \quad (3.3)$$

3.6 Inverse Kinematics

The inverse kinematics of a system are generally used to calculate joint angles q based on a given position and orientation of an end-effector x :

$$q = f^{-1}(x) \quad (3.4)$$

Solving the inverse kinematics problem is not as straight-forward as calculating the forward kinematics. Due to the kinematic equations being nonlinear, their solution is not always obtainable in closed form. Because the developed upper body model is a highly redundant system, the conventional inverse kinematics for a closed-form solution cannot be applied. The challenge of solving the redundancy of human poses is still a prominent research topic [7, 17, 30]. Instead of calculating a closed-form solution, we use a Jacobian

based approach. The Jacobian provides a mapping between joint angle velocities \dot{q} and Cartesian velocities \dot{x}

$$\dot{x} = J(q)\dot{q}, \quad (3.5)$$

where J is the Jacobian matrix $\partial f/\partial q$.

3.7 State Estimation Methods for Joint Tracking

Considering a state-space representation, the system model describes the dynamics of the system, or in this case how the links of the upper body model move in time. The observation model describes the relationship between the states and measurements. In this work, a linear Kalman filter and an extended Kalman filter are developed for joint tracking.

3.7.1 State Space Models

If it can be assumed that a tracked object, such as a joint of the human body, is executing linear motion, the linear Kalman filter can be used to estimate the states of a system. Below, two commonly used examples of discrete-time state space models describing the motion of an object in 3D space are presented [19]. For the sake of simplicity, the equations are derived to track a single joint's position. The models presented here are later used with the linear Kalman filter algorithm.

Zero Velocity Model

Assuming the velocity of the joint to be zero, the state vector for a problem with three spacial dimensions is given by $s = [x \quad y \quad z]^T$ and the state space model is given by

$$s_{k+1} = As_k + w_k \quad (3.6)$$

$$z_k = Cs_k + v_k, \quad (3.7)$$

where the state transition matrix is given by

$$A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3.8)$$

The observation matrix C takes into account the observed coordinates of the joint position and is given by

$$C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3.9)$$

Constant Velocity Model

Another approach is to model the joint to be moving with constant velocity and taking into account the joint velocities as states. For a 3D problem, the state space vector becomes 6-dimensional: $s = [x \ y \ z \ \dot{x} \ \dot{y} \ \dot{z}]^T$. The state space model has the same form as in the zero velocity model in Eq. (3.6) and (3.7), with the state transition matrix given by

$$A = \begin{bmatrix} 1 & 0 & 0 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 & \Delta t & 0 \\ 0 & 0 & 1 & 0 & 0 & \Delta t \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad (3.10)$$

where Δt is the sampling time. If only the positions, and not the velocities are observed, the observation matrix is given by

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \quad (3.11)$$

3.7.2 Linear Kalman Filter

The Kalman filter [32] is a recursive algorithm used to estimate a set of unknown parameters (in this case the states s) based on a set of measurements z . It uses a prediction and an

update step. The linear Kalman filter provides an optimal solution to the linear quadratic estimation problem. Assume the system and measurement models are linear and given by:

$$s_{k+1} = F_k s_k + B_k u_k + w_k \quad (3.12)$$

$$z_k = H_k s_k + v_k. \quad (3.13)$$

F_k is the state transition matrix, B_k is the input matrix, H_k is the observation matrix, w_k is the process noise, and v_k is the measurement noise. It is assumed that the process and measurement noises are zero-mean, Gaussian noise vectors with covariance matrices Q_k and R_k , i.e. $w \sim \mathcal{N}(0, Q_k)$ and $v \sim \mathcal{N}(0, R_k)$. The covariance matrices are:

$$Q_k = E(w_k w_k^T) \quad (3.14)$$

$$R_k = E(v_k v_k^T) \quad (3.15)$$

Consider that at time k the state estimate $\hat{s}_{k|k}$ and error covariance matrix $P_{k|k}$ are known and contain the information provided by all previous measurements. In the prediction step of the Kalman filter, these quantities can be propagated forward in time using:

$$\hat{s}_{k|k-1} = F_k \hat{s}_{k-1|k-1} + B_k u_k \quad (3.16)$$

$$P_{k|k-1} = F_k P_{k-1|k-1} F_k^T + Q_k \quad (3.17)$$

If a new measurement is available, then the update step can be performed:

$$y_k = z_k - H_k \hat{s}_{k|k-1} \quad (3.18)$$

$$\hat{s}_{k|k} = \hat{s}_{k|k-1} + K_k y_k \quad (3.19)$$

$$P_{k|k} = (I - K_k H_k) P_{k|k-1} \quad (3.20)$$

Equation 3.18 is a measure of the error between the measurement z_k and the current state estimate mapped into the measurement space. This measure is weighted by the Kalman gain:

$$K_k = P_{k|k-1} H_k^T (H_k P_{k|k-1} H_k^T + R_k)^{-1}. \quad (3.21)$$

3.7.3 Extended Kalman Filter

While the linear Kalman filter can be used for linear systems, the Extended Kalman Filter (EKF) extends the algorithm to work on nonlinear systems. Consider a nonlinear model:

$$s_{k+1} = f(s_k, u_k) + w_k \quad (3.22)$$

$$z_k = h(s_k) + v_k \quad (3.23)$$

The true state and measurement vectors can be approximated by linearizing the system about the current state estimate using a first-order Taylor series expansion:

$$s_{k+1} \approx f(\hat{s}_k) + F_k(s_k - \hat{s}_k) \quad (3.24)$$

$$z_k \approx h(\hat{s}_k) + H_k(s_k - \hat{s}_k) \quad (3.25)$$

F_k and H_k are the Jacobians of the system and measurement models, evaluated at the current state estimate:

$$F_k = \left. \frac{\partial f}{\partial s} \right|_{s=\hat{s}_k} \quad (3.26)$$

$$H_k = \left. \frac{\partial h}{\partial s} \right|_{s=\hat{s}_k} \quad (3.27)$$

After linearizing the system, the standard Kalman Filter can be applied. It should be noted that contrary to the linear Kalman filter, the EKF is not optimal. The filter is also still subject to the assumption of Gaussian noise for the process and measurement [33].

CHAPTER 4

MOTION CAPTURE USING A SINGLE KINECT SENSOR

In this chapter, the motion capture process with a single Kinect sensor is described and results are compared to a marker-based tracking system using a Vicon 3D Motion Capture system. The following section presents an evaluation of the baseline performance of depth measurements using the Kinect and its frequency-tracking abilities. Next, results from upper body joint angle tracking experiments with a single Kinect are presented. A test subject performed three different upper body motions: two planar motions (a two-handed wave and a “slow down” signal) and one non-planar motion (a one-handed “move along” signal). Finally, results from tracking the planar upper-body motion with additional weight lifting and under low light conditions are presented.

4.1 Kinect Baseline Performance Evaluation¹

4.1.1 Tracking of a Robot Arm

This section evaluates the robustness of the Kinect system by testing its ability to track a known motion generated by a 3-degree-of-freedom robot. Figure 4.1 shows the robot arm used in this test. The system is controlled using Siemens Simotion drives and PLCs. A MATLAB code is used to generate the trajectory that the robot arm executes [34].

Figure 4.2 shows the labeled components of the robot arm. Unlike conventional serial links, 2 motors are mounted on the base of the robot to drive the 2 links. The two motors can be driven in a coordinated manner to drive link 1. However, if the motors are moved differentially, then a 4-bar linkage rotates link 2. The two links can be moved laterally via a linear stage. The robot, motor driver, connection points, etc. are mounted onto the lower

¹The work in this section was performed in conjunction with Prachi Sahoo and Siddarth Sreeram.

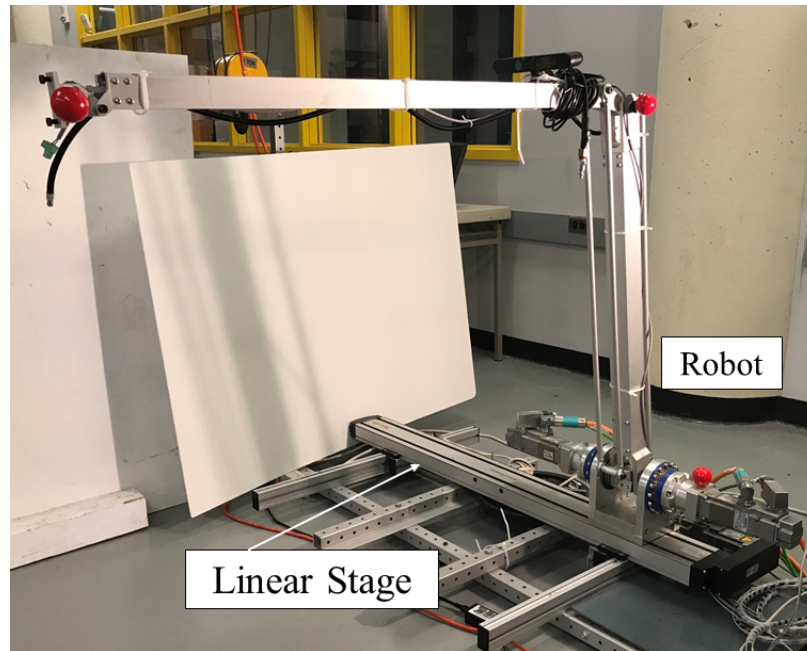


Figure 4.1: 3-DoF Robot Arm.

frame of the robot arm assembly.

Figure 4.3 shows the experimental setup used in this test. The Kinect is placed at an angle facing the robot arm. The robot arm has three red markers placed on it: one marker is on the base, one on the elbow, and one on the end effector. By default, the Kinect is able to track joints once it identifies a human body in its field of view. However, in this experiment a human is not tracked, instead the position of red markers is being tracked. The challenge was to track an object, in this case a robot arm, which did not mimic the general human silhouette, by processing the depth data acquired using the Kinect sensor.

To solve this problem, a visual marker detection was used. Red markers were chosen because red components can easily be detected in real time based on images acquired by the Kinect. This data can be processed to track the centroid of the red markers. Figure 4.4 shows a screen capture of a video recorded by the Kinect. This image shows a green “cross” on the red marker indicating that the Kinect is tracking the centroid of the red marker.

The MATLAB script that was written to capture the data and track the markers on the robot arm made use of functions in the Kinect Toolbox and the Image Processing Tool-

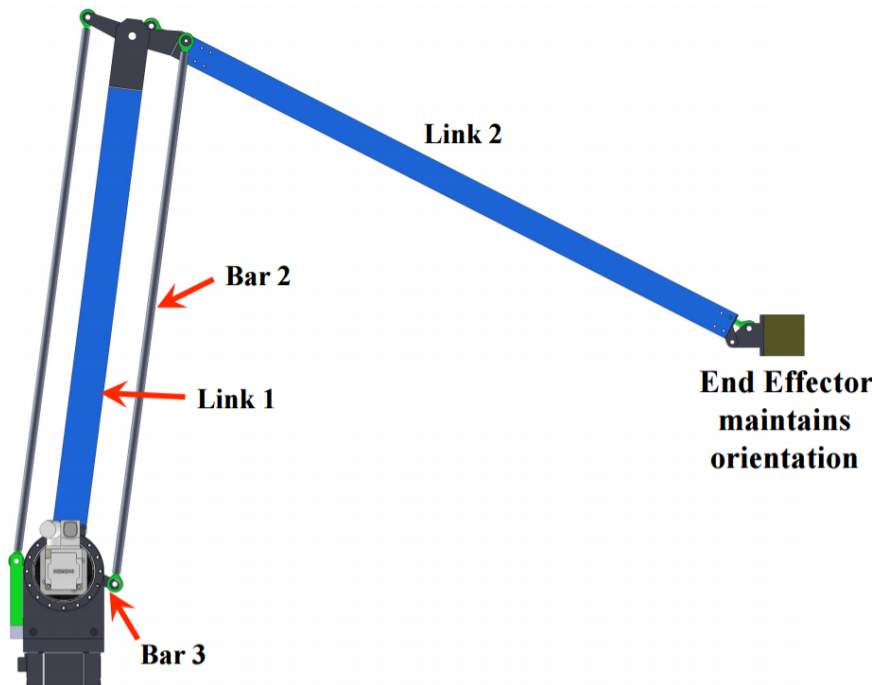


Figure 4.2: Schematic Diagram of Robot Arm.

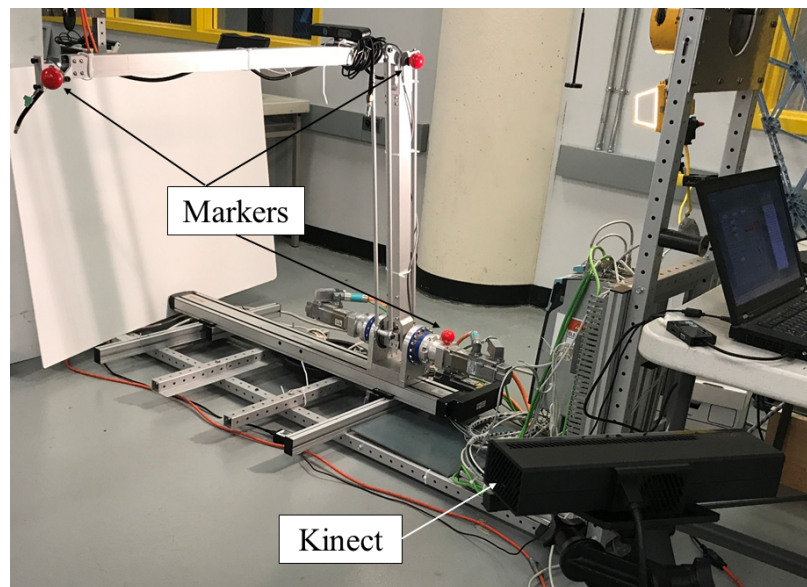


Figure 4.3: Labeled Experimental Setup.



Figure 4.4: Tracked Marker.

box [35]. A red color filter is first applied to the frame, followed by a simple conversion to binary and circle detection. This process accurately locates where the markers are in the Kinect's field of view. Once the X and Y coordinates of the pixel that coincides with the centroid of the tracked marker are obtained, a built-in function provided by the Kinect for Windows SDK can be used to map between locations on the color image and their corresponding locations on the depth image. This function can be used to obtain the X, Y, and Z coordinates of the tracked marker in 3D space.

This process was used on the binary image of each frame to track the red markers and record their X, Y, and Z coordinates with respect to the Kinect's origin. Because the Kinect's origin and the robot arm's origin were different, the data was transformed so that the Kinect's coordinate system matched that of the robot arm. A tracked position of the end effector $P_{EE}^K = [x \ y \ z \ 1]^T$, given in the Kinect's coordinate system, was transformed into the robot's coordinate system to yield P_{EE}^R using a 4×4 homogeneous transformation matrix A_R :

$$P_{EE}^R = A_R P_{EE}^K \quad (4.1)$$

A detailed description of how the transformation matrix can be derived from the Kinect

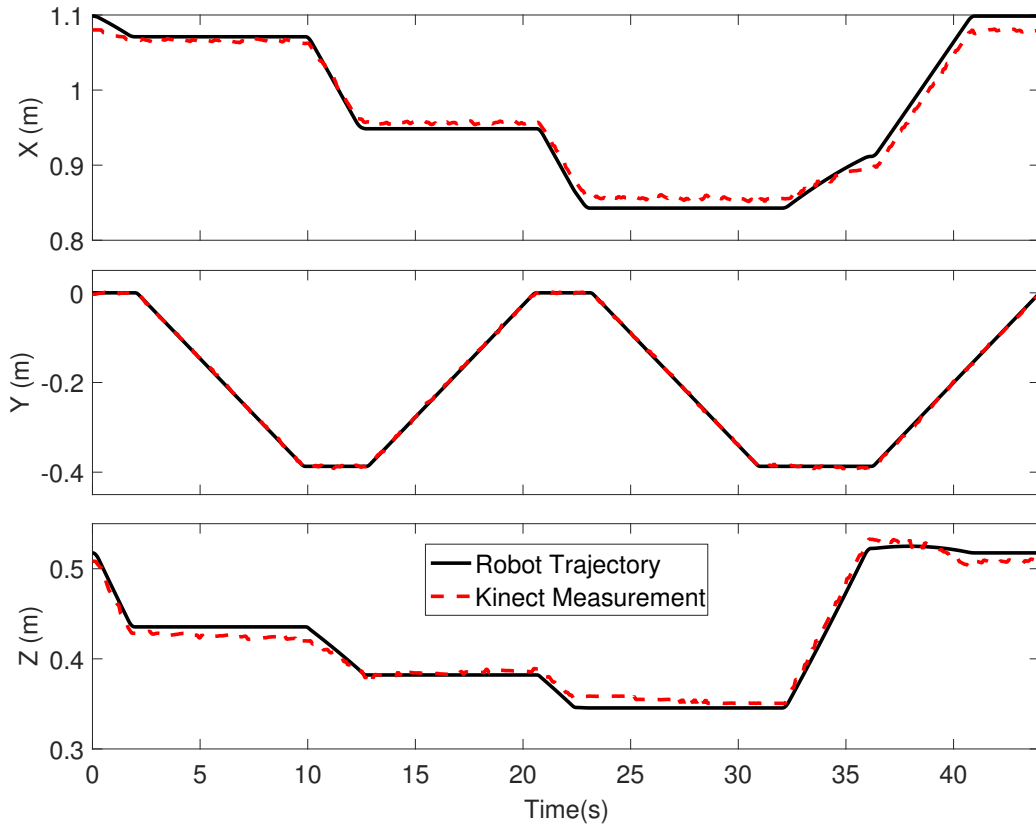


Figure 4.5: XYZ Coordinates of End Effector vs. Kinect Measurement for Trial 1.

measurements based on Corresponding Point Set Registration is laid out in Section 5.3.1.

The trajectory that was constructed to move the end effector was compared to the trajectory data collected by the Kinect. Three trials were conducted tracking a baseline trajectory that was originally developed for painting and sandblasting operations over large areas [34]. During the executed trajectory, the robot arm moved along the linear stage, and both the shoulder and the elbow joint were moved.

Figure 4.5 shows the XYZ coordinates of the end effector and their respective measured trajectories for trial 1. It can be observed that the Kinect tracks the Y coordinate almost perfectly. However, the X and Z coordinate show some deviation from the trajectory. Similar figures were generated for all trials and can be found in Appendix A.

Figure 4.6 shows the XYZ coordinates of the tracked end effector marker for three trials of the same robot motion. Figure 4.7 shows the error between the robot trajectory and

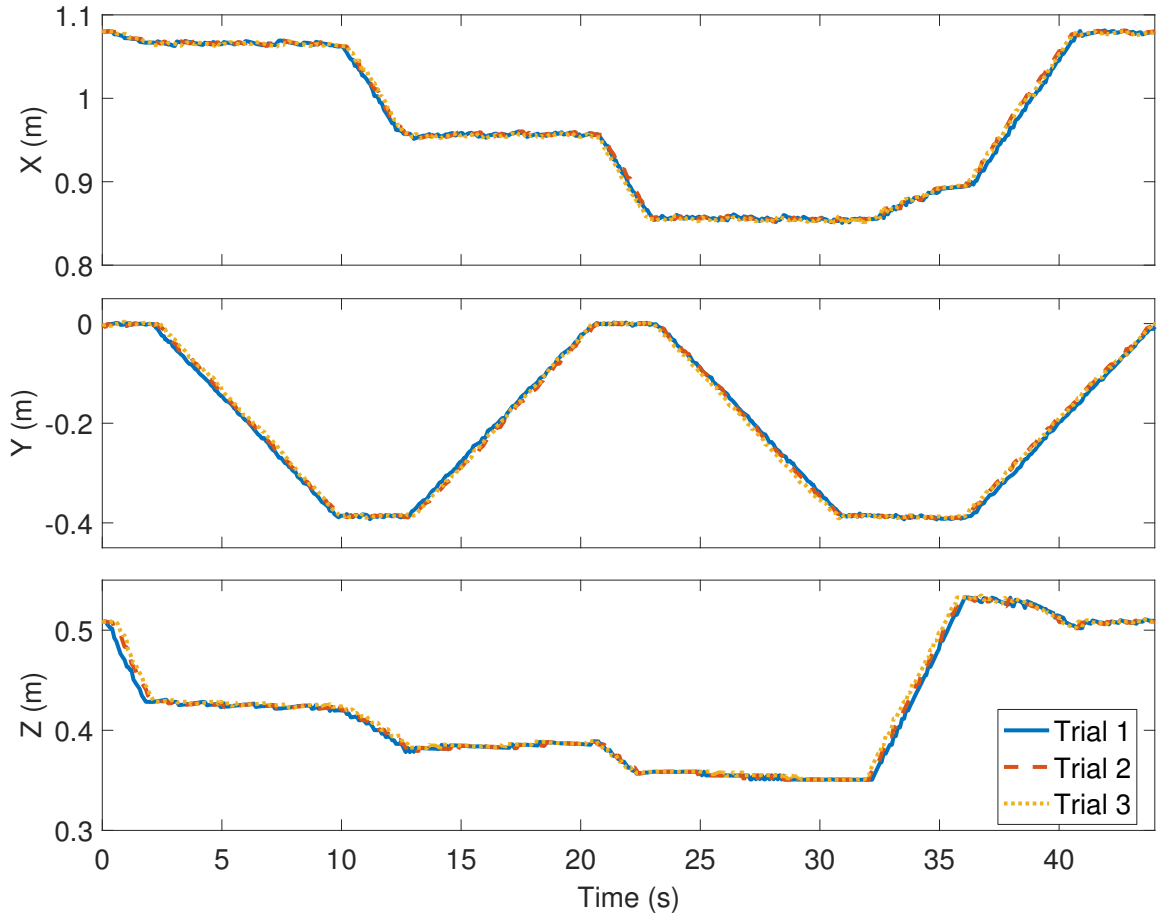


Figure 4.6: X, Y, Z Coordinates of End Effector for Three Trials.

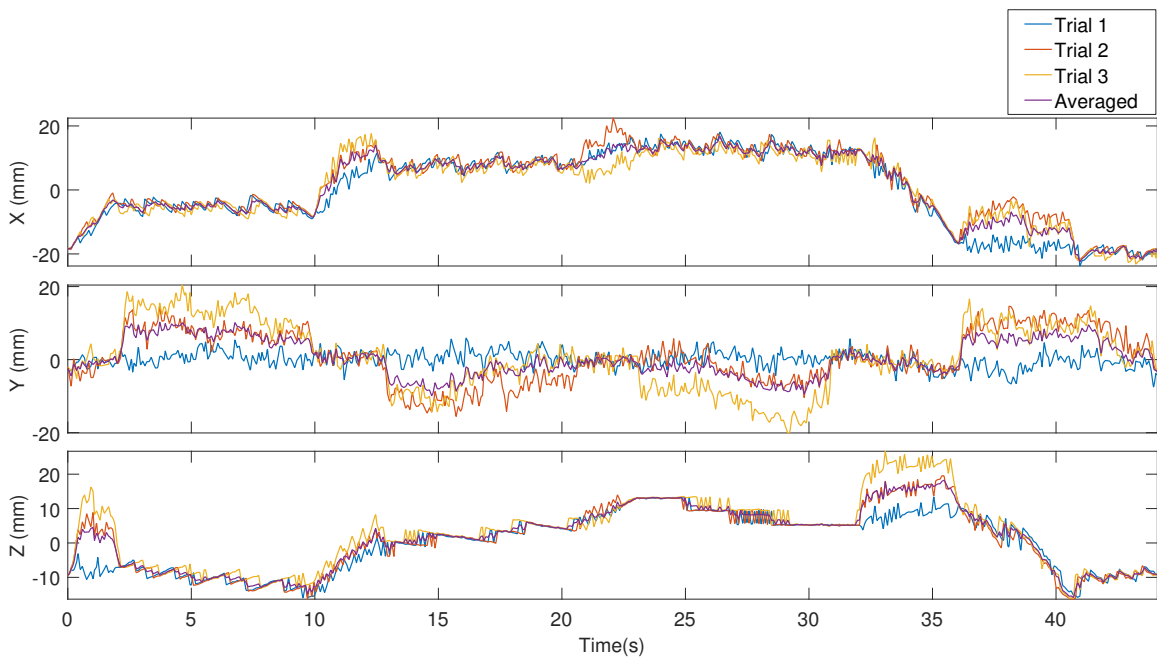


Figure 4.7: Error between Kinect Measurements and Robot Trajectory.

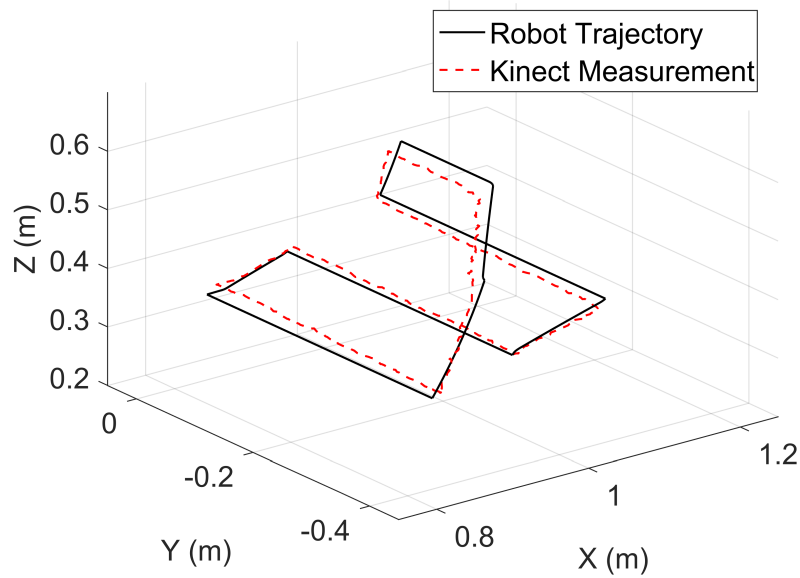


Figure 4.8: End Effector Trajectory in 3D Space.

Kinect measurements for the three trials. To analyze the accuracy, the mean and maximum absolute errors between the Kinect measurements and the true robot trajectory were calculated for each trial and are listed in Table 4.1. The end effector position could be tracked especially well in the Y direction. Mean absolute errors were smaller than 10 mm for all trials except the X component of trial 1, where the mean absolute error was 10.51 mm. Generally, the robot trajectory could be tracked especially well in the Y direction. The tracking performance was worst in the X direction. Maximum absolute errors in the tracked position were smaller than 27 mm for all trials.

The average end effector trajectory from the three Kinect measurements was calculated. Figure 4.8 shows the comparison between the XYZ coordinates of the programmed end effector trajectory and the averaged Kinect tracking data. To investigate the repeatability between the different trials, the mean and maximum absolute deviation from the average Kinect measurement was calculated for each trial. The values are listed in Table 4.1. The mean absolute deviations were smaller than 2 mm for the X and Z component of the motion for all trials, and smaller than 4.4 mm for tracking motion in the Y direction. The maximum

Table 4.1: Mean absolute error (MAE), maximum absolute error, mean absolute deviation (MAD) and maximum absolute deviation for robot arm tracking experiments with Kinect. All values in mm.

	MAE			Max. abs. error		
	X	Y	Z	X	Y	Z
Trial 1	10.51	1.63	7.48	23.77	7.47	15.91
Trial 2	9.86	5.61	7.89	22.45	15.55	19.55
Trial 3	9.49	6.93	8.72	23.17	20.42	26.67

	MAD			Max. abs. deviation		
	X	Y	Z	X	Y	Z
Trial 1	1.67	4.36	1.76	9.47	13.57	15.03
Trial 2	1.42	2.46	0.87	8.24	10.23	5.33
Trial 3	1.55	3.42	1.75	7.95	11.30	12.13

absolute deviation was smallest for tracking the X component of the trajectory, and did not exceed 15.1 mm for all trials. Thus the tracking of the end effector position with Kinect was repeatable in the conducted experiments.

It can be observed that the Kinect tracked the marker on the end effector with similar accuracy for each trial. Hence, it can be concluded that the Kinect tracks motion in a reliable and repeatable manner, and thus is inherently a robust tool to study and evaluate upper body motion.

4.1.2 Frequency Tests

It is important to understand the tracking limitations of the Kinect sensor. One important limitation is how reliably the Kinect can track frequencies of motions. Prior to human-motion tracking, a simple pendulum setup was utilized to verify the frequency of measurement by the Kinect and Vicon. Both systems were used to track a retro-reflective marker at the end of a pendulum. A pendulum setup was placed in front of the Kinect at a distance of 1.5 meters, as seen in Figure 4.9. The swinging motion of the pendulum bob with a marker was tracked by the Kinect and Vicon systems for varying lengths of the pendulum.

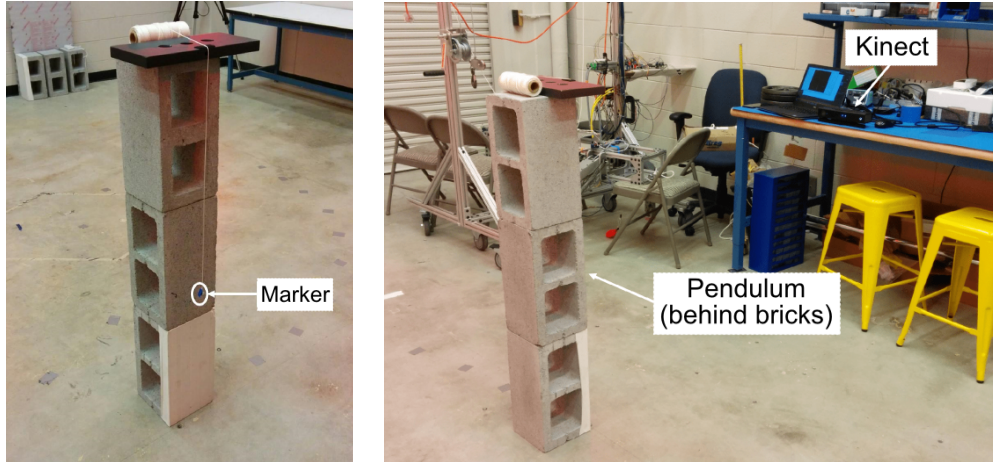


Figure 4.9: Pendulum setup for varying suspension lengths

Pendulum frequencies measured by the systems were compared with theoretical values.

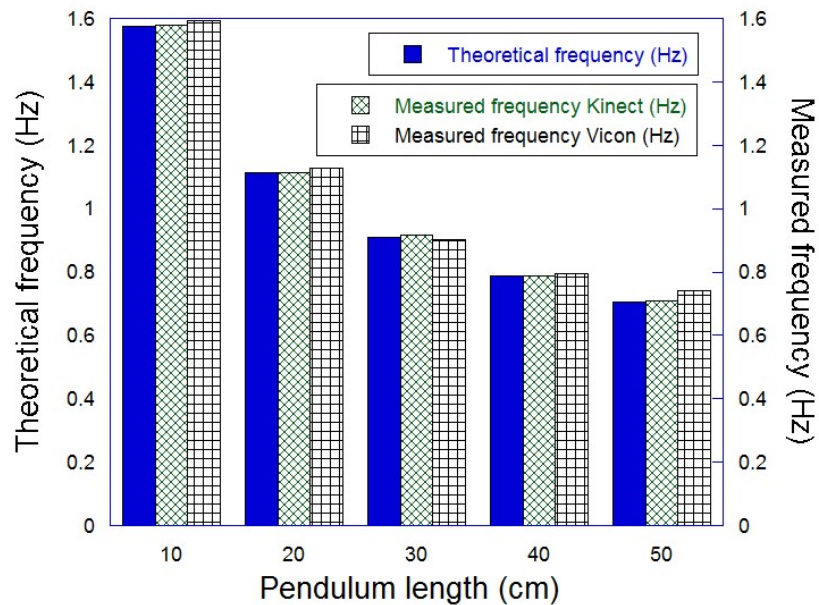


Figure 4.10: Theoretical and measured natural frequency for varying suspension lengths

Figure 4.10 shows the results of the frequency test for five pendulum suspension lengths. The Kinect tracked the pendulum motion with good accuracy for all tested lengths. This test verified that the Kinect is able to track motions at frequencies that are typically performed based on human physical limits. The mean average error in frequency measurement of the pendulum for the Kinect was 0.011 Hz, compared to the Vicon's error of 0.024 Hz.

Some of the error can be attributed to noise in the system, typically created by small reflective surfaces in the view of the camera system. Additional error may have arisen due to occlusion of the marker behind the brick structure of the pendulum, which is shown in Figure 4.9, during which time not all Vicon cameras would have captured data. However, the marker was in plain view of the Kinect sensor throughout data collection.

4.2 Comparison of Kinect and Vicon Motion Capture Systems

Motion capture and joint tracking analysis is used extensively in video game and graphic design, but applications in biomechanics and robotics are gaining in importance. Vicon 3D Motion Capture systems involve multiple high definition cameras which are accurate, but expensive, and infeasible to use outside of the confined space of a motion capture lab. Microsoft Kinect is a low-cost camera primarily used in the video game industry which can be used for human-body motion analysis.

The primary differentiating factor between Kinect and Vicon is the necessity of retro-reflective markers in the Vicon system. Light from the Vicon cameras is emitted and is reflected back from markers in the field of view. This yields the 3D position of each marker. However, the Kinect does not require markers for human-body tracking because a proprietary software possesses the ability to track human body joints without markers. The Vicon system is composed of multiple cameras set up around the perimeter of the measurement workspace at varying heights to obtain a full 360 degree view of the field. On the other hand, the Kinect is a single sensor that only measures what is in its field of view.

4.2.1 Joints Tracked by Kinect

The Kinect sensor tracks a human skeleton comprised of 25 joints. The joint positions are estimated from depth data. Table 4.2 lists the joints tracked by Kinect with their name and description. The position estimate of each joint is available at a frame rate of approximately 30 frames per second.

Table 4.2: Joints tracked by the Kinect v2 Sensor

No.	Name	Description	No.	Name	Description
1	SpineBase	Base of the Spine	14	KneeLeft	Left Knee
2	SpineMid	Middle of the Spine	15	AnkleLeft	Left Ankle
3	Neck	Neck	16	FootLeft	Left Foot
4	Head	Head	17	HipRight	Right Hip
5	ShoulderLeft	Left Shoulder	18	KneeRight	Right Knee
6	ElbowLeft	Left Elbow	19	AnkleRight	Right Ankle
7	WristLeft	Left Wrist	20	FootRight	Right Foot
8	HandLeft	Left Hand	21	SpineShoulder	Spine at the Shoulder
9	ShoulderRight	Right Shoulder	22	HandTipLeft	Tip of the Left Hand
10	ElbowRight	Right Elbow	23	ThumbLeft	Left Thumb
11	WristRight	Right Wrist	24	HandTipRight	Tip of the Right Hand
12	HandRight	Right Hand	25	ThumbRight	Right Thumb
13	HipLeft	Left Hip			

4.2.2 Vicon Marker Placement

The Vicon motion capture system uses retro-reflective markers that are attached to tracking objects, in this case, to the body of a human test subject performing the motion. Because this thesis is focused on upper body motions, nineteen (19) markers were strategically placed on the test subject to track joints of the upper body corresponding to those tracked by the Kinect sensor. For each of the joints, the position was obtained by calculating the average of two (three for the SpineBase joint) markers, as listed in Table 4.3. Figure 4.11 and Figure 4.12 show the marker placement on the test subject.

4.3 **Comparison of Kinect and Vicon Motion Capture for Upper Body Joint Angle Tracking²**

The following section describes a process for markerless tracking of three different upper-body motions and evaluates the accuracy of the Kinect's joint tracking abilities for these motions by comparing the joint angles to values obtained with the Vicon 3D Motion Cap-

²The work in this section was performed in conjunction with Siddarth Sreeram.

ture system. Because this thesis focuses on upper-body motions, the scope of the joint

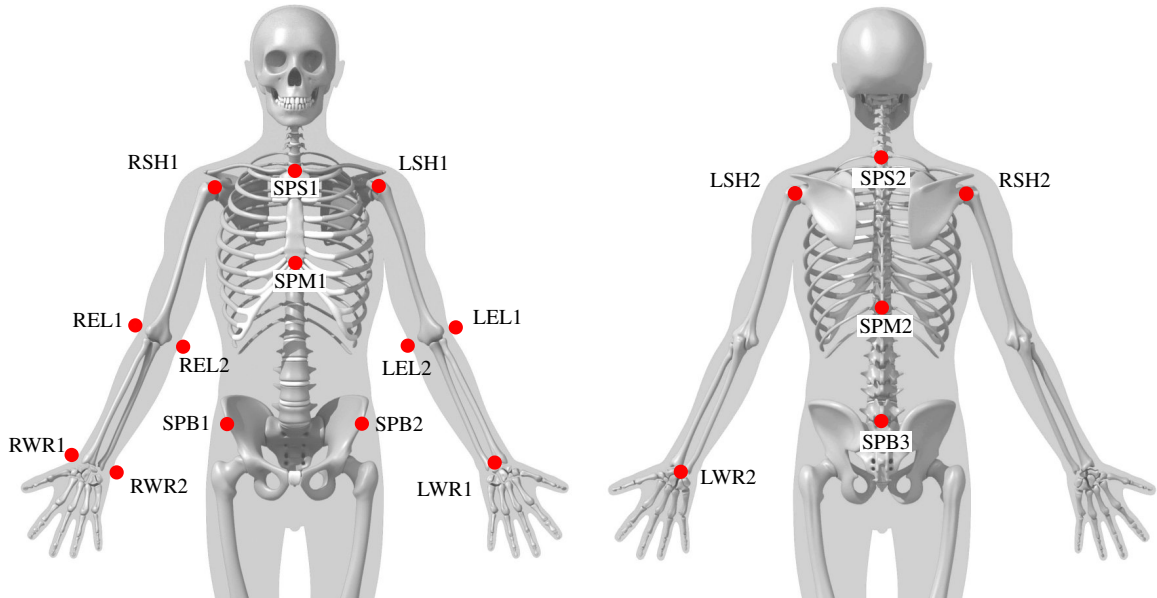


Figure 4.11: Marker Placement for Measurements with a Single Kinect Sensor³



Figure 4.12: Marker Placement on Test Subject

angle analysis in this study is limited to shoulder and elbow angle trajectories. The derived information from human upper-body motions can also potentially be used in robot applications that seek to mimic human upper-body movements.

³This image was adapted from https://www.c-motion.com/v3dwiki/index.php/Tutorial:_Plug-In_Gait_Full-Body

Table 4.3: Kinect Joints and corresponding Markers

Name	Markers	Name	Markers
SpineBase	SPB1, SPB2, SPB3	WristLeft	LWR1, LWR2
SpineMid	SPM1, SPM2	ShoulderRight	RSHO1, RSHO2
SpineShoulder	SPS1, SPS2	ElbowRight	REL1, REL2
ShoulderLeft	LSHO1, LSHO2	WristRight	RWR1, RWR2
ElbowLeft	LEL1, LEL2		

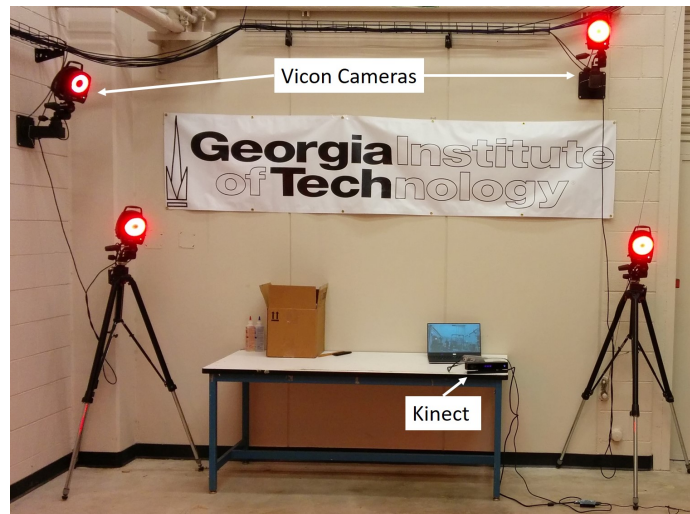


Figure 4.13: Experiment Setup: Vicon Cameras and Kinect v2 Sensor

4.3.1 Instrumentation and Setup

Tracking data was recorded at the Indoor Flight Facility at Georgia Tech. The room has 15 Vicon MX3+ cameras sampling at 100 Hz. Data was concurrently obtained using the Microsoft Kinect sampling at 30 Hz. The software used to process the data was Vicon Nexus 2.5 (Vicon Motion Systems, Oxford, UK) and the Microsoft SDK provided for Kinect developers. The test subject stood at a distance of 2 meters directly in front of the Kinect sensor. The experimental setup is shown in Figure 4.13.

4.3.2 Data Processing

Marker trajectories were filtered using a Woltring filter [36] in the Vicon Nexus 2.5 software pipeline which is commonly used to minimize marker trajectory noise. Gaps in marker data with gap sizes < 20 frames (< 0.2 seconds) were filled using spline interpolation. To calculate the joint positions corresponding to the Kinect's tracked joints, mean position data from two (three for the SpineBase) markers were calculated for each upper-body joint.

The coordinate system of the Kinect data was aligned with the Vicon coordinate system via coordinate transformation. To synchronize the timing of the collected motion data from Vicon and Kinect, the error in joint position data for the left and right wrist for each motion was minimized using least-squares fit. To reduce noise, the Kinect data was filtered with a low pass filter. As human movement is generally associated with low frequencies, the cutoff frequency of the filter was set to 3 Hz. The shoulder rotation data was calculated with respect to body planes. For the tested set of motions, it was assumed that the elbow joint has only one degree of freedom. Elbow joint angles were, therefore, calculated as the angle between two vectors (one from elbow to wrist and one from elbow to shoulder) [11].

4.3.3 Planar Motions

The planar motions analyzed in this study were the two-handed wave and the two-handed "slow down" motions. Figure 4.14 illustrates the two-handed wave and the two-handed "slow down" signal motions performed in this study. Because these motions result in upper-body joint rotations in mainly one plane, they are referred to as planar motions. Note that the wave motion in 4.14(a)-(c) occurs in the X-Z plane, while the "slow down" motion occurs in the Y-Z plane. Figure 4.15 shows that position tracking in the X and Z dimensions for the right wrist (the 2D coordinates) was nearly indistinguishable for the Kinect compared to the Vicon. The Y dimension represents the axis extending from the Kinect to the subject, and there was some discrepancy observed in that data. This arose from the error in the depth map stream gathered by the Kinect's infrared camera. Figure 4.16 presents joint

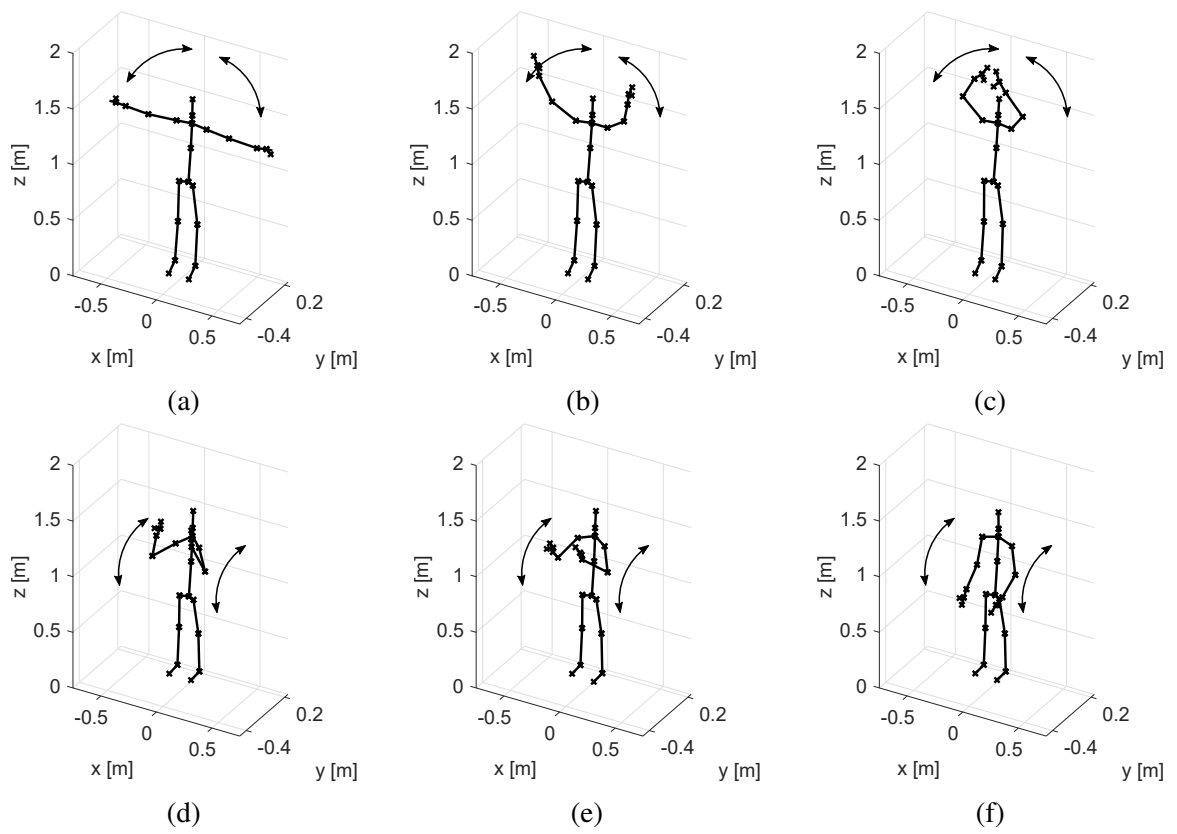


Figure 4.14: Illustration of the planar motions performed in this study: (a)-(c) Two-handed wave, (d)-(f) Two-handed "slow down" signal.

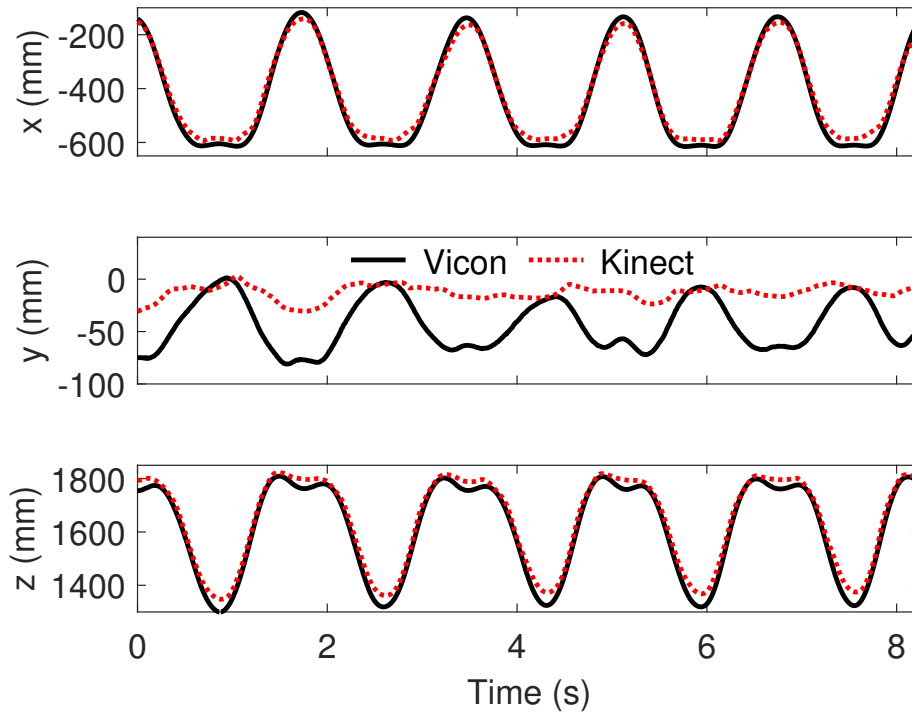


Figure 4.15: Right wrist joint trajectory for two-handed wave motion.

angle trajectories for the left elbow angle and left shoulder angle during the two-handed wave motion. Generally, the Kinect was able to track the joint angle phase, and tracked the amplitude with a small, and roughly constant offset.

Figure 4.17 presents joint angle trajectories for the right elbow angle and right shoulder angle during the two-handed “slow-down” motion. While the elbow angle is tracked well, the shoulder angle displays a larger offset in amplitude. However, the phase of the motion

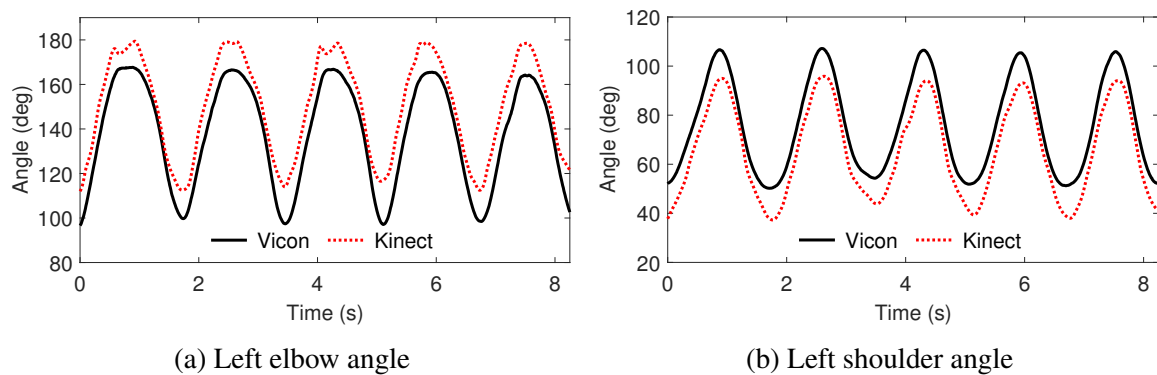


Figure 4.16: Joint angle trajectories for two-handed wave motion.

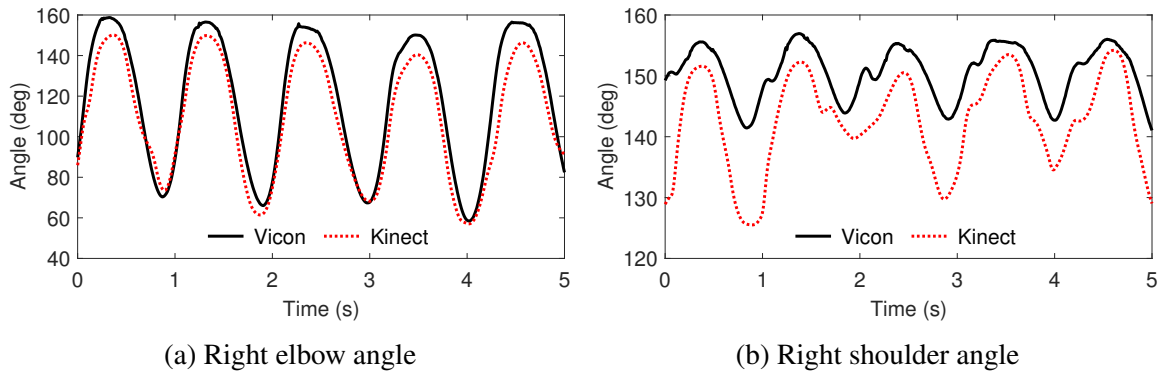


Figure 4.17: Joint angle trajectories for two-handed “slow down” signal motion.

was tracked well by the Kinect for both joint angles.

When looking at joint angle measurements in Figure 4.16, the frequency of joint angle measurement by the Kinect closely matches the Vicon data for a two-handed wave. However, there is some amplitude offset, simply due to the positioning of the markers on the subject’s elbow and shoulder. If the marker positions do not line up exactly with the coordinate joint mapping algorithm of the Kinect SDK, then some amplitude offset is expected in the joint angle measurements. Figure 4.17 shows similar results for the two-handed “slow down” motions, with some skips in the shoulder data, likely due to joint occlusion at those times. When the hands are at the highest point, it is possible that they occluded the shoulder from the view of the Kinect, causing a loss of smoothness in the data. The magnitude of elbow joint angle amplitude matches up very closely in both figures, whereas there are some discrepancies in the shoulder angle for the “slow down” motion.

In summary, the Kinect performed well for the two-handed wave motion joint tracking, but tracked the shoulder angle during the “slow down” motion less accurately, which is likely caused by occlusion of the elbow joint.

4.3.4 Non-planar Motion

A more complex, non-planar motion is important to investigate as to determine the joint tracking ability of the Kinect in 3D, spanning different planes and observing results when faced with problems such as joint occlusion. The non-planar motion conducted by the sub-

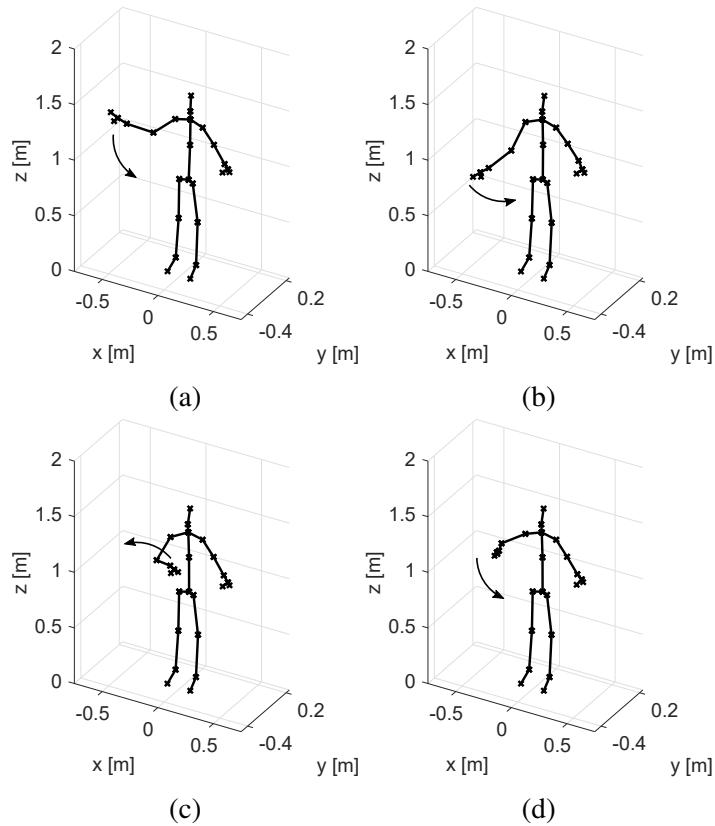


Figure 4.18: Illustration of the non-planar motion performed in this study: one-handed “move along” motion

ject was the one-handed “move along” motion. The motion was performed by pointing the left hand in a fixed direction (towards the left of the subject), and the right hand performing an elliptical counter-clockwise motion.

Figure 4.18 illustrates the one-handed, non-planar “move along” motion performed in this study. Figure 4.19 shows joint angle trajectories for the one-handed “move along” motion. While the right elbow could be tracked well with the Kinect, left elbow angle trajectories showed large offsets compared to the Vicon measurements. As expected, the left elbow experienced almost no change in angle because the left hand is stationary for this motion. Still, the graphs for the left hand do display some angle change, highlighting the sensitivity of both the Kinect and Vicon systems.

The right hand performs the elliptical motion, and the right elbow is tracked very well, with nearly no discrepancy in the data. The phase and amplitude measurements are accu-

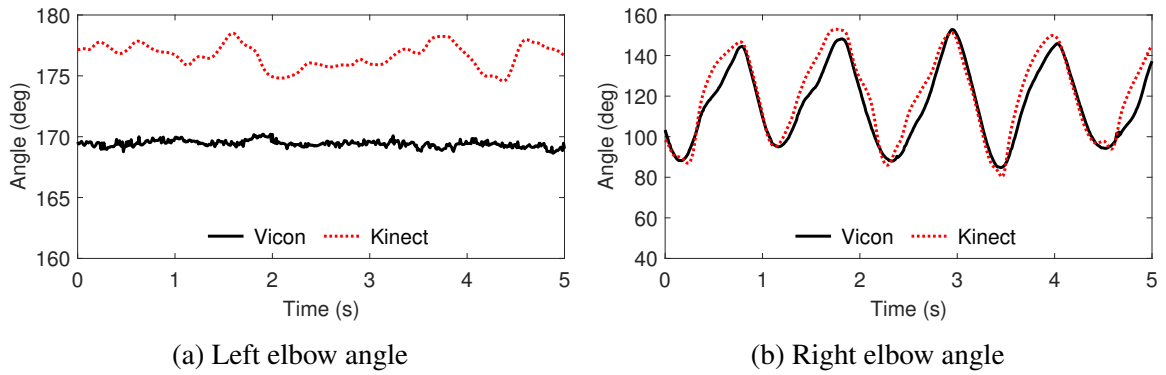


Figure 4.19: Joint angle trajectories for one-handed “move along” signal motion.

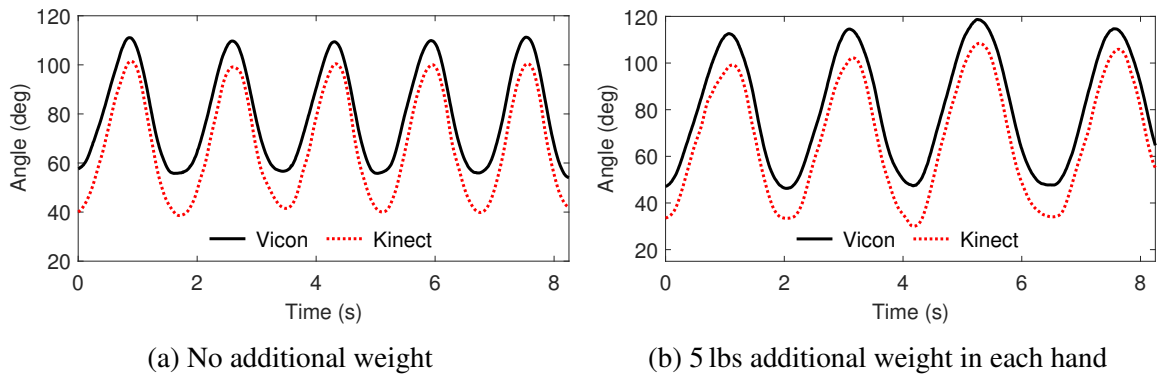


Figure 4.20: Right shoulder joint angle trajectories weighted tests.

rately measured by both systems.

4.3.5 Weighted Tests

Motions were performed with the subject carrying 2 lbs, 3 lbs and 5 lbs dumbbells in each hand to assess the performance of the camera systems for changing frequencies and objects in the subject’s hands, as well as the physical effect on the subject.

As the subject completed the two-handed wave with additional weights, the frequency of the motion declined, as seen in Figure 4.20. This is an intuitive result, as lifting additional weights leads to exhaustion and, therefore, less repetitions of the motion can be conducted in the same time period. From the data it is evident that both Kinect and Vicon systems are able to cope with objects in the test subject’s hands and still track joints well.

4.3.6 Rotation Tests

In an additional set of tests, the subject performed motions while gradually moving from the frontal to the sagittal plane, with respect to the Kinect sensor. This experiment was conducted to observe the performance of the Kinect when it cannot “see” the entire human body, and to obtain the approximate angle at which the tracking begins to fail.

Figure 4.21 shows right shoulder and right elbow joint angle trajectories for the two-handed wave motion conducted as the orientation of the subject, relative to the Kinect, was increased from 0° to 90°. The root-mean-square error (RMSE) was computed for different orientation angles relative to the Kinect sensor using:

$$RMSE_M = \sqrt{\frac{\sum_1^n (\hat{\beta}_i - \beta_i)^2}{n}}, \quad (4.2)$$

where M is a specific motion, n is the number of recorded frames, $\hat{\beta}$ is the joint angle calculated from Kinect measurements and β is the joint angle calculated from Vicon measurements. The results are listed in Table 4.4. For each motion, a 90° orientation toward the Kinect resulted in a very large RMSE. Generally, the RMSE increased as the motion was performed at a greater angle relative to the Kinect. However, good tracking was obtained for some of the motions up until 60° relative positioning.

4.3.7 Low Light Conditions

The effects of varying light conditions is an interesting area of study for camera systems such as the Kinect and Vicon, and these were briefly considered in this chapter. Low light, for the purposes of this study, is defined as a situation where the only sources of light in the room are the red light from the Vicon cameras, the small amount of light emitted from the cameras on the Kinect and the laptop computer that was used to run the Kinect.

First, the Vicon and Kinect systems were run concurrently with all other light removed

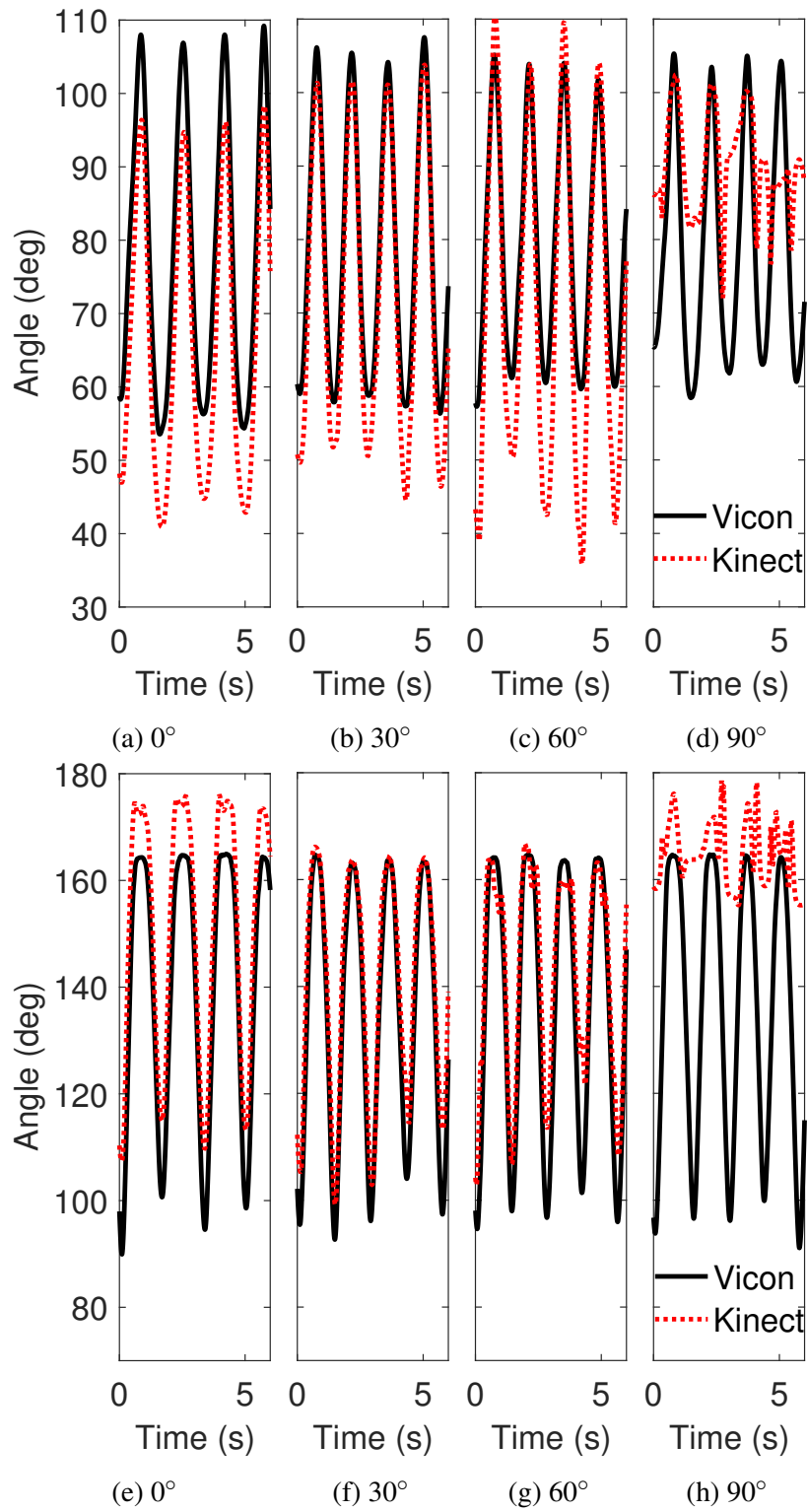



Figure 4.21: Joint angle trajectories for two-handed wave motion for different orientations towards Kinect: (a)-(d) Right Shoulder and (e)-(h) Right Elbow

Table 4.4: RMSE ($^{\circ}$) in Joint Angle Trajectories from Rotation Tests for two-handed Wave Motion and one-handed “Move Along” Signal for Left and Right (L, R) Elbow and Shoulder (E, S)

Motion	Joint angle	Orientation towards Kinect			
		0 $^{\circ}$	30 $^{\circ}$	60 $^{\circ}$	90 $^{\circ}$
Wave motion (4x600 frames)	LE	12.62	13.35	12.12	20.63
	RE	11.82	7.28	10.18	36.24
	LS	11.53	14.17	13.49	14.79
	RS	11.86	6.23	11.35	16.16
Move along signal (4x350 frames)	LE	3.37	4.64	13.76	5.78
	RE	6.19	5.75	10.35	71.82
	LS	6.68	18.56	24.43	16.61
	RS	10.26	4.81	3.52	54.10

 - Good tracking result

from the room. Following this, complete darkness was created by switching off the Vicon system and only running the Kinect in darkness. Figure 4.22 presents the left elbow angle for the two-handed wave motion under low light conditions. Joint angle trajectories from the Kinect showed large error compared to the Vicon measurements in amplitude, but the Kinect was able to accurately track the frequency of the motion. In complete darkness, the Kinect was still able to track joint positions. Because for this test the Vicon system had to be switched off completely, comparison between the two systems could not be conducted.

In low light conditions, the Vicon performed as expected, because its infrared cameras do not require ambient light to function well. The Kinect was also able to function and did track joints, but performed poorly. In complete darkness, the Kinect performed joint tracking much better than in the low light environment, likely due to reduced interference and reflections from surfaces in the room, thereby reducing noise in the signal.

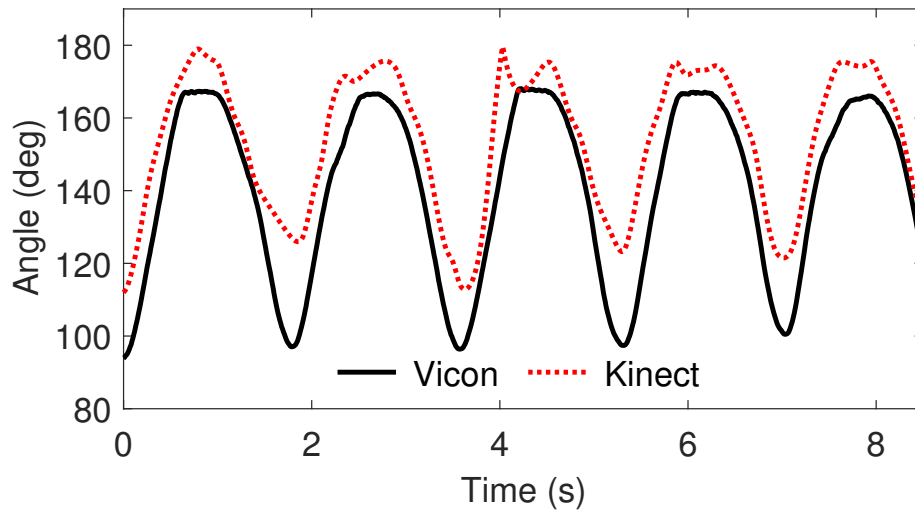


Figure 4.22: Left Elbow joint angle during two-handed wave motion in low light conditions

4.3.8 Summary

The Kinect proved to be a useful tool for several motion capture applications. For planar motions such as the two-handed wave and the two-handed “slow down”, the Kinect was expected to perform well because all joints were in its field of view. It did so, and joint angle measurements matched up closely with Vicon measurements. However, there was some loss of accuracy for the “slow down” motion, likely due to elbow joint occlusion.

For non-planar motions such as the two-handed “move along”, the left hand was stationary during the motion, but the Kinect still estimated small movements in the joints. The right arm joints were tracked sufficiently well during the elliptical motion, and joint occlusion did not seem to be much of an issue. The weighted tests showed that objects in the subject’s hands did not degrade tracking. Rotation tests showed that Kinect tracking fails when the subject is standing at more than 60° with respect to the Kinect. Lastly, low light conditions were problematic for the Kinect because of interference and reflections, and in fact it performed better in complete darkness due to reduced reflection from arbitrary surfaces in the room.

CHAPTER 5

DUAL-KINECT MOTION CAPTURE

This chapter presents a dual-Kinect system for real-time motion capture measurements. To evaluate the performance of the proposed system, it is used to track a human test subject conducting a set of three different motions (“two-handed wave”, “slow-down signal”, and “torso twist”). Further testing with loose-fitting clothes demonstrates the robustness of the proposed measurement system. During these tests, the test subject conducted motions commonly performed to test fit of garments, such as the torso twist, calf extensions, and squats.

The dual-Kinect system uses Kalman filters to fuse the two data streams and improve joint tracking. For analyzing the results in detail, a script that records the joint position estimates from both Kinect sensors has been implemented. To evaluate the tracking performance, data was concurrently obtained with a Vicon motion capture system.

The recorded data was used to study the joint position tracking performance for different filter parameters for a linear Kalman filter (LKF), and for the Extended Kalman filter (EKF) based on the kinematic human upper body model that was introduced in Chapter 3. Results from human motion capture experiments with the dual-Kinect system and both filters are compared to marker-based motion capture data collected with a Vicon system.

5.1 Dual-Kinect Motion Capture Process

In this section, the motion capture process with two Kinect sensors (referred to as Kinect 1 and Kinect 2) is described. First, data acquired from both Kinects is transformed into a common coordinate system. Then, the joint position estimates are combined using sensor fusion, taking into account the tracking state of each joint provided by the Kinects.

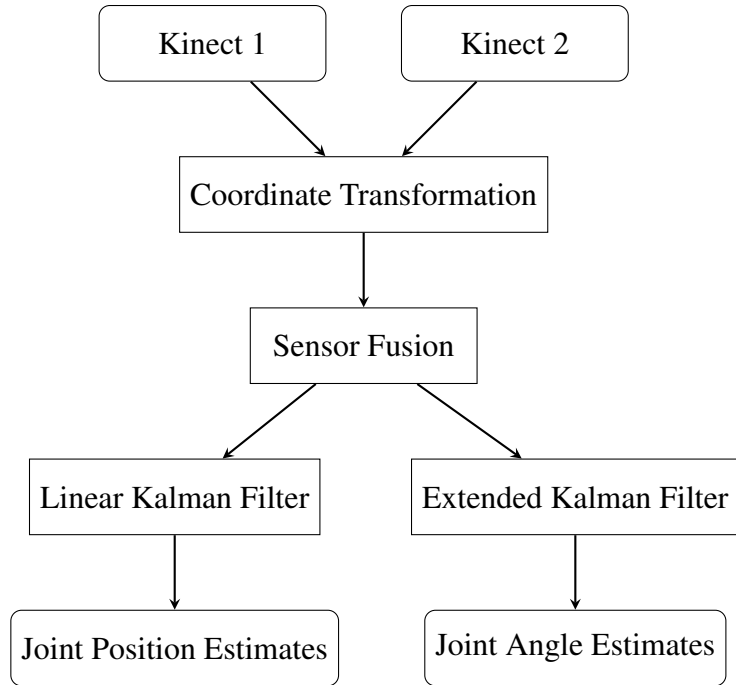


Figure 5.1: The Dual-Kinect Motion Capture Process

For real-time tracking, the fused data is subsequently fed into a linear Kalman filter (LKF), yielding joint position estimates based on both Kinect data streams. For offline analysis, the same data is fed into an Extended Kalman filter (EKF). The EKF estimates the joint angles of the upper body model. Figure 5.1 shows the workflow of the proposed motion capture system.

5.2 Implementation Details

For the real-time portion of the proposed system, the necessary computations must be carried out quickly enough to track motion at 30 frames per second. This is required for the tracking performance to be perceived without lag. To improve the out-of-the-box skeleton-tracking provided by Kinect, the Dual-Kinect system must yield more stable joint position estimates. Compared to a single-Kinect system, using data from two Kinects can increase the possible tracking volume and reduce problems caused by occlusion, especially for turning motions e.g. a torso twist.

5.2.1 Hardware and Implementation Restrictions

Development, data collection, and evaluation were carried out on two Laptops with Intel Cores i7-6820HQ CPUs. Because the Kinect for Windows Software Development Kit (SDK) for the second version of Kinect only supports one sensor, data was acquired with two laptops. Communication between the laptops was established via the User Datagram Protocol (UDP), used primarily for low latency applications. In order to directly process the data in MATLAB, the Kin2 Toolbox Interface for MATLAB [35] was used for data collection.

5.2.2 Dual-Kinect Configuration

To find an optimal orientation of the two Kinect sensors relative to each other, and to the test subject, nine (9) different sensor configurations were evaluated. First, both sensors were placed directly next to each other to define the zero position. The test subject stood facing the Kinect sensors at a distance of about 2 meters, while performing test motions. For the first six test configurations, both Kinects were then gradually moved outwards on a circular trajectory around the test subject, as illustrated on the left side of Figure 5.2.

The angle γ between each sensor and the zero position was increased in 15° steps. For Configurations 7-9, one Kinect sensor was kept at the zero position, while the second Kinect was placed at varying positions on a circular trajectory towards the right of the test subject in 30° steps. The angle δ was measured between the two Kinects, as illustrated on the right side of Figure 5.2.

For each sensor configuration, the test subject performed a set of three test motions (a wave motion, a "slow down" signal, and a torso twist). Table 5.1 lists all tested sensor configurations with their respective angles.

Because this thesis is focused on upper body motions, the fused tracking data of the wrist joints was chosen as a measure of tracking quality. Evaluation of the tracking data from the different test configurations showed that with the combined data from both Kinects,

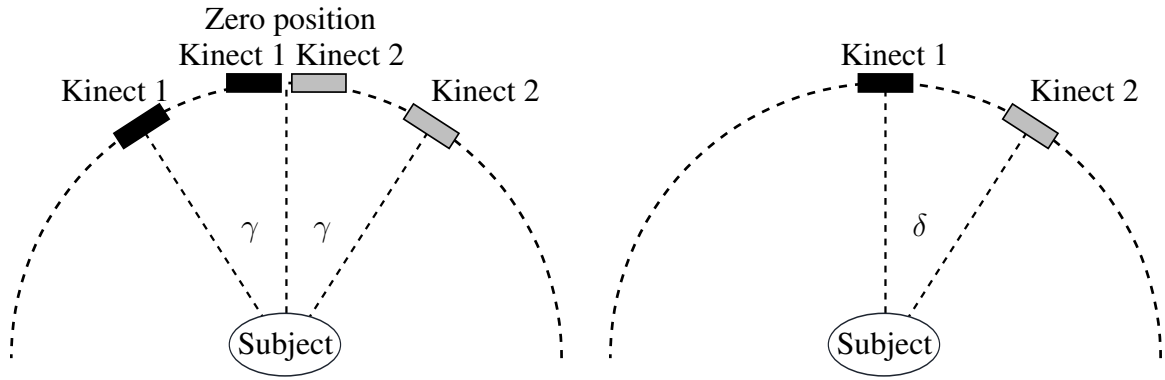


Figure 5.2: Different Kinect Configurations Tested in this Study

Table 5.1: List of tested Dual-Kinect Configuration Variants

Configuration No.	Angle between Kinect sensors and zero position	Configuration No.	Angle between Kinect sensors and zero position
1	$\gamma = 0^\circ$	6	$\gamma = 75^\circ$
2	$\gamma = 15^\circ$	7	$\delta = 30^\circ$
3	$\gamma = 30^\circ$	8	$\delta = 60^\circ$
4	$\gamma = 45^\circ$	9	$\delta = 90^\circ$
5	$\gamma = 60^\circ$		

the wrist joint could be tracked closely for Configurations 1-5 and Configurations 7-8. However, for Configurations 6 and 9, the wrist trajectory could not be tracked reliably, especially at extreme positions during the torso twist motion. The wrist joint trajectories from the Dual-Kinect system for all test configurations are presented in Appendix D.

Setting up the Kinects according to Configuration 4, at an angle of 90° with respect to each other, and at an angle of $\gamma = 45^\circ$ to the test subject, produced very good tracking results. The dual-Kinect system was able to cover a large range of motion without losing the wrist position. This configuration was chosen to evaluate the filter performance and comparing the Kinect tracking results to the Vicon motion capture data. The configuration is shown in Figure 5.3.

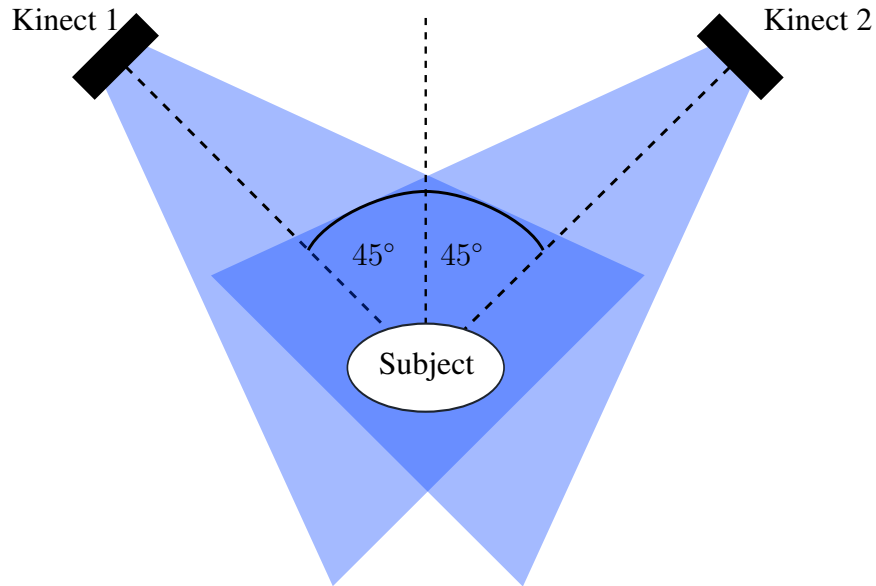


Figure 5.3: Chosen Dual-Kinect Configuration

5.3 Sensor Calibration and Sensor Fusion

Prior to data collection, the two Kinect sensors were calibrated to yield the rotation matrix and translation vector needed to transform points from the coordinate system of Kinect 2 into a common coordinate system, in this case, the coordinate system of Kinect 1.

5.3.1 Calibration

Considering the need for a fast, real-time calibration without any additional calibration objects, the two Kinects can be calibrated using the initial 3D position estimates of the 25 joints. To ensure no joint occlusion, the test subject is required to stand with straight legs and both arms fully extended, pointing sideways in a T-shape (= T-Pose) for less than two seconds, while 50 frames are acquired by both Kinect sensors. Then, the joint position estimates are averaged and fed into the calibration algorithm, which is based on an approach similar to the multiple Kinect Calibration described by Córdova-Esparza et al. [20]. The coordinate transformation is calculated via Corresponding Point Set Registration [37].

Considering two sets of 3D points Set_A and Set_B , with Set_A given in coordinate frame

1 and Set_B given in coordinate frame 2, solving for R and t from:

$$Set_A = R \cdot Set_B + t \quad (5.1)$$

yields the rotation matrix R and translation vector t needed to transform the points from coordinate frame 2 into coordinate frame 1. The problem of finding the optimal rigid transformation matrix can be divided into the following steps:

1. Find the centroids of both datasets.
2. Bring both datasets to the origin.
3. Find the optimal rotation R .
4. Find the translation vector t .

The rotation matrix R is found using Singular Value Decomposition (SVD). Given N Points P_A and P_B from dataset Set_A and Set_B respectively, with $P = [x \ y \ z]^T$, the centroids of both datasets are calculated using:

$$centroid_A = \frac{1}{N} \sum_{i=1}^N P_A^i \quad (5.2)$$

$$centroid_B = \frac{1}{N} \sum_{i=1}^N P_B^i \quad (5.3)$$

The equations needed to find the rotation matrix R are given by:

$$H = \sum_{i=1}^N (P_A^i - centroid_A)(P_B^i - centroid_B) \quad (5.4)$$

$$[U, S, V] = SVD(H) \quad (5.5)$$

$$R = VU^T \quad (5.6)$$

The translation vector t can then be found using:

$$t = -R * centroid_B + centroid_A \quad (5.7)$$

With the derived rotation matrix and translation vector, the joint position data from Kinect 2 can be transformed into the coordinate system of Kinect 1. Both datasets are further processed in the sensor fusion step to yield a fused joint position.

5.3.2 Sensor Fusion

The joint positions collected from both Kinects are used to calculate a weighted fused measurement. In addition to the 3D coordinates of the 25 joints, the Kinect sensor assigns a tracking state to each of the joints, with 0 = 'Not Tracked', 1 = 'Inferred', 2 = 'Tracked'. This information is used to intelligently fuse the data collected by both Kinects. If the tracking state of a joint is 'Tracked' by both Kinects, or the tracking state of the joint is 'Inferred' in both Kinects, then the average position is taken. If a joint is 'Tracked' by one Kinect, but 'Inferred' or 'Not Tracked' by the other, then the fused position only uses data from the 'Tracked' joint. The fused position p_{fused} of each joint can, therefore, be calculated using the position estimates p_1 from Kinect 1 and p_2 from Kinect 2 as follows:

$$p_{fused} = w_1 p_1 + w_2 p_2, \quad (5.8)$$

with weighting factors w_1 and w_2 assigned using the tracking state information for each joint obtained from both Kinects:

$$w_1 = \frac{TrackingState1}{TrackingState1 + TrackingState2} \quad (5.9)$$

$$w_2 = \frac{TrackingState2}{TrackingState1 + TrackingState2} \quad (5.10)$$

5.4 Linear Kalman Filter for Kinect Joint Tracking

To improve tracking of the 25 joints, two versions of a linear Kalman filter were designed based on the state space models presented in Section 3.7.1. The state vector is taken to be the true 3D coordinates of the 25 joints for the zero-velocity model, and the 3D coordinates and velocities of the 25 joints for the constant-velocity model. For the sake of simplicity, the derived Kalman filter equations are presented for only one joint. The same equations can be applied to all 25 joints.

5.4.1 Linear Kalman Filter Implementation

After completing the coordinate transformation and sensor fusion steps described in the previous sections, the fused joint position is fed into the Kalman filter as a measurement. Algorithm 1 summarizes the linear Kalman filter algorithm used for the joint position tracking with the Dual-Kinect system.

Algorithm 1 Linear Kalman Filter Algorithm

1: **function** LINEAR KALMAN FILTER

 Initialize: $k = 0$, Given: $\hat{s}_{0|0}, P_{0|0}, H, Q, R$

Predict

2: State Estimate: $\hat{s}_{k|k-1} = F\hat{s}_{k-1|k-1}$

3: Error Covariance: $P_{k|k-1} = FP_{k-1|k-1}F^T + Q$

Update

4: **if** new measurement available **then**

5: Measurement: z_k , set $k = k + 1$

6: Kalman Gain: $K_k = P_{k|k-1}H^T(HP_{k|k-1}H^T + R)^{-1}$

7: State Estimate: $\hat{s}_{k|k} = \hat{s}_{k|k-1} + K_k(z_k - H\hat{s}_{k|k-1})$

8: Error Covariance: $P_{k|k} = (I - K_kH)P_{k|k-1}$

9: **return** $\hat{s}_{k|k}$

10: **end if**

11: Go to Step 2

12: **end function**

The filter equations remain the same for both the zero and the constant-velocity model.

Depending on the chosen underlying state space model, the state vector, as well as state transition matrix F and the observation matrix H are set accordingly. For the zero-velocity model, the state vector consists of the joint positions $s = [x \ y \ z]^T$, and the matrices take the following form:

$$F = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad H = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (5.11)$$

For the constant-velocity model, the states are the joint positions and the joint velocities $s = [x \ y \ z \ \dot{x} \ \dot{y} \ \dot{z}]^T$, and F and H are calculated as follows:

$$F = \begin{bmatrix} 1 & 0 & 0 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 & \Delta t & 0 \\ 0 & 0 & 1 & 0 & 0 & \Delta t \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \quad (5.12)$$

In both cases, the measurements are the fused joint positions from the Dual-Kinect system.

5.5 Extended Kalman Filter for Kinect Joint Tracking

To implement the extended Kalman filter, nonlinear dynamics of upper body motions are taken into account. The joint positions are calculated using the transformation matrices derived from the kinematic human upper body model presented in Chapter 3. Instead of the joint position and translational joint velocities used with the linear Kalman filter, the joint angles and angular joint velocities are taken to be the states of the system:

$$s = [\theta_1 \ \dots \ \theta_{26} \ \dot{\theta}_1 \ \dots \ \dot{\theta}_{26}]^T \quad (5.13)$$

Assuming constant angular joint velocities, the system has the following description in sampled time:

$$s_{k+1} = f(s_k) + w_k = F s_k + w_k \quad (5.14)$$

$$z_k = h(s_k) + v_k \quad (5.15)$$

The process noise w_k and the measurement noise v_k are assumed to be zero mean, Gaussian noise with covariance Q_k and R_k , respectively. The state transition matrix is given by:

$$F = \begin{bmatrix} \mathbb{I}_{26 \times 26} & \Delta t \mathbb{I}_{26 \times 26} \\ 0_{26 \times 26} & \mathbb{I}_{26 \times 26} \end{bmatrix} \quad (5.16)$$

with sampling time Δt . In the measurement model, the 3D positions of the upper body joints can be calculated using the DH-Parameters and transformation matrices for the upper body model derived in Chapter 3. Recalling the transformation matrices

$$T_i^{i+1} = \begin{bmatrix} \cos \theta_i & -\sin \theta_i \cos \alpha_i & \sin \theta_i \sin \alpha_i & a_i \cos \theta_i \\ \sin \theta_i & \cos \theta_i \cos \alpha_i & -\cos \theta_i \sin \alpha_i & a_i \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (5.17)$$

the spatial configuration of the upper body model is defined for given link lengths L_1, \dots, L_{10} and joint angles $\theta_1, \dots, \theta_{26}$. Using the transformation matrices $T_{i-1}^i = T_{i-1}^i(\theta_i)$, the position of the i^{th} joint $p_i = [x_i \ y_i \ z_i]^T$ can be expressed as a function of i joint angles:

$$\begin{bmatrix} p_i \\ 1 \end{bmatrix} = T_0^1 T_1^2 \dots T_{i-1}^i \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \quad (5.18)$$

$$= h(\theta_1, \dots, \theta_i) = h(s) \quad (5.19)$$

The system can be linearized about the current state estimate using the Jacobian:

$$H_k = \left. \frac{\partial h}{\partial s} \right|_{s=\hat{s}_k} \quad (5.20)$$

For each time step k , the linearized function is evaluated at the current state estimate.

The form of the underlying transformation matrices T_{i-1}^i is dependent on the body segment lengths L_1-L_{10} . Therefore, $h(s)$ is initialized with corresponding values for the body segment lengths of each individual test subject obtained during the Dual-Kinect calibration process.

5.5.1 Extended Kalman Filter Implementation

Algorithm 2 summarizes the extended Kalman filter algorithm used for upper body joint tracking.

Algorithm 2 Extended Kalman Filter Algorithm

1: **function** EXTENDED KALMAN FILTER

 Initialize: $k = 0$, Given: $\hat{s}_{0|0}$, $P_{0|0}$, Q , R

 Initialize $h(\theta)$ with body segment lengths L_1-L_{10}

Predict

2: State Estimate: $\hat{s}_{k|k-1} = F\hat{s}_{k-1|k-1}$

3: Error Covariance: $P_{k|k-1} = FP_{k-1|k-1}F^T + Q$

Update

4: **if** new measurement available **then**

5: Measurement: z_k , set $k = k + 1$

6: Linearize h: $H_k = \left. \frac{\partial h}{\partial s} \right|_{s=\hat{s}_k}$

7: Kalman Gain: $K_k = P_{k|k-1}H_k^T(H_kP_{k|k-1}H_k^T + R_k)^{-1}$

8: State Estimate: $\hat{s}_{k|k} = \hat{s}_{k|k-1} + K_k(z_k - H_k\hat{s}_{k|k-1})$

9: Error Covariance: $P_{k|k} = (I - K_kH_k)P_{k|k-1}$

10: **return** $h(\hat{s}_{k|k})$

11: **end if**

12: Go to Step 2

13: **end function**

5.5.2 Handling Missing Data

One advantage of the underlying state space model for the Kalman filter is that a missing observation can easily be integrated into the filter framework. If at time step k a joint's position is lost by both Kinect sensors (tracking state 'Not Tracked' for Kinect 1 and Kinect 2), then the vector $z_k - H_k \hat{s}_{k|k-1}$ and the Kalman gain K_k are set to zero. Thus, the update only follows the state space model:

$$\hat{s}_{k|k} = F \hat{s}_{k-1|k-1} \quad (5.21)$$

$$P_{k|k} = F P_{k-1|k-1} F^T + Q \quad (5.22)$$

This approach can be applied to the implementations of both the linear Kalman filter and the extended Kalman filter.

5.6 Experimental Setup

5.6.1 Tracked Motions

Joint tracking with the Dual-Kinect system utilizing the Kalman filters was tested with three test motions: a two-handed wave, a two-handed "slow down" signal, and a torso twist. The first two motions were previously tested with the single-Kinect motion capture (see Section 4.3). To investigate the effect of joint occlusion on the Dual-Kinect system, the third motion from the previous study was replaced with a torso twist motion. The test subject rotated her upper body from side to side about 90 degrees. This motion causes joint occlusion of the elbow, wrist, and hand.

Starting from the T-Pose, the test subject performed five repetitions of all three test motions. To clearly distinguish the between different motions in the recorded data, the subject returned to the T-Pose for about two seconds before switching to a new motion. Data was recorded continuously until five repetitions for each of the three motions had

been completed, and the subject had returned to the T-Pose.

5.6.2 Marker-based Tracking

To evaluate the performance of the Dual-Kinect system, tracking data for the three test motions was compared to marker-based tracking data recorded with a Vicon 3D motion capture system at the Indoor Flight Facility at Georgia Tech. For the marker-based motion capture with the Vicon system, the full body Plug-in-Gait marker setup was used. The marker setup uses 39 retroreflective markers and can be used with the Plug-in-Gait model, which is a well-established, and commonly-used, model for marker-based motion capture.

A complete list of the markers used in the experiments and details on the marker placement can be found in Appendix B. Figure 5.4 shows the locations of the markers for the full body Plug-in-Gait model. Figure 5.5 shows the subject standing in the T-Pose while facing the Dual-Kinect setup. Figure 5.6 shows the test subject wearing the motion capture suit with the attached markers.

5.6.3 Marker Trajectory Data Processing

Motion capture data from the Vicon system was processed in the Vicon Nexus 2.5 and Vicon BodyBuilder 3.6.3 software (Vicon Motion Systems, Oxford, UK). Marker trajectories were filtered using a Woltring filter [36]. Gaps in the marker data with durations < 20 frames (< 0.2 seconds) were filled using spline interpolation. In order to compare the performance of the Dual-Kinect system to the marker-based Vicon tracking, joint center locations corresponding to the joints tracked by the Kinect system were calculated from the marker trajectories in Vicon BodyBuilder.

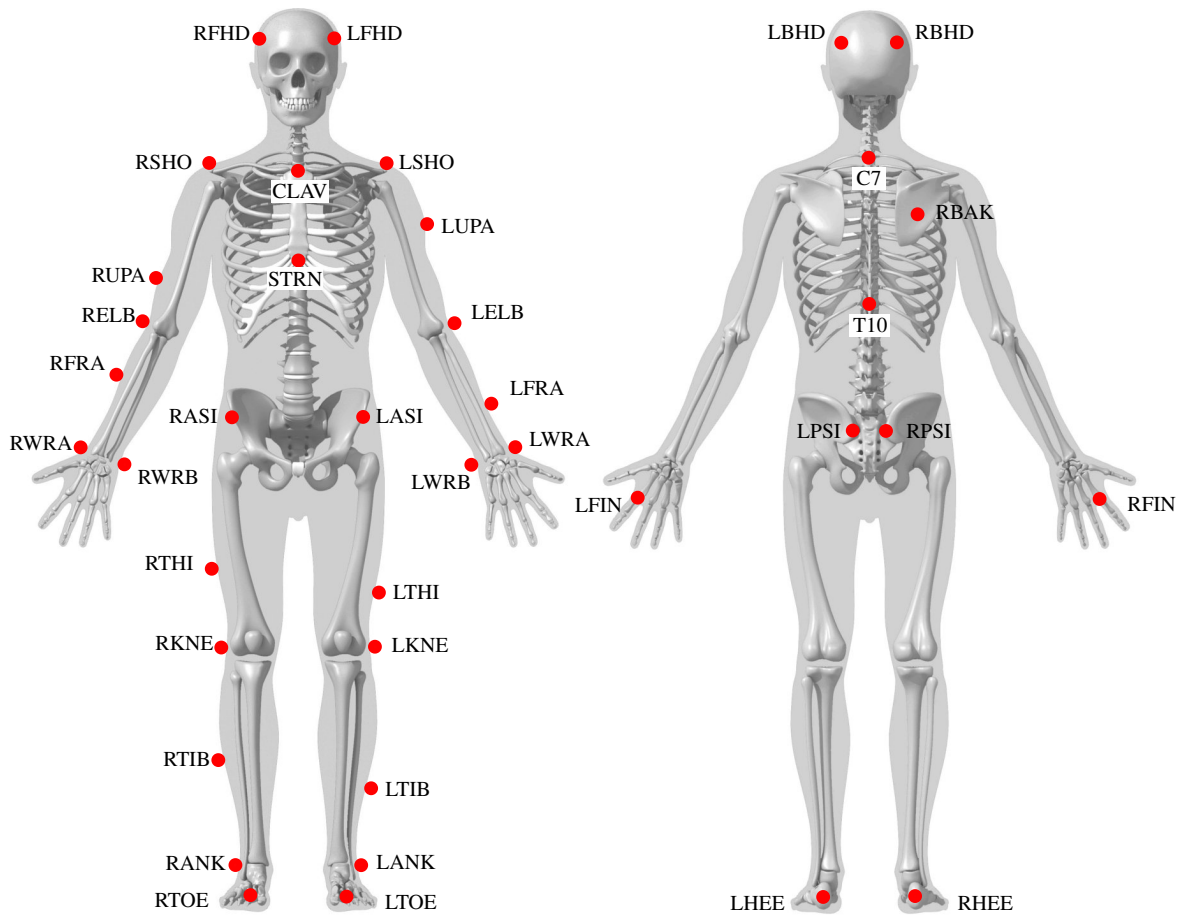


Figure 5.4: Marker Placement for Full Body Plug-in-Gait Model¹



Figure 5.5: Test Subject Standing in T-Pose and Dual-Kinect Test Setup

¹This image was adapted from <http://www.idmil.org/mocap/Plug-in-Gait+Marker+Placement.pdf> and https://www.c-motion.com/v3dwiki/index.php/Tutorial:_Plug-In_Gait_Full-Body



Figure 5.6: Marker Placement for Full Body Plug-in-Gait Model on Test Subject

5.7 Results and Comparison with Vicon Motion Capture

In this section, results from tracking experiments with two variants of the linear Kalman filter and the Extended Kalman filter (EKF) are presented. While the first variant of the linear Kalman filter (LKF1) uses a zero-velocity model, the second variant (LKF2) uses a constant-velocity motion model. The position estimates are compared to the raw data from the Kinect sensors, and to joint position data obtained from marker-based motion capture. The joint positions derived from the Vicon system were assumed to be the true positions of the joints.

5.7.1 Linear Kalman Filter

During the experiments, it was noted that the differences between the two variants of the linear Kalman filter were in many cases small, but became larger as the process covariance

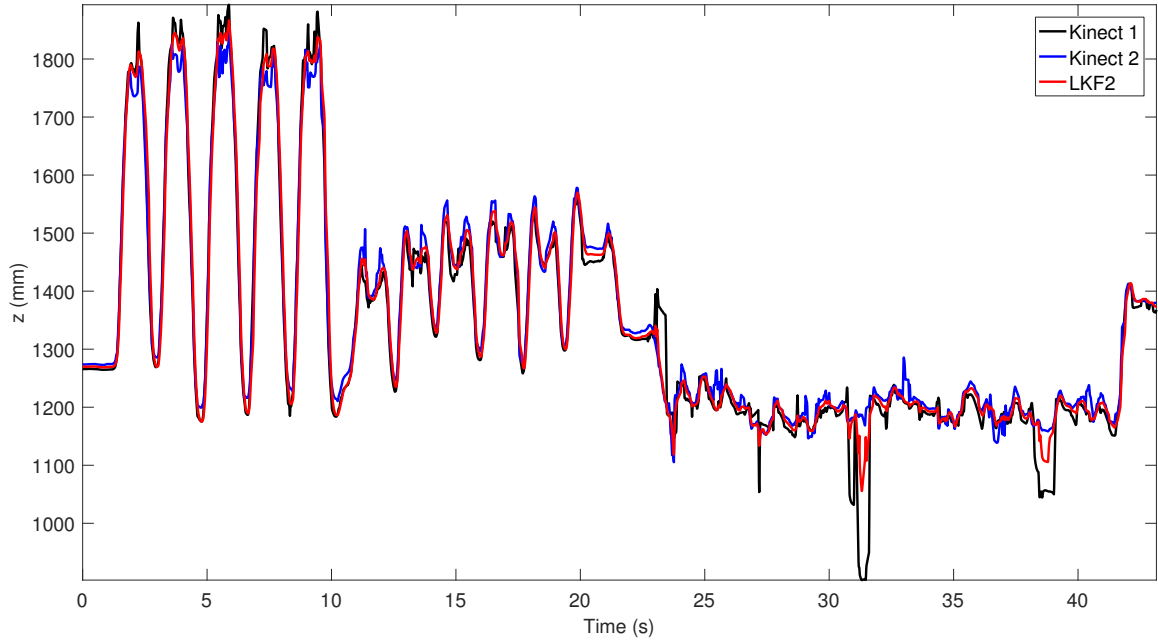


Figure 5.7: Z Component of the Left Wrist Joint Trajectory with Sensor Fusion and Linear Kalman Filter with Constant Velocity Model ($Q = 0.005$)

was decreased. This result is to be expected, as a smaller process covariance means the filter relies more on the underlying motion model and less on actual observations. Figure 5.7 shows the z component of the left wrist joint position for the recorded test motions estimated with the linear Kalman filter using the constant-velocity model (LKF2). The position estimate is compared with the raw data acquired by Kinect 1 and 2.

Figure 5.8 shows the difference between the raw data and the filtered data for the z component of the left wrist position estimate. The greatest deviation between the raw data and the LKF2 output was observed during the torso twist motion, as the wrist moved behind the torso during the motion, and was therefore occluded. The average deviation between the Kinect 1 and the LKF2 output was 19.6113 mm, and the maximum deviation between Kinect 1 and the LKF2 output was 246.0466 mm. The average deviation between the Kinect 2 and LKF2 was 16.3035 mm and the maximum deviation between Kinect 2 and LKF2 was 131.5598 mm.

To compare the joint tracking data from Kinect with Vicon data, the filter outputs were aligned with the Vicon data in terms of motion timing and were transformed into the Vi-

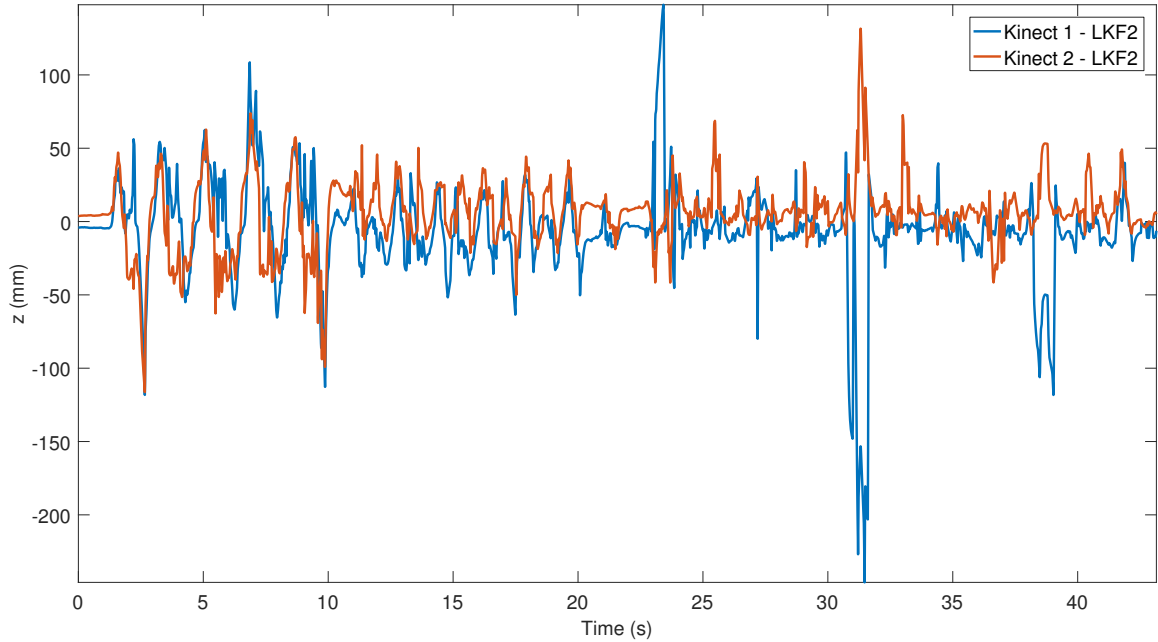


Figure 5.8: Error Between Kinect 1 and 2 and LKF2 Outputs for the Z Component of the Left Wrist Joint Trajectory

Table 5.2: Mean Absolute Error (MAE) and Maximum Absolute Error (MAD) for LKF2 for Tracking the Position of the Left Wrist. All values in mm.

	x	y	z
MAE	36.9908	16.1586	22.9192
MAD	163.7479	157.4331	125.5755

con’s coordinate system. Because the Kinect samples at a rate of approximately 30 Hz, the filter outputs were interpolated using linear interpolation to match the Vicon’s sampling rate of 100 Hz.

Figure 5.9 shows the position estimate of the left wrist from the LKF2. The results are compared to the joint trajectory obtained with the Vicon system. Figure 5.10 shows the difference between the Vicon and the LKF2 data for tracking the left wrist position. The mean and maximum deviations between the LKF2 output and the Vicon data are listed in Table 5.2. The mean deviation was smallest in the y component of the position estimate, and was worst in the x direction. The maximum deviation also occurred in the x direction.

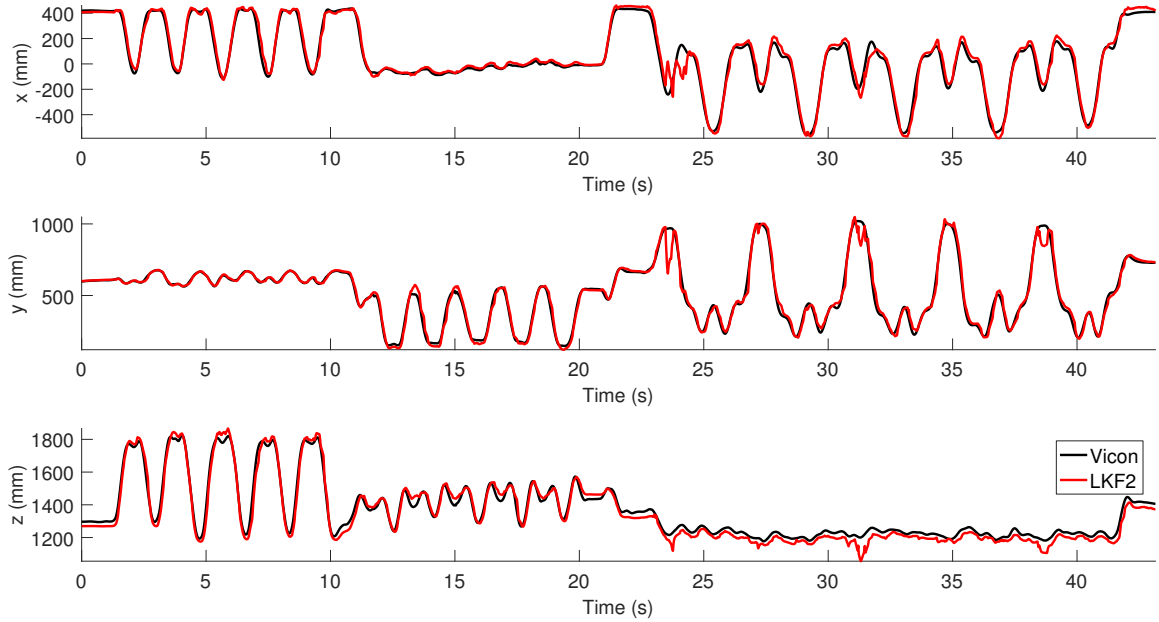


Figure 5.9: Left Wrist Joint Trajectory with Sensor Fusion and LKF2 ($Q = 0.005$) - Comparison with Vicon Motion Capture Data

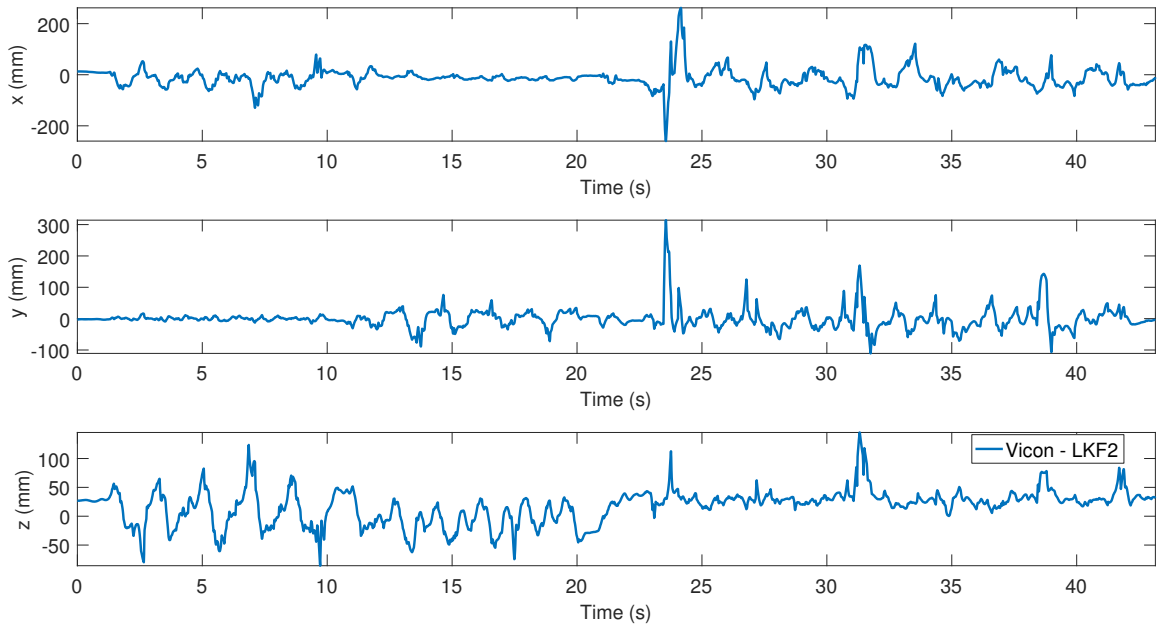


Figure 5.10: Error Between the LKF2 Output and the Vicon Data for the Left Wrist Position

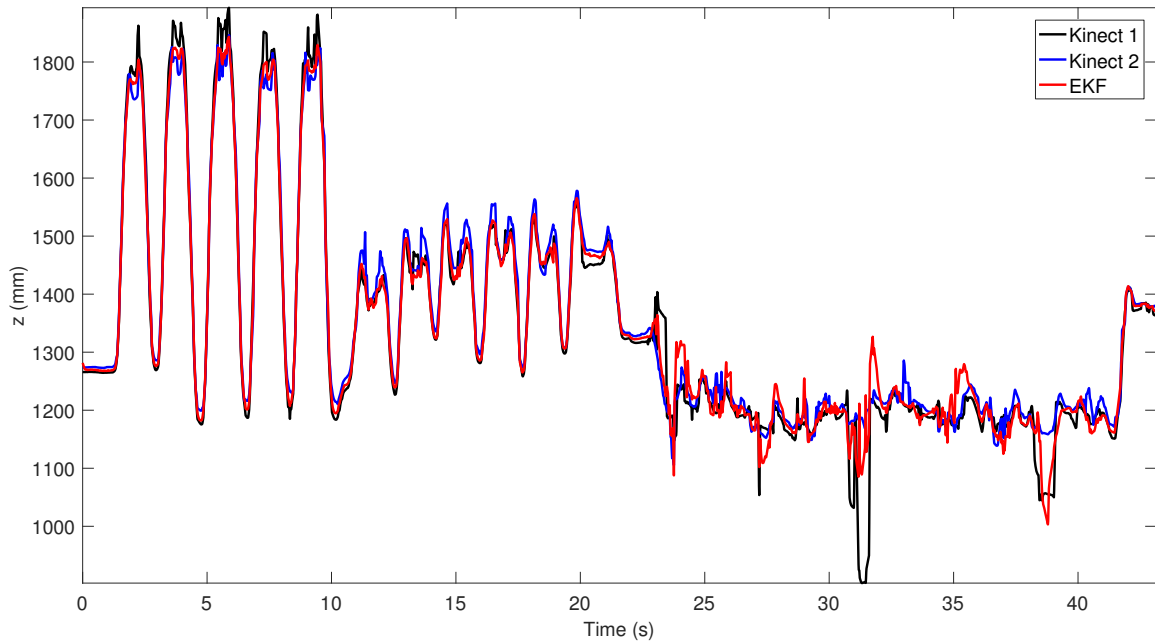


Figure 5.11: Z Component of the Left Wrist Joint Trajectory with Sensor Fusion and Extended Kalman Filter ($Q = 0.0001$)

Figure 5.9 shows that the left wrist position was closely tracked for the wave motion (from $t=0$ s until $t=10$ s) and the “slow down” motion (from $t=11$ s until $t=21$ s). During the torso twist motion starting at $t=23$ s, however, there was some discrepancy between Kinect and Vicon tracking data for extreme positions, when the wrist moved out of the field of view of both Kinect sensors. Generally, the wrist could be tracked well for the majority of the test motions.

5.7.2 Extended Kalman Filter

Figure 5.11 presents the z component of the left wrist joint trajectory from the EKF output, as well as the raw data acquired by Kinect 1 and 2. The wrist position could be tracked closely for the first two motions (two-handed wave and “slow down signal”). However, the EKF outputs from tracking the torso twist motion were not as smooth as the linear Kalman filter outputs. To better compare the tracking performance of the different filter variants, the same data sets obtained from Kinect 1 and 2 were used.

Figure 5.12 compares the wrist position estimate from the EKF with the LKF2 outputs

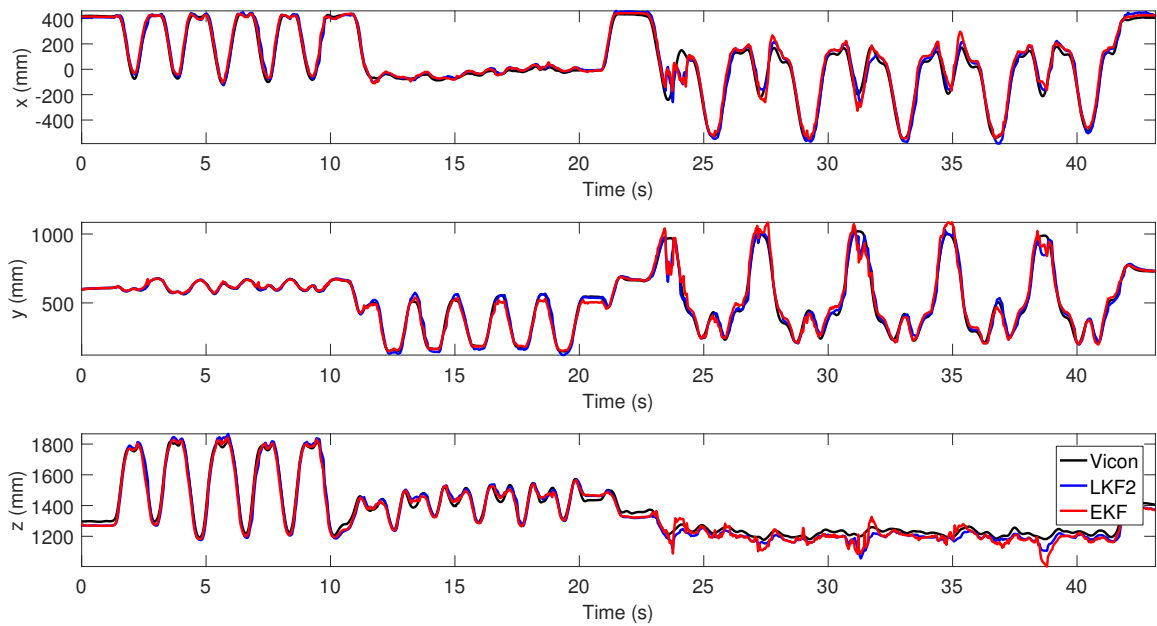


Figure 5.12: Left Wrist Joint Trajectory from EKF - Comparison with Vicon and LKF2 Outputs

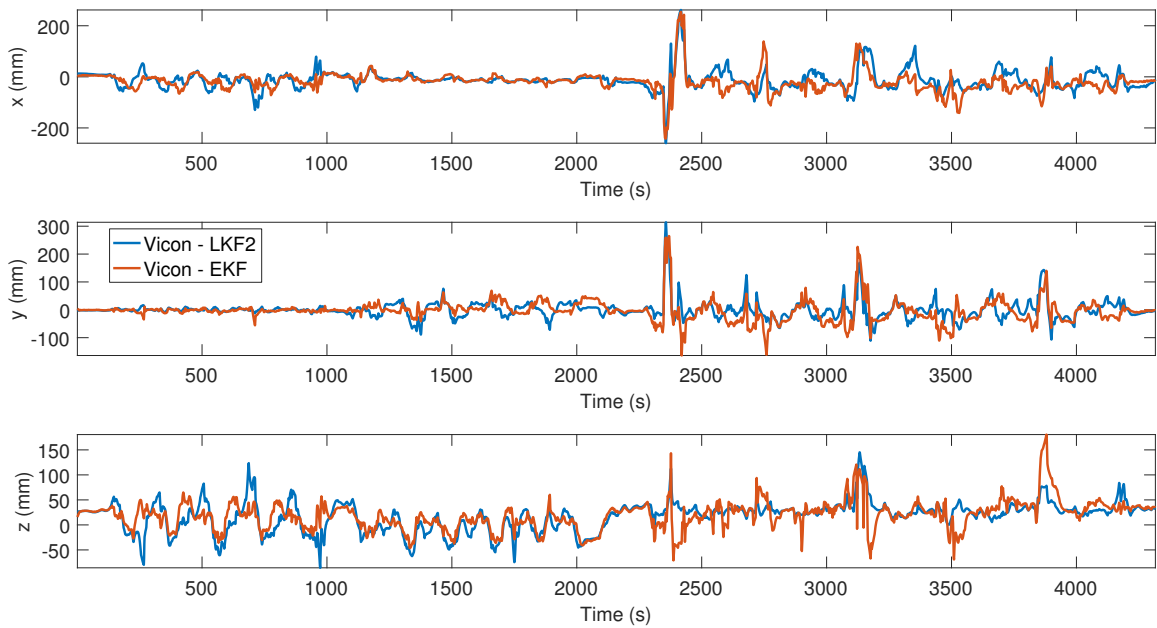


Figure 5.13: Error Between Vicon and LKF2, and Vicon and EKF Outputs for the Left Wrist

Table 5.3: Mean Absolute Error (MAE) for all Filter Variants Averaged Over Ten Upper Body Joints

Filter variant	MAE (mm)		
	x	y	z
LKF1	36.4419	37.0706	30.4291
LKF2	36.5739	36.9309	31.0161
EKF	37.5273	40.2169	32.3851

and the data obtained with the Vicon system. Figure 5.13 shows the deviation between each filter output and the Vicon data. For the first two tracked motions, differences between the filter outputs are very small. For the torso twist motion, the linear Kalman filter provides a more stable and smoother tracking of the joint position.

To evaluate accuracy of the tracking with the different variants of the Kalman filters, the mean absolute errors in x, y, and z position between the filter outputs and joint position data collected with the Vicon system were calculated for ten joints considered in the kinematic upper body model (see Chapter 3): SpineMid, SpineShoulder, ShoulderLeft, ElbowLeft, WristLeft, HandTipLeft, ShoulderRight, ElbowRight, WristRight, and HandTipRight. The values are listed in Appendix E.

Table 5.3 lists the mean absolute error in x, y, and z position averaged over the ten joints considered in the upper body model. In general, the different filter variants tracked the motion of the joints with similar accuracy, with the linear Kalman filter using a zero-velocity model (LKF1) performing slightly better than the linear Kalman filter using a constant-velocity model (LKF2) and the Extended Kalman filter (EKF). The most accurate results in terms of least mean absolute error averaged over all joints were achieved while tracking the z coordinate of the position (along the vertical axis). In general, mean absolute error was greatest in the y direction (corresponds to the axes extending from the Kinect sensors to the test subject).

The Kinect’s out-of-the-box joint tracking algorithm is not based on a kinematic model

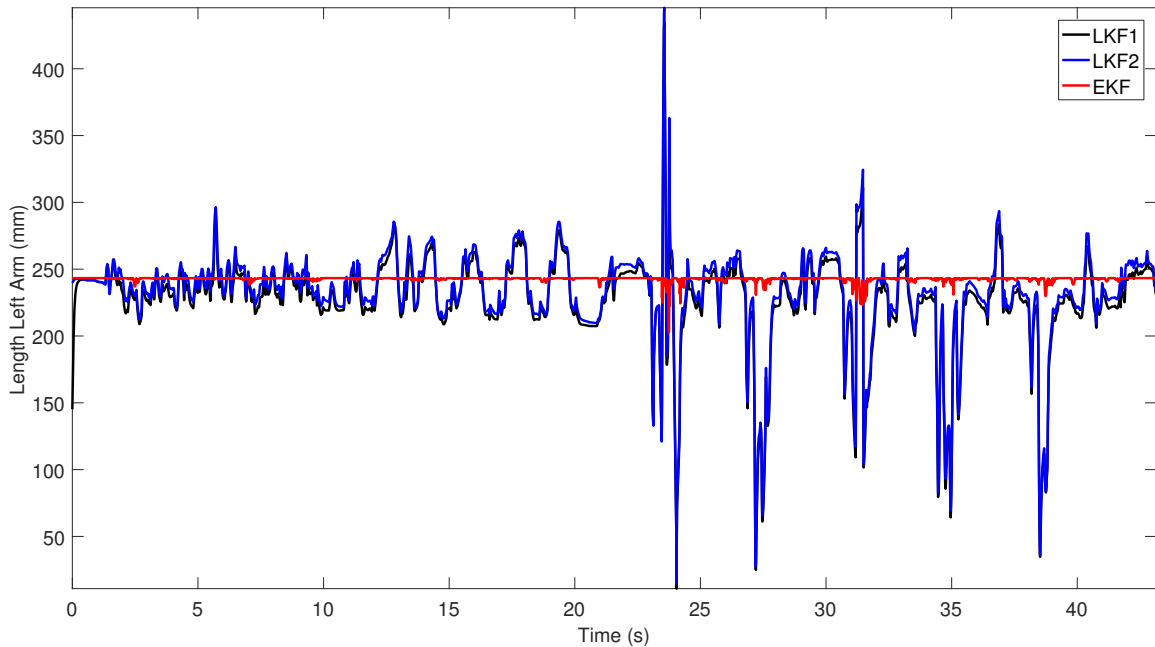


Figure 5.14: Length of the Left Arm during Motion Capture Trial

for the human body. As a consequence, the distances between neighboring tracked joints i.e. the limb lengths of the estimated skeleton are not kept constant. This can lead to unrealistic variation of the body segment lengths and “jumping” of the joint positions. The extended Kalman filter used in this study uses the novel kinematic human upper body model described in Sections 3.3 and 3.4. By using the model, constant limb lengths are enforced during the joint tracking.

Figure 5.14 shows the length of the left arm calculated from the different filter outputs. The arm length was measured from elbow joint to wrist joint. The outputs from the EKF show that by definition, the arm length was kept constant throughout the motion, while the estimates from the linear Kalman filters show that the estimated arm length varied over time.

5.8 Tracking with Garments of Different Fit

This section presents an experimental protocol to evaluate how the fit of clothing affects motion capture and joint tracking with the dual-Kinect system. Most motion capture sys-

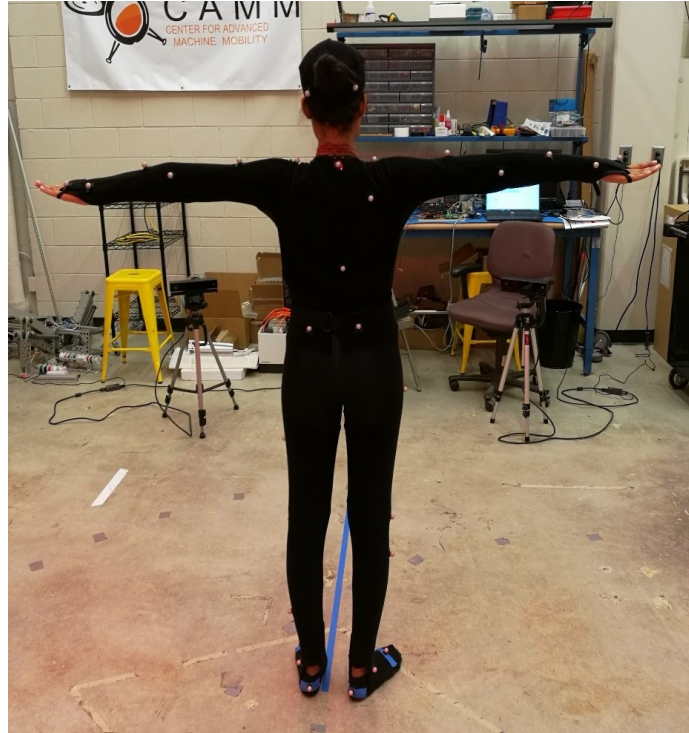


Figure 5.15: Test Subject wearing a tight-fitting Motion Capture Suit with retroreflective Markers

tems require extremely tight fitting clothes, very little clothing, or a special suit to track joint position and angles accurately. Moreover, a large number of these systems are marker based systems that use retroreflective markers to track joints. In the event that the test subject wears glasses, light colored clothing, or reflective jewelry, the data becomes noisy.

Figure 5.15 shows a test subject in a motion capture suit with retroreflective markers that are required by the Vicon system. Given that the Kinect sensor uses RGB and depth data to track a human-shaped silhouette, it does need a reasonable view of the joint motions that compose the human body motion. It is clear that the clothing worn by the test subject obscures the visible joint motion to some degree. This section demonstrates that the dual-Kinect system can track human motion even when relatively loose clothing is worn by the test subject.

The Kinects were placed according in Configuration 4 (see Section 5.2.2), at an angle of 90° with respect to each other, and at an angle of $\gamma = 45^\circ$ to the test subject. The test

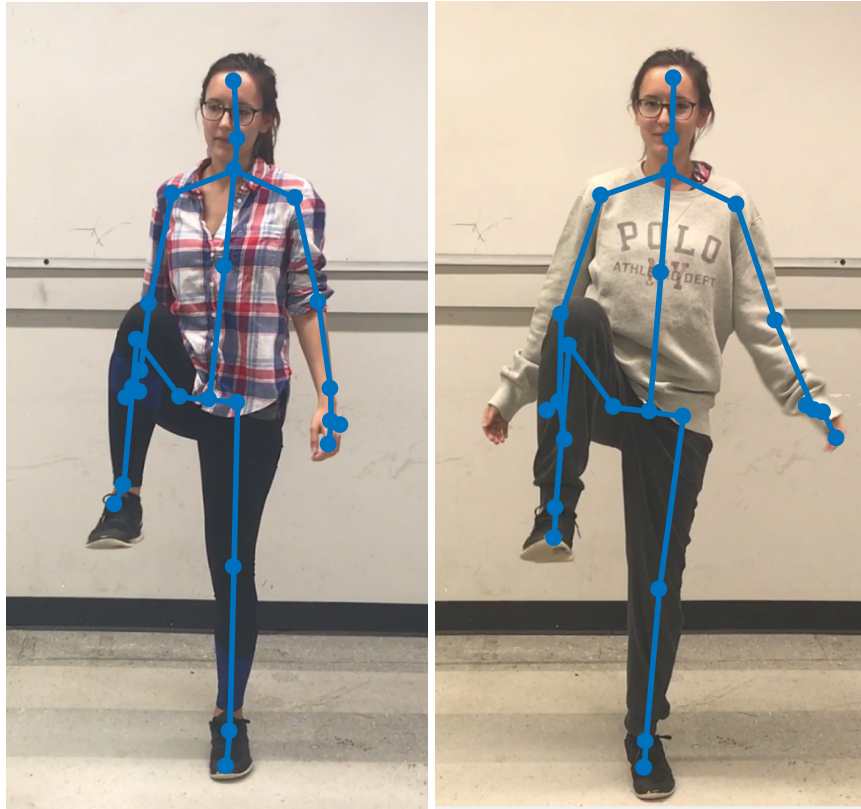


Figure 5.16: Test Subject performing Test Motion with Tight-Fitting (left) and Loose-Fitting Clothes (right)

subject executed characteristic motion performed by people to test fit of garments, such as the torso twist, calf extensions, and squats. Joint position data was collected for two trials, one with fitted clothing, and the other with loose clothing. Figure 5.16 shows the outfits used for the two trials. The tighter outfit is shown on the left, while the looser outfit is shown on the right. The skeleton tracked by the dual-Kinect system is overlaid on the RGB frame of a video recording of the test motions.

Figure 5.17 shows the joint position plot for the SpineBase from the two trials. The subject performed two calf extensions and a squat. In the z component of the tracked joint, the squat motion can be clearly identified starting from $t=20$ s until $t=22.5$ s for tracking with both tight-fitting and loose-fitting clothes. Because the test subject changed starting positions in between the two trials, there was an offset in the x and y component of the tracked position. It could be observed that loose fitting clothing did not significantly de-

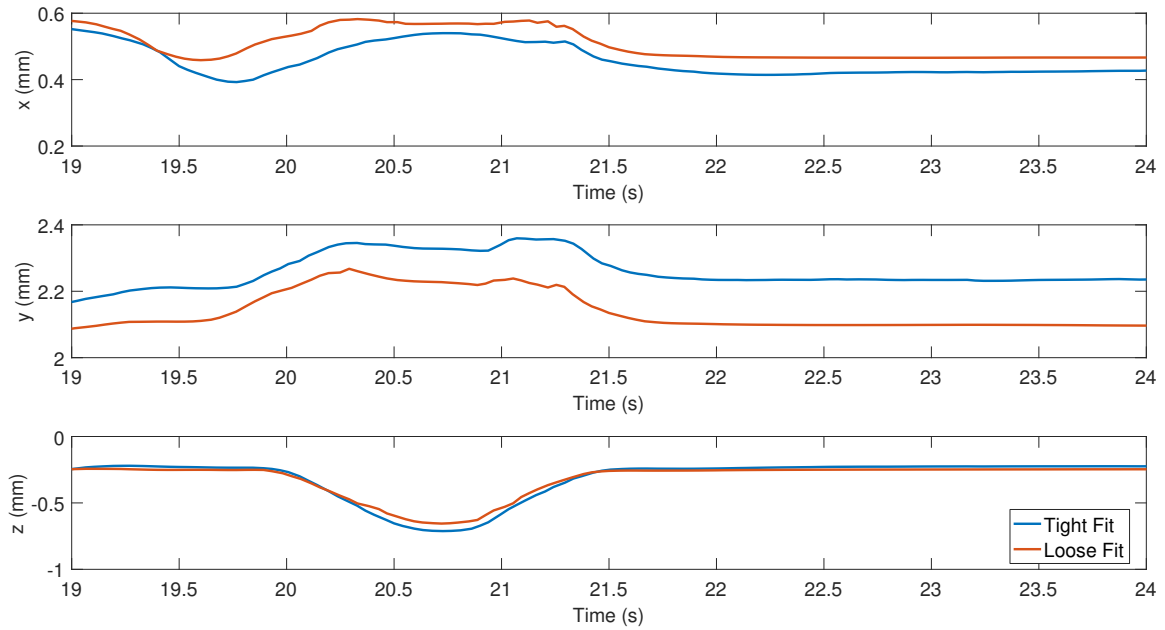


Figure 5.17: SpineBase Trajectory for Test Motions with Tight-Fitting and Loose-Fitting Clothes

grade the tracking ability of the dual-Kinect system. Because the tracking does not fail with the loose fit of the clothing, it can be concluded that, in general, the dual-Kinect system is a robust tool to capture motions performed by clothed test subjects.

5.9 Graphical User Interface for Real-Time Joint Tracking with Dual-Kinect

To visualize the real-time tracking with the Dual-Kinect system, a graphical user interface (GUI) was implemented in MATLAB. Figure 5.18 shows the implemented GUI. Figure 5.19 shows example results for tracking the test motions ((a)-(c) torso twist, (d)-(f) two-handed wave motion). The tracked skeletons from both Kinect sensors, as well as the combined resulting skeleton are plotted for each time frame. The GUI can be used for calibration, recording tracking data, and replaying the tracked results.

A red colored joint indicates that the Kinect sensor has either lost the joint's position completely, or the tracking state of the joint is 'Inferred'. As shown in Figure 5.19, the fused data compensates for occlusion of the joints of the right arm, and uses the more realistic position data from Kinect 2 to calculate the position estimation.

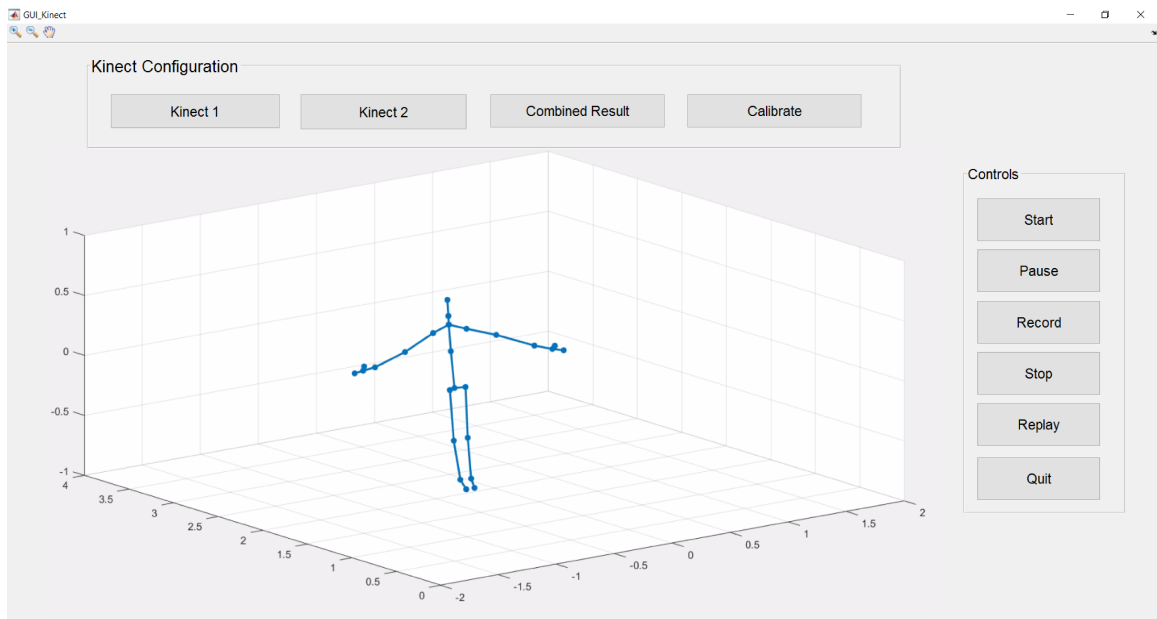
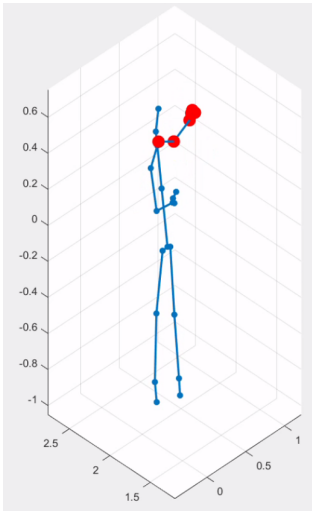
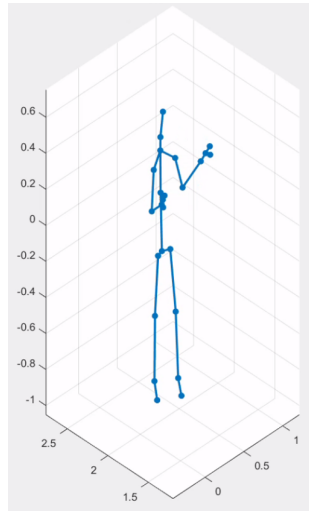


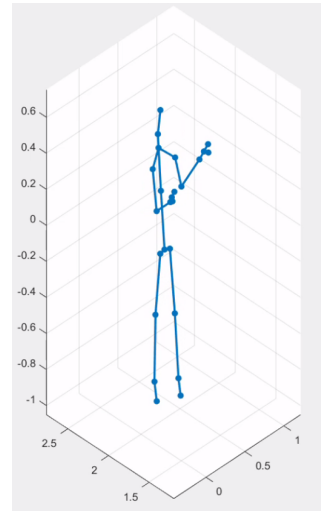
Figure 5.18: GUI for Real-Time Tracking with the Dual-Kinect System



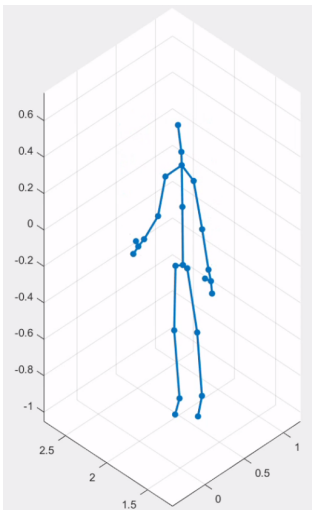
(a) Kinect 1



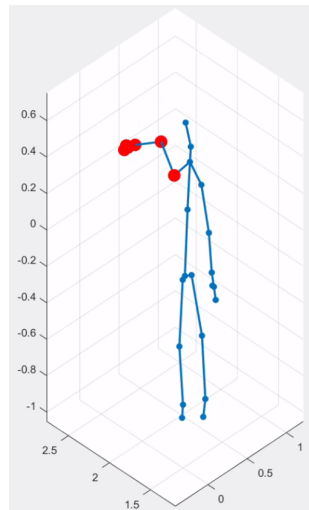
(b) Kinect 2



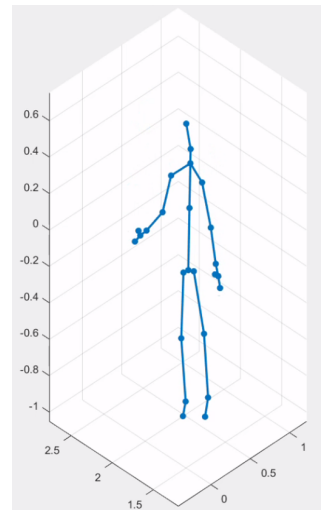
(c) Combined Result



(d) Kinect 1



(e) Kinect 2



(f) Combined Result

Figure 5.19: Real-time Tracking Results with Dual-Kinect and Linear Kalman Filter: (a)-(c) Torso Twist, (d)-(f) Wave Motion

CHAPTER 6

CONCLUSION

6.1 Discussion

Chapter 1 presented a background and motivation for the problem of human motion capture and human modeling in the context of evaluating clothing fit. Chapter 2 explored prior work in human body and human motion modeling and reviewed established models for the lower and upper body. In Chapter 3, a novel kinematic human upper body model was developed. The model is composed of a torso segment and two arms, and has 26 degrees of freedom. A robotics based approach using Denavit-Hartenberg parameters was used to define the upper body pose via joint angles.

Chapter 4 contained a baseline evaluation of markerless motion capture with a single Kinect sensor, and presented results from tracking a known trajectory of a robot arm and human motion capture experiments. The human motion capture experiments were compared to data acquired with a marker based Vicon 3D system. It was concluded that in general, the Kinect could track motions in a reliable manner, and with acceptable accuracy for the intended application of evaluating clothing fit. However, due to it relying on only one camera viewpoint, occlusion can lead to problems while tracking more complex human motions.

In an attempt to overcome this limitation of the Kinect sensor, a dual-Kinect setup using two sensors was developed in Chapter 5. Experiments with different sensor placements tracking human joints during a set of three test motions were conducted. A method of fusing data from two Kinects was presented. In order to further improve the joint tracking, a Kalman filter and an extended Kalman filter derived on the developed kinematic upper body model were designed and integrated into the dual-Kinect setup. The tracking performance

was evaluated by comparing the data to joint positions calculated from marker tracking with the Vicon system. Both Kalman filters lead to reduced jitter in the Kinect data. In addition, the extended Kalman filter ensures constant body segment lengths, and thus produces more realistic joint position estimation than the raw Kinect data which does not enforce this constraint.

A drawback of the EKF is that poor choices of initial conditions and noise properties can lead to the filter becoming unstable. Currently, the linear Kalman filter has been implemented into the real-time joint tracking setup with the dual Kinect system. However, the EKF has only been tested offline with recorded datasets.

The proposed setup offers a low-cost, markerless, and portable alternative to marker based motion tracking. It eliminates the disadvantage of tedious marker setup and subject preparation time. Human upper body motion recorded with the proposed setup can be used as input for virtual character animation. Joint trajectories for motions typically performed while trying on clothes can be animated based on experimental data. To evaluate clothing fit, a cloth model can be put on top of the virtual character.

In summary, the main contributions of this thesis are:

1. Development of a novel, robotics-based model for the human upper body with 26 degrees of freedom.
2. Development of a dual-Kinect motion tracking system that enables real-time human motion capture.
3. Design and implementation of two linear Kalman filters and an Extended Kalman filter based on the developed upper body model.
4. Experimental validation of the developed tracking system through comparison with a marker based Vicon motion capture system.

6.2 Future Work

The proposed dual-Kinect system can easily be used to create pose libraries for a variety of human motions, e.g. for ergonomic workspace assessments or physical rehabilitation. In order to model the whole human body, the proposed upper body model can be extended. This can be achieved by adding joints and links representing the lower limbs, the neck, and head. To further increase the task space of the tracking system and even achieve a 360°-view, additional Kinect sensors can be added to the dual-Kinect setup while using the same sensor fusion method. The use of the proposed model in conjunction with a clothing model can be further explored with e.g. finite-element models or mass-spring-damper models for cloth drape simulation.

Appendices

APPENDIX A
ROBOT TRACKING TRAJECTORIES

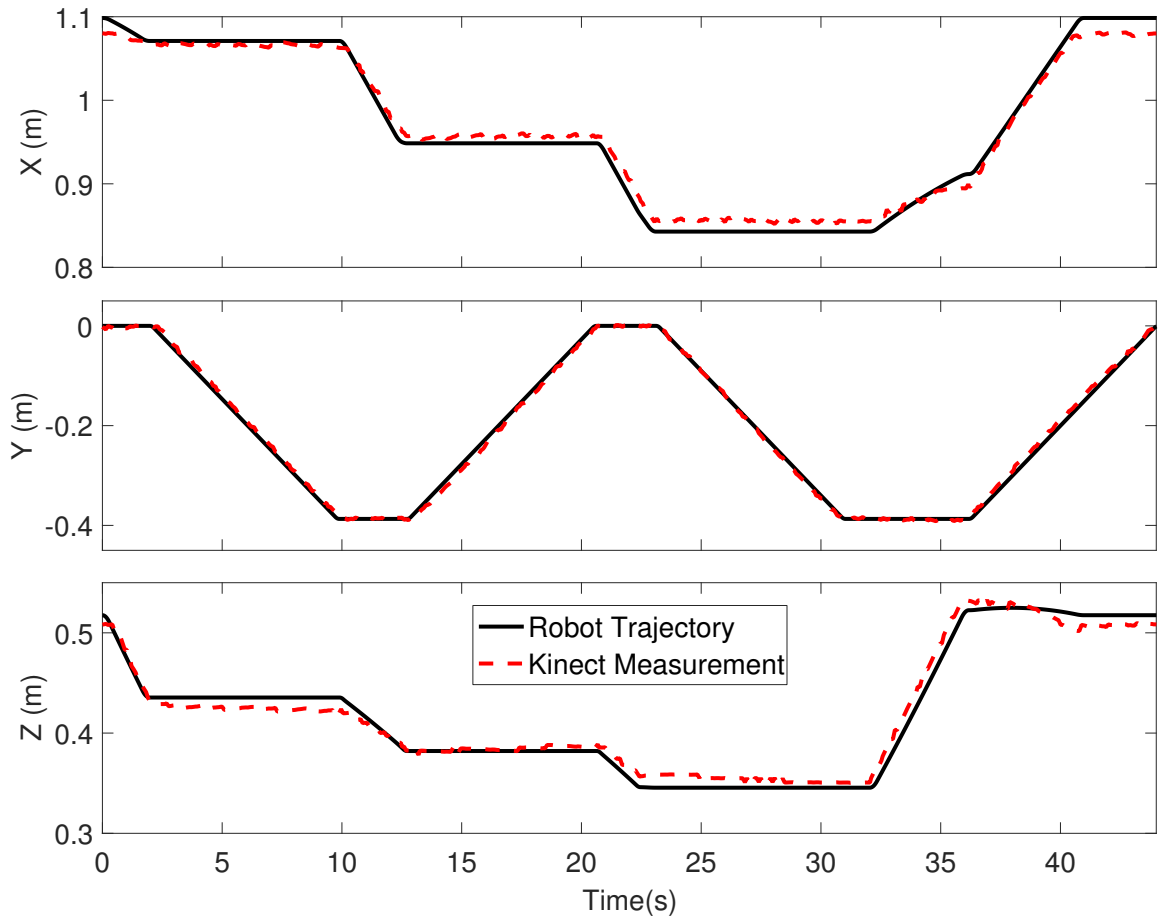


Figure A.1: XYZ Coordinates of End Effector vs. Kinect Measurement for Trial 2.

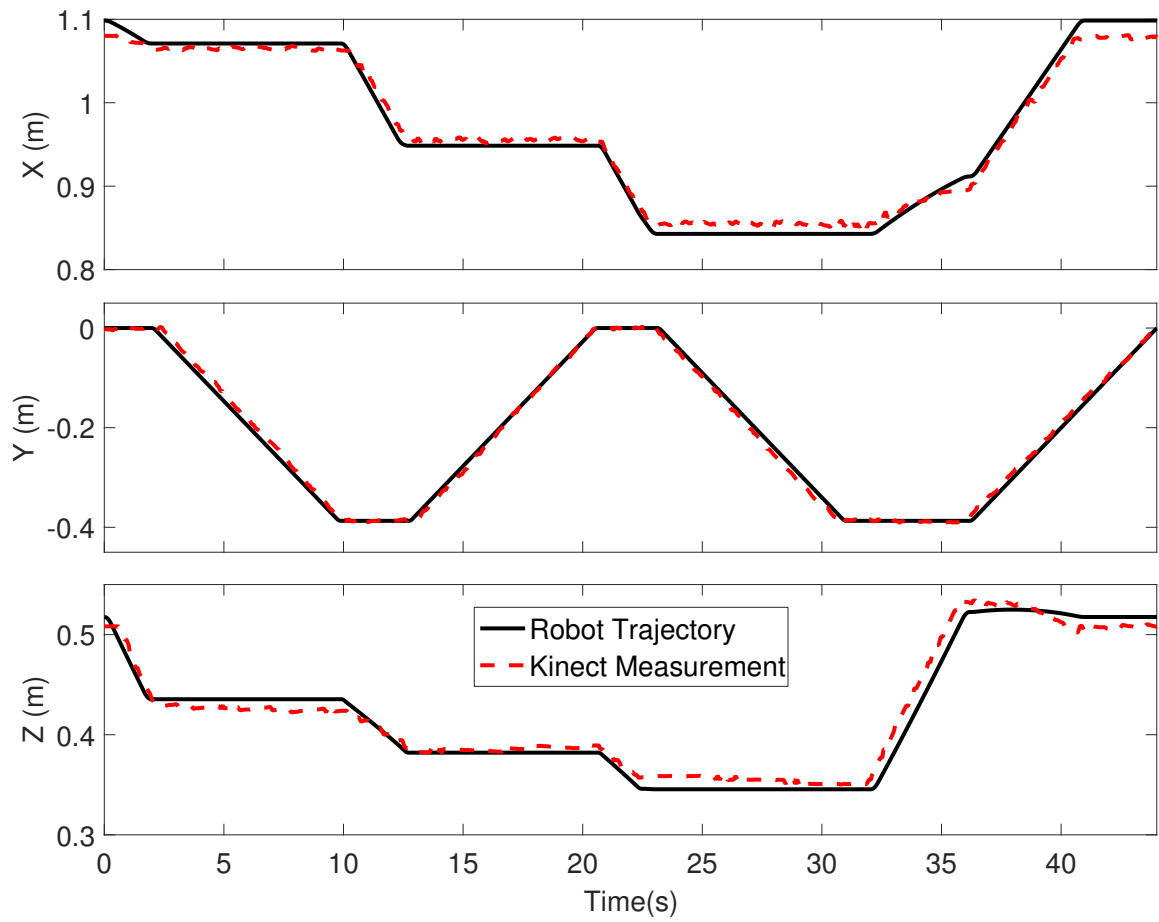


Figure A.2: XYZ Coordinates of End Effector vs. Kinect Measurement for Trial 3.

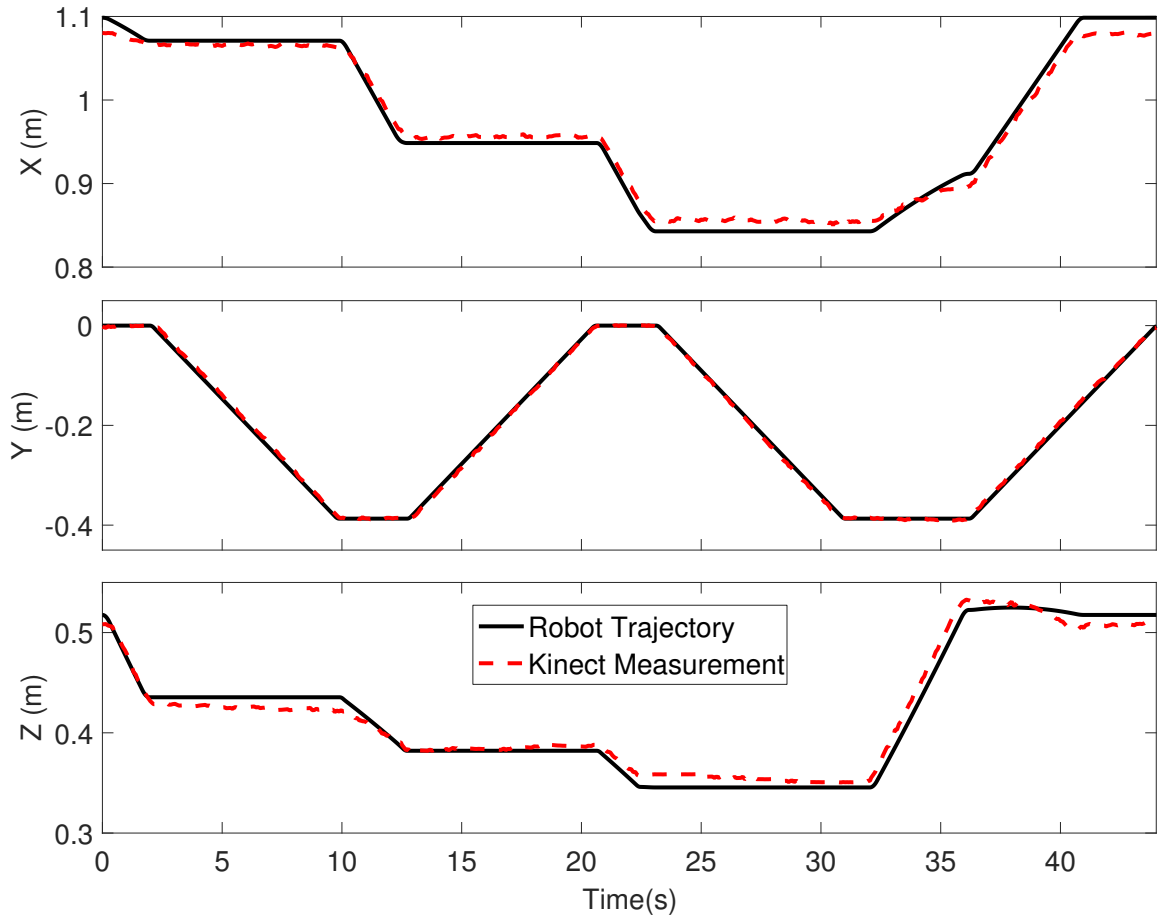
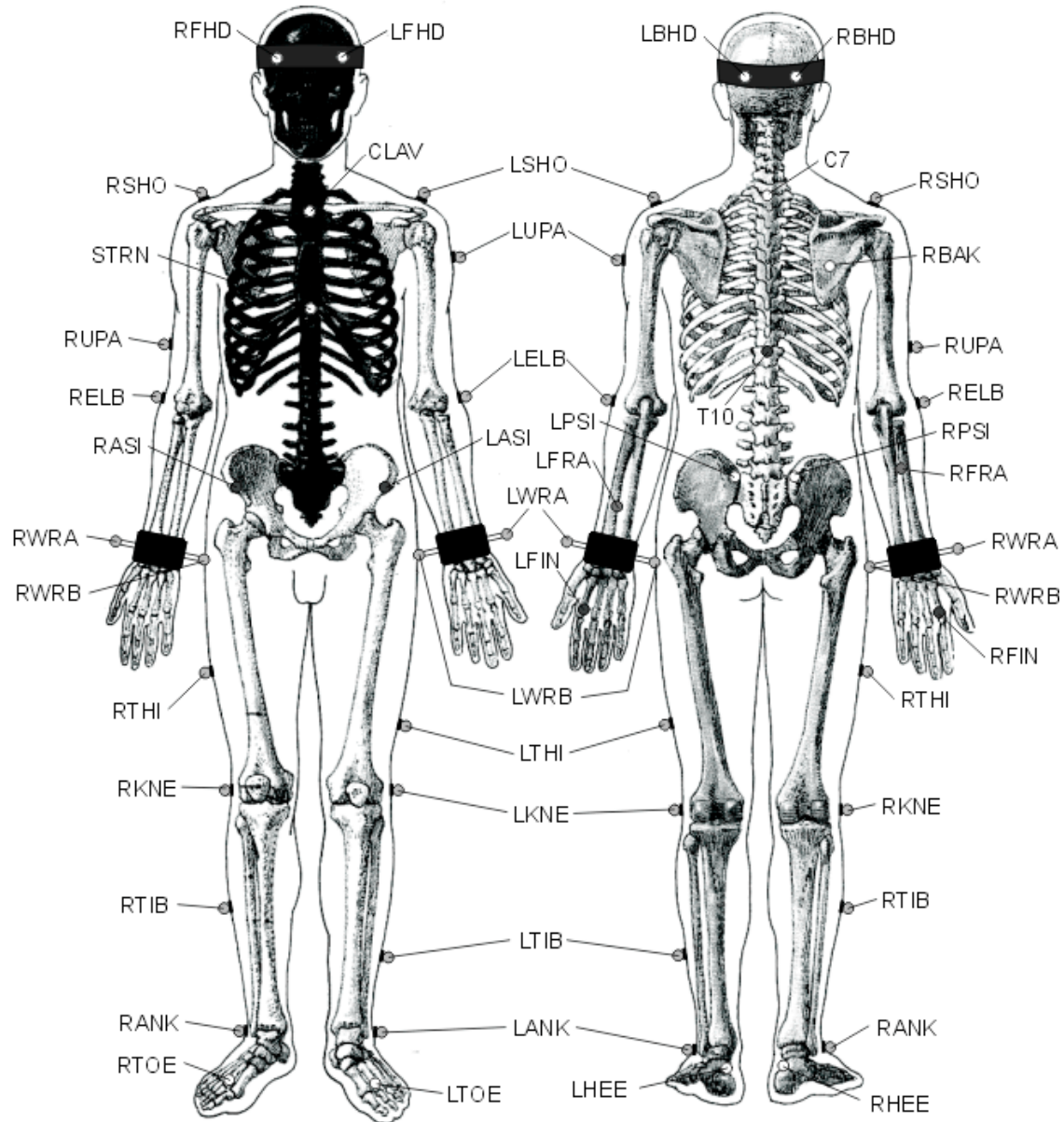


Figure A.3: XYZ Coordinates of End Effector vs. averaged Kinect Measurement (Trials 1-3).

APPENDIX B
PLUG-IN-GAIT MARKER PLACEMENT

Plug-in-Gait Marker Placement



The following describes in detail where the Plug-in-Gait markers should be placed on the subject. Where left side markers only are listed, the positioning is identical for the right side.

Upper Body

Head Markers

LFHD	Left front head	Located approximately over the left temple
RFHD	Right front head	Located approximately over the right temple
LBHD	Left back head	Placed on the back of the head, roughly in a horizontal plane of the front head markers
RBHD	Right back head	Placed on the back of the head, roughly in a horizontal plane of the front head markers

The markers over the temples define the origin, and the scale of the head. The rear markers define its orientation. If they cannot be placed level with the front markers, and the head is level in the static trial, tick the "Head Level" check box under options on "Run static model" in the pipeline when processing the static trial. Many users buy a headband and permanently attach markers to it.

Torso Markers

C7	7 th Cervical Vertebrae	Spinous process of the 7th cervical vertebrae
T10	10 th Thoracic Vertebrae	Spinous Process of the 10th thoracic vertebrae
CLAV	Clavicle	Jugular Notch where the clavicles meet the sternum
STRN	Sternum	Xiphoid process of the Sternum
RBAK	Right Back	Placed in the middle of the right scapula. This marker has no symmetrical marker on the left side. This asymmetry helps the auto-labeling routine determine right from left on the subject.

C7, T10, CLAV, STRN define a plane hence their lateral positioning is most important.

Arm Markers

LSHO	Left shoulder marker	Placed on the Acromio-clavicular joint
LUPA	Left upper arm marker	Placed on the upper arm between the elbow and shoulder markers. Should be placed asymmetrically with RUPA
LELB	Left elbow	Placed on lateral epicondyle approximating elbow joint axis
LFRA	Left forearm marker	Placed on the lower arm between the wrist and elbow markers. Should be placed asymmetrically with RFRA
LWRA	Left wrist marker A	Left wrist bar thumb side
LWRB	Left wrist marker B	Left wrist bar pinkie side

The wrist markers are placed at the ends of a bar attached symmetrically with a wristband on the posterior of the wrist, as close to the wrist joint center as possible.

LFIN	Left fingers	Actually placed on the dorsum of the hand just below the head of the second metacarpal
------	--------------	--

Lower Body

Pelvis

LASI	Left ASIS	Placed directly over the left anterior superior iliac spine
RASI	Right ASIS	Placed directly over the right anterior superior iliac spine

The above markers may need to be placed medially to the ASIS to get the marker to the correct position due to the curvature of the abdomen. In some patients, especially those who are obese, the markers either can't be placed exactly anterior to the ASIS, or are invisible in this position to cameras. In these cases, move each marker laterally by an equal amount, along the ASIS-ASIS axis. The true inter-ASIS Distance must then be recorded and entered on the subject parameters form. These markers, together with the sacral marker or LPSI and RPSI markers, define the pelvic axes.

LPSI	Left PSIS	Placed directly over the left posterior superior iliac spine
RPSI	Right PSIS	Placed directly over the right posterior superior iliac spine

LPSI and RPSI markers are placed on the slight bony prominences that can be felt immediately below the dimples (sacro-iliac joints), at the point where the spine joins the pelvis.

SACR	Sacral wand marker	Placed on the skin mid-way between the posterior superior iliac spines (PSIS). An alternative to LPSI and RPSI.
------	--------------------	---

SACR may be used as an alternative to the LPSI and RPSI markers to overcome the problem of losing visibility of the sacral marker (if this occurs), the standard marker kit contains a base plate and selection of short "sticks" or "wands" to allow the marker to be extended away from the body, if necessary. In this case it must be positioned to lie in the plane formed by the ASIS and PSIS points.

Leg Markers

LKNE	Left knee	Placed on the lateral epicondyle of the left knee
------	-----------	---

To locate the "precise" point for the knee marker placement, passively flex and extend the knee a little while watching the skin surface on the lateral aspect of the knee joint. Identify where knee joint axis passes through the lateral side of the knee by finding the lateral skin surface that comes closest to remaining fixed in the thigh. This landmark should also be the point about which the lower leg appears to rotate. Mark this point with a pen. With an adult patient standing, this pen

mark should be about 1.5 cm above the joint line, mid-way between the front and back of the joint. Attach the marker at this point.

LTHI	Left thigh	Place the marker over the lower lateral 1/3 surface of the thigh, just below the swing of the hand, although the height is not critical.
------	------------	--

The thigh markers are used to calculate the knee flexion axis location and orientation. Place the marker over the lower lateral 1/3 surface of the thigh, just below the swing of the hand, although the height is not critical. The antero-posterior placement of the marker is critical for correct alignment of the knee flexion axis. Try to keep the thigh marker off the belly of the muscle, but place the thigh marker at least two marker diameters proximal of the knee marker. Adjust the position of the marker so that it is aligned in the plane that contains the hip and knee joint centers and the knee flexion/extension axis. There is also another method that uses a mirror to align this marker, allowing the operator to better judge the positioning.

LANK	Left ankle	Placed on the lateral malleolus along an imaginary line that passes through the transmalleolar axis
LTIB	Left tibial wand marker	Similar to the thigh markers, these are placed over the lower 1/3 of the shank to determine the alignment of the ankle flexion axis

The tibial marker should lie in the plane that contains the knee and ankle joint centers and the ankle flexion/extension axis. In a normal subject the ankle joint axis, between the medial and lateral malleoli, is externally rotated by between 5 and 15 degrees with respect to the knee flexion axis. The placements of the shank markers should reflect this.

Foot Markers

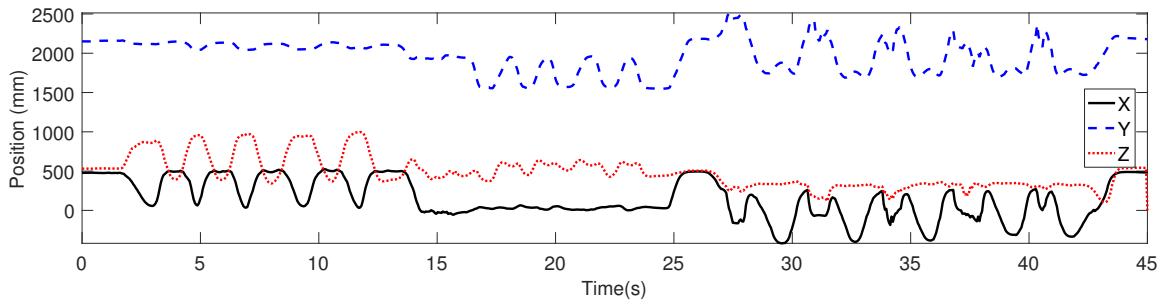
LTOE	Left toe	Placed over the second metatarsal head, on the mid-foot side of the equinus break between fore-foot and mid-foot
LHEE	Left heel	Placed on the calcaneus at the same height above the plantar surface of the foot as the toe marker

APPENDIX C
PLUG-IN-GAIT MARKER IDENTIFIERS AND LOCATION

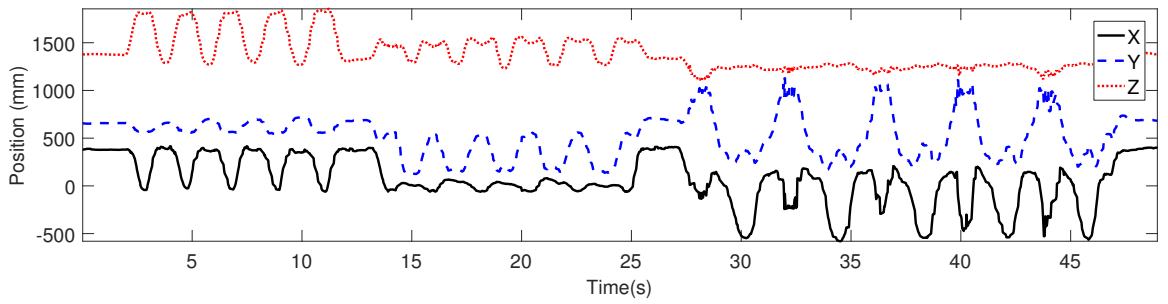
Table C.1: Marker Identifiers and Marker Placement for Plug-in-Gait model used for tracking with Vicon 3D Motion Capture System

Marker Number	Marker Identifier	Placement
1	LFHD	Left Front Head
2	RFHD	Right Front Head
3	LBHD	Left Back Head
4	RBHD	Right Back Head
5	C7	7th Cervical Vertebrae
6	T10	10th Thoracic Vertebrae
7	CLAV	Clavicle
8	STRN	Sternum
9	RBAK	Right Back
10	LSHO	Left Shoulder
11	LUPA	Left Upper Arm
12	LELB	Left Elbow
13	LFRA	Left Forearm
14	LWRA	Left Wrist Marker A
15	LWRB	Left Wrist Marker B
16	LFIN	Left Fingers
17	RSHO	Right Shoulder
18	RUPA	Right Upper Arm
19	RELB	Right Elbow
20	RFRA	Right Forearm
21	RWRA	Right Wrist Marker A
22	RWRB	Right Wrist Marker B
23	RFIN	Right Fingers
24	LASI	Left Anterior Superior Iliac Spine
25	RASI	Right Anterior Superior Iliac Spine
26	LPSI	Left Posterior Iliac Spine
27	RPSI	Right Posterior Iliac Spine
28	LKNE	Left Knee
29	LTHI	Left Thigh
30	LANK	Left Ankle
31	LTIB	Left Tibial Marker
32	LTOE	Left Toe
33	LHEE	Left Heel
34	RKNE	Right Knee
35	RTHI	Right Thigh
36	RANK	Right Ankle
37	RTIB	Right Tibial Marker
38	RTOE	Right Toe
39	RHEE	Right Heel

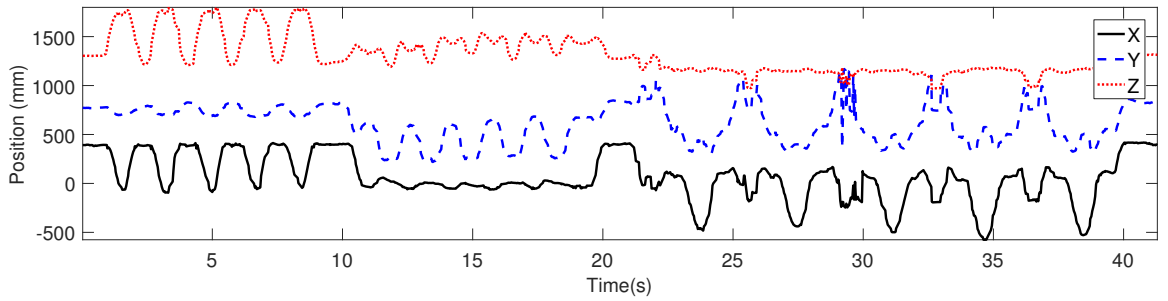
APPENDIX D
DUAL-KINECT SETUP TEST TRAJECTORIES



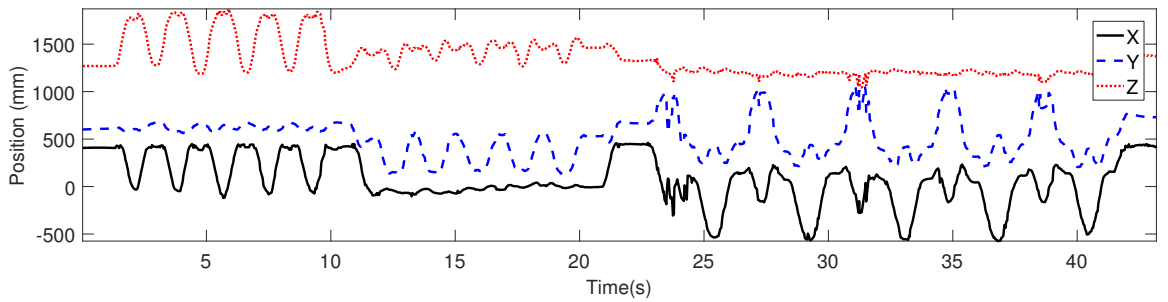
(a) Setup 1 ($\gamma=0^\circ$)



(b) Setup 2 ($\gamma=15^\circ$)

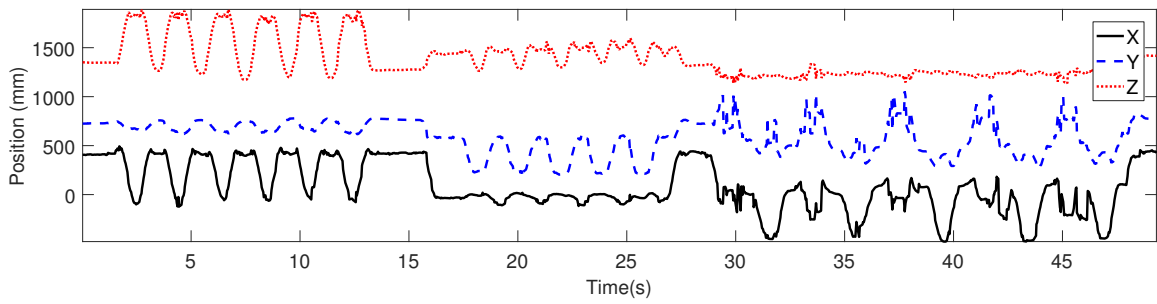


(c) Setup 3 ($\gamma=30^\circ$)

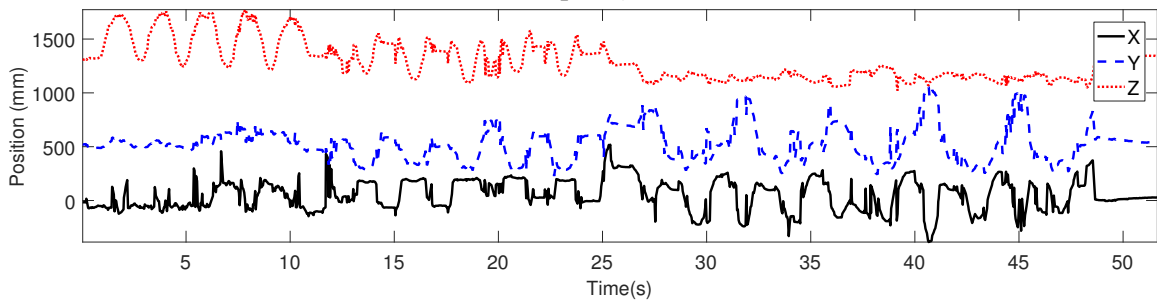


(d) Setup 4 ($\gamma=45^\circ$)

Figure D.1: Fused Left Wrist Trajectory for Dual-Kinect Setup 1-4

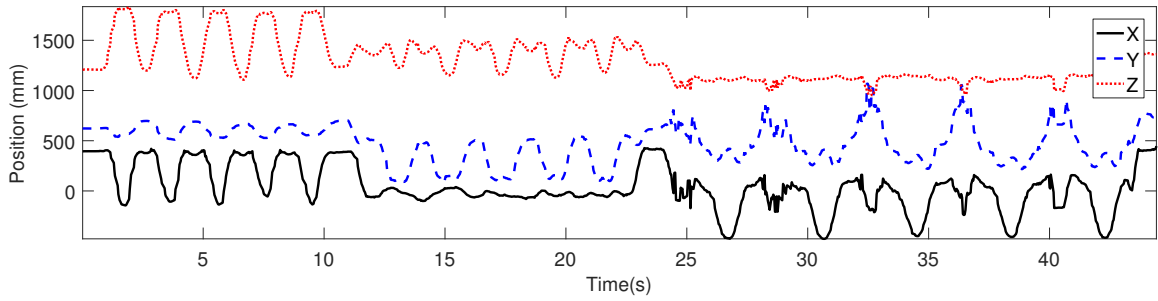


(e) Setup 5 ($\gamma=60^\circ$)

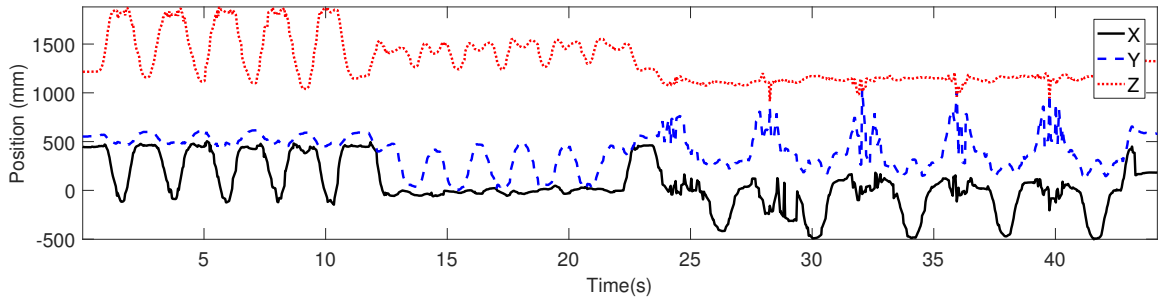


(f) Setup 6 ($\gamma=75^\circ$)

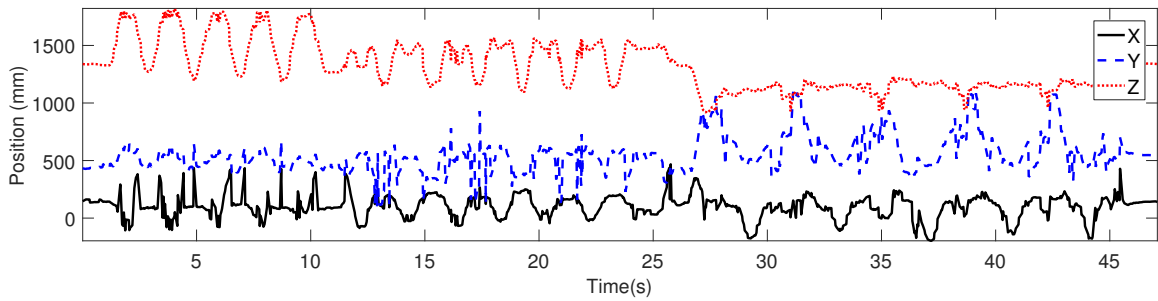
Figure D.1: Fused Left Wrist Trajectory for Dual-Kinect Setup 5-6



(a) Setup 7 ($\delta=30^\circ$)



(b) Setup 8 ($\delta=60^\circ$)



(c) Setup 9 ($\delta=90^\circ$)

Figure D.2: Fused Left Wrist Trajectory for Dual-Kinect Setup 7-9

APPENDIX E
TRACKING ACCURACY OF MOTION CAPTURE EXPERIMENTS WITH
DUAL-KINECT

Table E.1: Accuracy of the Different Filter Outputs for Setup 4

Joint	Mean Absolute Error (mm)											
	LKF1			LKF2			EKF					
	X	Y	Z	X	Y	Z	X	Y	Z	X	Y	Z
SpineMid	21.7044	59.9314	23.4772	21.2944	59.9417	23.556	21.3348	60.7895	23.556	21.3348	60.7895	23.5577
SpineShoulder	24.3905	23.0314	17.314	23.9639	23.1039	17.5668	28.0918	32.7701	17.5668	28.0918	32.7701	28.3603
ShoulderLeft	46.7217	30.6348	27.916	46.5161	30.3626	28.2068	61.6301	27.5947	28.2068	61.6301	27.5947	30.5579
ElbowLeft	38.8949	41.3346	24.6375	38.6788	39.2196	24.6786	34.334	42.8241	24.6786	34.334	42.8241	20.5734
WristLeft	27.763	20.1574	27.8804	28.1157	19.1075	29.0177	28.5107	22.6613	29.0177	28.5107	22.6613	27.4909
HandTipLeft	50.8307	46.2039	50.1849	53.5383	46.1913	52.4592	57.2518	57.1384	52.4592	57.2518	57.1384	53.4151
ShoulderRight	24.7811	31.5799	38.2539	24.4557	31.4344	38.415	27.0514	28.7534	38.415	27.0514	28.7534	40.457
ElbowRight	42.5645	45.1523	28.2156	40.7346	44.1187	28.3209	33.9066	47.0176	28.3209	33.9066	47.0176	25.0735
WristRight	35.9739	31.5865	21.8422	33.2698	31.6339	21.8758	31.1033	34.9596	21.8758	31.1033	34.9596	17.7353
HandTipRight	50.7945	41.0936	44.5696	55.1716	44.1955	46.0646	52.058	47.6606	46.0646	52.058	47.6606	48.8299

REFERENCES

- [1] Jintu Fan, Winnie Yu, and Lawrence Hunter. *Clothing appearance and fit: Science and technology*. Elsevier, 2004.
- [2] Phoebe R Apeageyi. “Application of 3D body scanning technology to human measurement for clothing fit”. In: *this issue* (2010).
- [3] Kwang-Jin Choi and Hyeong-Seok Ko. “Research problems in clothing simulation”. In: *Computer-aided design* 37.6 (2005), pp. 585–592.
- [4] Shweta Satish Devare Phadke, Rajak Revati, and Rauf Iqbal. “Work Related Musculoskeletal Symptoms among Traffic Police: Cross Sectional Survey Using Nordic Musculoskeletal Questionnaire”. In: ().
- [5] C Ha et al. “The French Musculoskeletal Disorders Surveillance Program: Pays de la Loire network”. In: *Occupational and Environmental Medicine* 66.7 (2009), pp. 471–479. eprint: <http://oem.bmj.com/content/66/7/471.full.pdf>.
- [6] Jessica G Ramsey, CPE Kristin Musolin, and Charles Mueller. “Evaluation of carpal tunnel syndrome and other musculoskeletal disorders among employees at a poultry processing plant”. In: *National Institute for Occupational Safety and Health, Health Hazard Evaluation, Report 2014-0040* (2015), p. 3232.
- [7] Edward YL Gu. *A journey from robot to digital human: mathematical principles and applications with MATLAB programming*. Vol. 1. Springer Science & Business Media, 2013.
- [8] Amir Mobini, Saeed Behzadipour, and Mahmoud Saadat Foumani. “Accuracy of Kinect’s skeleton tracking for upper body rehabilitation applications”. In: *Disability and Rehabilitation: Assistive Technology* 9.4 (2014), pp. 344–352. eprint: <http://dx.doi.org/10.3109/17483107.2013.805825>.
- [9] Anne Schmitz et al. “Accuracy and repeatability of joint angles measured using a single camera markerless motion capture system”. In: *Journal of Biomechanics* 47.2 (2014), pp. 587–591.
- [10] Brook Galna et al. “Accuracy of the Microsoft Kinect sensor for measuring movement in people with Parkinson’s disease”. In: *Gait & Posture* 39.4 (2014), pp. 1062–1068.

- [11] A. Fernández-Baena, A. Susín, and X. Lligadas. “Biomechanical Validation of Upper-Body and Lower-Body Joint Movements of Kinect Motion Capture Data for Rehabilitation Treatments”. In: *2012 Fourth International Conference on Intelligent Networking and Collaborative Systems*. 2012, pp. 656–661.
- [12] Ross A. Clark et al. “Validity of the Microsoft Kinect for assessment of postural control”. In: *Gait & Posture* 36.3 (2012), pp. 372–377.
- [13] C. C. Martin et al. “A real-time ergonomic monitoring system using the Microsoft Kinect”. In: *2012 IEEE Systems and Information Engineering Design Symposium*. 2012, pp. 50–55.
- [14] F. Destelle et al. “Low-cost accurate skeleton tracking based on fusion of kinect and wearable inertial sensors”. In: *2014 22nd European Signal Processing Conference (EUSIPCO)*. 2014, pp. 371–375.
- [15] Sung-Hee Lee, Eftychios Sifakis, and Demetri Terzopoulos. “Comprehensive Biomechanical Modeling and Simulation of the Upper Body”. In: *ACM Trans. Graph.* 28.4 (2009), 99:1–99:17.
- [16] Walter Maurel. “3D modeling of the human upper limb including the biomechanics of joints, muscles and soft tissues”. PhD thesis. Ecole Polytechnique Federale De Lausanne, 1999.
- [17] Derek James Lura. “The Creation of a Robotics Based Human Upper Body Model for Predictive Simulation of Prostheses Performance”. PhD thesis. University of South Florida, 2012.
- [18] Alexandra Pfister et al. “Comparative abilities of Microsoft Kinect and Vicon 3D motion capture for gait analysis”. In: *Journal of Medical Engineering & Technology* 38.5 (2014), pp. 274–280. eprint: <http://dx.doi.org/10.3109/03091902.2014.909540>.
- [19] Ville Stohne. “Real-time filtering for human pose estimation using multiple Kinects”. MA thesis. KTH, School of Computer Science and Communication (CSC), 2014.
- [20] Diana Margarita Córdova-Esparza et al. “Multiple Kinect V2 Calibration”. In: *Automatika—Journal for Control, Measurement, Electronics, Computing and Communications* 60.1 (2017).
- [21] J. K. Aggarwal and Sangho Park. “Human motion: modeling and recognition of actions and interactions”. In: *Proceedings. 2nd International Symposium on 3D Data Processing, Visualization and Transmission, 2004. 3DPVT 2004*. 2004, pp. 640–647.

- [22] A. Maciel, L. P. Nedel, and C. M. Dal Sasso Freitas. “Anatomy-based joint models for virtual human skeletons”. In: *Proceedings of Computer Animation 2002 (CA 2002)*. 2002, pp. 220–224.
- [23] Signe Brunnstrom. *Clinical kinesiology*. FA Davis Company, 1972.
- [24] Margareta Nordin and Victor Hirsch Frankel. *Basic biomechanics of the musculoskeletal system*. Lippincott Williams & Wilkins, 2001.
- [25] Richard Drake, A Wayne Vogl, and Adam WM Mitchell. *Gray’s Anatomy for Students E-Book*. Elsevier Health Sciences, 2009.
- [26] Frank C Anderson and Marcus G Pandy. “Dynamic optimization of human walking”. In: *Journal of biomechanical engineering* 123.5 (2001), pp. 381–390.
- [27] Yujiang Xiang, Jasbir S Arora, and Karim Abdel-Malek. “Physics-based modeling and simulation of human walking: a review of optimization-based and other approaches”. In: *Structural and Multidisciplinary Optimization* 42.1 (2010), pp. 1–23.
- [28] Marcus G Pandy. “Computer modeling and simulation of human movement”. In: *Annual review of biomedical engineering* 3.1 (2001), pp. 245–273.
- [29] S. Kajita and K. Tani. “Study of dynamic biped locomotion on rugged terrain-derivation and application of the linear inverted pendulum mode”. In: *Proceedings. 1991 IEEE International Conference on Robotics and Automation*. 1991, 1405–1411 vol.2.
- [30] Yuting Wang. “Closed-form inverse kinematic solution for anthropomorphic motion in redundant robot arms”. PhD thesis. Arizona State University, 2013.
- [31] J. Denavit and R. S. Hartenberg. “A kinematic notation for lower-pair mechanisms based on matrices”. In: *Trans. ASME E, Journal of Applied Mechanics* 22 (1955), pp. 215–221.
- [32] Rudolph Emil Kalman et al. “A new approach to linear filtering and prediction problems”. In: *Journal of basic Engineering* 82.1 (1960), pp. 35–45.
- [33] Andrew Leonard. “Vehicle Tracking Using Ultra-Wideband Radar”. PhD thesis. Georgia Institute of Technology, 2016.
- [34] John A Harber. “A dual hoist robot crane for large area sensing”. PhD thesis. Georgia Institute of Technology, 2016.
- [35] Juan R. Terven and Diana M. Córdova-Esparza. “Kin2. A Kinect 2 toolbox for MATLAB”. In: *Science of Computer Programming* 130 (2016), pp. 97–106.

- [36] Herman J Woltring. “A Fortran package for generalized, cross-validatorspline smoothing and differentiation”. In: *Advances in Engineering Software (1978)* 8.2 (1986), pp. 104–113.
- [37] Paul J. Besl and Neil D. McKay. *Method for registration of 3-D shapes*. 1992.