DEVELOPMENT OF A FEEDBACK-CONTROLLED ELBOW SIMULATOR: DESIGN VALIDATION AND CLINICAL APPLICATION

by

Laurel Kuxhaus

B.S. Engineering Mechanics, Michigan State University, 2001

B.A. Music, Michigan State University, 2001

M.S. Mechanical Engineering, Cornell University, 2003

Submitted to the Graduate Faculty of

the Swanson School of Engineering in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

University of Pittsburgh

2008

UNIVERSITY OF PITTSBURGH

SWANSON SCHOOL OF ENGINEERING

This dissertation was presented

by

Laurel Kuxhaus

It was defended on

January 17, 2008

and approved by

Mark E. Baratz M.D., Surgeon, Orthopaedic Surgery, Allegheny General Hospital

Rory A. Cooper Ph.D., Professor, Bioengineering

Mark Carl Miller Ph.D., Associate Research Professor, Mechanical Engineering

Mark S. Redfern Ph.D., Professor, Bioengineering

Jeffrey S. Vipperman Ph.D., Associate Professor, Bioengineering

Lee E. Weiss, Ph.D., Research Professor, The Robotics Institute, Carnegie Mellon University

Dissertation Director: Jeffrey S. Vipperman Ph.D., Associate Professor, Bioengineering

DEVELOPMENT OF A FEEDBACK-CONTROLLED ELBOW SIMULATOR: DESIGN VALIDATION AND CLINICAL APPLICATION

Laurel Kuxhaus, Ph.D.

University of Pittsburgh, 2008

This work involves three topics that advance the functionality of an elbow simulator in the Orthopaedic Biomechanics Laboratory at Allegheny General Hospital. To draw clinically and scientifically meaningful conclusions from future cadaver studies conducted with the simulator, its design must be validated and the accuracy of the data collection methods demonstrated.

The simulator was designed to offer physiologically-correct adjustable moment arms throughout the elbow's range of motion. To validate this, muscle moment arms were measured in three cadaver elbow specimens. Flexion-extension moment arms were measured at three different pronation/supination angles: fully pronated, fully supinated, and neutral. Pronation-supination moment arms for four elbow muscles were measured at three different flexion-extension angles: 30°, 60°, and 90°. The numeric results compared well with those previously reported. The biceps and pronator teres flexion-extension moment arms varied with pronation-supination position, and vice versa. This represents the first use of closed-loop feedback control in an elbow simulator, one of the first reports of both flexion-extension and pronation-supination moment arms in the same specimens, and demonstrates the adjustability of the moment arms that the elbow simulator can produce.

Towards accurate motion analysis of the radial head, two areas were investigated. The first identified the phenomena of *camera-switching*, which occurs in motion analysis when data

from one or more cameras is temporarily excluded from the computation of a marker's three-dimensional position. Tests with static markers showed that camera-switching could cause up to 3.7 mm of perceived movement. The second area of investigation set the stage for future studies with cadaver elbows. A protocol was developed to quantify both the travel of the native radial head, radial head implants, and the finite helical axis during pronation-supination movement. The tracking of implant motion employs a unique circle-fitting algorithm to determine the implant's center. A video-based motion analysis system was used to collect marker position coordinates actuated by a precision micrometer table. MATLAB code was designed and implemented to compute both the radial head position and finite helical axis from these data. Immediate future work will use these algorithms to evaluate radial head implants in comparison to the native radial head.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

PREFACE

It takes a village to write a dissertation. I am thankful for the support of the people in my village throughout the years. First and foremost, my thanks go to my village, led by my advisors, Drs. Jeffrey S. Vipperman and Mark Carl Miller. Without their mutual support, ideas, and encouragement, this work would not have been possible. I give my thanks to the other "elders" in the village – Drs. Redfern, Baratz, Weiss, and Cooper - as well for their unique perspectives and suggestions to improve this work. I am indebted to my colleagues in the lab for their support and assistance with daily tasks, in particular Pat Schimoler for his help with the elbow experiments and programming, Sunghwan Kim for his help implementing the circle-fitting code, and Mike Hudzik, Derek Dazen, Angela Flamm, and Karol Galik for their contributions. Additionally, the last set of data collection would not have been possible without the help of Dr. Rich Debski's Shoulder Lab.

The villagers outside of the laboratory have been invaluable as well. My roommates (Lisa, Beth, and Alexis) who have put up with the stress and disarray at home are troopers. Over the years, my past and present quintet colleagues in the Aeolian Winds of Pittsburgh have contributed both to my sanity and insanity, though Barb, Josie, Steve, and Alex (and Michelle!) have done a good job of keeping me grounded, especially in these last few months. Likewise, the support of my musical colleagues in the (North) Pittsburgh Philharmonic and the Edgewood Symphony should not be overlooked. My double-reed friends, especially Sharlotte, bassoon-

Mark, and oboe-Steve, have kept me just a little bit musically-nerdy. The performances we took, along with Barb and piano-Mark, to the International Double Reed Society conference were unforgettable. My village has many talented and inspiring people who are medical doctors, computer programmers, and other professionals by day, but who can make beautiful music together at night.

There are a few villagers who many not be geographically close, but who have been no further than a phone call away. In particular, Charles in Baltimore, Eric in Indiana, Anna and Paul in Ithaca, Jahi and Andrea in Ann Arbor, Audra in New Jersey, Katie in Las Vegas, Stephanie in Atlanta, Penny in Wisconsin, Joe in Berkeley, Justin in Seattle, and the rest of my family (mostly) in Michigan. Surely there are others whose names escape me at the moment. I am also indebted to my pet frog, Kermit. She has been a great companion for nearly 10 years.

1.0 INTRODUCTION AND MOTIVATION

The long-term goal of this work is to improve elbow function. Towards that goal, a physiologic elbow simulator has been built at Allegheny General Hospital (AGH) in Pittsburgh, PA. This thesis presents three distinct topics necessary for the use of the simulator. The first topic is the refinement of the AGH Elbow Simulator and validation of its ability to accurately reproduce physiologic muscle moment arms throughout the elbow's range of motion. The second is a description of the effects of *camera-switching* on the tracking of small motions with a Vicon motion analysis system. Third, as a first application, the software to compute radial head movement from motion data collected while the elbow is actuated via the simulator is developed and validated.

1.1 MOTIVATION

The overall goal of the studies presented here was to improve and validate a previously-built elbow joint simulator [163] and demonstrate its application to clinical research. The elbow is critical to activities of daily living, particularly to feeding and grooming. Without proper elbow function, the hand cannot be positioned properly to interact with objects including telephones, computers, and more basic objects such as shoelaces and shirt buttons. A myriad of conditions, such as stroke or spinal cord injury, can cause an individual to lose control of a limb, yielding a

serious long-term disability. Through a better understanding of the neuromuscular control required to manipulate the elbow, the long-term applications of these studies can restore or create proper neural control of limbs in those who suffer from disease, injury, or other impairments. Specifically, as injuries to the elbow are common, [19] these studies can contribute to the restoration of necessary elbow function via Functional Electrical Stimulation (FES). Cumulative trauma disorders, including cubital tunnel syndrome, account for 56% of all occupational injuries in the United States. [172] These disorders frequently compromise an individual's upper extremity control. The results of this work have the potential to advance and enhance the understanding of neuromuscular control in general through the specific study of elbow movement. Additionally, evaluations of radial head replacements become more meaningful when performed in a physiologic elbow simulator – the forces on the muscles serve as important constraints to radial head motion. Without these forces, the radial head implants may not perform in a manner that accurately represents the *in vivo* case.

One general approach to the study of limb control is to create a simulator for a particular joint which accurately recreates human motion and force capabilities. This is particularly advantageous in that it can reproduce physiologic motions using the physiologic structures of the muscles and tendons, which creates physiologically realistic forces in and around the joint. For the elbow, a physiologic simulator offers opportunity to evaluate the performance of implants, such as radial head implants, and also to measure strain in the elbow ligaments during both daily living motions and extreme motions, such as those used in baseball pitching. The foundations in place from the previously-constructed AGH Elbow Simulator offered opportunities for improvement in three areas. The success of the elbow simulator and the future clinical studies relies on the successful coordination and implementation of the simulator's control system,

including all relevant hardware, with a sufficiently accurate means of measuring the elbow's motion. Thus the first aim of this work was to validate a physical aspect of the design of the simulator. To this end, muscle moment arms of three cadaver specimens were measured in the simulator with the actuators in force control. This is both the first demonstration of closed-loop feedback control with the simulator and also illustrates an application of the simulator to a clinically-meaningful result. The second aim of this work was to evaluate the suitability of a Vicon motion analysis system to track the extremely fine motion of the radial head on the capitellum in conjunction with the simulator's actuation of the elbow. The third aim, applicable to the clinical use of the simulator, is to develop, implement, and validate software capable of computing both the position of the radial head and the finite helical axis of rotation during pronation/supination movements.

1.2 SIGNIFICANCE

This work contributes to both the basic science of joint simulator design and to the clinical applications of radial head replacement.

1.2.1 Basic science significance

The open literature lacks a closed-loop elbow simulator. The design, assembly, and refinement of a simulator's hardware represents a significant accomplishment. The custom pulleys which maintain physiologically accurate moment arms are unique, and the structure of the frame will allow testing of cadaveric elbows in both a vertical and horizontal orientation. The long-term

potential to improve radial head implant design exists. The design and implementation of the control scheme used to measure moment arms represent the first elbow joint simulator operating via closed-loop control. The techniques developed to accurately track the motion of the radial head extend standard motion analysis procedures. In addition, this work contributes to the fundamental understanding of the control of elbow movements.

1.2.2 Clinical significance

The measurement of elbow moment arms in cadaver specimens at different elbow joint angles is a new contribution to the biomechanical knowledge base, and has the ability to improve the clinical understanding of elbow function. The results of this work can benefit a wide range of applications, including prosthetic limb design, the development of new rehabilitation strategies for those with compromised elbow function, and the immediate benefit of the clinically-oriented studies of radial head implants. These applications can improve the health of individuals with compromised elbow function.

1.2.3 Specific contributions of this work

This work makes several contributions to the basic science and clinical literature. The improvements to the elbow simulator's design advance the study of joints in general, and the elbow in particular. The technique used to measure the moment arms presents is a novel modification of a standard technique. The moment arm study is one of the first instances of moment arm measurement in the same specimens for both flexion-extension and pronation-supination motions, and also is the first to present results which show that, for some muscles,

moment arm about one axis may depend on the position around the other axis. The work with the Vicon motion analysis system is the first to quantify the amount of error induced by camera-switching. The use of closed-loop feedback control in an elbow joint simulator is similarly novel. The clinical results of the radial head travel offer significant insight into the outcomes radial head implant design.

1.3 OVERVIEW

This dissertation presents a collection of three studies that have contributed to the advancement of the AGH Elbow Simulator. Following a broad review of background material (Chapter 2.0) and a targeted literature review (Chapter 3.0), the methods (Chapter 4.0) and results (Chapter 5.0) from the simulator design validation by measuring elbow moment arms are presented. The methods used to evaluate the Vicon system accuracy are presented in Chapter 6.0 , and the results follow in Chapter 7.0 . Chapter 8.0 describes the methods used to develop, implement, and validate software to compute the position of the radial head and finite helical axis. The corresponding results are shown in Chapter 9.0 . Finally, Chapter 10.0 gives some concluding remarks and suggestions for future work.

2.0 BACKGROUND

Though outwardly simple, the interactions of the bones and muscles of the elbow are complex. Activities of daily life depend on a functional elbow: it plays a crucial role in hand positioning and also load transfer. With elbow impairment, activities such as feeding, grooming, and hygiene can be severely limited, jeopardizing the individual's independence. Even small improvements in elbow mobility can yield large gains in functional capability. This chapter presents an overview of basic elbow anatomy, structure, and function; an overview of radial head anatomy and prosthesis design; an introduction to motion analysis; an introduction to moment arms; and an introduction to joint simulators.

2.1 ELBOW ANATOMY

The structure of the elbow joint includes three bones which are attached by numerous ligaments and tendons and are actuated by several muscles that cross the joint. These structures act in concert to control forearm position and also to transfer loads from the hand and forearm to the upper arm.

2.1.1 Bones and kinematics

Three bones interact at the elbow joint: the humerus, the radius, and the ulna (see Figure 1.) Thus there are three bony articulations at the joint: radiohumeral, unlohumeral, and radioulnar. the trochlear groove rides in the ulnar notch; the radial head moves against the capitellum; and the radial head is also constrained by the proximal ulna. The humeroradial joint, between the capitellum and the radial head, and the humeroradial joint, between the trochlea and the trochlear notch, contribute to flexion/extension motion. The radioulnar joint between the radial head and the radial notch on the ulna allows pivoting about a longitudinal axis in the forearm, that is, pronation/supination movement. Though there are three articulations that comprise the elbow joint, it is widely considered to have only two degrees of kinematic freedom: flexion/extension (f/e) and pronation/supination (p/s), as shown in Figure 2. Full extension is defined as 0° of flexion (i.e. the flexion angle is measured with respect to the vertical and is not the included angle between the forearm and the upper arm.) Neutral (0° of pronation or supination) is defined when the radial styloid is directly over the ulnar styloid with the elbow in 90° of flexion.

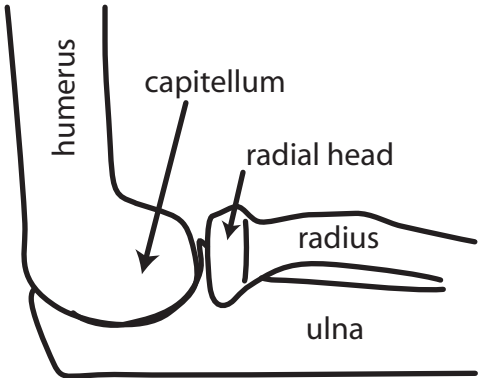


Figure 1: Elbow bones.

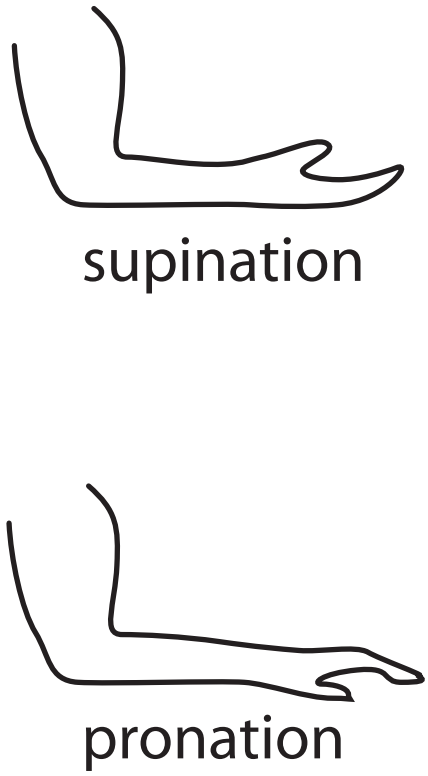


Figure 2: Pronation and supination.

The elbow is one of the most intricately constructed joints of the human body, and as such, is one of the most stable. [186] The entire elbow joint complex is constrained anteriorly by the coronoid process and radial head, and posteriorly by the olecranon. [19] The entire joint complex is enclosed in a single capsule and stabilized by both medial and lateral ligaments.

Constrained by the shape of the proximal end of the ulna, the f/e axis of the elbow is nearly constant, like a perfect hinge. [11] Elbow laxity can create up to 3-4° of variation in the axis. [71] The center of rotation has been found to be in an area 2-3mm in diameter at the center of the trochlea [11], though others have since found this locus to be larger. [82] Morrey *et al.* have suggested that the f/e axis is internally rotated 3-8°. [11] It is also important to note that, when fully extended, the long axis of the ulna is not aligned with the long axis of the humerus. The acute angle between the two is referred to as the *carrying angle*. For the moving elbow, the precise definition of the angle is controversial despite the conceptual agreement. Typical elbow

range of motion includes 150° of flexion (from 0° to 150°), and the carrying angle for men is 10-15°. In women, the carrying angle is about 5° greater. [11]

During p/s, the radius rotates in a circular motion around the long axis of the ulna. Translation of the radius at the proximal end is considered insignificant, and movement at the distal end is of a sliding nature. The distal end of the radius, at the radioulnar joint, rotates around the ulnar styloid,. At the proximal end, the radial head is more constrained. Thus this axis of rotation is generally thought to pass from the center of the radial head (the proximal end of the radius) through the tip of the ulnar styloid. [11] The axis remains constant regardless of elbow position. [114] The axis of forearm rotation was shown to pass approximately 4 mm to the center of the capitellum and 8 mm to the ulnar styloid. [72, 114, 256] Note that this axis is oblique to the long axes of both the radius and the ulna. This motion is critical for wrist and hand positioning. Full range of p/s motion is about 160° (from 75° pronation to 85° supination.) [11]

The radial head plays a key role in elbow joint stability and load transfer. The radial head is a stabilizer against valgus load and aids the coronoid in providing an anterior buttress for the humerus. [117, 183, 189, 214] Captier *et al.* described the biometry of the radial head and found that in most cases (57%) the radial heads were elliptical in cross-sectional shape, which implies adaptation of the radial neck to accommodate the differences in kinematics between radial heads of circular and elliptical cross-section. [43] This adaptation will also affect the load transfer across the joint. Morrey *et al.* have shown that up to 90% of body weight can be transferred across the radial head. [181]

2.1.2 Ligaments and tendons: an overview

The elbow's stability is due in part to the congruity of the bony articulations and in part due to the soft tissue constraints. For example, at 90° flexion, the bony articulations are mostly responsible for stability. However, in full extension, the anterior capsule is taut and thus plays a role in varus and valgus stability. [183] The collateral ligament complexes and the anterior capsule serve as stabilizers. [189] The multi-bundle structure of these complexes has been documented. [21] The lateral collateral ligament helps stabilize the lateral ulnohumeral joint [189], specifically by preventing supination of the ulna. [19] Sojbjerg *et al.* have shown that the annular ligament plays a role in varus and valgus elbow stability, both with and without the radial head in place. [227] The medial collateral ligament is the primary valgus stabilizer for the flexed elbow. [189] The posterior band of this ligament prevents pronation of the ulna in conjunction with the bony articulations. [19] The interosseous membrane and triangular fibrocartilage complex contribute to axial stability of the forearm. In fact, these structures may be responsible for preserving radial head length even in the absence of a radial head. [58, 188]

2.1.3 Main muscles that cross the elbow joint

The major muscles that cross the elbow joint include the biceps brachii, brachialis, brachioradialis, extensor carpi radialis longus, triceps, flexor carpi radialis, and anconeus. [19] Other forearm muscles, such as the flexor carpi ulnaris, cross the elbow, but do not notably contribute to elbow motion. [19] Muscle forces and force directions have been reported as have muscle tensions and potential excursions. [31, 100, 125, 252] Muscle moment arms have also been measured and reported (see Chapter 3.0 for an overview.)

The biceps brachii originates on the scapula with two heads, and covers anteriorly the brachialis as it continues down the arm. The long head originates on the superior glenoid labrum in nearly 50% of the population, and from the supraglenoid tubercule otherwise. [255] It inserts onto the radial tuberosity, and thus wraps around the radius during pronation. Mechanically, it acts as an elbow flexor and a strong supinator. [184]

The brachialis originates on the anterior distal half of the humerus and inserts onto the ulnar tuberosity. It acts as a flexor. While large in cross-sectional area, its position relative to the elbow joint gives it poor mechanical advantage and thus limits moment-generating capability. However, since it inserts into the ulna, the insertion site is unaffected by pronation or supination movement. [184]

The triceps brachii has three heads at its origin: the outer two of which originate on the posterior side of the humerus, and the long head, which originates from the infraglenoid tubercule of the scapula. The lateral head originates on the humerus just lateral and superior to the radial groove. The medial head originates on the humerus medial and inferior to the radial groove. The insertion site is on the posterior surface of the olecranon. Functionally, it is the elbow's main extensor. It is also interesting that, although two of the origin heads are not affected by shoulder position, the long head does cross the glenohumeral joint and can aid in humerus extension and adduction. [184]

The pronator teres originates on the medial epicondyle of the humerus and inserts onto the mid-radius. This geometry gives it function as both a strong pronator and a weak flexor. During extreme supination, it will wrap around the radius. [184]

2.2 RADIAL HEAD FRACTURES AND RADIAL HEAD REPLACEMENT

Complete elbow function is critical to daily life functions such as feeding, grooming, and hygiene. Elbow function can be impaired through both acute and chronic injury and trauma. In the case of extreme injury, the radial head may be replaced with an implant.

2.2.1 Radial head fractures

Radial head fracture is the most common adult elbow injury, occurring in 17-19% of elbow trauma cases and accounting for 33% of elbow fractures. [182] Radial head fractures alone account for 5.4% of all adult fractures. [59, 129, 182] About 15-20% of these fractures involve the neck [191, 244] and radial head fracture is twice as common in women. [16, 45] Comorbidities may include fractures of the olecranon and/or coronoid and disruptions of the medial collateral ligament, triangular fibrocartilage complex and/or interosseous membrane. Treatment varies accordingly [214, 215, 247] and can include excision, surgical reconstruction, and complete radial head replacement.

Clinicians have been disappointed with treatment outcomes. [182] As the radial head also aids the triangular fibrocartilage complex and the interosseous membrane in preventing proximal migration of the radius with respect to the ulna, [140] displaced radial head fractures are routinely reduced and internally stabilized. [116, 174, 214] This treatment, however, can lead to unsatisfactory clinical results. [116, 214] Radial head excision has been advocated for the comminuted fracture, but complications can occur such as an unstable elbow or chronic wrist pain. [53, 246] In the extreme case, the radial head may be replaced. In short-term clinical follow up, metallic radial head implants have fared acceptably for the unreconstructable fracture.

[147, 180] However, radiographic review has suggested that a metallic radial head implant did not restore physiological load transfer across the elbow. [180] In another study, two out of thirty-six metallic implants were removed because of painful loosening. [147] However, it has also been shown that metallic radial head arthroplasty can offer improved valgus stability, approaching that of the native radial head and far superior to that offered by silicone implants. [143]

2.2.2 Radial head prostheses: overview

Radial head prostheses have had increasing levels of success. In the mid-1970's, fifty-five percent of patients receiving radial head implants had complications. Nonetheless, overall, the results were good in sixty percent of the patients. [186] Radial head prostheses should prevent the complications of radial head excision, transfer normal loads, and reestablish the original elbow kinematics. [53, 241] A number of studies have examined transfer of loads of prostheses by measuring elbow stability and laxity. [126, 143, 154, 167, 200, 205, 207, 227] Silicone radial head prostheses became popular but caused a reactive synovitis and did not re-establish normal ulnar-humeral load transmission. [101, 143, 207] In cadaveric studies, metallic radial head replacements restored the axial forearm stiffness to almost normal, [126, 143, 205] but decreased the humero-radial contact area by 68%. [154]

The kinematics of the native and replaced radial head have been studied much less than the load transfer. Excessive relative motion between the radial head and capitellum may lead to wear of capitular cartilage and may cause other soft tissue damage and distal radial-ulnar joint pathologies. Studies have shown that the contact between the noncircular shape of the native radial head and the radial fossa of the ulna causes translation of the radial head during pronation-

supination. [249-251, 260] It has been observed that the radial head may translate anteriorly with pronation and posteriorly with supination.[260] Due to the noncircular shape of the radial head, it has been speculated [251] that the radial head should translate predominantly in the medial-lateral direction. Note that all three experiments above were qualitative, providing no quantitative measures. [91]

Preservation of original kinematics also requires preservation of the normal axis of forearm rotation (i.e., the pronation-supination axis, the helical axis, the screw displacement axis). The location of the axis rotation as a function of prostheses alignment and orientation remains unknown.

There is controversy whether the annular ligament should be reconstructed with radial head replacement. Sojbjerg *et al.* [227] concluded that this ligament is the major stabilizer of the lateral side of the elbow. However, the experiment was performed amidst confusion about the nomenclature of the lateral collateral ligament (LCL) and they inadvertently cut the lateral collateral complex. To confirm this, a subsequent study [200] showed that it was the LCL ligament that was the important stabilizer of the lateral side of the elbow. The annular ligament did not have an effect on valgus laxity but was responsible for 2.3° of varus laxity, or about 20 percent of laxity achieved by transaction of the medial collateral ligament. Weiss and Hastings [260] applied forces to the radial head to study radioulnar divergence. They concluded the main contributors to the proximal radioulnar stability were the annular ligament and the middle third of the interosseous membrane.

2.2.3 Radial head prostheses: history and design

When indicated, a radial head prosthesis can help restore an individual's upper extremity function and increase independence in daily life. The first radial head prosthesis in the English literature appeared in 1941, when Speed described ferrule caps in patients with acute radial head fractures. [231] These caps were first tested on dogs prior to its documented use in two human patients. Shortly thereafter, metallic radial head caps were reported. [44] Later developments introduced acrylic radial head prostheses [53] which had a notable incidence of fracture (2 of 14 implants.) The next major innovation was the introduction of the silicon implant by Swanson *et al.* [241] and did not perform well. Complications included fractures due to inadequate load transfer capabilities and silicone-based synovitis at the elbow, which required removal within two years. [185, 254] The instability and unsuitability of the silicone inspired the design of metallic radial head implants, which are still popular today.

Today's surgeon can choose from a variety of commercially-available radial head replacement systems. Some are a monoblock design, meaning that there is only one rigid body component of the prosthesis. Others are bipolar, allowing the prosthesis to dynamically adjust to the kinematic requirements. Some implants are designed to be cemented into the radius while others are not. Figure 3 shows a representative sample of the implants available, and the salient characteristics of each implant are described below. Despite these advances, Beredjiklian *et al.* concluded that, among commercially-available implants, the size ranges seemed to overestimate the size of the radial head. Inappropriate sizing can lead to unintended changes in proximal radial length, which can have adverse effects on the resultant elbow kinematics and load transfer capabilities. [24]

Figure 3: Commercially-available radial head prostheses.

2.2.4 Radial head prostheses: commercially available

A brief overview of select radial head prostheses is given here. A summary, along with photos of each implant discussed, is shown in Figure 3. All implants insert into the canal of the proximal radius. Some are intended to be cemented in place while others are designed to freely rotate in the canal. While some implants have only one piece, others offer interchangeable stems and heads which can be assembled *in vivo*. Monoblock implants are those in which the stem and head remain in fixed relative positions throughout motion. Bipolar implants permit rotation of the head with respect to the stem.

The Swanson radial head prosthesis (Wright Medical Technology, Arlington, TN) is a monoblock design (Figure 3, inset a). It is designed to be press-fit into the radial shaft such that it does not freely rotate. Its geometry copies that of the original Swanson silicone implant. Its salient design features include availability in five sizes and a shortened stem, which shortens the overall length but increases cross-section to ensure easy surgical insertion.

The rHead Radial Implant System (Small Bones Innovations, New York, NY) has a cobalt-chrome surface (Figure 3, inset b). There are multiple stem lengths and head sizes to choose from, including heads of varying thickness. The roughened stem is designed to be inserted with our without cement at the surgeon's discretion, and is curved to facilitate easy insertion. A bipolar option, the rHead RECON, is also available (Figure 3, inset c) and allows for 10° of motion.

The Ascension Modular Radial Head system (Ascension Orthopedics, Austin, TX) is made of cobalt-chrome and available in six sizes head sizes (Figure 3, inset d). The stem is available in 4 sizes and is grit-blasted to permit insertion without cement. The implant can be

assembled either *in situ* or prior to insertion, and has a notch to permit easy annular ligament reconstruction.

The Liverpool Radial Head replacement (Biomet, Warsaw, IN) is a monoblock implant which offers two head diameters and thirteen length options. Unique to this design, the head includes an angled articulating surface and an offset stem intended to reproduce the natural curve of the radius (Figure 3, inset e.) This makes the insertion angle critical to the proper function of the implant. The stem is coated to encourage bony ingrowth, and is thus recommended for uncemented fixation.

The Judet prosthesis (Tournier SA, Saint-Ismier, France) is also bipolar. Its cobalt chromium stem is available in two sizes and designed for cemented insertion. The proximal stem is angled to reproduce natural anatomy (Figure 3, inset f.) The heads are offered in two diameters and have a polyethylene insert to allow bipolarity. When fixed to the stem, the bipolar articulation allows for 35° of rotation in all directions, which ensures capitellar contact throughout the range of motion.

The Biomet ExploR (Biomet Orthopedics, Inc., Warsaw, IN) is a two-piece implant system that was developed as an alternative to the one-piece implants. Shown in Figure 3, inset g, this design permits insertion without removal of the radial collateral ligament. It is made of cobalt chrome (CoCr) and Titanium alloys, and is available in three head diameters with five stem diameters and five stem lengths.

The Evolve Radial Head implant System (Wright Medical Technology) is a modular prosthesis made of CoCr (Figure 3, inset h). One-hundred-fifty size combinations are possible with its ten stem sizes and fifteen head sizes. The extended head lengths can accommodate a

wide range of damaged radius lengths. The implant can be assembled *in situ* and is designed to be uncemented so that the shaft can rotate freely inside the radial canal.

The Solar Radial Head system (Stryker, Rutherford, NJ) is a monoblock design and produced in five sizes (Figure 3, inset i). Made of CoCr, this implant is intended only for cemented insertion, and is advertised as being superior for high-load patients.

The Katalyst Radial Head Implant (KMI, Carlsbad, CA) is a modular implant made of CoCr. This system is unique in that it offers a telescoping shaft so that its length can be adjusted after insertion (Figure 3, inset j) to offer between 2mm and 10mm of spacing. Three head sizes and two stem widths are offered. The stem is designed to be press-fit into place without cement. The neck design is bipolar to allow for dynamic adjustment of the prosthesis to maintain capitellar contact. It permits up to 15° of motion at the head-stem junction.

2.3 MOTION ANALYSIS BACKGROUND

2.3.1 History and development

Motion analysis is an important tool for biomechanical studies. The overall goal of motion capture is to record movement in a compact and usable manner. [98] The first reported instance of motion analysis was in the late 1800s by Braun and Fisher [239] who strapped Geissler tubes to the subject's limb segments and recorded the motion with four still cameras. The illumination was interrupted at regular intervals with a large tuning fork. The computational time between the data acquisition and the final results took several months. The use of interrupted light and photography continued into the mid-twentieth century. In the 1960's, a new method was

developed that combined reflective markers, strobe lighting and photography to track the motion of specific markers only. [239] Electrogoniometers were soon developed and incorporated and dramatically reduced computation time. Automated tracking systems emerged in the late 1960s in Europe and made three-dimensional motion analysis a reality. [239] The Vicon company released the first automated tracking and computation system in the late 1970s. Since then, considerable work has been done to study the optimal marker sets to use for a given movement, such as the Helen Hayes marker set [132] and the Cleveland Clinic marker set [40] The Euler/Cardan method and the helical axis method became the most frequently-used methods to calculate joint angles, though methods based on planar projections, direction cosines, and that described by Grood and Suntay [107] still received considerable attention.

Through the use of digital or digitized video, motion can be precisely quantified. Historically, the points tracked have been either sensors or passive reflective markers, though modern-day techniques include the exploration and development of markerless motion capture. [50, 61] Human motion analysis has become a useful tool in animation [203], sports performance analysis [195], and in the biomechanical analysis.

The general principle behind motion analysis is that, by viewing the same points (markers) by multiple cameras simultaneously, it is possible to mathematically compute the precise three-dimensional positions of the points. Originally applied to the analysis of gait, early motion analysis systems were cumbersome and demanded extensive user interaction to track the locations of markers in space. As computing power increased, modern-day systems became user-friendly and can reconstruct the desired points' locations in a matter of seconds. Today, human motion analysis is used to quantify the motion of body segments or whole body motion

using one or more cameras. Additional techniques are being explored to recognize human activities from image sequences. [3]

Current motion analysis systems offer quick computation of marker coordinates. Typical video systems use charge-coupled device (CCD) cameras to track the three-dimensional coordinates of marker points, which can be passive (retroreflective) or active (light-emitting). Recognition of the markers in the video frame can be performed either via pattern recognition software or with the use of dedicated hardware circuits. Active markers are typically pulsed sequentially, permitting easy tracking by the system. Passive marker systems tend to be favored as they do not require the wires, batteries, and other electronic accessories needed by active marker systems. [54]

2.3.2 Known challenges with motion analysis

While motion capture and analysis offers valuable insights into human movement and performance, it has weaknesses. Motion capture data by definition contains a large amount of information irrelevant to the motion of interest. By nature, the data is cumbersome to manipulate and relies heavily on sophisticated, but sometimes imprecise, image-processing techniques. Another challenge is the data collection itself – techniques for processing have evolved much more quickly than the hardware to collect the motion. [98] Whether based on mechanical or magnetic sensors, or specially-designed cameras which view carefully-lit markers, even today's most advanced systems are expensive and intrusive.

Certain challenges with motion analysis have been studied, particularly those with direct application to the study of gait. Soft tissue motion, or the relative motion between the skin and the underlying bones, has been extensively analyzed. [234] Other previously-studied sources of

error in motion capture can include errors due to imperfect camera calibration, lens distortion, and computational errors in the process of reconstructing markers' locations. Calibration is the process of computing the mapping between image coordinates, as seen by the cameras, and ray directions in space. [235] Accurate camera calibration is critical to the successful computation of quantitative results from motion analysis. It has been shown that noise in marker centroid estimation can have a dramatic effect on the camera calibration parameters. [149] This affects translation much less than the rotation. Aside from the direct linear transformation technique [51], typical calibration routines are not available to the public [206] and thus it is difficult for the user to customize the calibration parameters for a nonstandard application of motion analysis.

One issue that is particularly of concern for those tracking extremely small motions is that the visible effects of small errors in position are significant. [98] It is important to note that, particularly for animation, that the same magnitude of error occurring at a different time or in a more fortunate direction would not be visibly perceivable. These jitters and wobbles can come from the computations involved in the reconstruction of the markers' trajectories or the failure to enforce trajectory smoothness across consecutive frames. The importance of true high-frequency motions to both animation and biomechanical analysis means that high-frequency filtering is not an adequate solution.

2.4 JOINT MOMENTS AND MOMENT ARMS

Muscles apply forces to the bones at their insertion sites, and movement occurs as these bones rotate about joints. Since muscles rarely, if ever, pass through the axes of rotation of the joints upon which they act, muscles actually exert moments about the joints they cross.

2.4.1 Moment arms: definition

By definition, a moment $\vec{M}$ is a turning effect that a force has about an axis. [229] The magnitude of this effect is equal to the product of the magnitude of the force and the shortest (perpendicular) distance between the force's line of action and the axis of rotation. That is:

$$\left|\vec{M}\right| = \left|\vec{d}_{perp} \times \vec{F}\right| \qquad (1)$$

where $\vec{F}$ is the applied force vector and d_{perp} is the moment arm, or shortest distance between the force vector and the axis of rotation. The moment arm is in effect a measure of mechanical advantage: increasing this distance can increase the magnitude of the resulting moment without requiring an increase in the amount of force applied.

2.4.2 Moment arms: application to biomechanics

Movement is created when muscles create moments about joints which cause rotation. A muscle's mechanical advantage is its moment arm. The farther its effect is from a joint, the greater movement effect it can have. Given that muscles originate and insert asymmetrically about a joint's axis of rotation, a muscle's moment arm is dependent on joint angle and thus changes throughout the muscle's range of motion. Muscle moment arms can be measured in cadaver specimens and can be used to evaluate the movement-generating capability of a muscle. Since the same cannot be measured in live humans, measurements made in cadaver specimens may not accurately represent the same quantity in the living population. However, given that

moment arms are a geometric property, the results obtained from cadavers are believed to be representative of those in the living population.

The magnitude of a muscle's moment arm is of interest to biomechanists and clinicians alike. Quantifying the moment arms of muscles can offer insights into the moments that are produced at a joint, which can in turn influence prosthesis design. Clinically, a surgeon may be interested in a muscle's moment arm so that it may be accurately reconstructed after trauma.

2.5 BACKGROUND: JOINT SIMULATORS

Human joint simulators that use cadaveric specimens can be grouped into two categories: physiologic and kinetic. In a physiologic simulator, the physiologic structures control the specimen through both actuation and constraint. In physiologic simulators, forces applied via tendons actuate movement. The interface of bones and actuators with the supporting structures incorporates clearly defined anatomic boundary conditions that recreate adjacent joint movements and muscle lines of actions. Muscle coactivations (in antagonist-agonist sets) can be included. The joint can then perform movements that cannot be distinguished from the joint movements of live subjects. Researchers have successfully used physiologic joint simulators in the wrist, elbow, shoulder, hip, knee and ankle. [64, 130, 179, 222, 262] Physiologic simulators are advantageous in that, when accurately controlled, they can produce movement identical to human movements while simultaneously placing forces with anatomically correct locations across the joint, in the joint and in the surrounding tissues. However, these simulators are challenging to control as the joints of the human body have redundant actuators. That is, multiple muscles can produce the same effect at each joint. This redundancy means that there

are infinitely many muscle activation combinations to produce a given movement or force at each joint. Thus there is no available controller to replicate the human nervous system.

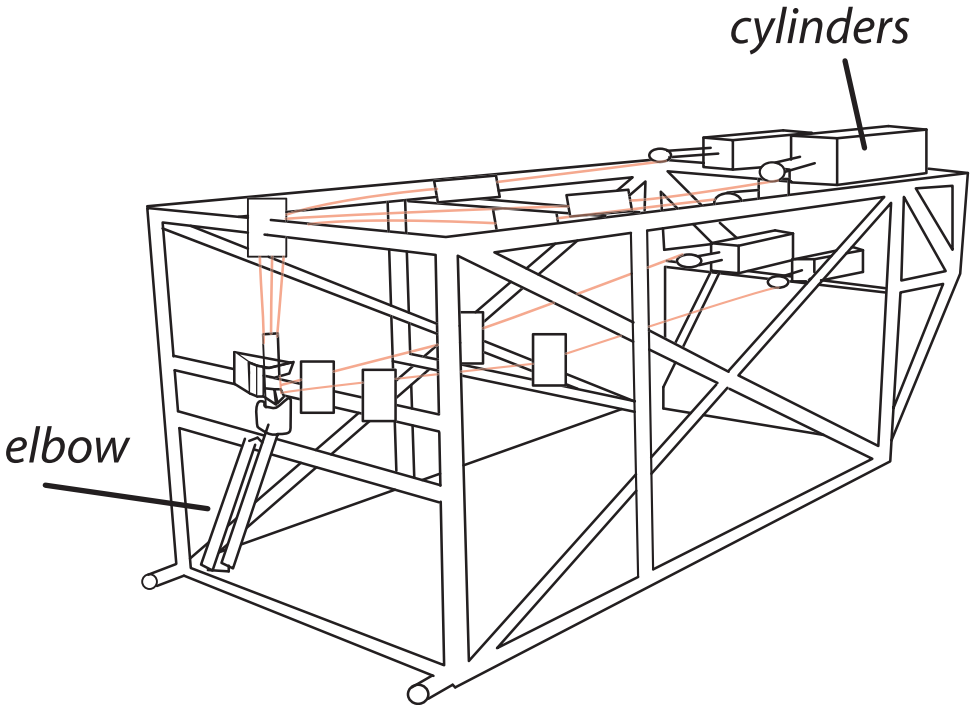
Kinetic simulators impose external movements and loads onto joints. Kinetic joint simulators typically employ robotic manipulators. The robotic manipulator can place known loads on joints or mandate well-known joint movements. Typical construction of a kinetic simulator includes the alteration of an industrial robot, [76, 89] or custom construction to adapt to the anatomy of a particular joint. [17] Kinetic joint simulators have the ability to accurately recreate the same set of motions with the same specimen before and after simulated surgical interventions or injuries. Kinetic simulators also eliminate the issues dealing with muscle redundancy that are inherent to physiologic simulators as the muscles themselves are not used directly to create forces or motions. However, kinetic simulators require the input of a movement or a set of external forces and moments that may not be physiologically correct. Models of human movement are often used to drive kinetic simulators, [171] so that the model's fidelity affects the realism of the simulation.

Simulators of each type may also incorporate features of the other type in order recreate specific testing conditions, such as Bach's inclusion of an specific muscle forces in a kinetic knee simulator. [18] Some experimental studies require components from both physiologic and kinetic simulators, creating a hybrid simulator. For instance, Maletsky constructed a physiologic knee simulator but wanted to adjust the moment at the knee to include the gastrocnemius/soleus support across the ankle. [164] Therefore, a rotational actuator (a kinetic component) was added at the ankle to apply a torque to the distal tibia. Conversely, Bach wanted to include quadriceps forces with tests performed in a kinetic simulator and added an actuator (a physiologic component) to a kinetic knee simulator. [18]

2.5.1 Elbow simulators

The only elbow joint simulator reported to date is a physiologic elbow simulator at the University of Western Ontario, London, Canada. [130] This simulator has been used to study the contribution of muscle forces to elbow stability, [14, 73] the contribution of radial head prostheses to valgus stability of the elbow [142] and various ligament reconstruction procedures in cadaveric elbows. [15, 139] While this simulator has expanded the understanding of elbow reconstruction, it has several limitations: 1) It operates in an open-loop manner, meaning that no feedback is used to adjust the control scheme and ensure physiologically-correct forces and motions, 2) Therefore, many iterations are required to create the desired arm motion, 3) The arm is led through a prescribed flexion-extension movement with a single fixed non-physiologic external moment, 4) Synergism between muscles is based on static EMG data, muscle cross-sectional areas (CSA), and moment arms that are assumed to be fixed, and 5) There is no antagonistic muscle control. The success of the Syracuse Wrist Simulator [262] with a straightforward PID control algorithm demonstrates that feedback control of an upper-extremity joint simulator is attainable and addresses many of the shortcomings of the above-described elbow simulator. In particular, since it uses feedback control, the desired motion can be achieved more quickly in a manner that is more representative of *in vivo* limb control. The Syracuse simulator also illustrates that the experiments performed on such a device lead to clinically-meaningful results. Thus we have been inspired to develop a closed-loop feedback-controlled elbow simulator for use with cadaveric elbows.

An elbow simulator at Allegheny General Hospital (AGH) has been designed and the hardware constructed and assembled (Figure 4). [163] The frame was designed to hold five servoelectric actuators controlled by a central controller card (ACR8020, Parker-Hannifin,



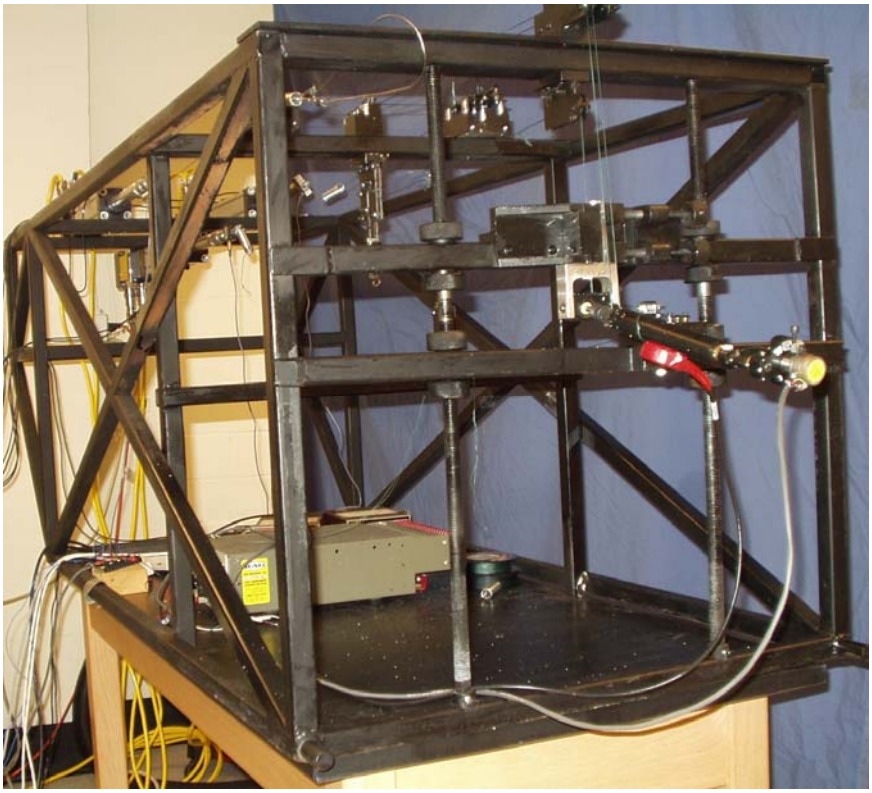


Figure 4: Drawing and photo of the AGH Elbow Simulator.

Rohnert Park, CA.) Elbow specimens are clamped to the front of the frame. The simulator was designed to accommodate both right and left elbow specimens with simple hardware adjustments (see Appendix A). Custom pulleys are adjustable to maintain physiologically-accurate moment arms throughout the elbow's range of motion. [151] A unique feature of this simulator design is that the frame can rotate 90° in either direction to test elbow specimens under varus and valgus loads, such as those experienced during baseball pitching of curveballs and screwballs. Figure 4 shows a drawing and photo of the simulator.

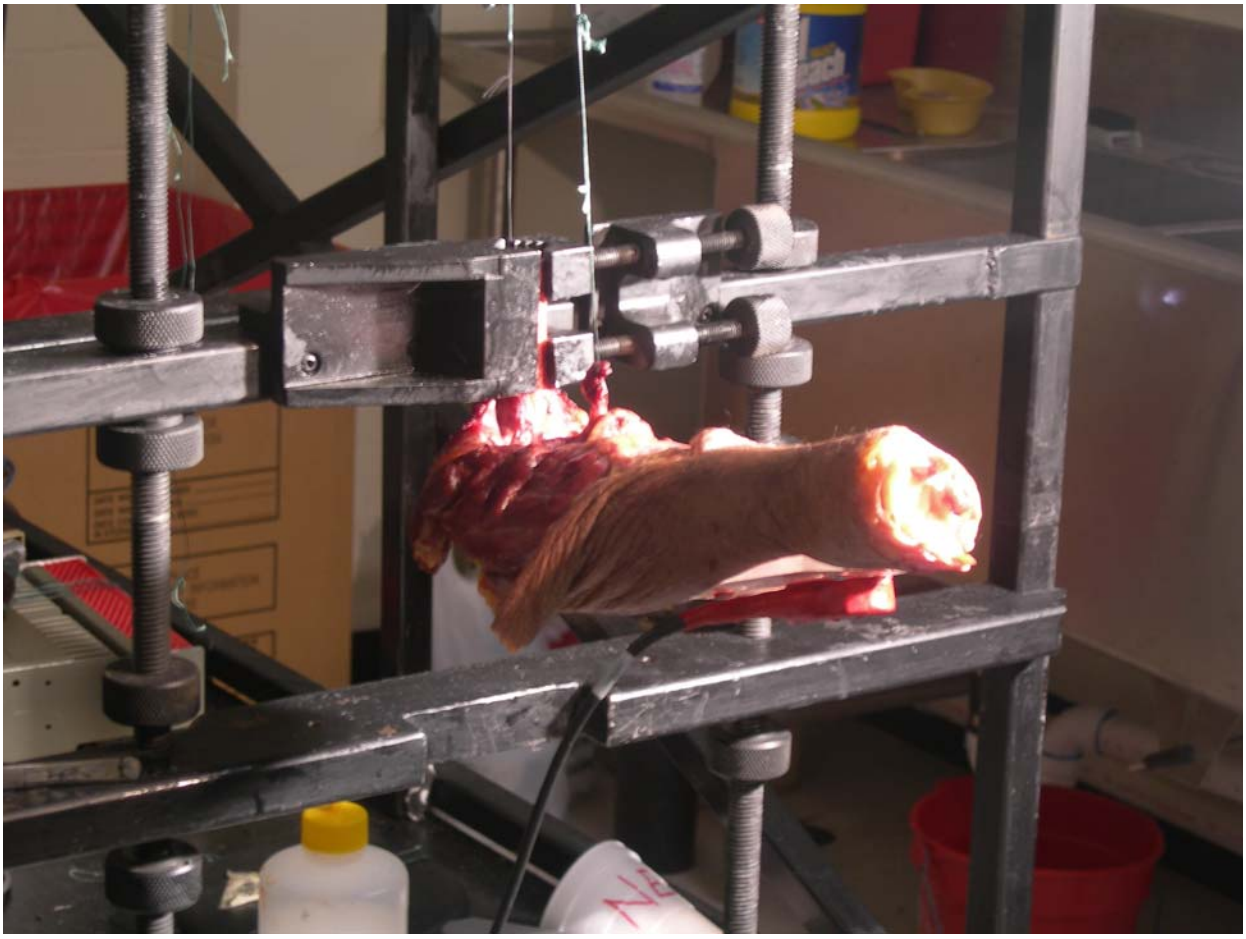


Figure 5: Photo of a cadaver specimen in the AGH elbow simulator.

2.5.2 Background: Motor control.

The neural control of movement has received considerable attention and is critical to the development of a joint simulator. [133, 173] The process of controlling limb movements is complex – muscles can work in agonist-antagonist pairs to produce joint motions, while the brain acts as a controller, receiving feedback about joint position and muscle forces from the Golgi tendon organ. [223] Yet motor control may be explainable using simple principles. [109] Investigations into the complex neuromusculoskeletal control have taken various approaches. Some investigators have begun their investigations at various levels in the musculoskeletal system and its pathways, [103, 118, 169, 197, 228, 258] while others have studied the movement outcomes. [67, 68, 87, 236] Among those who seek to model the brain's control methods, models of cerebrocerebellar control that include phase lags representing neuromuscular activation and transcortical round trip (long loop) delays show great promise for joint simulator control. Dynamic simulations of upright balance have shown that the recurrent integrator proportional integral derivative (RIPID) feedback model stabilizes the long loop proprioceptive feedback loops. [168] The algorithm was improved by providing hybrid force feedback and gain scheduling to create feedback recurrent integrator proportional integral derivative control (FRIPID). [127] Not only did the algorithm provide stability to large and small disturbance inputs, but predicted joint torques and EMG activity were found to be very comparable to experimentally measured data. While these and other algorithms have been validated in simulation, experiments with a physical (cadaveric) elbow model have not been done to evaluate control performance.

Feedback plays a critical role in the neural control of movement. [213] Both feedback and feedforward control have been used in some models. [238] In fact, a feedforward

neuroprosthetic controller was evaluated using an artificial neural network. Though the results relative to the neuroprosthetic were mixed, the paradigm was useful and can be used to evaluate future control designs. [160] Evidence also suggests that positive feedback, or feedback that causes the system to respond in the same direction as the perturbation [198] can provide stable control in gait and locomotion [208, 209] even though it is often associated with instability and oscillation. [38] Experimental studies using an exoskeleton that could provide velocity-dependent force resistance also suggest that both feedback and feedforward control strategies are used in gait. [152] Another study suggests that the brain tends to associate motor action at a joint with the same joint's intended (sensory) motion. [224] *In vivo* experimentation found that muscles under positive feedback control remained stable, even with high loop gains and delays up to 40ms. [209] Positive feedback has also been found in a brainstem tactile sensorimotor loop in rats [197], suggesting that it is not an anomaly to find positive feedback in biologic systems. Others have suggested that the Golgi tendon organ, which reacts to muscle force and usually provides negative feedback to the spinal cord, can also provide positive feedback under the right conditions. [79]

Feedforward control has been used in conjunction with a feedback controller to improve performance in functional electrical stimulation. [150] There is evidence suggesting that optimal feedback control [245] may play a role in linking motor behavior, limb mechanics, and neural control and offer insight to how the brain plans and controls movement. [219] Still others have suggested that voluntary movement control is composed of three elements: a muscle activation pattern, a kinematic plan, and a volitional set, or interaction of multiple reflex mechanisms (which rely on feedback) that can assist if the actual trajectory deviates from the planned one. [102, 103]

One commonly-used feedback control strategy is Proportional-Integral-Derivative (PID) control. When properly tuned, a PID controller is stable, has a short rise time, and eliminates steady-state error. [198] Thus it is a good choice for joint simulators and has been successfully used in a wrist joint simulator. [262] Somewhat in contrast, model-reference control has been developed and validated in a simulation, [57] but has not been applied to the control of cadaver specimens. Model-reference control is a form of adaptive control in which the gains of the controller change in response to the inputs and feedback. Simply put, this means that the controller compares its output to the output predicted by a model and uses the difference between the two as the error in its computations. It can also combine both feedforward and feedback control. The results of these computations are used to adjust the controller's own gains.

Model-reference control has been used in simulations of joint movement [56] and has been implicated in other physiologic processes with regard to the brain's development of internal models as a basis for motor control, [120, 121, 137] suggesting that it may be used in the cerebellum, which plays a critical role in movement control. Colbaugh and Glass initially suggested its application to biomechanical systems in 1993, as this control scheme requires no *a priori* implicit knowledge of the system's internal dynamics. This is particularly beneficial in biomechanical studies, in which the internal dynamics of the specimen (precise musculotendon lengths and material properties) are unknown and difficult to obtain *a priori*. This method was validated in simulation, but not in actual specimens. [56] In addition, the notion of an "internal model"-based movement strategy is prevalent in motor control research, [67] which makes the natural suggestion that a control law that includes a model may accurately simulate the neuromusculoskeletal control system. Adaptive control has been proposed for a control system for artificial hearts [145] and electrically stimulated muscles in cats. [25] Others have shown

that a model-reference adaptive controller can be used to control the knee joint of paraplegics if the reference model parameters were chosen appropriately. [112] In addition, model reference adaptive control has been shown to be a superb choice for functional neuromuscular stimulation, which holds tremendous potential to improve the lives of paralyzed individuals. [4, 5]

Upper extremity movements are an appropriate paradigm in which to study neuromuscular control because the motor skills of the hand and arm include the most finely controlled movements humans can produce. The motor cortex contains a disproportionately large area devoted to hand control. Thus the study of neuromuscular control has extensively examined upper extremity movements. Various control strategies have been proposed to model neural control of the upper extremity, including simple PID control, optimal control, fuzzy logic, neural networks, recurrent integrator proportional integral derivative (RIPID), and FRIPID. [35, 124, 127] Neural commands for muscle actuation and mechanoreceptor feedback are known to contain higher and lower level pathways of information transmission. [104] Force, position, velocity and stiffness are known to be central to control [99, 106, 113, 240] and have formed the basis of theories of neurocontrol. [26, 84, 113] St-Onge *et al.* suggest that the timing and amplitude of EMG bursts are in effect representations of the long-lasting dynamic response of the central, reflex, and mechanical components of the neuromusculoskeletal system to a short-duration shift in that system's equilibrium state. [237] This "equilibrium-point hypothesis" has been studied by Ghafouri and Feldman, who claim that internal models are a natural result of the system under the belief that shifts in its equilibrium point are what instigates movement. [95] Proponents of this hypothesis have furthered its strength by suggesting that it is necessary to include stiffness as part of the postural model. [221] However, Kistemaker and colleagues found the task of designing a controller for realistic movements of an arm model in was not as simple

as proponents of the equilibrium point hypothesis claim. [144] Although these theories differ on the extent to which lower- and higher-level neural pathways influence movement, all theories can be tested on a sufficiently robust simulator. Simulations and models of neuromuscular control include both pathway levels and contain all the features of standard control theory used to analyze robotic and mechanical systems. [56, 127] Our robust simulator design has provisions for these while using the extensive tools of standard hardware control theory.

2.5.3 Background –Studies of elbow movement control

Elbow movement has been extensively studied in the literature particularly with respect to its role in sports performance such as baseball pitching. [115, 157] Some investigators have quantified the kinematics of elbow movements [7, 48, 158] and some have used mathematical models. [2, 46, 100, 162, 211, 259, 263] Others have studied how the kinematics are affected by fatigue. [62, 80, 118] Winters and Kleweno found that biarticular muscles' change in force production capability with joint configuration is mainly due to their change in length, not their change in moment arm. [265] Elbow moment arms have been measured by Murray and others. [163, 192-194] The effects of muscle geometry [265] and the mechanical properties of muscle, [264, 266] have contributed to our understanding of the elbow. However, the study of these physical attributes cannot offer insight about elbow movement control. Measured EMG suggests that major elbow muscle activity is determined by the magnitude of the moments created at the elbow joint. [90] Allin and Inbar dynamically characterized the forearm in healthy humans using the biceps, triceps, and pronator teres muscles by means of EMG measurements and electrode stimulation of the elbow muscles. They measured wrist and elbow rotation, and determined that a third-order model with one simple pole and one complex pair best fit the system. [5] This

approach was based on a Hill-type fourth order model in which a pole and zero were sufficiently close to cancel each other. They later used this information to evaluate potential control strategies for functional neuromuscular stimulation, and found that the model-reference adaptive controller was a practical solution. [4] In a fundamentally different approach to the study of the biomechanics and control of joints, the Matsuoka research group has modeled the hand and its muscles extensively. [96] This technique could be applied to the elbow joint as well. Several control strategies have been postulated to represent how humans control their own elbows. One approach towards understanding elbow control strategies is rooted in the topic of muscle activations and coactivations. It has been shown that the amount of coactivation between the *biceps brachii* and *triceps brachii* does not vary with elbow joint movement speed. [20] Ives and colleagues have studied rapid elbow flexion movements in humans and have concluded that specific agonist muscle activation patterns are set according to the antagonist muscle's neuromuscular constraints and the perceived expectation of the movement dynamics. [122] They have also concluded that males, but not females, can modulate their braking capacity during rapid-release movements. [123] However, they measured *biceps brachii* activity as the agonist, with the *triceps brachii* as the antagonist. Since the *brachialis* is the main elbow flexor, their conclusions may be based on an inappropriate assumption. Still other researchers have measured EMG activity during wrist torque generation activities (that is, forearm pronation-supination torque) and have reported that, with the exception of the *brachioradialis* and *brachialis,* there are no invariant muscle coactivation patterns across all conditions. [34] While this may contradict others' findings about muscle synergies, the same study also found that superposition of flexion/extension and pronation/supination motions did not alter the relative proportions of the muscle activity magnitudes, though it did affect their magnitudes. This

illustrates that muscles may behave in a way that seems counterintuitive – for example, *pronator teres* may be active during flexion in order to resist the supination component of the *biceps brachii*, and this may vary with the task being performed. [39] While the work presented in this thesis does not directly offer new insights into motor control, it does set the stage for future experiments to investigate the motor control of the elbow.

One important application of studies of elbow control is to meaningfully evaluate the function of prostheses such as radial head replacements. Researchers have studied radial head replacements in isolation. However, the absence of a feedback-controlled elbow joint simulator hampers progress as the role of neural control in complex elbow movements is unknown. A more complete understanding of the control strategies used will allow us to better study the effects of different implant designs and will ultimately aid clinicians in choosing the best design for their patients. The work presented in this dissertation drives towards this goal.

3.0 LITERATURE REVIEW RELEVANT TO THE SUBSEQUENT STUDIES

This chapter presents an overview of the existing English-language literature of the three major topics of this work. Section 3.1 presents an overview of previous work measuring elbow moment arms. Section 3.2 summarizes the findings related to motion analysis. Finally, in Section 3.3, a review of radial head replacement studies is given.

3.1 MOMENT ARM MEASUREMENT

There are three main approaches to the measurement of moment arms: the geometric, direct-load, and tendon-displacement methods. The tendon-displacement method is the most prevalent.

The *geometric measurement method* is based on the literal definition of a moment arm as the perpendicular distance between a muscle's line of force (action) and the joint's axis of rotation. This method requires the three-dimensional reconstruction of this axis of rotation and the spatial orientation of the muscle paths. The muscle paths can be determined in several ways. One way is via biplanar x-rays of cadaver specimens with metallic markers inserted into the tendons and muscles. Another is to digitize the bones and muscle paths during dissection. Serial cross-sectioning is a third method that can be used to determine muscle paths, and can be done directly by measuring serial cross-sections, or by computation from images such as those generated by computed tomography (CT) or magnetic resonance imaging (MRI) scans.

In cadavers, the direct dissection method is limited as it only permits moment arm measurement in one joint position and thus cannot be used to investigate the relationships between moment arms and joint positions in the same specimens. With the use of imaging measurements in both cadavers and humans, these relationships can be investigated using the geometric measurement method. It has been used in the elbow [6, 8] and with the trunk muscles. [170] Several investigators have taken a simplified two-dimensional approach with this method. [170, 196, 232]

Another way to measure muscle moment arms is the *direct load method* which is only suitable for use in cadaver specimens. This method has its origins in the definition of a moment arm as the ratio of the moment to the force developed by a muscle. In this method, a known load is applied to a muscle. The resultant force and moments produced at the distal segment are measured and used to estimate of the moment arm. The accuracy of this method is dependent on the sensitivity of the load cell used to measure the forces and moment and the ability to support the cadaver specimen in a sufficiently rigid manner to ensure that no movement is created with the applied forces. Thus, this method is challenging to implement, although it has been used in a few cadaver studies. [94, 108]

The *tendon-displacement method* is based on the definition of a muscle moment arm that stems from the principle of virtual work. The principle of virtual work states that the work done by a force over an infinitesimal displacement (dr) is equal to the work done by the resulting moment through a corresponding infinitesimal rotation, $d\theta$. (Equation 2) When applied to a muscle's moment arm, this states that a muscle's moment arm determines the excursion of the musculotendon during joint rotation. That is, when subjected to a constant force, muscle

moment arms, d_{perp}, are calculated as the derivative of the tendon excursion r, in mm, with respect to the joint angle, θ, in radians.

$$d_{perp} d\theta = dr \qquad (2)$$

$$\boldsymbol{d_{perp} = dr/d\Theta} \qquad (3)$$

This is the most commonly-implemented method of moment arm measurement, though like the geometric method, it cannot be used to measure moment arms in live humans. Of the three measurement methods, this method most easily permits the study of the variation in moment arms with joint position. It has been used to measure moment arms about the finger joints, [12, 32, 155] wrist, [159] elbow, [194] shoulder, [201] ankle, [146] knee, [65] and hip. [257]

Amis *et al.* measured moment arms for elbow flexion as analyzed by direct dissection of cadavers. The "probe and grid" method was used to create an accurate outline of the elbow's cross-section. Briefly, this method involved rigid mounting of the specimen to a two-dimensional measured grid. This grid, together with a vertical ruler, permitted the three-dimensional measurement of muscle and tendon locations. The muscle paths were computed and were then used to derive the moment arms. These results compared well with previously-reported data. [6]

An *et al.* used a serial sectioning method to investigate moment arms. Cross-sections of the bones and muscles were used to calculate centroids of each, and thus assumed to be the lines of action of the muscles. Relative to the elbow joint, they calculated moment arms relative to the center of the trochlea perpendicular to the forearm. This assumes muscle uniformity and perpendicularity of the lines of action to the cross-sections. While six joint positions were

studied, each position could only be studied in one cadaver. The moment arms were normalized by arm length. [8] Limited p/s moment arm values were later reported from the same group. [10] Later, the same research team developed a model incorporating the muscle length-tension relationship to calculate the muscle forces across the elbow joint. [9] The model also determined the moment arms.

The most extensive work pertaining to the moment arms of the muscles of the elbow is presented in Murray. [192] In addition to measuring flexion-extension moment arms, other architectural properties of the elbow muscles were quantified including musculotendon length, muscle length, fascicle length, sarcomere length, pennation angle, and mass. Anthropometric dimensions of the cadaver specimens and muscle attachment sites were measured. The relationship between the anthropometric dimensions, muscle attachment sites, and moment arms was studied. It was concluded that the distance between the joint's axis of rotation and a muscle's nearest attachment point explained 96% of the variation in peak flexion moment arms. Bone length alone did not correlate with peak moment arm, except in cases in which it correlated well with the nearest muscle attachment site.

Kawakami *et al.* used serial cross-section images from MRI scans to calculate elbow flexion moment arms with the elbow at full extension. [136] The results from this study do not agree with the trends seen in other reports in the literature, and are also not within range of moment arms measured in this thesis. This may be because the MRI images were calculated with the elbow in full extension, which exceeds the range of values reported by others.

Ettema *et al.* measured the moment arms of twenty-three muscle segments in the upper extremity, and was the first published study to measure both f/e and p/s moment arms in the same specimens. [81] Embalmed cadaver specimens were used with the geometric method.

The muscles were removed and their origin and insertion sites were carefully marked. Elastic strings were then attached to connect each muscle's marked origin and insertion point. The elbow was then posed in various combinations of f/e and p/s and the elastic string length was measured with a hand-held ruler, and the muscle moment arms were computed using the tendon-displacement method. The results of this study are limited due to the accuracy with which the "musculotendon" lengths could be measured with the hand-held flexible metal ruler. Pigeon *et al.* [204] reported moment arm variation with flexion-extension position by aggregating previously-reported results from the literature and fitting polynomials to the results. This approach is somewhat limited because it does not account for inter-specimen variability in moment arms.

Several investigators have used models to estimate moment arms. A model created using the SIMM software (Musculographics, Santa Rosa, CA) was created by Charlton *et al.* [49] The moment arms computed with this model by tendon excursion compared well with the calculations based on the direct vectors. Another model of the elbow joint complex was developed by Hutchins [119] and reported in Gonzalez *et al.* [100] This model was developed for the purpose of evaluating different control schemes and the moment arms were reported to validate the model. Dunning *et al.* also reported f/e moment arms for select muscles during the validation of the Ontario elbow simulator. [75] Lemay and Crago [153] developed a dynamic model to simulate forearm and wrist movements with the goal of complementing their studies of functional electrical stimulation and to predict clinical results of surgeries such as tendon transfers.

The p/s moment arms have been reported with less frequency than the f/e moment arms. In addition to the results mentioned above, [8, 81] Gardinier and Gonzalez reported p/s moment arms measured in an embalmed cadaver elbow for select f/e angles. [92]

3.2 MOTION ANALYSIS ACCURACY

Motion analysis is widespread and can produce extremely accurate results. However, in order to achieve high degrees of accuracy, motion analysis systems must be used carefully. In particular, the calibration of the working volume in which the motion occurs is critical. [111] The accuracy of early systems in use for gait analysis was reported as 1 part in 1000 for static images, and 1 part in 300 for dynamic tracking. [243]

Errors in motion analysis can be attributed to limitations of the equipment in use. [54] The resolution of systems using charge-coupled device (CCD) cameras is essentially limited by the pixel grid of the CCD chip. Sub-pixel accuracy is possible by estimating the centroid of a marker image and the more pixels a marker covers, the higher the resolution. [128] Image distortion is sometimes to blame for inaccurate results. Random errors may be induced due to electrical noise, marker flickering, the digitizing process itself, and shape distortion that may occur at high movement velocities. Compensation for some of these errors can be achieved by filtering or smoothing of the results. [54]

Motion analysis is used to develop animation. [97, 98] Its reliability for gait research and other large-volume applications is well-established [1, 138, 216] Accuracy of optical marker systems has been quantified [54, 156, 164] and software algorithms have been described to track the markers in space. [85, 190] Others have described the importance of calibration methods

[206] and camera parameters. [166, 248] A calibration technique using only a rigid bar, as opposed to a calibration frame, has been proposed. This method yields sub-millimeter accuracy in a cubic volume with a diagonal length of 1.5m. [29] The study of accuracy with respect to small linear displacements has shown mean position errors as small as 0.034mm in a small volume (0.7 m x 0.5 m x 0.3 m) using an unspecified three-camera system. [161]

Previous work has assessed relative error in the systems [52, 88, 165, 175, 212, 235], motion artifact due to soft tissue motion [41, 42] and compared several commercially available systems. [77, 78, 212] The comparisons included an analysis of the change in relative distance between two points caused by the object moving among different cameras' viewable ranges. Tracking sensors through gaps and occlusions is particularly challenging, [69, 135] as is tracking dense arrays of markers. [28, 41] Motion reconstruction, editing, and tracking methods have also received considerable attention. [85, 97, 166, 206] The accuracy of motion analysis systems is of particular interest to those in computer animation [36, 37, 98] and is known to affect clinical results. [1, 216] Noise is known to affect camera calibration parameters, [149] and the cameras' internal properties can also affect the results. [235, 248] Liu *et al.* have also shown that, for small motions, the use of diamond-shaped markers can give improved accuracy. [156] Vander Linden *et al.* showed that the Motion Analysis system was capable of 0.4° accuracy for rotational movements. [253] While the error in tracking motion spanned by multiple camera ranges has been studied, [36, 37, 69, 135, 175, 268] and algorithms have been proposed to predict the coordinates of markers that are temporarily obscured, [66] no work to date has quantified the error in position caused by the removal of markers from the view of one or more cameras, whether the removal is caused by obstruction or by data omission during the image reconstruction process. This issue is critical to the tracking of fine movements such as the travel

of the radial head on the capitellum during elbow motion. [35, 91, 176] Due to anatomical constraints, markers must be smaller than those conventionally used in gross motion analysis and the nature of the motion may not permit all cameras to view all markers at all times. For example, the body being tracked may itself obstruct a camera's view during certain motions. The desired outcome is an accurate continuous trajectory such that the measurement of fine motions is not influenced by the change in the number of cameras contributing to the reconstruction of the marker location.

3.3 RADIAL HEAD FUNCTION AND REPLACEMENT

It is of both clinical and scientific interest to quantify the kinematics of radial head implants. The hemiresection interposition arthroplasty of the ulnar head has merited a kinematics study. [217] The kinematic behavior of radial head prostheses remains largely unexplored in the literature.

3.3.1 Forces and load transfer

Loads across the native and replaced radial head have been studied. The bony structures of the elbow are believed to provide 75% of the joint's stability. [183] An early study indicated that, during fast elbow motions, the humero-radial force could be as high as 1.16 kN. [7] This estimate was based on a video analysis during high accelerations and thus may have overestimated the total possible joint forces. Beingessner *et al.* found an inverse relationship between radiocapitellar joint stability and radial head fracture size by sequentially removing

wedges of the radial head and applying joint loading. A change in stability was defined as a decrease in the shear force at the radial head. [22] Both elbow and wrist joint contact forces were reported during pick-and-place activities. Humero-radial forces of up to 800 N were reported. [46] A study of cadaver elbows with postereolateral instability revealed that the soft tissue components are critical to the elbow's stability, and that the lateral collateral ligaments were the most critical. [199] In fact, the authors noted that postereolateral instability did not occur until these ligaments had been transected.

The forces within the muscles that cross the elbow joint have also been studied. The maximum strengths for the elbow flexors, as a function of pronation-supination position, has been quantified. [131] The forces produced by the major muscles that cross the elbow joint depend on the size of the resultant elbow flexion and extension moments, but were independent of varus and valgus moments. [90] The maximum isometric arm forces generated in a horizontal plane were predicted in a model and correlated well with measured EMG data. [125]

3.3.2 Elbow kinematics

As with the studies pertaining to forces and loading, the kinematics of the entire elbow joint have received markedly more attention than the study of the individual joint components that make up the elbow. It has been noted that, due to the sparse amount of studies, there is no standard computational method to evaluate arm motion. [13] The three-dimensional range of motion of the elbow was quantified, including variations in f/e with p/s. [48] An early study suggested that total elbow prostheses be uniaxial. [158] The axis of rotation of the elbow has been quantified. In the early twentieth century, Fisher used a Reuleaux method and reported that the instantaneous center of rotation moved within an area of approximately 2-3 mm in diameter at

the center of the trochlea. [11] These findings were later confirmed using biplanar x-rays. [187] While it is generally believed that the f/e axis of rotation does not change during movement, [269] up to 8° of variation across individuals has been shown. [11] This is of particular clinical relevance in that, when treating patients using a hinged external fixator, the axis of the fixator must be aligned with that of the elbow to minimize motion resistance. [162] A separate study determined that orthopaedic surgeons can use fluoroscopy to determine the elbow's axis of rotation with an accuracy of 3.7°. [30] Elbow range of motion has been quantified during wheelchair propulsion and was shown to vary with propulsion speed. [27] A study of isometric muscle force at various elbow positions showed that the shape of an elbow muscle's moment-angle curve is dependent on the position of the other joints in the upper limb. [265]

The axis of rotation of the forearm, or the p/s axis, is considered to be constant and independent of elbow flexion position. [91, 114] This axis passes distally through the center of the distal ulna and proximally through the center of the radial head. Hollister *et al.* also noted that the interosseous membrane crosses this axis of rotation. While this membrane does not limit rotation, it can offer stability to the forearm if the bony structures are disrupted. The forearm axis of rotation was also used to quantify elbow instability. During passive motion, King and colleagues transected the collateral ligaments and reported increased instability, which could be mostly counteracted by muscle activations via the Ontario Elbow Simulator. [70] The flexor-pronator mass is also reported to have an effect on the elbow's valgus stability. [202]

Many studies of elbow kinematics have been within the context of muscle activation patterns and other neuromuscular control work. For example, Corcos *et al.* reported that fatigued movements involved lower muscle torques and slower speeds. [62] The kinematics of overarm throws have been reported within a study of the neuromuscular control behind natural throwing.

[63] A recent work suggests that when the forearm is modeled as two segments (radius and ulna), the axis of rotation for p/s and the translation of the radius with respect to the ulna during p/s can be quantified. This model, in conjunction with motion analysis in live humans, can detect changes in forearm axis of rotation with injury and recovery. Its accuracy remains limited, as a 20% variation in calculated p/s rotation was attributed to skin marker movement. [55] Three-dimensional movements of the upper arm have been studied during kinematically redundant arm movements. This study found that movement velocity decreases the accuracy of the performed forearm motion. [2] Surprisingly, the subjects' head movements did not show this relationship with velocity, which casts doubt on some models of movement control.

Though the kinematics of radial head implants have been previously investigated. More prevalent is the analysis of whole-elbow kinematics. One study used whole-arm kinematics in conjunction with manipulator-hand contact forces to identify the body segment parameters of mass and centers of mass. [148]

3.3.3 Radial Head Kinematics

In addition to studies of radial head prostheses, King *et al.* found that a capitellocondylar unconstrained total elbow prosthesis could restore acceptable elbow joint kinematics in cadaver specimens provided that the collateral ligaments both had sufficient integrity to ensure joint stability. [141] While this finding is not directly related to a radial head prosthesis, it nonetheless illustrates the important of soft tissues in the role of joint kinematics. Both passive motions and active motions in the Ontario Elbow Simulator were tracked with an electromagnetic tracking device. The outcome measure of total joint kinematics may be clinically important, but may not be accurate indicators of how each component of the prosthesis performs.

It has been shown that radial head fractures of less than one-third of the head's diameter may, with proper fixation, be biomechanically sound. In addition, with sufficient collateral ligament reconstruction, total elbow motion remained similar to the intact (uninjured) case. [23]

This kinematics of the radial head have been observed qualitatively, but have been quantified much less than the load transfer across the radiohumeral joint. Skalski *et al.* [225] qualitatively found that the prosthesis positioned itself to the center of the capitellum. Galik *et al.* have quantified the translation and travel of radial head prostheses in cadaver specimens during pronation and supination motions. [91] The same research team found that the transection of the annular ligament affects radial head travel. [177] Gupta *et al.* performed a computer simulation and analysis of an intact radial head, a silicone prosthesis, and a proposed ultra-high molecular weight polyethylene (UHMWPE) prosthesis, and found that the UHMWPE prosthesis transferred more force than the silicone while offering more stability to the joint. [110]

3.3.4 Prior studies of radial head kinematics completed at AGH

Two studies used a precursor to the AGH elbow simulator. [176, 178] These studies developed the analytic techniques required for the control experiments and produced the ability to accurately record each aspect of forearm movement. The first study sought to determine whether a bipolar prosthesis offered advantages to a monoblock prosthesis. It quantified radial head translation and variation of the axis of rotation during pronation-supination for the arthroplastic case. The radial head was replaced by a monoblock prosthesis with a smooth straight uncemented stem (Evolve, Wright Medical) and then a bipolar radial head (Katalyst, KMI Inc.)

Seven fresh-frozen elbows (including forearms) were mounted in a custom load frame (a precursor to the current simulator's frame.) The ulna rested on a support to fix the flexion angle

at 90° but was not otherwise constrained. Constant forces (22 N) were applied to the *brachialis* and *triceps* tendons to compress the elbow joint while p/s motion was achieved by manual actuation of the *pronator teres* and *biceps* tendons. Movements were tracked with a passive marker motion-analysis system (PEAK 5, Vicon-Peak Inc.). At the conclusion of each complete set of tests, each specimen was dissected and local coordinate systems for each body were determined using landmarks measured by a coordinate measuring machine. (Numerex, Zeiss) An elbow joint coordinate system was used similar to Dunning *et al.* [74] with the origin of the radiohumeral system at the center of the capitellum. The travel of the radial head from the location at the neutral position was then calculated, where neutral was defined by the p/s angle at which the radial styloid was directly above the ulnar styloid. The location of the p/s axis of the forearm (the finite helical axis, FHA) at each 10° increment was also calculated and the distances of this axis from the ulnar styloid and center of the capitellum were determined over two full p/s cycles. The spread index [72] and dispersion angles [267] were calculated. Tests of system accuracy showed translation precision of at least 0.3 mm and rotational precision of 0.5°.

To determine if the radioulnar joint can be defined by a single axis, positional and angular variability (dispersion) of the axis were calculated. Each elbow was tested first with a monoblock prosthesis and then with a bipolar implant. The average location of the radial head with respect to the center of the capitellum changed more than 1 mm in both the medial-lateral (M-L) and anterior-posterior (A-P) directions. Specimens achieved different extents of pronation-supination rotation, with 45º of pronation to 45º of supination forming the common range of motion analyzed in this study.

A repeated measures analysis of radial head location on the capitellum showed no differences between the native, monoblock, and bipolar cases ($p=0.19$). The average M-L and

A-P travel of the prosthetic radial heads is shown in Table 1, where travel is defined as the overall peak medial (anterior) location minus overall peak lateral (posterior) location. A paired t-test found a significant difference in medial-lateral travel. The location of the average p/s axis is shown in Table 2. No statistical differences in the location of the axis were found. The positional and angular variabilities were less than the system accuracy, thus the axis can be considered constant. A second study quantified the effect of annular ligament transection at 90º of elbow flexion in eleven cadaveric elbows found that transection of the annular ligament leads to greater radial head travel. [176]

Table 1: Radial head travel

	M-L [mm]	*A-P [mm]*
Monoblock	2.3 ± 1.2	3.5 ± 1.9
Bipolar	1.6 ± 0.7	3.0 ± 1.4
p-value	0.03	0.37

Table 2: Pronation-supination axis location.

	Capitellum		*Ulnar Styloid*	
	M-L [mm]	*A-P [mm]*	*M-L [mm]*	*A-P [mm]*
Monoblock	1.2±1.9	-1.2±1.9	-0.7±3.1	6.3±4.4
Bipolar	0.7±1.5	-2.1±1.8	-0.5±3.6	4.9±2.1
p-value	0.51	0.14	0.77	0.87

The monoblock implants travel a greater distance than the bipolar in the medial-lateral direction than in the anterior-posterior direction. Statistically, the amount of travel of the two implants is different only in the medial-lateral direction. Looking at the pronation-supination axis location, the axis is only significantly different between the two implants at the capitellum in

the anterior-posterior direction. This necessarily implies that the entire axis position is skewed differently between the two implants since one component of its location is changed.

3.4 SUMMARY

The studies described above show a need for three areas of investigation. The measurement of moment arms is needed to validate one aspect of the design of the AGH Elbow Simulator. The accuracy of motion analysis techniques for the study of very small motions is critical to emerging areas of interest in biomechanics, such as the tracking of the radial head in cadaver specimens. Finally, the development of a method to track the kinematics of the radial head is essential to the comparison of different implant types.

4.0 METHODS: SIMULATOR DESIGN IMPROVEMENT AND VALIDATION

The hardware design implemented by Magnusen [163] has been updated with several improvements. As a validation of the mechanical design, elbow moment arms have been measured in cadaver specimens. The methods are described and discussed in this chapter. First, improvements pertaining to the hardware are described. Then the methods used to validate the design intent of producing physiologically-realistic moment arms are described. The results are later reported in Chapter 8.

4.1 HARDWARE IMPROVEMENTS

As described in Magnusen's work, the simulator includes a robustly-designed frame capable of being used in three rotational positions to permit elbows to be tested in three different positions: upright, with a valgus load, and with a varus load. No modifications have been made to the frame itself, as seen in Figure 4. The same controller card (ACR8020, Parker-Hannifin, Rohnert Park, CA) remains in place. For this work, four actuators (BE series, Parker-Hannifin, Rohnert Park, CA) were used. These servoelectric are in-line with the motors and actuated via precision ballscrews to ensure smooth and accurate motion. Load cells were added in line with the actuators to monitor muscles forces. For the biceps and pronator teres muscles, 25-lb load cells

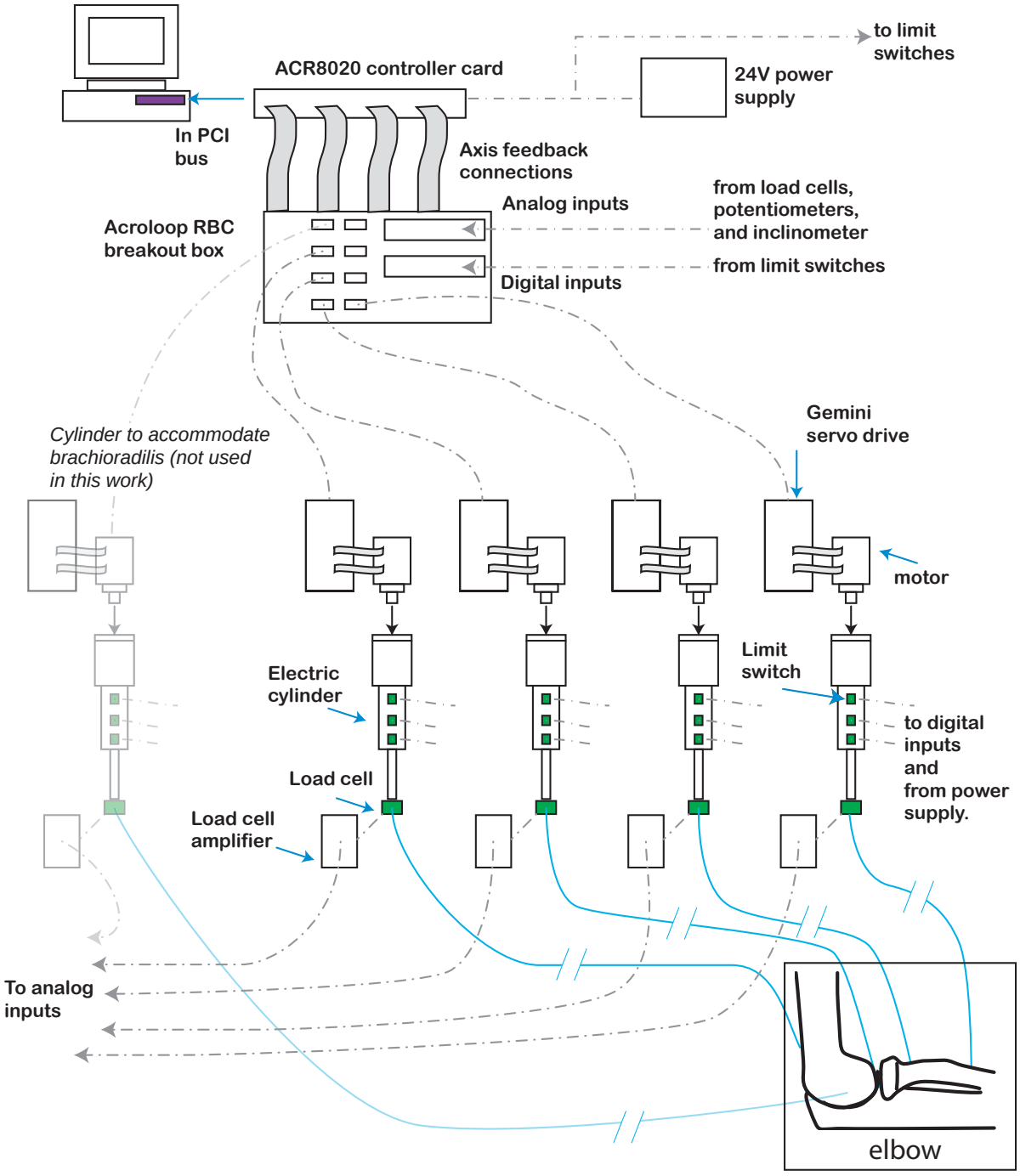


Figure 6: Schematic of simulator hardware and connections.

were used (MDB-25, Transducer Techniques, Temecula, CA). Fifty-pound load cells were used for the brachialis (MLP-50) and the triceps (MDB-50, Transducer Techniques).

The MDB-series load cells offer nonlinearity and hysteresis of 0.05% of the rated output. The MLP load cell offers nonlinearity and hysteresis of 0.1% of the rated output. An RBC Breakout box (Parker-Hannifin, Rohnert Park, CA) replaced the older RBD breakout box. This upgrade offered the advantage of easy access to the analog inputs via screw terminals and eliminates the custom secondary breakout box that Magnusen's work required. The now-obsolete secondary breakout box featured terminal blocks and over 120 wire connections of 26-gague wire, which were easily broken. Further improvements include the addition of limit switches (SMH-1N and SMC-1N, Parker-Hannifin, Rohnert Park, CA) to prevent the actuators from reaching the ends of their travel. These switches can be monitored and controlled via a custom PLC program in the ACR-View Software. An updated schematic of the overall system is shown in Figure 6.

Other minor hardware improvements were made to the system. A new computer (Dimension 3000, Dell, Round Rock, TX) houses the ACR8020 (Parker-Hannifin, Rohnert Park, CA) controller card, which facilitates easy backup of the controller programs and offers an improved user interface. The controller software has been upgraded to ACR-View version 5.2.0.0 from Acroview version 4.0, a precursor to the ACR-View software. This upgrade offered improvements to the controller interface and fixes several software bugs from the earlier versions. A notable advance includes the ability to monitor inputs, outputs, and parameters in real time via the Numeric Status Panel. In addition, the entire system was grounded in a star-grounding fashion to eliminate any ground loops. An emergency-stop button was added to facilitate the quick halt of motion if needed.

Slight alterations to the custom pulleys for the pronator teres and brachioradialis muscles were made to permit physiologically-correct relative positioning of the muscle origins. The older designs fixed the pulley assemblies for the pronator teres at equal heights and offered no adjustment in the direction of the cylinders' motion (parallel to the ulna with the forearm parallel to the ground.) An additional slot was added to the assemblies to give two discrete options for this, shown in Figure 7. In addition, custom shims were created to change the relative heights of the pronator teres and brachialis muscles.

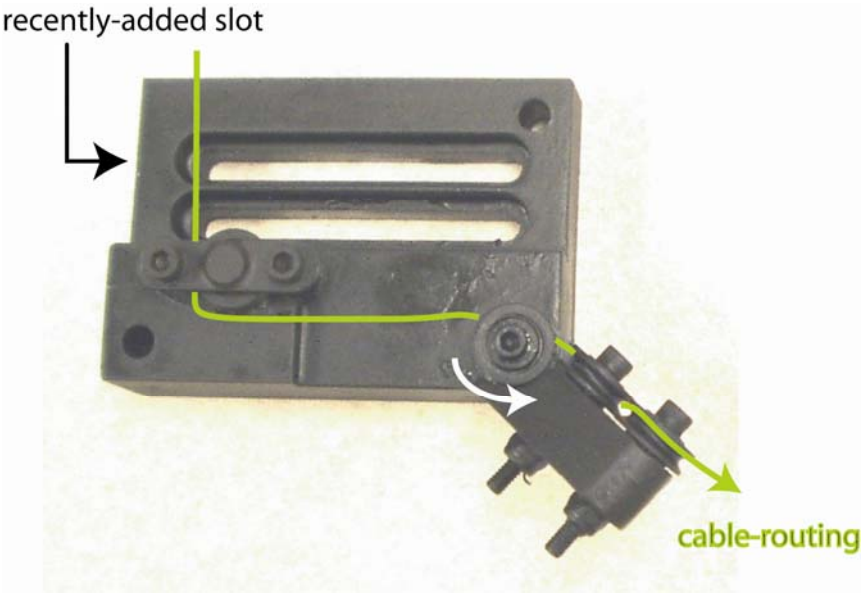


Figure 7: Photo of pulley assembly with cable-routing and swiveling indicated.

4.1.1 Calculated feasible moment arm ranges

Given that the pulleys in the simulator are adjustable, the simulator is capable of producing a range of feasible moment arms. Based on the muscle insertion data from Serig and Arkivar

[220], the range of feasible moment arms was calculated using MATLAB (The Math Works, Natick, MA) code. (See Appendix I.) For each muscle, feasible moment arms were computed for the pulleys at both extremes of their adjustment range. It is important to note that these values were based on only one set of anatomic muscle origins and insertion measurements. [220] The results are presented in Chapter 8.0 .

4.2 VALIDATION: MEASUREMENT OF MUSCLE MOMENT ARMS

One key feature of the AGH Elbow Simulator is its ability to reproduce physiologically-accurate muscle moment arms. Validation of this capability is critical. To validate, muscle moment arms were measured in three specimens using the tendon-displacement method, as described in Chapter 3.1. This method has its roots in the definition of the moment arm and the principle of virtual work. This method is the most prevalent in the estimation of muscle moment arms in cadaver specimens and has been used in the finger, shoulder, hip, knee, ankle, and elbow joints. [12, 32, 65, 146, 155, 194, 201, 257]

4.2.1 Description of tendon-displacement method

The tendon-displacement method is derived from the principle of virtual work [230] and basic geometry. In this method, muscle moment arms are calculated as the derivative of tendon displacement with respect to joint angle, as used to develop (Equation 1 in Chapter 3.0 .) This technique is widely accepted and the errors associated with the method are generally considered insignificant. This method is also the one in which the variation of moment arm with joint angle

is the most apparent since it is straightforward and simple to repeat tests at multiple joint angles. The tendon-displacement method is particularly advantageous experimentally because it requires no *a priori* assumptions about muscle insertion locations and is not prone to imprecision associated with the geometric approach. Implementation is simple in comparison to the geometric methods, which require the processing of image data from either MRI or CT scans.

4.2.2 Application to joint simulator validation

To validate one aspect of the AGH elbow simulator's design, moment arms were measured in three cadaver elbows. Information about the specimens is shown below in Table 3. The muscle moment arms of these specimens were measured in both f/e and p/s.

Table 3: Cadaver specimen details.

Specimen	Side	Sex	Age	Cause of Death
1	Left	F	60	not listed
2	Right	F	75	lymphoma
3	Right	F	70	dementia

The specimens were carefully dissected and the tendonous insertions of the biceps brachii, triceps brachii, pronator teres, and brachialis were exposed. The muscle bellies were removed. Krakow whip-stitches with suture (minimum size 0) ensured secure attachment to the tendons. A potentiometer (P1401a, Novotechnik, Southborough, MA) was used to measure p/s angle. These precision potentiometers offer linearity of 0.25% and resolution of at least

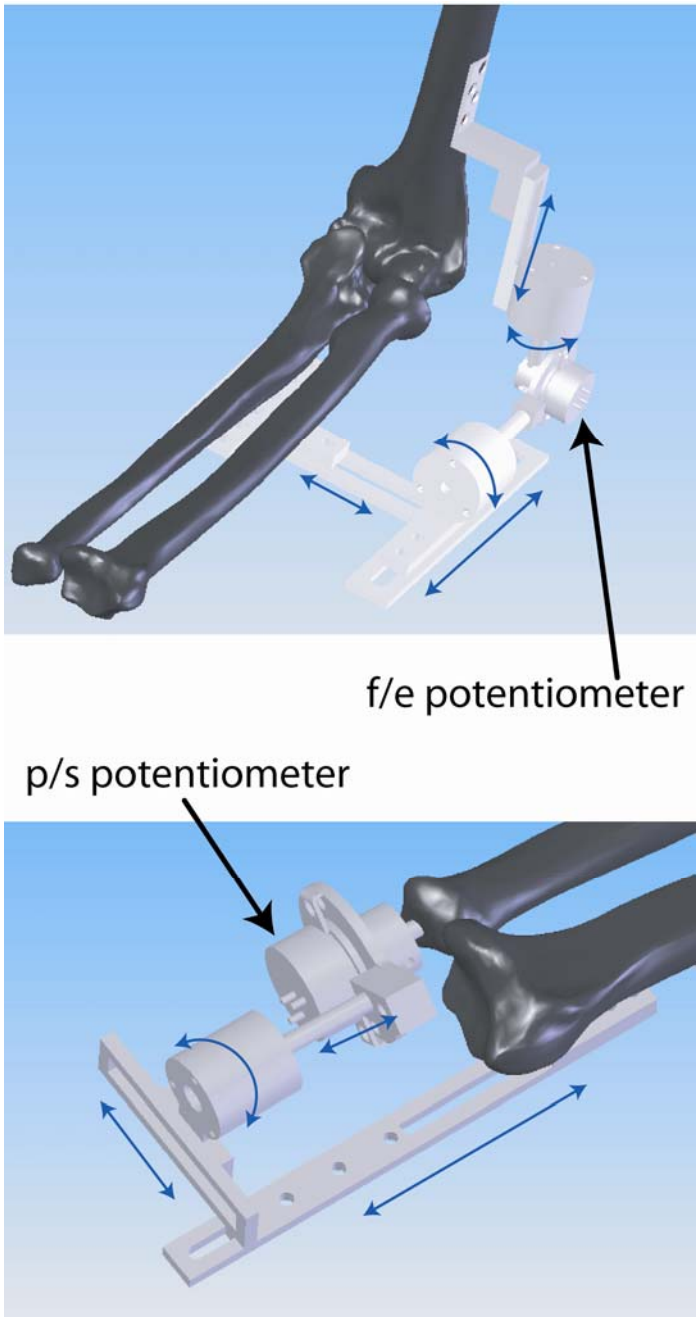


Figure 8: Illustrations of flexion/extension potentiometer mount (top) and pronation/supination potentiometer mount (bottom).

0.015°. Custom fixtures (see Figure 8, bottom) held the potentiometer in place. This fixture was installed with the aid of an Axis Finder (Figure 9) modeled after Hollister *et al.* [114] to ensure alignment with the p/s axis. The Axis Finder is simple to use. Briefly, it is rigidly attached via screws to the stationary bone (for p/s, the ulna). The adjustment rods are maneuvered so that the pointed rod can be inserted into the moving bone (for p/s, the radius). Then the desired movement is performed. Once the pointed rod is aligned with the joint axis of rotation, it will not move during the joint rotation. Adjustments to the rods are made until this criterion is reached. The pointed rod is then pushed into the bone, and the rest of the fixture removed. The potentiometer mounting fixtures described below permit attachment using the pointed rod from the Axis Finder as a guide to ensure proper alignment. To measure f/e angle, a potentiometer (P1401a, Novotechnik, Southborough, MA) was used for the first two specimens, and a newly-available inclinometer (X3Q, US Digital, Vancouver, WA) was used for the third. This recently-released inclinometer is capable of measuring tilt about two axes and is easy to attach to the ulna. Its maximum error is ±0.20° and built-in damping ensures a smooth output. The f/e potentiometer was mounted using custom fixtures (Figure 8, top) designed to permit alignment with the axis of rotation. The inclinometer was mounted via a plate rigidly attached to the ulna via screws. The inclinometer was implemented as soon as the product was available and offers considerable time savings in attachment to the specimen over the custom f/e potentiometer fixture.

Once placed in the simulator, high-strength line (80lb Superbraid, Stren, Spirit Lake, IA) linked the tendons to the motors. The custom pulleys were adjusted by appearance to replicate the muscle origins. After calibrating the position-measuring devices (potentiometer(s) and inclinometer as available), the cylinder positions were adjusted to maintain a minimum of 13.34

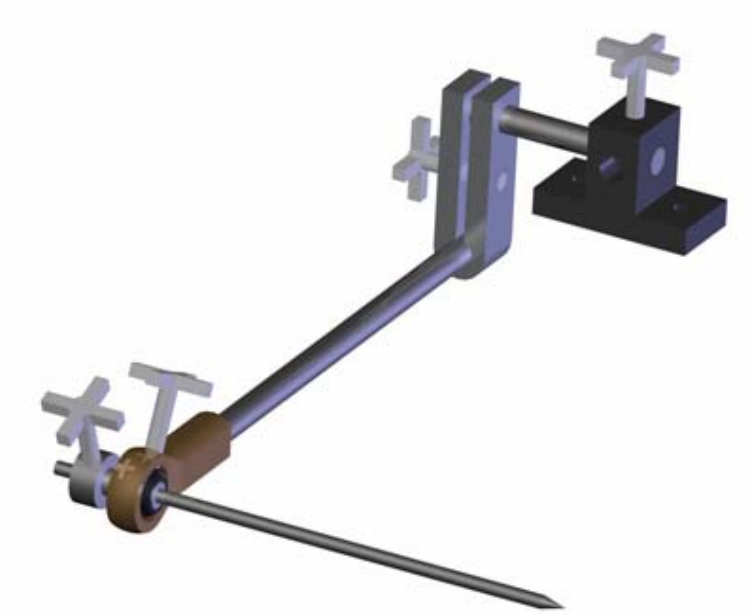


Figure 9: Axis finder.

N load on each of the four muscles with the elbow in an extreme (fully flexed, or fully supinated) position. Equal loading on all muscles in these positions was not possible without an externally-applied moment to counteract the moment produced by the loading. Following the initial loading of each muscle, force control on each cylinder was used to maintain constant force. Discrete proportional control was used. Manual actuation was used to move the elbow through the desired motion (f/e or p/s). That is, a person slowly moved the elbow through the prescribed motion while the servoelectric cylinders moved in reaction in order to maintain a constant force on each muscle.

Cylinder position (equivalent to tendon position), forces from the load cells, p/s angle, and f/e angle were measured and recorded. The speed of the motion was about 0.1 Hz (10 seconds per cycle) for f/e and 0.2 Hz (5 seconds per cycle) for p/s. The sampling rate was determined by the computational requirements of the control algorithms, which were permitted

to run as fast as possible. With these programs running at full speed, the sampling rate was set to 40 Hz. The controller program was run on the ACR8020 card via the ACR-View software. Data were retrieved and stored via a custom interface created in MicroSoft Visual Basic (MicroSoft, Redmond, WA). The ACR-View program used to control the cylinders is given in Appendix H.

Measurements of both f/e and p/s moment arms were each made while the other degree of freedom was varied. In total, six cases were observed. Flexion-extension moment arms were measured with the forearm: 1) fully supinated, 2) fully pronated, and 3) in the anatomic neutral position. Pronation-supination moment arms were measured with the elbow at 30°, 60°, and 90° of flexion. Five complete cycles of motion were measured for each of the six cases. For the f/e motion at neutral p/s, a custom clamp ensured that the p/s angle did not vary. For the p/s motions at varying f/e angles, the distal end of the forearm rested on a support to ensure minimal f/e motion.

Custom MATLAB (The MathWorks, Natick, MA) code, included Appendix K, was used to complete the calculations. Briefly, the data were imported and the truncated to exclude the portions that contained no motions (i.e.: the ends of the trials.) Similar to the methods used by Bremer *et al.* and Gardinier and Gonzalez, [33, 92] sixth-order curves were fit to both the tendon excursion data and the joint angle data. These curves were then analytically differentiated with respect to sample number to form the rate of change of length of the tendons and the rate of change of joint angle. These equations were numerically evaluated and their quotient is equivalent to $dr/d\theta$ from Equation 1. Since the movement speed may have varied within and across trials, average moment arms for each case were computed every 5° of dominant motion. That is, for f/e moment arms measured at a neutral p/s position, all trials were averaged at every five degrees of elbow flexion. The standard deviations across trials were also computed to

assess intra-case variability. These results are shown in Chapter 5.0 . Peak moment arms were computed for all cases based on the average values. To summarize these average results, 6^{th}-order polynomial curves were fit to the averages for each specimen for each case, and these coefficients are reported in Chapter 5.0 as well.

It was anticipated that the measured moment arms would exhibit magnitudes and trends similar to values previously-reported. [6, 8, 10, 49, 75, 81, 92, 100, 153, 192-194, 204] Based on the anatomical fact that the biceps and pronator teres muscles attach to and wrap around the radius during p/s motion thereby changing the effective insertion points, it was predicted that both f/e and p/s moment arms would vary with forearm position angle. It was also anticipated that, during p/s motion, the moment arms of the triceps and brachialis would be relatively constant and near zero because these muscles attach to the ulna, which remains stationary during p/s motion.

4.2.3 Limitations of tendon-displacement method

One limitation of the tendon displacement method for measuring moment arms is that it cannot be used in live human subjects. While it is a significant limitation, it is not of particular concern for this work, which seeks to validate the AGH Elbow Simulator's reproduction of realistic moment arms. Additionally, since the tendon-displacement method requires differentiation, it is sensitive to the calculation techniques. These calculations can be performed either numerically or analytically. As described above, a completely analytic approach can present computational difficulties, and a numeric approach can introduce noise as a result of numeric differentiation. The approach taken in this work combines both analytic and numeric techniques, which is unique in comparison to other studies. It is believed that this approach of differentiating smooth

curves will yield improved results over a purely numeric differentiation approach. The numeric evaluation of the parametric (with respect to sample number) derivatives to compute the resultant moment arms may be smoother with a purely analytic solution. However, averaging across all trials of each case, as well as the computation of results at every 5°, acts as a filter reducing the noise effects. This is in lieu of filtering the raw data, which has been required by others who do not use curve-fitting. [192] Discontinuity was avoided by judicious truncation of the raw data to include only the data points which occurred during elbow motion. That is, $d\theta$ could never become zero. Additionally, Murray [192] showed that the difference in calculated elbow f/e moment arms between a purely analytic and a purely numerical approach (with filtering) was less than 3 mm overall across the f/e range.

From an experimental standpoint, the tendon-displacement method relies on complete dissection of the muscles from the surrounding fascia to ensure translation during joint rotation. This was ensured by careful dissection, which included removing the muscle bellies so that only the insertion tendons remained. Murray commented that the brachialis and triceps present challenges due to their proximity to the joint capsule. [192] While the exposure of these tendons was challenging, the present technique removing the muscle belly helped ensure translation during joint rotation.

Another source of error with the tendon displacement method is the assumption concerning the joint axes locations. Murray performed a sensitivity analysis to evaluate the effect of errors in the f/e axis on the calculated moment arms by computing the f/e axis two different ways: by 1) assuming the axis passed through the center of the capitellum and the trochlear groove, and 2) by assuming the axis passed through both the lateral and medial epicondyles. These results indicated a maximum effect on the calculated moment arm of 3 mm

for the biceps, brachialis, and pronator teres, and 4 mm for the triceps. With the use of the custom potentiometer fixtures in the present work, alignment with both the f/e and p/s axis is ensured as the fixtures will bind if the potentiometer's axis of rotation is not aligned with that of the joint. For the one case in which the inclinometer was used, slight misalignments were possible. However, the inclinometer is robust against small (<10°) deviations in mounting position. Thus, slight misalignments would have no affect on the output.

Despite these potentials for error and uncertainty, the tendon-displacement method remains the best choice for the cadaver studies described herein, and reasonable precautions have been taken to minimize the influence of these known pitfalls.

5.0 RESULTS: MOMENT ARMS

In this chapter, the results from the moment arm experiments are presented and discussed. In short, the measured values compare well with previously-reported values. In addition, for some muscles, the moment arm is a function of both the flexion/extension (f/e) and pronation/supination (p/s) position.

5.1 FLEXION/EXTENSION MOMENT ARMS

The results from the computations illustrating the ranges of feasible f/e moment arms that the AGH Elbow Simulator can produce are shown in Figure 10. Recall that these ranges represent the feasible moment arms for all possible origins for one set of muscle insertion points. The shaded region represents these calculated values. Previously-reported moment arm values from the open literature are shown as well. The range of feasible biceps moment arms falls within the range of previously-reported values, though is slightly higher than the bulk of the previously-reported values. The peak predicted values occur near the same flexion angle as the peak values from the literature. Given that the calculations assumed a fully-supinated forearm, these results represent the feasible f/e moment arms only with the forearm in the supinated position. For the brachialis, the range of predicted feasible moment arms falls within the range of the bulk of the

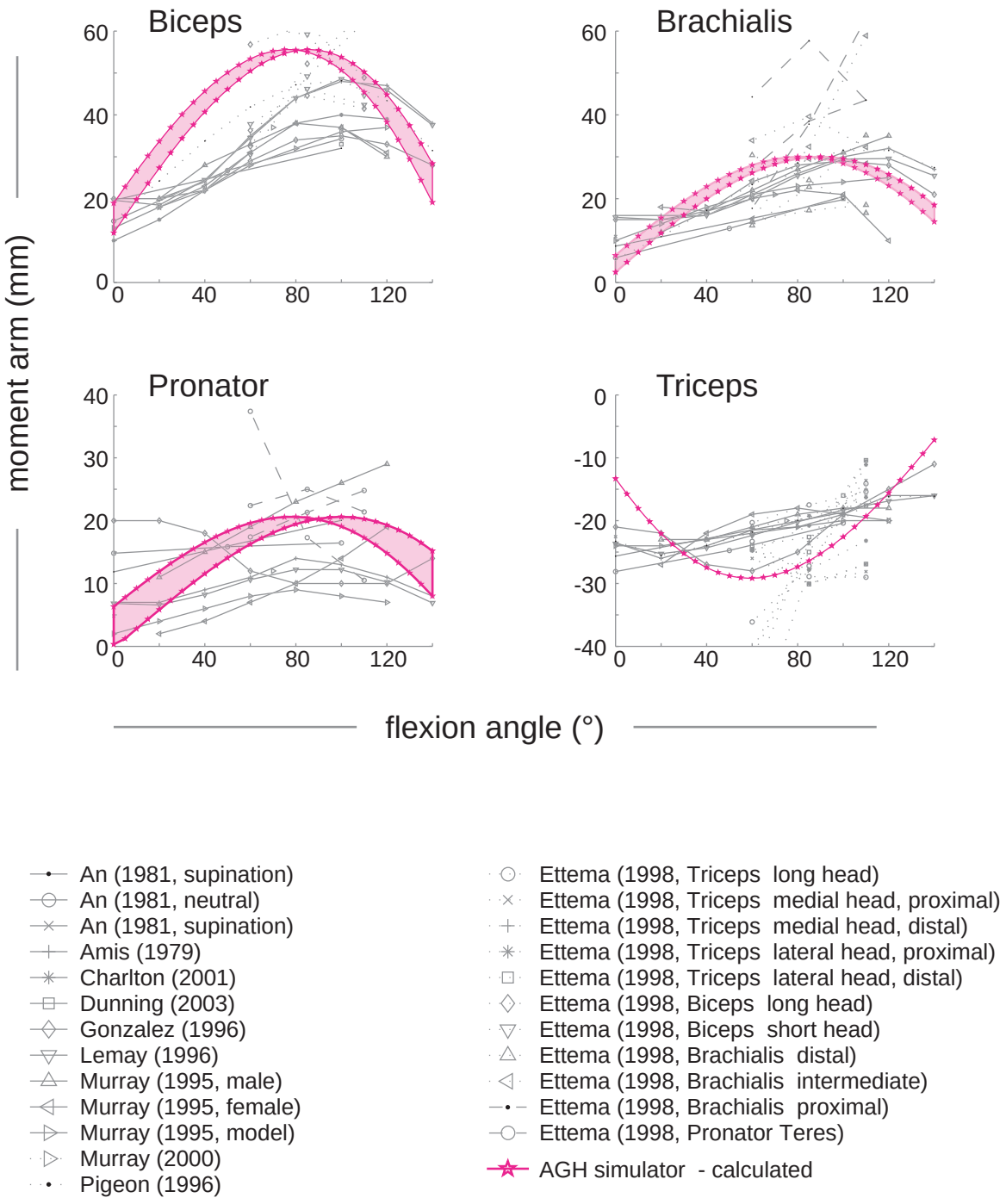


Figure 10: Predicted and previously-reported f/e moment arm values.

previously- reported values. The maximal values previously reported, from the Ettema study, [81] are markedly different from the others, and thus it is not of great concern that the simulator's predicted values do not encompass them. There is great variability in the range of previously-reported moment arm values for the pronator teres. The AGH Elbow Simulator's feasible moment arms are within this range. There is less variability, with the exception of the Ettema study, [81] for the triceps moment arms. Since no adjustability is built in to the simulator for the triceps pulleys, the triceps moment arms are fixed for each f/e position. However, these values again lie near the bulk of the previously-reported values for triceps moment arms.

The measured results for the biceps f/e moment arms are shown in Figure 11. The dark lines show the averages across all trials, and the shaded regions show the corresponding standard deviations across trials of the same type for each specimen. It is notable that, across all specimens, the f/e moment arm varies with p/s angle. That is, the peak moment arm occurs at a smaller flexion angle in supination than when in neutral or pronation. This observation is implied from the anatomical structure of the biceps. When the forearm pronates and supinates, the biceps tendon wraps around the radius bone, which alters its effective insertion point. The variability seems to decrease from Specimen 1 to Specimen 3. This could be attributed to refinement of the manual technique required to move the elbow through its range of motion, as learned by the human operator. If this variability truly occurred within each specimen, it could represent the viscoelastic effects of the tendon, though that is unlikely given that the same variability is not seen across all specimens. Table 4 shows the peak values and corresponding flexion angles for all specimens, and Table 5 shows the coefficients of the polynomial fitted to the averages for each case. The peak moment arms occur within 10° across all specimens. With regard to the polynomial coefficients, the coefficients of the higher-order terms are quite small

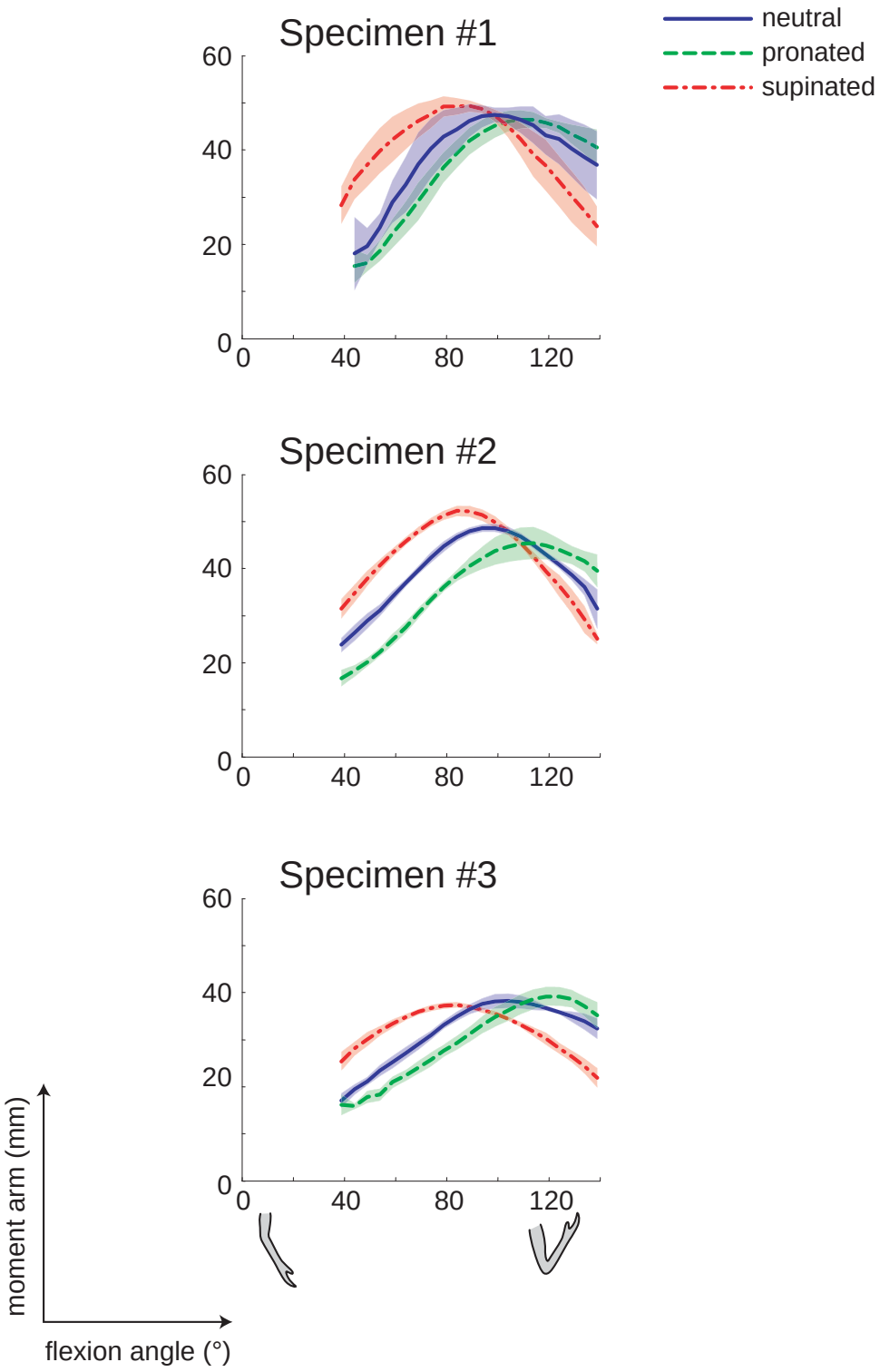


Figure 11: Biceps f/e moment arms for three specimens at three different p/s angles.

Table 4: Biceps f/e peak moment arms

	Biceps - Peak Flexion Moment Arms					
	Pronated		Neutral		Supinated	
Specimen	Angle (º)	Moment Arm (mm)	Angle (º)	Moment Arm (mm)	Angle (º)	Moment Arm (mm)
1	94	46.45	84	47.42	74	49.32
2	99	45.42	84	48.63	69	52.23
3	104	39.17	89	38.26	69	37.37

Table 5: Biceps f/e average curve-fit coefficients.

Biceps Curve Fit Coefficients for Moment Arm (mm) vs. Angle (degrees)								
Spec.	p/s pos.	x^6	x^5	x^4	x^3	x^2	x^1	x^0
1	Pro	2.8E-10	-1.7E-07	4.1E-05	-5.7E-03	4.3E-01	-1.6E+01	2.5E+02
2	Pro	-8.4E-11	4.7E-08	-9.9E-06	9.0E-04	-3.0E-02	3.3E-01	1.5E+01
3	Pro	4.3E-10	-2.4E-07	5.4E-05	-6.2E-03	3.9E-01	-1.3E+01	1.7E+02
1	Sup	-9.1E-10	5.0E-07	-1.1E-04	1.3E-02	-7.8E-01	2.6E+01	-3.2E+02
2	Sup	-7.3E-10	3.8E-07	-8.1E-05	8.6E-03	-4.9E-01	1.5E+01	-1.6E+02
3	Sup	-2.6E-10	1.3E-07	-2.8E-05	3.0E-03	-1.7E-01	5.9E+00	-6.0E+01
1	Neu	8.7E-10	-5.1E-07	1.2E-04	-1.5E-02	1.1E+00	-3.7E+01	5.1E+02
2	Neu	-8.8E-10	4.7E-07	-9.9E-05	1.1E-02	-6.0E-01	1.8E+01	-1.9E+02
3	Neu	-4.9E-10	2.8E-07	-6.2E-05	7.1E-03	-4.3E-01	1.4E+01	-1.6E+02

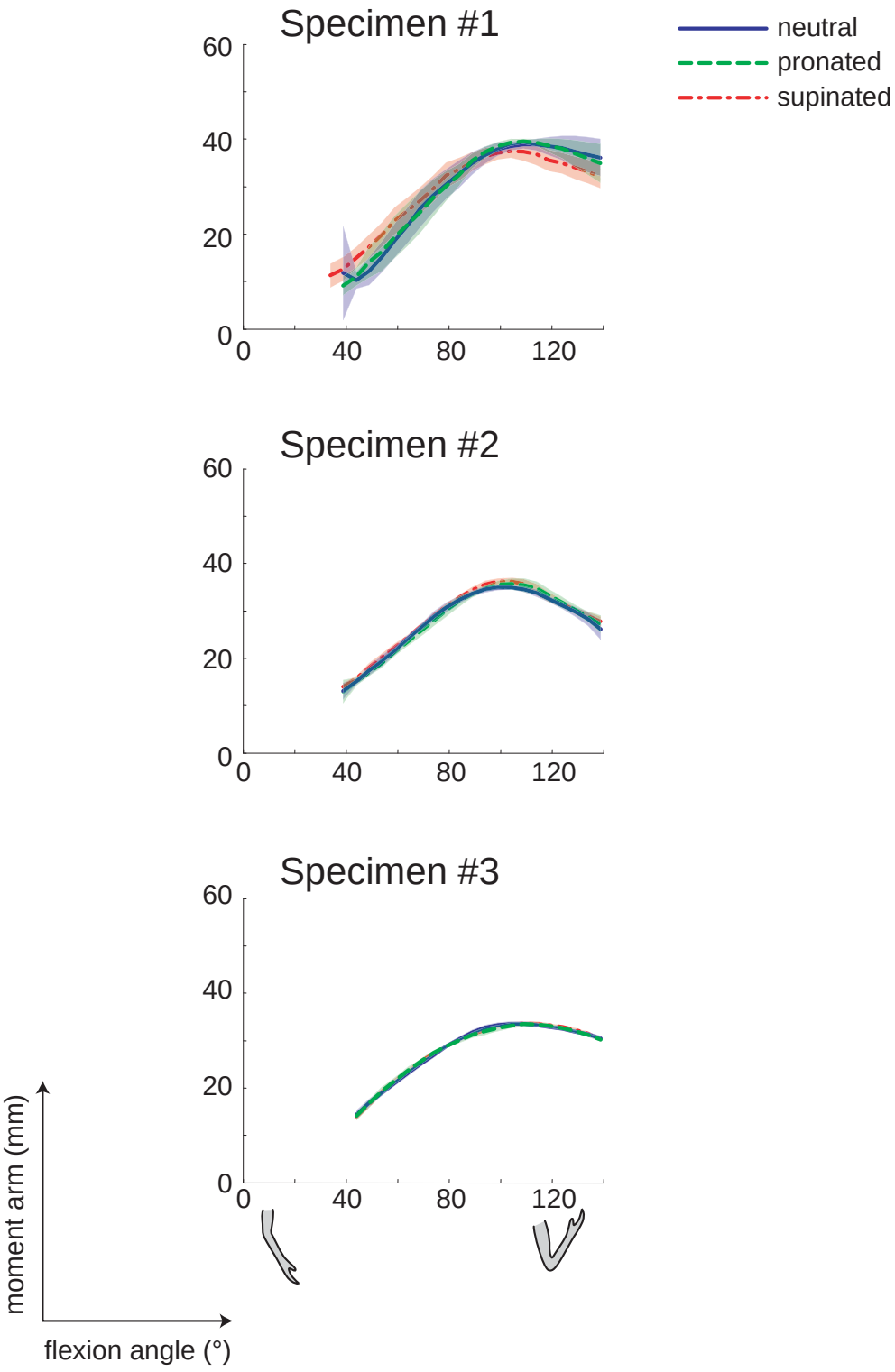


Figure 12: Brachialis f/e moment arms for three specimens at three different f/e angles.

Table 6: Brachialis f/e peak moment arms

	Brachialis - Peak Flexion Moment Arms					
	Pronated		Neutral		Supinated	
Specimen	Angle (º)	Moment Arm (mm)	Angle (º)	Moment Arm (mm)	Angle (º)	Moment Arm (mm)
1	94	39.48	94	38.90	89	37.50
2	89	35.71	89	34.96	89	36.25
3	94	33.45	89	33.55	94	33.61

Table 7: Brachialis f/e curve-fit coefficients

Brachialis Curve Fit Coefficients for Moment Arm (mm) vs. Angle (degrees)								
Spec.	p/s pos.	x^6	x^5	x^4	x^3	x^2	x^1	x^0
1	Pro	-7E-11	6E-08	-2E-05	2E-03	-1E-01	4E+00	-5E+01
2	Pro	-8E-11	6E-08	-2E-05	2E-03	-1E-01	4E+00	-5E+01
3	Pro	1E-10	-6E-08	1E-05	-1E-03	7E-02	-1E+00	9E+00
1	Sup	-6E-12	1E-08	-3E-06	2E-04	-8E-04	-2E-02	7E+00
2	Sup	-2E-10	1E-07	-3E-05	3E-03	-2E-01	7E+00	-8E+01
3	Sup	-1E-10	6E-08	-1E-05	1E-03	-1E-01	4E+00	-5E+01
1	Neu	8E-10	-5E-07	1E-04	-1E-02	9E-01	-3E+01	4E+02
2	Neu	-3E-10	2E-07	-4E-05	4E-03	-2E-01	6E+00	-7E+01
3	Neu	-5E-10	3E-07	-6E-05	7E-03	-4E-01	1E+01	-2E+02

in comparison to the coefficients of the second-order term. Notably, the sixth-order terms are less than 1% of the magnitude of the fifth order terms. It is not until the second-order coefficients that the coefficients become >1% of those of the next order. This suggests that, for the biceps, a second-order curve could be used to describe the resulting behavior, which agrees with the general trends shown in the measured results.

Plots of the measured moment arm results for the brachialis muscle are shown in Figure 12. As expected given the anatomy of the brachialis, namely that it attaches to the ulna and does not interact with the radius during p/s motion, the moment arms do not vary much with p/s position. This is confirmed in Table 6, which shows the peak moment arm values across specimens and p/s positions. For this muscle, the peaks do not vary more than 5° across p/s position. Table 7 shows the coefficients for the sixth-order polynomial fit to the average. As with the biceps, the coefficients of the higher-order terms are small in comparison to the second-order term, and thus the second-order behavior that dominates, which agrees with the overall trends in the results. As with the biceps, the measured results here compare well with previously-reported results in the open literature.

Based on the anatomy of the triceps muscle, as seen in Figure 13, the flexion moment arm does not change much with p/s position. As with the biceps, the variability, represented by the standard deviation (the shaded region of the plots) decreases from Specimen 1 to Specimen 3. The most likely reason for this is improved operator performance of the manual actuation of the forearm. The measured moment arms for Specimen 1 also increase dramatically as flexion angle decreases near 50° flexion. This behavior is likely a result of the end effects of the polynomial fits during the calculation of the moment arms, and could be reduced by more judicious selection

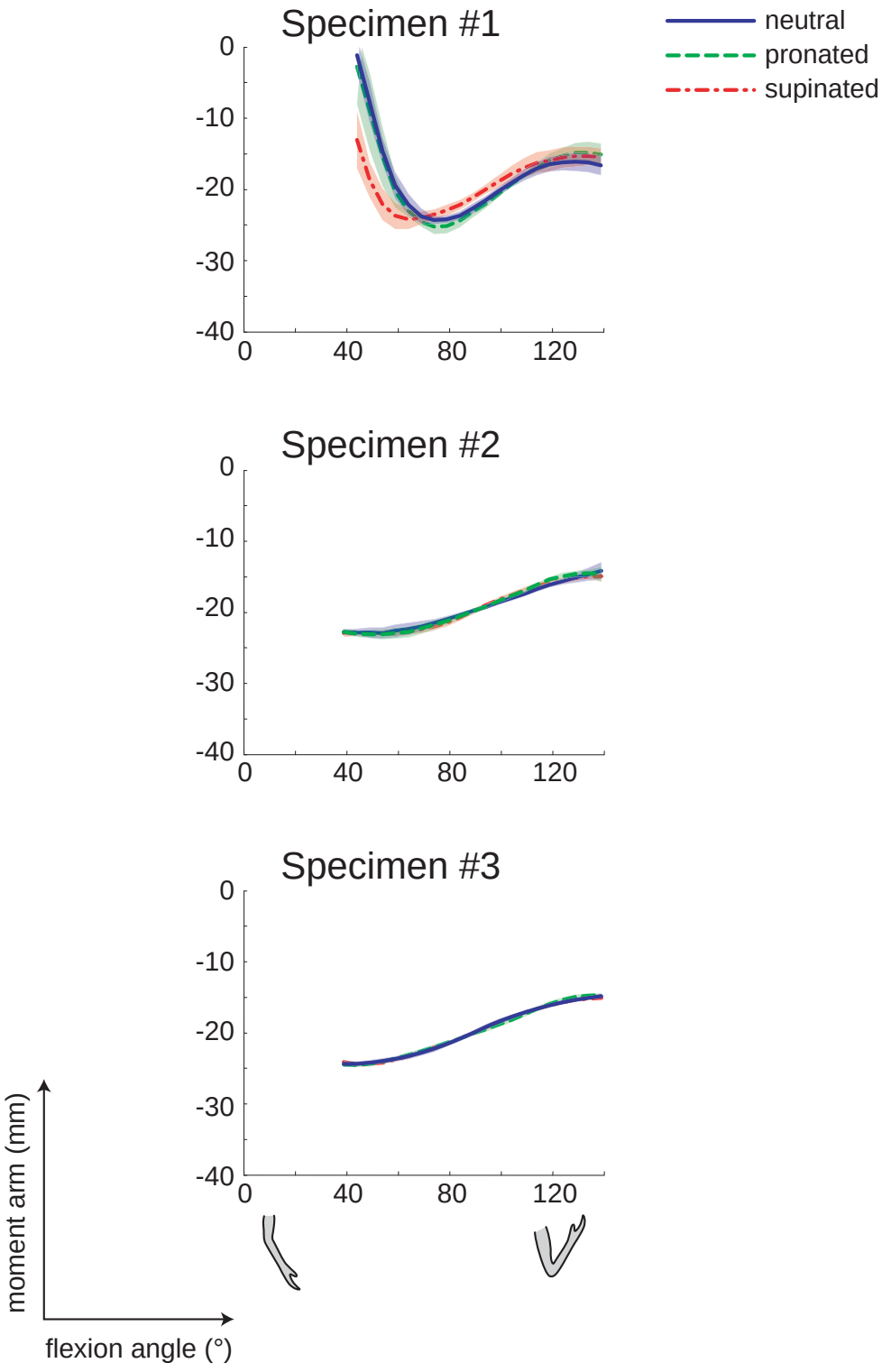


Figure 13: Triceps f/e moment arms for three specimens at three different f/e angles.

Table 8: Triceps f/e peak moment arms.

	Triceps - Peak Flexion Moment Arms					
	Pronated		Neutral		Supinated	
Specimen	Angle (º)	Moment Arm (mm)	Angle (º)	Moment Arm (mm)	Angle (º)	Moment Arm (mm)
1	59	-25.27	59	-24.34	49	-24.22
2	34	-23.08	39	-22.94	29	-22.99
3	29	-24.57	24	-24.40	29	-24.45

Table 9: Triceps f/e curve-fit coefficients

Triceps Curve Fit Coefficients for Moment Arm (mm) vs. Angle (degrees)								
Spec.	p/s pos.	x^6	x^5	x^4	x^3	x^2	x^1	x^0
1	Pro	-6.5E-10	3.5E-07	-7.4E-05	8.2E-03	-5.0E-01	1.6E+01	-2.1E+02
2	Pro	-4.2E-10	2.2E-07	-4.7E-05	5.1E-03	-3.1E-01	9.9E+00	-1.2E+02
3	Pro	3.1E-10	-1.6E-07	3.4E-05	-3.7E-03	2.1E-01	-5.9E+00	6.8E+01
1	Sup	-1.4E-10	7.7E-08	-1.7E-05	2.0E-03	-1.3E-01	5.0E+00	-7.9E+01
2	Sup	-3.1E-10	1.6E-07	-3.1E-05	3.2E-03	-1.8E-01	5.5E+00	-7.2E+01
3	Sup	-8.7E-11	4.7E-08	-1.0E-05	1.0E-03	-5.9E-02	1.8E+00	-2.0E+01
1	Neu	2.2E-10	-1.3E-07	3.1E-05	-3.8E-03	2.5E-01	-7.4E+00	8.3E+01
2	Neu	-2.2E-10	1.1E-07	-2.5E-05	2.7E-03	-1.7E-01	6.0E+00	-8.0E+01
3	Neu	-8.9E-11	3.7E-08	-5.3E-06	2.9E-04	-2.1E-03	-2.6E-01	1.3E+01

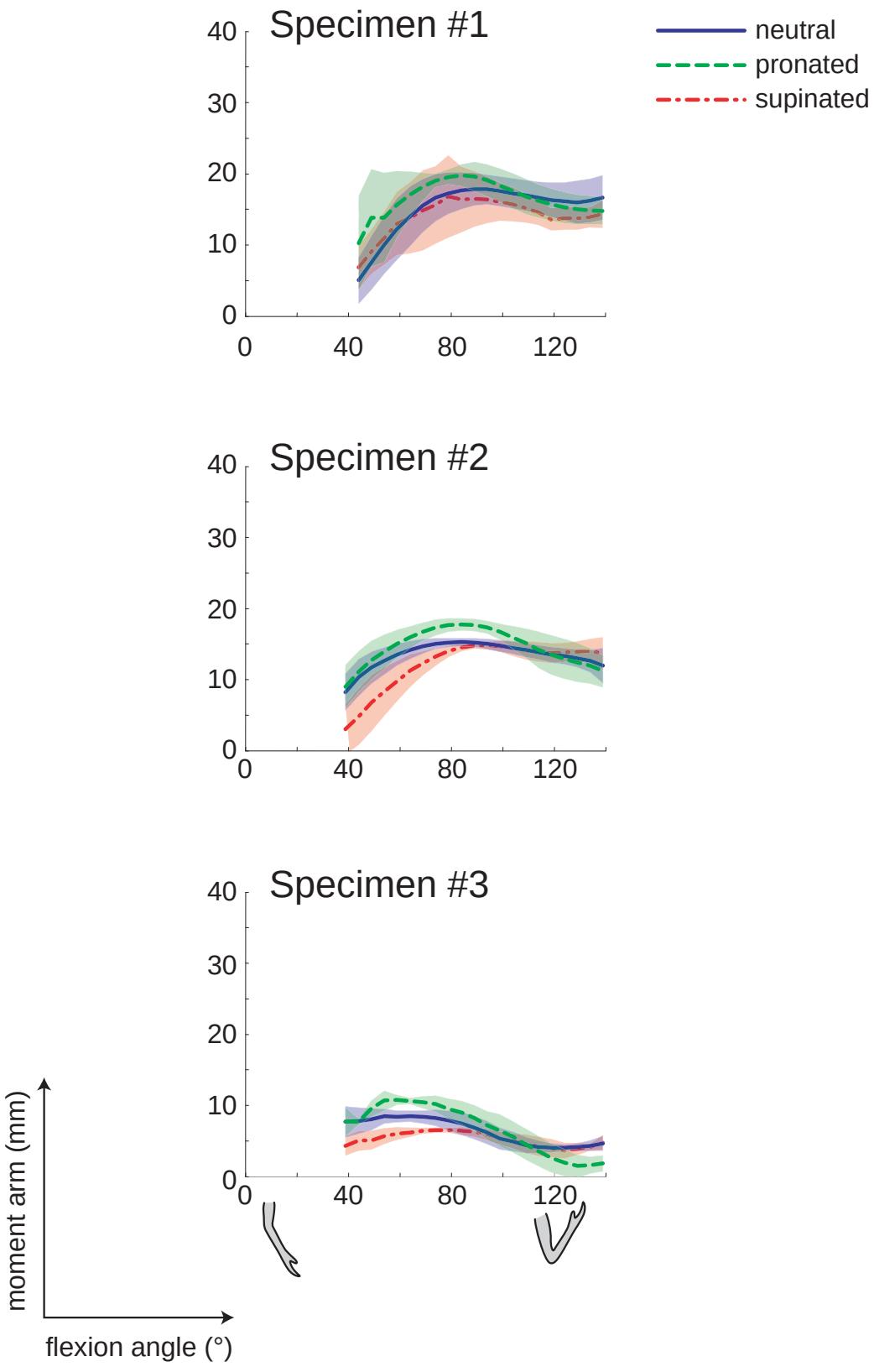


Figure 14: Pronator teres f/e moment arms for three specimens at three different f/e angles.

Table 10: Pronator teres f/e peak moment arms.

	Pronator - Peak Flexion Moment Arms					
	Pronated		Neutral		Supinated	
Specimen	Angle (º)	Moment Arm (mm)	Angle (º)	Moment Arm (mm)	Angle (º)	Moment Arm (mm)
1	69	19.79	74	17.82	64	16.81
2	69	17.76	69	15.24	79	14.81
3	44	10.74	39	8.46	59	6.52

Table 11: Pronator teres f/e curve-fit coefficients

Pronator Teres Curve Fit Coefficients for Moment Arm (mm) vs. Angle (degrees)								
Spec.	p/s pos.	x^6	x^5	x^4	x^3	x^2	x^1	x^0
1	Pro	-6.8E-10	3.9E-07	-8.7E-05	9.8E-03	-5.7E-01	1.4E+01	-1.0E+02
2	Pro	-4.8E-11	2.3E-08	-4.8E-06	4.9E-04	-2.5E-02	5.2E-01	-2.5E+01
3	Pro	4.4E-11	-3.0E-08	8.0E-06	-1.1E-03	7.7E-02	-2.8E+00	1.4E+01
1	Sup	5.9E-10	-3.5E-07	8.3E-05	-1.1E-02	7.8E-01	-3.0E+01	4.6E+02
2	Sup	-1.3E-11	1.5E-08	-5.3E-06	8.3E-04	-6.1E-02	2.1E+00	-5.1E+01
3	Sup	9.9E-11	-5.4E-08	1.2E-05	-1.4E-03	9.6E-02	-3.3E+00	2.2E+01
1	Neu	-4.6E-10	2.6E-07	-5.9E-05	6.5E-03	-3.5E-01	6.8E+00	6.6E-01
2	Neu	-9.1E-12	5.0E-09	-1.1E-06	9.8E-05	-1.7E-03	-1.6E-01	-1.8E+01
3	Neu	-1.5E-11	1.3E-08	-3.9E-06	5.4E-04	-3.6E-02	1.2E+00	-3.9E+01

of initial data points for each trial (not shown). As indicated in Table 8, the peak moment arms across specimens are very close in value, though the flexion angle at which they occur is much lower in Specimen 1 than in the others. This could be due to anatomic variability, or due to slight inaccuracies in the calibration of the f/e angle measurement device. Given that the f/e angles for the other muscles measured on this specimen do not differ dramatically from the other specimens, this phenomena is unlikely. As with the biceps and brachialis, looking at the coefficients of the curves fit to the average values in Table 9, the coefficients for the highest-order terms are quite small, suggesting that the lower-order (2^{nd} and below) terms dominate, agreeing with the plots in Figure 13.

The measured moment arms of the pronator teres display the greatest variability within cases, as shown by the shaded areas in Figure 14. As with the biceps, the pronator teres' flexion moment arm varies with forearm p/s position. It is greatest in pronation and least in supination, as illustrated in Table 10. This is intuitive given the muscle's anatomy. With an insertion on the lateral side of the radius, the pronator teres wraps around the radius in supination, changing its effective insertion point and reducing its effective ability to flex the elbow. As with the other muscles, one explanation for the decreased variability from Specimen 1 to Specimen 3 could be improvements in the manual actuation of the forearm through the range of f/e motion. In general, the variability of the moment arms of the pronator teres is larger than that of the other muscles. This could be due to inconsistent wrapping of the tendon around the radius which would have been possible with the absence of the surrounding tissue constraints. It could also be due to the swiveling action of the pulleys designed to maintain physiologically accurate moment arms, as described in Chapter 4.0 . It is possible that, across trials, the pulleys did not swivel at the same flexion angle. Since pulley swiveling is dependent on tendon excursion, a change in

the amount of tendon excursion at a given angle which would necessarily affect the computed moment arms. This could be monitored in future work with a rotary potentiometer, assuming that there is no slip between the high-strength cables and the pulley surfaces. The coefficients of the curve fit to the average values are shown in Table 11. Similar to the other muscles, the highest-order coefficients are quite small. In particular, it seems that the second- and first-order terms dominate the behavior of the measured muscle moment arms.

5.2 PRONATION/SUPINATION MOMENT ARMS

The range of previously-reported p/s moment arm values is shown in Figure 15. Pronation/supination moment arms have been reported with less frequency than their f/e counterparts. There is a large variability in the reported values across all muscles. This could be due to the anatomic variability of the cadaver specimens or poor measurement methods. For most cases, the p/s moment arms for the triceps and brachialis are negligible, which is expected. The anatomy dictates that these muscles should not move much during p/s motion, which implies that their excursions, and thus their moment arms, should be zero. The pronator teres and biceps moment arms are nearly equal and opposite, which supports the idea that p/s motion can be achieved using just these two muscles.

Similar to the f/e moment arms, the biceps p/s moment arms show a dependence on f/e position, as seen in Figure 16. This is again anticipated because of the way the biceps tendon wraps around the radius during p/s motion, which changes its effective insertion point. It is not surprising that the moment arm increases with flexion angle. As with the f/e moment arms, the variability (standard deviation) decreases from the first to the third specimen. This is likely due

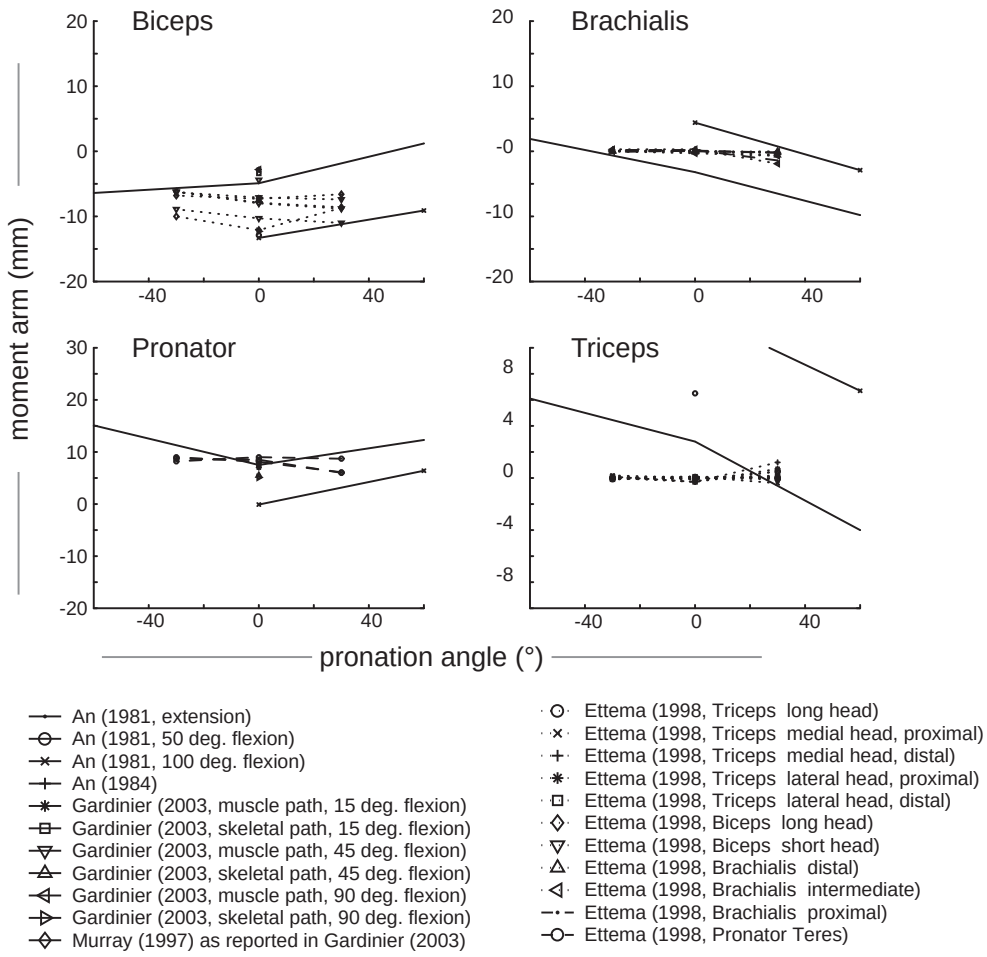


Figure 15: Previously-reported p/s moment arm values.

to improved operator technique during the manual actuation of the p/s motion. There is great variability in the peak moment arms for the biceps muscle, as shown in Table 12. This could be explained by inconsistencies in wrapping during p/s motion. With the surrounding muscles removed, the tendon would have had the opportunity to wrap around the radius differently each time. Looking at the coefficients of the average values in Table 13, the first-order term seems to be the dominant term. The coefficients of the terms higher than third-order are all quite small. However, given the shape of the plots, a second-order or higher fit would be more appropriate to capture the peaks of the moment arm values.

As seen in Figure 17, the brachialis p/s moment arm is essentially zero for all f/e angles. This is anticipated given its attachment to the ulna, which does not move during p/s at a fixed f/e angle. This compares well with the previously-reported results shown above. The variability within cases (between trials) is quite small as well. The peak values are included in Table 14 for completeness, though it is not anticipated that these peaks would be consistent across specimens or across f/e positions. These peaks are likely caused by intra-trial variability, and are not necessarily meant to imply that the brachialis moment arm has an extremum. The polynomial coefficients, in Table 15, are all quite small, with the constant coefficient being of the largest order. This suggests that its behavior dominates.

Similar to the brachialis, it is anticipated that the triceps will have no p/s moment arm due to its attachment to the ulna, which remains stationary during p/s motion. This is seen in Figure 18. The inter-trial variability is more apparent in these figures, and suggests that some slight adjustments to the triceps position were needed to maintain constant force. Small adjustments are not surprising, since the biceps and pronator muscles, which actuate p/s motion, can also

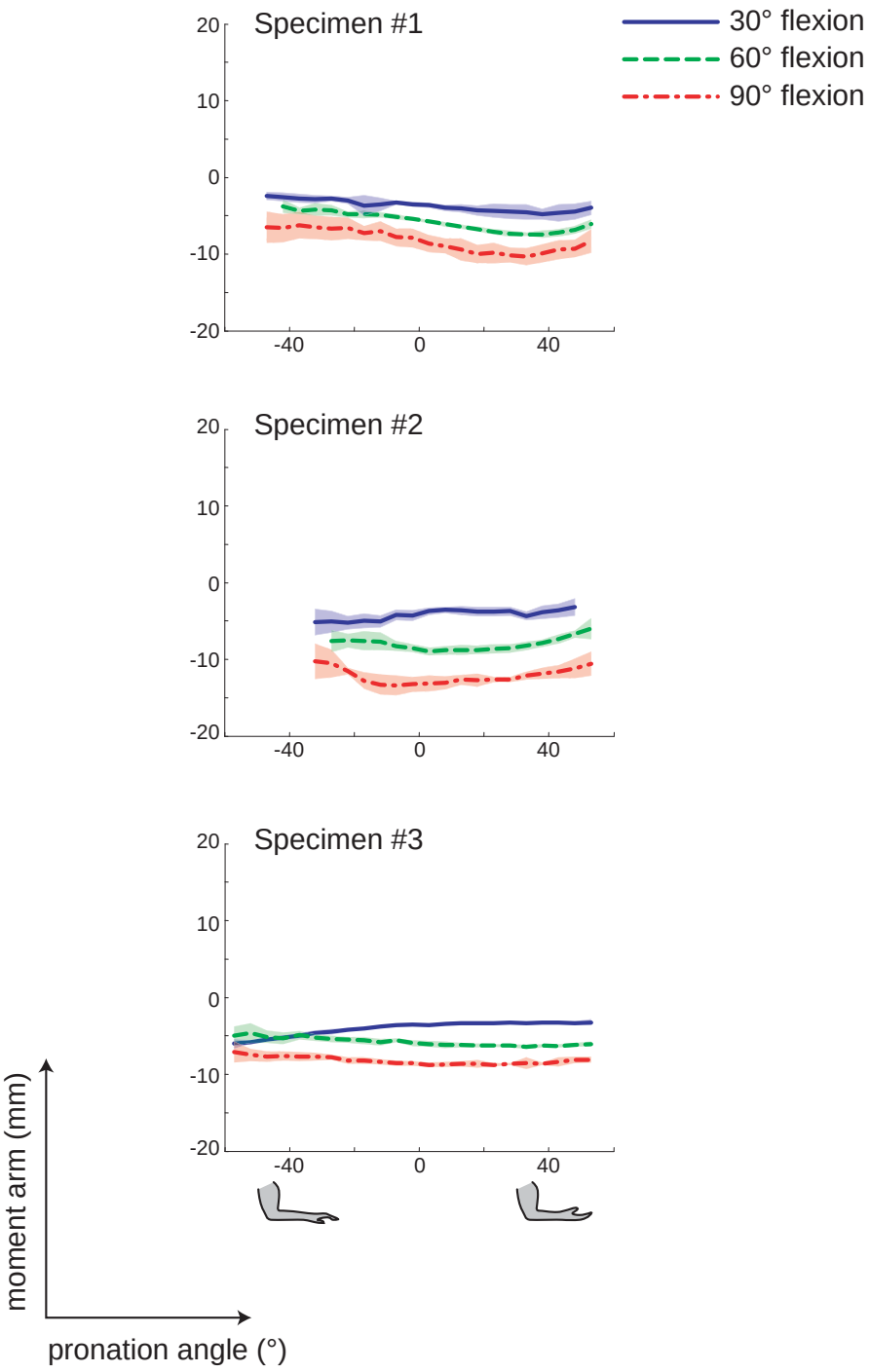


Figure 16: Biceps p/s moment arms at three different f/e angles.

Table 12: Biceps p/s peak moment arms.

	Biceps - Peak Pronation Moment Arms					
	30° flexion		60° flexion		90° flexion	
Specimen	Angle (º)	Moment Arm (mm)	Angle (º)	Moment Arm (mm)	Angle (º)	Moment Arm (mm)
1	14	-4.75	14	-10.18	14	-7.49
2	-51	-5.44	-26	-13.49	-6	-8.91
3	-56	-5.00	-1	-8.83	9	-6.53

Table 13: Biceps p/s curve-fit coefficients

Biceps Curve Fit Coefficients for Moment Arm (mm) vs. Angle (degrees)								
Spec.	f/e pos.	x^6	x^5	x^4	x^3	x^2	x^1	x^0
1	30°	-1.2E-10	4.2E-09	5.3E-07	-6.2E-06	-4.5E-04	-2.6E-02	-3.7E+00
2	30°	2.4E-10	-1.9E-08	7.4E-07	-4.4E-06	-1.5E-03	4.2E-02	-4.0E+00
3	30°	2.1E-11	-8.2E-10	-2.6E-08	4.9E-06	-4.4E-04	1.7E-02	-3.5E+00
1	60°	-3.1E-11	-4.2E-09	7.6E-07	1.5E-05	-1.1E-03	-6.3E-02	-5.5E+00
2	60°	-5.1E-10	5.0E-08	-9.7E-07	-2.0E-05	1.7E-03	-2.6E-02	-8.7E+00
3	60°	-1.9E-11	-6.2E-10	6.0E-08	7.7E-06	-4.4E-06	-2.8E-02	-5.9E+00
1	90°	-1.1E-11	-5.6E-09	2.5E-07	4.1E-05	-2.1E-04	-9.2E-02	-8.3E+00
2	90°	-5.3E-10	4.4E-08	2.9E-07	-9.5E-05	2.4E-03	1.6E-02	-1.3E+01
3	90°	-9.1E-12	-3.1E-10	6.5E-09	2.9E-06	3.6E-04	-1.3E-02	-8.6E+00

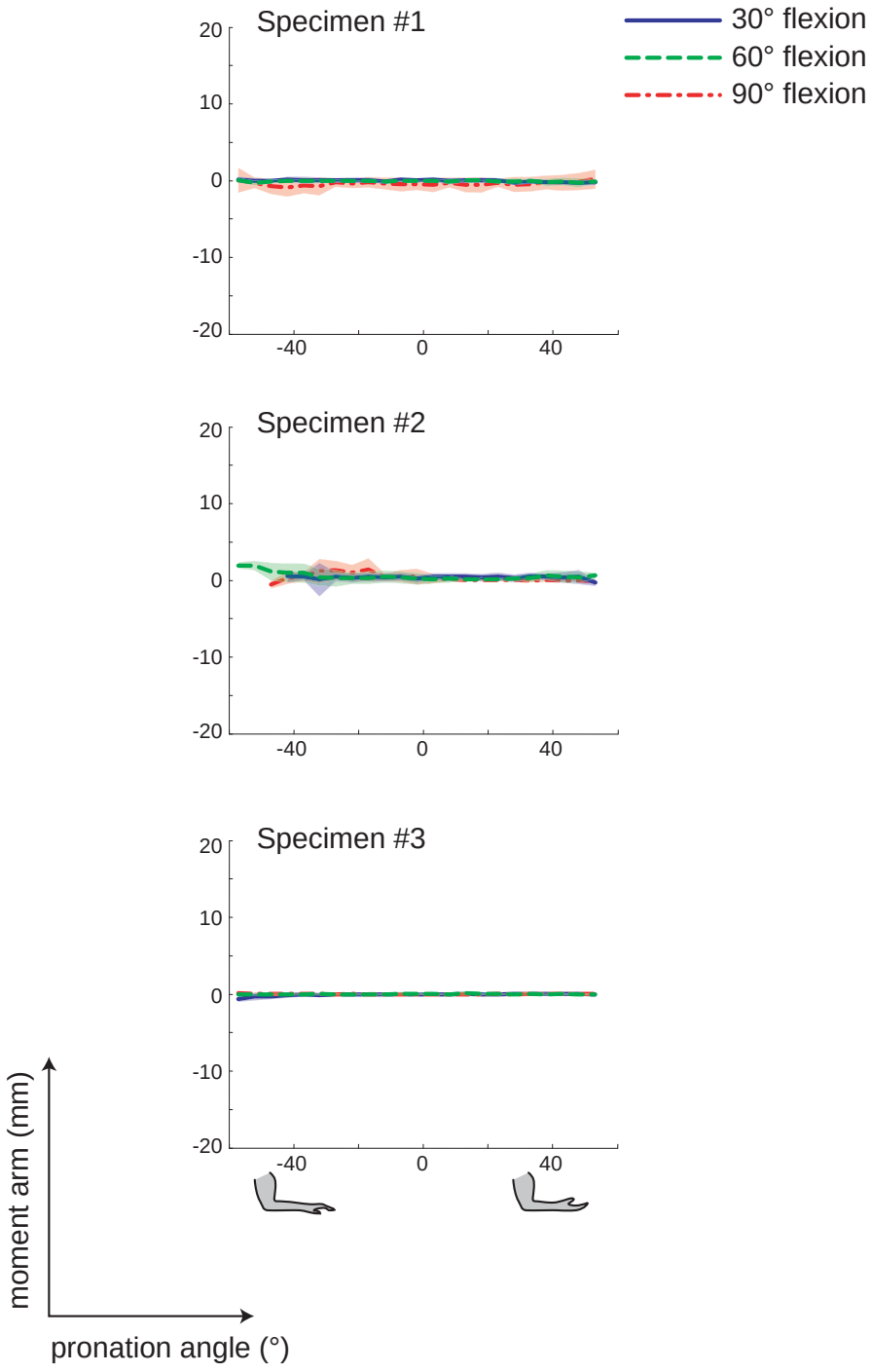


Figure 17: Brachialis p/s moment arms at three different f/e angles.

Table 14: Brachialis p/s peak moment arms.

	Brachialis - Peak Pronation Moment Arms					
	30° flexion		60° flexion		90° flexion	
Specimen	Angle (º)	Moment Arm (mm)	Angle (º)	Moment Arm (mm)	Angle (º)	Moment Arm (mm)
1	14	0.36	-21	0.86	14	0.13
2	-21	0.61	-51	1.40	-56	0.54
3	-56	0.13	-31	0.04	-11	0.07

Table 15: Brachialis p/s curve-fit coefficients

Brachialis Curve Fit Coefficients for Moment Arm (mm) vs. Angle (degrees)								
Spec.	f/e pos.	x^6	x^5	x^4	x^3	x^2	x^1	x^0
1	30°	-4.7E-11	3.6E-10	2.6E-07	-9.0E-07	-4.0E-04	-4.0E-03	1.4E-02
2	30°	-1.9E-10	6.6E-09	5.2E-07	-1.5E-05	-4.3E-04	7.7E-03	5.2E-01
3	30°	-1.6E-11	2.1E-10	7.0E-08	1.9E-07	-1.1E-04	7.3E-04	-1.1E-02
1	60°	9.7E-12	2.4E-10	3.6E-08	-3.3E-06	-1.2E-04	2.2E-03	-5.3E-02
2	60°	-1.6E-10	-7.8E-09	6.6E-07	2.0E-05	-3.7E-04	-1.4E-02	2.0E-01
3	60°	4.6E-12	1.2E-10	-2.7E-08	-1.0E-06	2.8E-05	1.7E-03	3.7E-03
1	90°	-4.0E-11	9.3E-10	2.4E-07	-2.7E-06	-2.7E-04	3.5E-03	-5.2E-01
2	90°	2.8E-10	-9.4E-09	-1.4E-06	4.5E-05	1.3E-03	-4.7E-02	2.8E-01
3	90°	9.5E-14	-9.6E-11	-9.1E-09	-4.5E-08	5.1E-05	4.3E-04	-2.1E-02

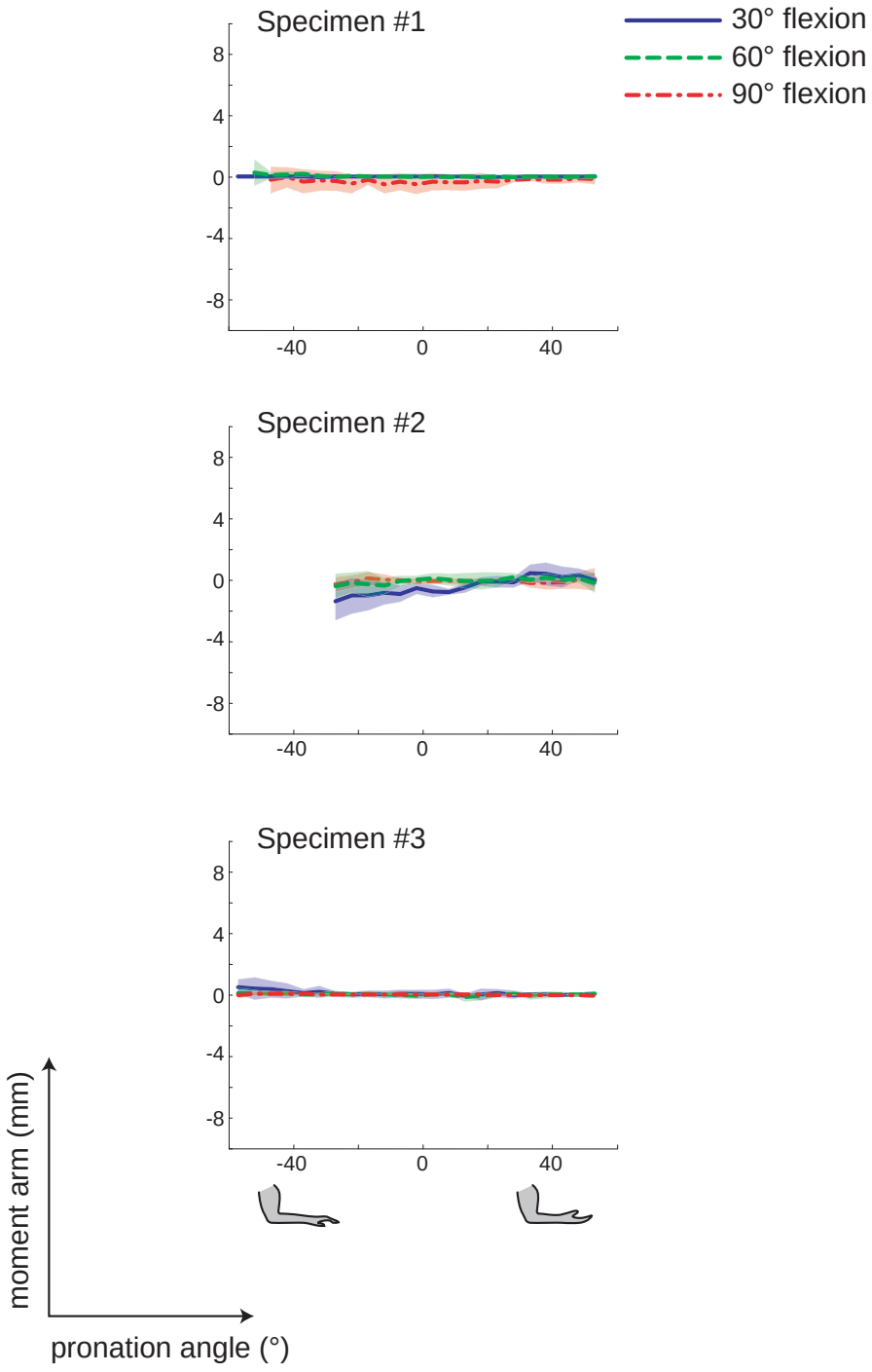


Figure 18: Triceps p/s moment arms at three different f/e angles.

Table 16: Triceps p/s peak moment arms

	Triceps - Peak Pronation Moment Arms					
	30° flexion		60° flexion		90° flexion	
Specimen	Angle (º)	Moment Arm (mm)	Angle (º)	Moment Arm (mm)	Angle (º)	Moment Arm (mm)
1	-1	0.02	-36	0.45	-51	0.12
2	-46	1.15	-46	0.36	-46	0.30
3	-56	0.22	-56	0.05	-51	0.12

Table 17: Triceps p/s curve-fit coefficients

Triceps Curve Fit Coefficients for Moment Arm (mm) vs. Angle (degrees)								
Spec.	f/e pos.	x^6	x^5	x^4	x^3	x^2	x^1	x^0
1	30°	-6.0E-13	-4.9E-11	1.1E-08	2.3E-07	-1.6E-05	-2.4E-04	4.1E-03
2	30°	1.7E-10	-1.2E-08	-5.6E-07	3.2E-05	4.6E-04	5.8E-03	-7.0E-01
3	30°	-1.1E-11	1.7E-09	-4.9E-09	-5.8E-06	1.1E-04	1.5E-03	3.8E-02
1	60°	3.9E-12	-1.0E-10	-1.1E-08	-2.5E-07	7.0E-05	-9.3E-04	-1.3E-02
2	60°	2.9E-10	-2.8E-08	2.7E-07	3.3E-05	-5.0E-04	-2.1E-03	1.0E-02
3	60°	-4.6E-13	-2.6E-10	6.4E-09	2.1E-06	1.1E-05	-3.9E-03	-5.0E-03
1	90°	-6.5E-11	4.3E-09	1.3E-07	-1.1E-05	5.4E-05	6.5E-03	-3.5E-01
2	90°	5.4E-11	6.6E-10	-3.8E-07	1.4E-05	-4.7E-05	-5.7E-03	-4.1E-02
3	90°	1.9E-12	2.9E-10	-1.0E-08	-9.1E-07	1.2E-05	-3.0E-04	4.8E-03

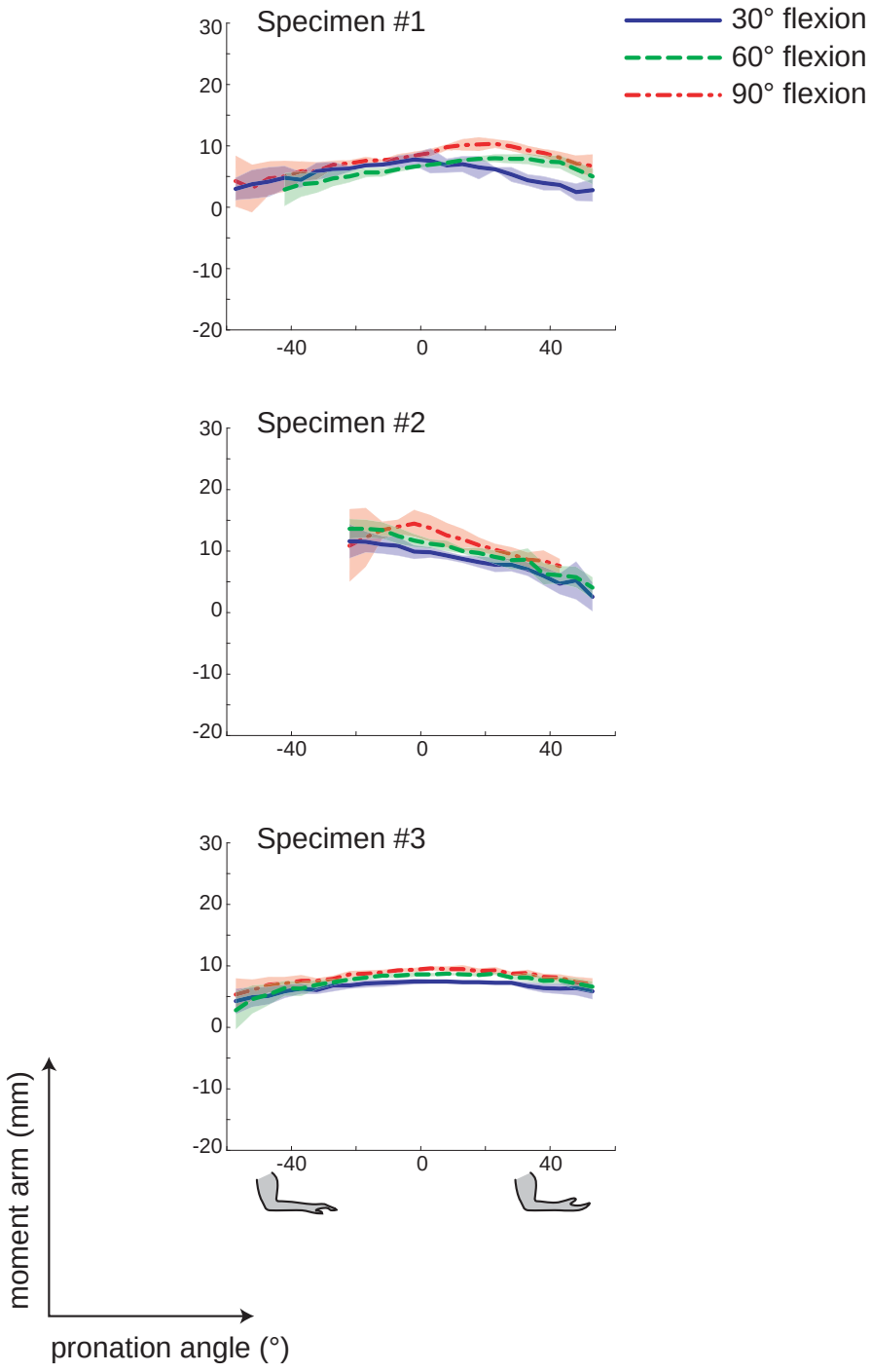


Figure 19: Pronator teres p/s moment arms at three different f/e angles.

Table 18: Pronator teres p/s peak moment arms.

	Pronator - Peak Pronation Moment Arms					
	30° flexion		60° flexion		90° flexion	
Specimen	Angle (º)	Moment Arm (mm)	Angle (º)	Moment Arm (mm)	Angle (º)	Moment Arm (mm)
1	-16	7.90	-1	10.56	9	7.92
2	-41	11.67	-21	14.29	-41	13.66
3	-26	7.46	-16	9.56	-11	8.62

Table 19: Pronator teres p/s curve-fit coefficients

Pronator Teres Curve Fit Coefficients for Moment Arm (mm) vs. Angle (degrees)								
Spec.	f/e pos.	x^6	x^5	x^4	x^3	x^2	x^1	x^0
1	30°	2.7E-11	5.4E-09	1.6E-07	-2.1E-05	-2.1E-03	1.4E-03	7.2E+00
2	30°	3.1E-10	-2.0E-09	-1.7E-06	3.6E-05	1.7E-04	-1.0E-01	1.0E+01
3	30°	2.7E-13	-1.6E-10	-4.7E-08	1.2E-06	-6.9E-04	6.8E-03	7.4E+00
1	60°	2.1E-10	-6.5E-09	-1.0E-06	8.5E-06	3.4E-04	5.1E-02	6.7E+00
2	60°	4.2E-10	-1.2E-08	-2.1E-06	7.2E-05	4.6E-04	-1.3E-01	1.2E+01
3	60°	-8.8E-12	8.0E-10	8.1E-08	-5.4E-07	-1.1E-03	1.9E-02	8.5E+00
1	90°	1.3E-10	7.3E-09	-7.3E-07	-4.6E-05	-2.8E-04	9.7E-02	8.7E+00
2	90°	4.1E-09	-4.1E-07	9.0E-06	2.2E-04	-8.7E-03	-9.4E-02	1.4E+01
3	90°	2.0E-11	1.7E-09	-7.2E-08	-6.8E-06	-1.0E-03	1.8E-02	9.4E+00

produce f/e moments. The triceps may have been counteracting such moments. The peak moment arms and coefficients of the polynomial fitted to the average values for each case are shown in Table 16 and Table 17 for completeness. As with the brachialis, it is not surprising that the peak values occur at different p/s angles, nor is it surprising that the coefficients of the polynomial are quite small in general. The results from Specimen #2 do not include the full range of p/s motion that is seen in the other specimens. This is due to the truncation of the results to eliminate the end effects of the differentiation. Specimen #2 had a smaller range of motion in comparison to the others.

Figure 19 shows the results for the pronator teres. There is variation in moment arm magnitude with flexion angle, however, the pattern is less stark. As with the f/e moment arms for the pronator teres, this could be due in part to inconsistent pulley swiveling at different times across trials. The custom pulleys were designed to swivel to maintain physiologically-accurate moment arms, but it is possible that friction in the pulley assembly, or inconsistencies in the manual actuation, caused the pulleys to swivel at different p/s angles across trials. This same phenomena could account for the unexpected variability in location of the peak moment arm, shown in Table 18. As with all other muscles, the highest-order terms have the smallest coefficients in the polynomial fit to the average values, as seen in Table 19. It is also possible that this variability is due to differences in the way the tendon wrapped around the radius between trials. Since much of the surrounding tissue was removed, the tendon was less constrained and may have wrapped around the radius differently than it would have *in vivo*. Similar to the results from the brachialis, the plot from Specimen #2 is more truncated due to its more limited range of motion. Another explanation for the decrease in variability between

Specimens #1 and #3 is that, for the third specimen, more tissue was left intact on the forearm than with previous specimens, due to a different preparation technique.

5.3 DISCUSSION

To summarize, the measured f/e moment arms vary with p/s position for the biceps and pronator muscles, and vice-versa, as expected from the way these muscles wrap around the radius during p/s motion. Similarly, these same muscles' p/s moment arms show dependence on f/e angle, as expected for the same reason. Also due to their anatomy, the p/s moment arms of the triceps and brachialis are essentially zero. In general, the variability within cases decreases as the number of specimens tested increases, which suggests that there may be a learning effect of the manual operator actuating the motion. The differences between right and left specimens should be mitigated by the adjustability of the simulator. There is no reason to suspect that this may have influenced the variability in the results.

The ranges of feasible moment arm values for the AGH Elbow Simulator are reasonable in comparison to the data reported in the literature in the sense that the predicted values lie within the broad ranges of previously-reported values for each muscle. It is important to note that the calculations of the simulator's capability were based only on one set of reported muscle origins and insertions that were measured from one specimen. Since muscle origins and insertions vary with size, using a set of different values could yield changes in the ranges of feasible moment arms. While calculating feasible moment arms with different sets of muscle origin and insertion points would change the results, it is expected that all sets of adult muscle origin and insertion data would yield results comparable to those previously-reported.

The measured moment arm values compare well with both the previously-reported values from the open literature and with the ranges of feasible f/e moment arm values that for which the AGH Elbow Simulator was designed. Feasible p/s moment arm values for the AGH Elbow Simulator were not computed due to the difficulties in accounting for the wrapping of the biceps and pronator teres tendons around the radius during p/s motion. Nevertheless, the measured p/s moment arms compare well with the limited number of previously-reported values available.

A physical aspect of the AGH Elbow Simulator has been validated with a new variation on the tendon-displacement method for moment arm measurement. A limitation of this work is that a small number of specimens was tested, and all specimens were from elderly female donors. While this small sample may not be sufficient to make generalizations about moment arm behavior, it is sufficient to show that the AGH Elbow Simulator can create physiologically-accurate moment arms. Since all specimens exhibit the same trends with respect to the dependence of the biceps and pronator teres on p/s position for f/e moment arms, and vice versa, it is reasonable to conclude that these trends are real. Future testing should include male specimens, and specimens of a wider age range, to confirm these findings.

The standard deviations, as indicated by the shaded regions on the plots, are sometimes large near the ends. This behavior is likely due to the end effects of the polynomial fits. Near the ends, the velocity of the movement was necessarily slower, which means that the rate of change in tendon length, or elbow position angle, were lower than during the mid-range of the motion. Thus the limit of the $d\theta$ term heads towards zero, which in turn makes the calculated moment arm approach infinity. Thus neglecting additional terms at the beginning and the end of the trials could reduce these end effects.

The good comparison with previously-reported moment arm values validates one aspect of the AGH Elbow Simulator's design. It illustrates that physiologically-accurate moment arms can be produced with the simulator and demonstrates that our closed-loop force controller offers a practical variation on the traditional tendon-displacement method to measure moment arms. While this one measure cannot validate all the simulator's design goals, the ability to replicate physiologically-correct moment arms is a unique feature of the AGH Elbow Simulator. In future studies, this will yield results that are more anatomically and physiologically realistic than can be produced on any other elbow simulator currently in use.

The general disagreement between the values reported by Ettema *et al.* [81] and those reported by other researchers is cause for concern. This study is one of the few previously-published to investigate the dependence of elbow muscle moment arms on both f/e and p/s position. Several factors could have contributed to their extreme results, including their methodology. Replacing the musculotendons with elastic bands, and thus removing all surrounding tissue that would ordinarily influence tendon paths could have yielded moment arms substantially larger than those feasible with the constraints of skin, muscle, and fascia left intact. While their approach was certainly novel, several other factors could have yielded more realistic results. Using an instrument more precise than a hand-held ruler to measure the musculotendon lengths could have increased the precision of their results, though it is unlikely that the errors associated with this technique would cause the dramatic departures from others' reported results. These authors also used embalmed cadaver specimens. Due to their unique method of replacing the musculotendons with elastic bands, it is unlikely that the embalming of the bones had a marked effect on the results. It is interesting to note that these authors compared their results to a work by Kawakami *et al.* who measured moment arms in live people using a geometric method

from collected CAT scans. Given the age of Kawakami's subjects (29 ± 5 years), it is anticipated that these reported moment arms may be larger than those measured in elderly cadavers. [136] However, these reported results are markedly larger than those reported elsewhere in the literature. Since these were only measured in full extension (flexion angle equal to 0°), which was beyond the range of motions feasible for the specimens tested in the present study, the values from Kawakami *et al.* [136] are not included in Figure 10.

The curve-fitting in the present work also merits discussion. The polynomials were fit to both the "r" (tendon excursion) and "θ" curves so that the derivatives could be computed analytically. Sixth order curves were chosen based on preliminary work which illustrated that lower-order curves deviate notably from the measured data near the ends of the collected data range. Given the suspected end effects as seen with the larger standard deviation magnitudes near the ends, it is possible that using even higher-order curves may additionally improve the results. The final moment arm ($dr/d\theta$) was computed numerically for simplicity. This technique is not without fault – numeric rounding errors during the evaluation of both polynomials may have introduced noise into the results. However, the averaging across trials, and doing so only every 5°, helped mitigate these effects. The polynomial coefficients reported in this chapter are those of 6^{th}-order polynomials fitted to the resultant moment arm averages for each f/e and p/s combination for each specimen. While these may differ slightly from the analytically-computed ones, these differences should not have a marked effect on the end results. It is worth noting that the sixth-order coefficients on all of the p/s curves are quite small, on the order of 10^{-10} or 10^{-11}. This implies that polynomials of a lower order may have been suitable for the p/s cases, despite the diligence in choosing polynomial order.

Gardinier *et al.* also reported a selection of peak p/s moment arms, but only for a few muscles, notably, the pronator teres and brachialis. [92] Their results indicated that p/s moment arms for these muscles change with f/e position, which is in agreement with the results described here.

As with any work, there is the potential to develop future studies based on the results presented here, and room for improvement in the current techniques to offer more accurate and meaningful results. Firstly, the f/e moment arms could be measured at discrete p/s angles, instead of at the extremes and midpoint of the motion. This may yield more accurate comparisons across specimens, since the amount of feasible p/s motion varies from specimen to specimen. While a more advanced controller (such as PI, or PID) may in theory offer improved force control, the current hardware configuration would produce inferior results due to the increased loop time and corresponding delay in system response. This added delay is a result of increased computation time from the additional gains. The 40 Hz sampling rate was the maximum achievable speed. Repeatability tests could assess the variability of moment arms from day-to-day, which would also include the variability associated with the manual positioning of the adjustable pulleys for each specimen. A rotary potentiometer could be installed to monitor the swiveling action of the pronator teres pulley assembly. This may account for the intra-specimen variability, particularly for the p/s moment arm values.

In conclusion, the moment arm values measured in the AGH Elbow Simulator compare well with those previously-reported in the biomechanics literature. This validates one aspect of the simulator's design and offers insights into the behavior of the moment arms of the elbow.

6.0 METHODS: EFFECTS OF NOISE AND CAMERA-SWITCHING ON MOTION ANALYSIS

During video-based motion analysis, multiple cameras must view the same marker in space in order for software algorithms to compute the marker's position. When using more than the minimum number of cameras to compute a marker's position, it is feasible that a more accurate result for any given frame of data could be obtained by discarding the video information from one or more cameras. Without a requirement for consistent usage of camera images in adjacent frames of data, automated tracking algorithms are free to use different cameras for each frame's analysis, which can lead to oscillation between different sets of cameras. This automated use of different camera sets across successive frames of data is referred to as *camera-switching*. The point of the present study is that camera-switching can alter a marker's calculated trajectory and that the resulting errors are significant enough to manifest themselves in the final calculations of radial head travel.

6.1 THE EFFECTS OF NOISE AND CAMERA-SWITCHING

Experiments were performed to determine the effects of both noise and camera-switching on calculated radial head travel. To gauge the effects of noise, random noise was added to simulated elbow motion and the radial head travel was calculated. To observe the effects of

camera-switching, experiments deliberately controlled camera-switching. In addition, the correspondence between increased root-mean-square (RMS) error in the relative positions of markers on one array was compared to the group of cameras used in the computation of the position coordinates.

6.1.1 Effects of noise added to radial head travel

To evaluate the effect of noise in calculated marker position on the outcome measure of travel on the radial head, a physical simulation was created. Similar to the actual cadaver application, four arrays of markers were rigidly attached to a mock elbow, as described in Appendix D. Each array contained four markers. The arrays were attached to the humerus, the distal radius, the radial head, and the ulna. Figure 20 shows the mock elbow with the marker arrays attached. The lower image highlights the positions of the markers and bones. The mock elbow was put in the AGH Elbow Simulator in a position of 90° flexion. A static image was captured using a six-camera Vicon 612 motion analysis system. The position coordinates from each ball from the first frame in which all markers were visible were then mathematically rotated 360° around an axis parallel to the ulna and passing through the center of the capitellum in increments of 0.5°. This 360° was not intended to represent anatomically realistic motion. Rather, the intent was to create an ideal data platform on which to impose noise. In addition, the relative positions of the markers were precisely measured on a coordinate measuring machine (Numerex, Zeiss.) This allowed the calculation of the center of the capitellum relative to the humerus markers. The root mean square error of the markers' relative positions, as output from the Vicon system, was computed.

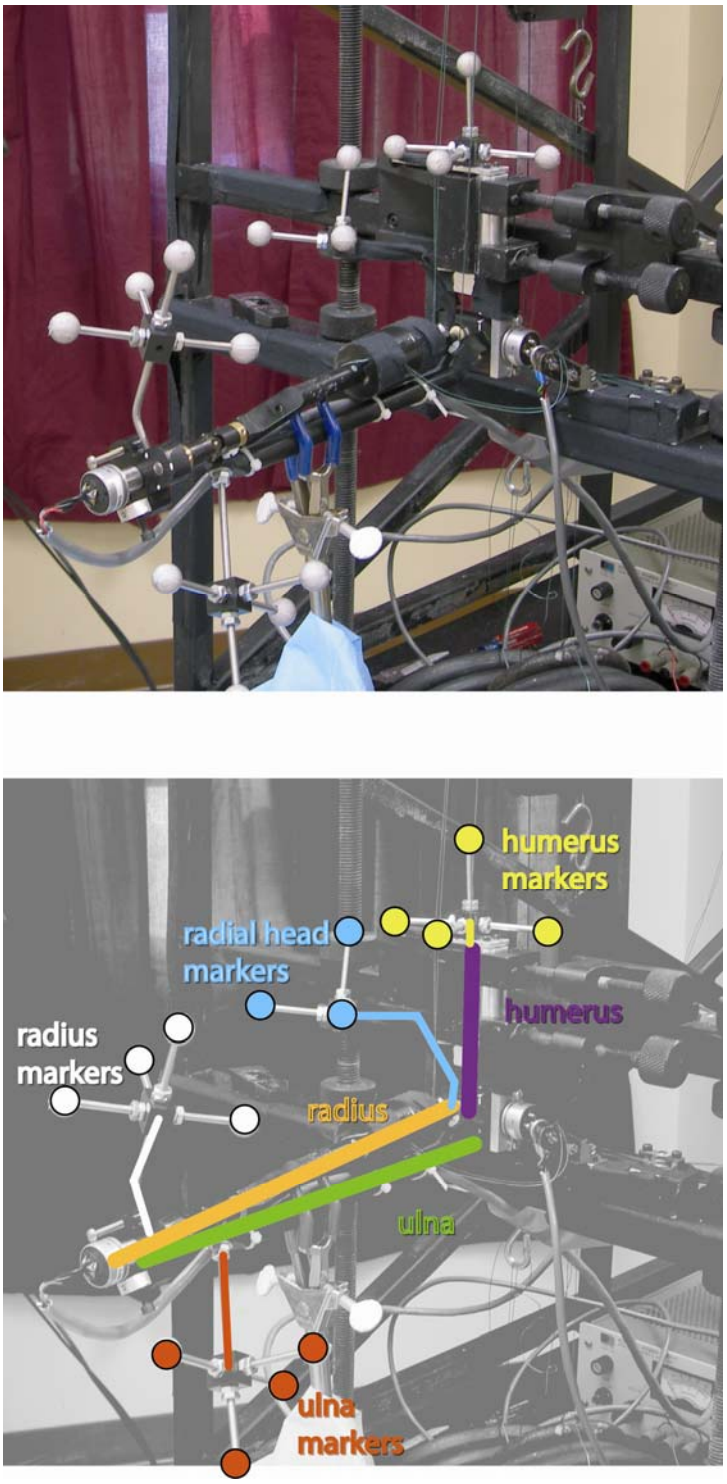


Figure 20: Mock elbow with arrays attached. The bottom image highlights the marker and bone positions.

Noise was then added to the markers' positions. Random noise of maximum magnitude 1 mm was used. The normally-distributed random noise was generated via the MATLAB (The Math Works, Natick, MA) function *randn*. Two noise cases were created: 1) noise was added to a single ball of the array on the radial head, and 2) noise was added to all four of the balls of the array on the radial head. The radial head travel on the capitellum was then calculated for these cases with the added noise. The predication was that the noise added to all balls of the radial head array would create marked deviations from the true path of the simulated radial head motion.

6.1.2 Correspondence between increased RMS and cameras used in calculation

The root mean square error (RMS) of each array, a measure of the position error relative to the centroid of the balls of each array, was calculated. To compare sharp changes in RMS with

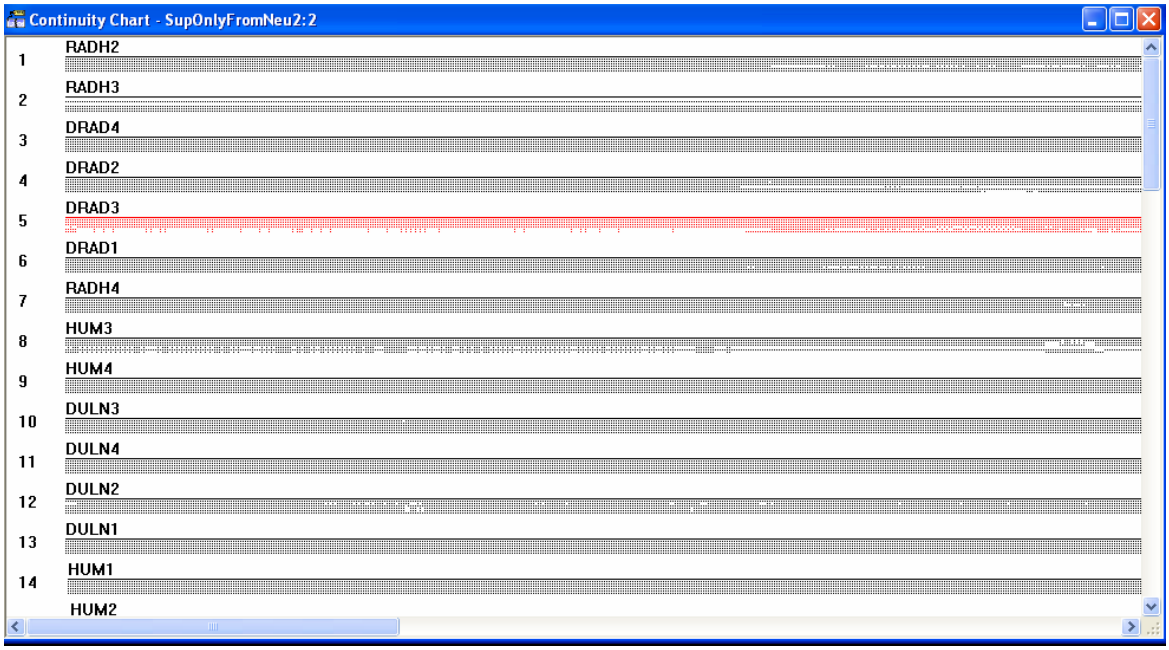


Figure 21: Example Continuity Chart

changes in the cameras used to compute the trajectories, the Continuity Chart was used. This output communicated which cameras were used by the automatic tracking algorithm in the Vicon Workstation software at each time instant. This information is not available for export other than via a screen capture. An excerpt from a Continuity Chart is shown in Figure 21. Each cluster of lines represents one marker. There are six lines in each cluster, with each line corresponding to one camera. For every frame of data recorded, dots in the continuity chart indicates which cameras were used to compute the marker's location. Gaps indicate that the camera was not used in the calculations for that particular frame. For example, in Figure 21, the red cluster of lines (labeled "DRAD3") gives information about one of the distal radius markers. At the start of the trial, the data from Cameras #1, #2, #3, #5, and #6 were used to compute this marker's location. Shortly thereafter, the data from camera #4 was used, and cameras #5 and #6 were neglected most of the time, though they were used in occasional, intermittent, and non-adjacent frames. This is an example of *camera-switching*, when the software algorithms disregard image data from some cameras some of the time.

Visual inspection of the Continuity Chart was compared to the computed RMS to seek correlation between dramatic changes in the RMS error and changes in the cameras used to compute a marker's position coordinates. Since the Continuity Chart was not available for quantitative export, only qualitative conclusions can be inferred.

6.2 EXPERIMENTS WITH CONTROLLED CAMERA-SWITCHING

Based on the observations made in the experiments described above, a study was designed to test the hypothesis that camera-switching influences the measured position of the markers. These

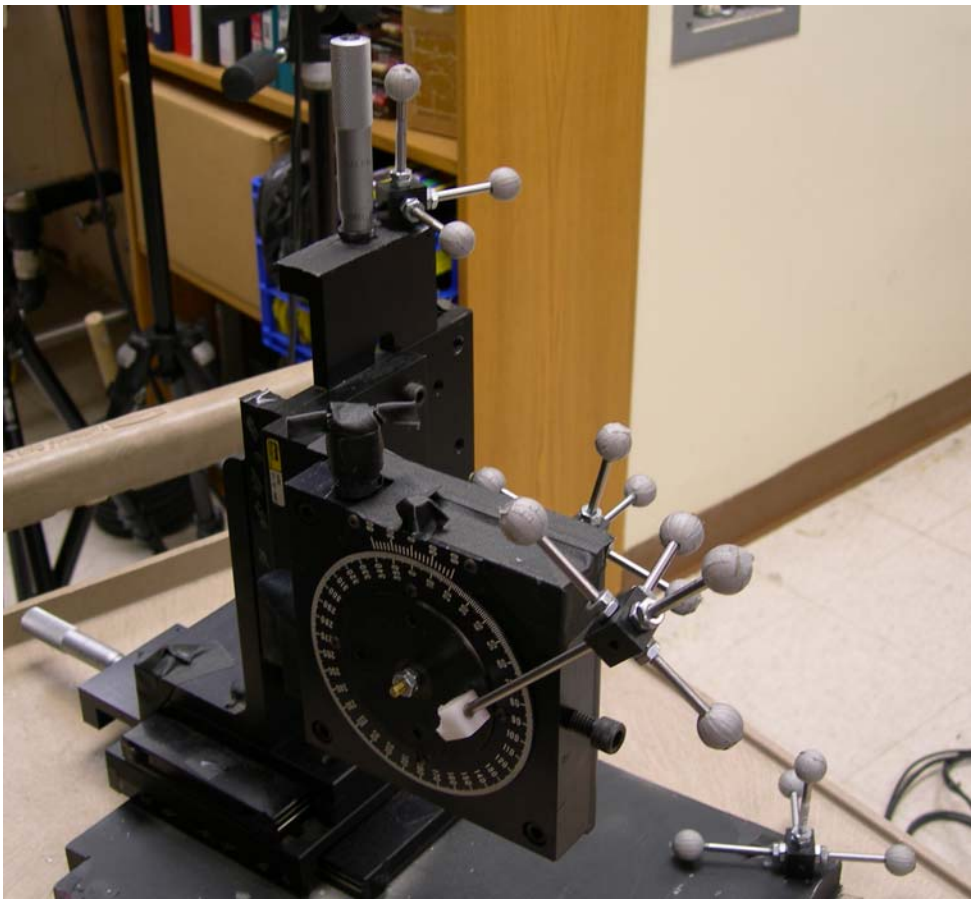


Figure 22: Custom apparatus with arrays of retroreflective markers.

experiments deliberately removed and reinstated markers from cameras' views during collection of data. This was achieved by blocking select cameras' fields of view in a controlled manner. Tests were performed on both stationary and moving markers. Marker arrays of four retroreflective balls each were used, as in the specimen testing. [35, 91, 176-178]

These arrays were attached to a custom apparatus (see Figure 22) which included three translational and one rotational precision microtables (#4499 and #10000, Parker Daedal, Harrison City, PA.) The markers were 9 mm in diameter. Threaded rod fixation with cyanoacrylate adhesive assured rigidity of the construct.

Six Vicon M-cameras (612, ViconPeak, Oxford, UK) were arranged in a semi-circular pattern at varying heights surrounding the test apparatus. There were no obstructions between any camera and the apparatus. No camera could see any other cameras' infrared strobe rings and all cameras could see all sixteen markers. The motion capture volume was approximately 1.25m x 1.25m x 1.25m. The cameras had recently been backfocused. as recommended by Vicon technical support. to ensure a clean image. The windows in the room were covered to prevent interference from bright sunlight. Camera data were collected at 120 Hz on two different days. The mean (range) calibration residuals for each day were 0.401 mm (0.322 mm) and 0.413 mm (0.284 mm). Vicon Workstation, v.4.5 (ViconPeak, Oxford, UK) software was used.

6.2.1 Data collection

Three types of data were collected: static, translational, and rotational. During the static trials, there was no movement of the markers. The translation and rotation movements were performed with the precision micrometer tables. During the translation trials, the markers were moved a distance of 12.7 mm, and during the rotation trials, the precision table was moved through an angular rotation of 23° measured with an accuracy of 0.1°. Five camera block states were used. Figure 23 shows the camera arrangement and illustrates the five camera block states. Case A used all cameras all the time. Cases B and C blocked three of the cameras. Cases D and E selectively blocked, then restored to view, three cameras during the markers' motion. Cases D and E were used only for the static and translational trials. A black felt sheet was used to obstruct selected cameras' views of the test apparatus in cases B, C, D, and E. The timing and completeness of the camera obstruction was verified with the software package using the Continuity Chart.

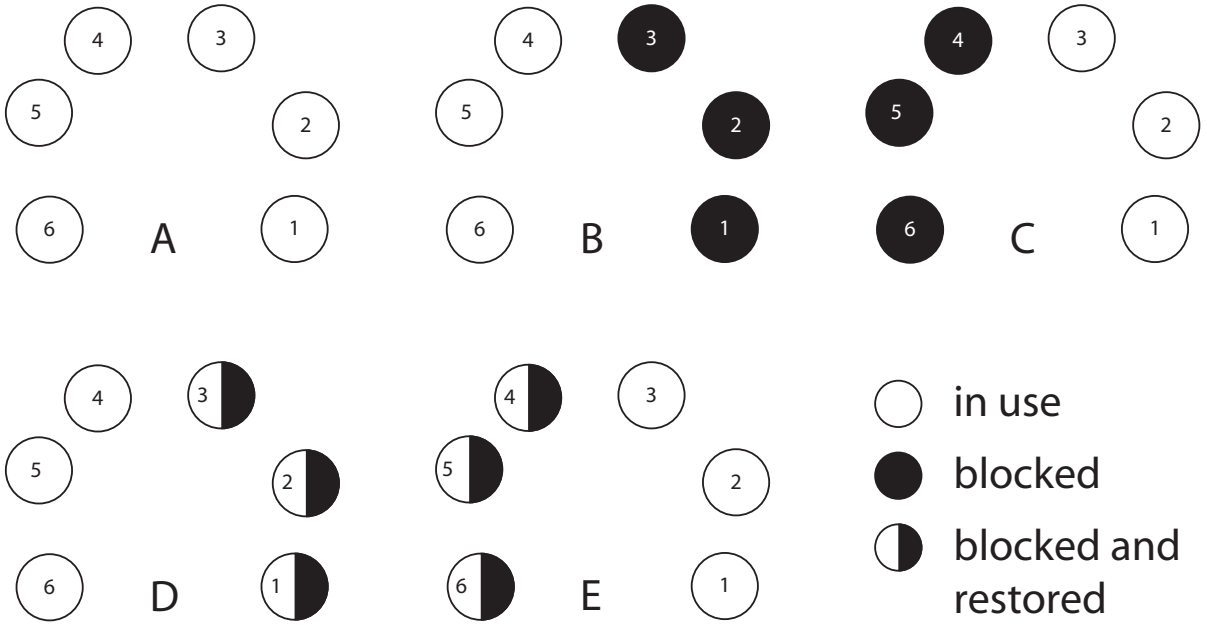


Figure 23: Illustration of camera arrangement and camera block states.

6.2.2 Data analysis

Following data collection, the marker trajectories were reconstructed using the commercially-provided software (Workstation). The reconstruction parameters were adjusted for each trial to minimize the total number of trajectories, as is standard practice for motion analysis. The trajectories were defragmented, if necessary, prior to export. Further calculations were performed in MATLAB (The MathWorks, Natick, MA.) The maximum measured distance, in mm, was calculated for both the static and translation trials, and the maximum measured change

in angle was computed for the rotation trials. These calculations were repeated after applying a median filter [261] (width = 11) in MATLAB to the position coordinates. Median filters are commonly used in image processing and by nature preserve sharp changes in the data while removing the outliers. For this application, a median filter was particularly advantageous as the interest was in observing sharp changes in position that occur for extended time periods.

Repeated-measures ANOVAs were conducted for each type of data (static, translational, and rotational) with the significance level set at $\alpha=0.05$. For all cases, the first repeated measure was distance (or angle), and had three levels: actual (truth), computed, and filtered. The second independent variable was camera block state, as defined above, to give five levels for the static and translation trials, and three for the rotation. A contrast on the main effects indicated which cases were distinguishable from others.

7.0 RESULTS: EFFECTS OF NOISE AND CAMERA-SWITCHING ON MOTION ANALYSIS

The results of the study of the Vicon system accuracy indicate that the perception and quantification of fine movements can be altered with changes in which cameras' images are used in the reconstruction of the position of a marker.

7.1 NOISE AND OBSERVED CAMERA-SWITCHING

Based on the observations below, camera-switching may interfere with the accurate computation of the travel of the radial head. These results are based on observational evidence including a simulation of noise added to known marker position coordinates and the association between the results shown on the Continuity Chart and the calculated RMS error of the radial head array of four markers.

7.1.1 Simulation with noise

Adding noise to the idealized data had a marked affect on the results. The qualitative results of the ideal data with added noise are shown in Figure 24. The large heavy circle represents

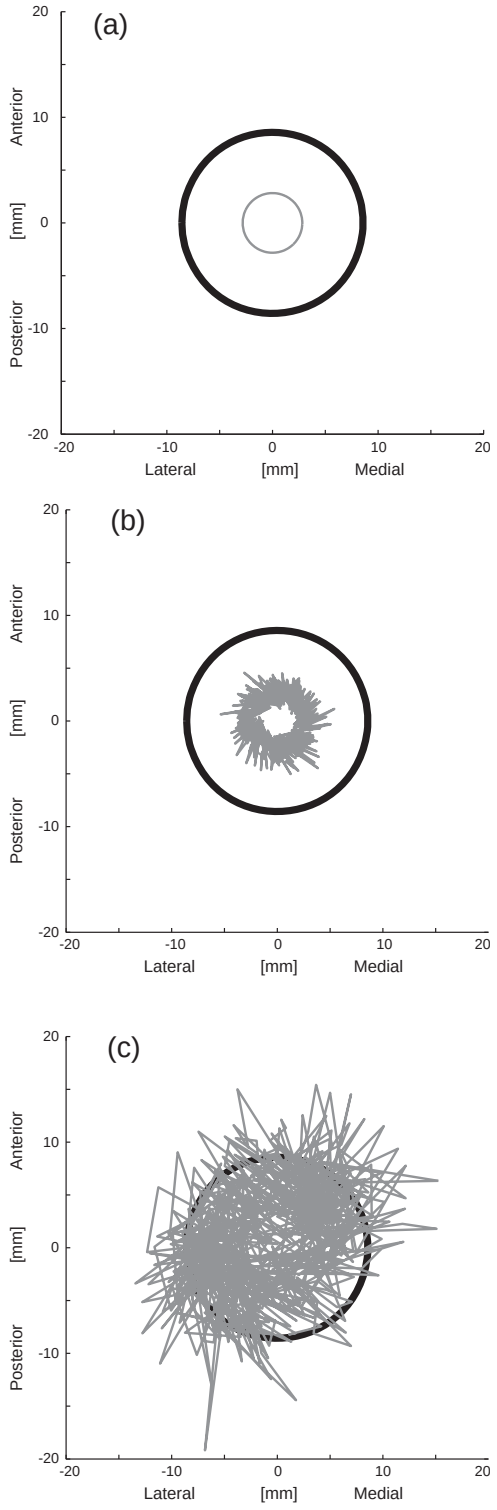


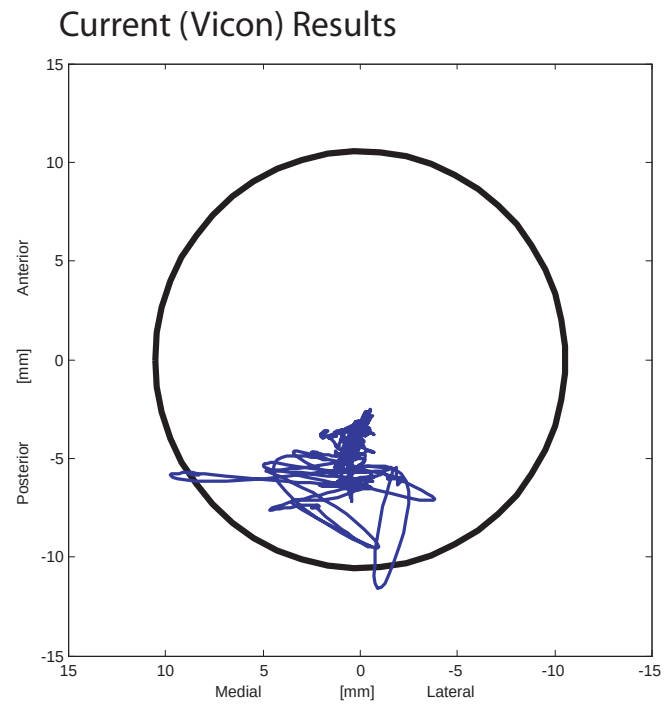
Figure 24: Calculated radial head travel from simulation based on data collected with six Vicon cameras a) without noise, b) with noise added to one ball, and c) with noise added to one array of markers.

the capitellum, and the smaller is the calculated travel of the radial head, projected on to the capitellum. Subplot (a) shows the simulated data as if the radial head were traveling in a perfect circle around the center of the capitellum; (b) shows the results when 1 mm of random noise is added to one marker's position; and (c) shows the radial head travel when 1 mm of noise is added to all markers used in tracking the radial head. Note that in (c), the calculated travel of the center of the radial head extends to outside of the capitellum and the true circular motion is completely obscured.

When compared to results from preliminary tests with cadaveric specimens, the effects of noise in the marker position data on the calculation of the radial head travel could explain some of the inconsistent and unexpected patterns in radial head travel. Figure 25 shows an example of such preliminary tests. The top subplot shows calculated radial head travel based on data collected with the Vicon system. For comparison, a representative example from previous work performed with a PEAK5 motion capture system is also included (bottom.) The large erratic variation in the trail collected with the Vicon system, when compared to the simulation results with noise presented above, suggests that there may be noise in the marker position coordinates.

7.2 CONTROLLED CAMERA-SWITCHING

The measured results from the static trials with selective camera blocking indicate that change in static marker position of up to 3.7 mm can occur when different sets of cameras are used to calculate the marker's position in space (Figure 26a). Recall that the effect of 1 mm of



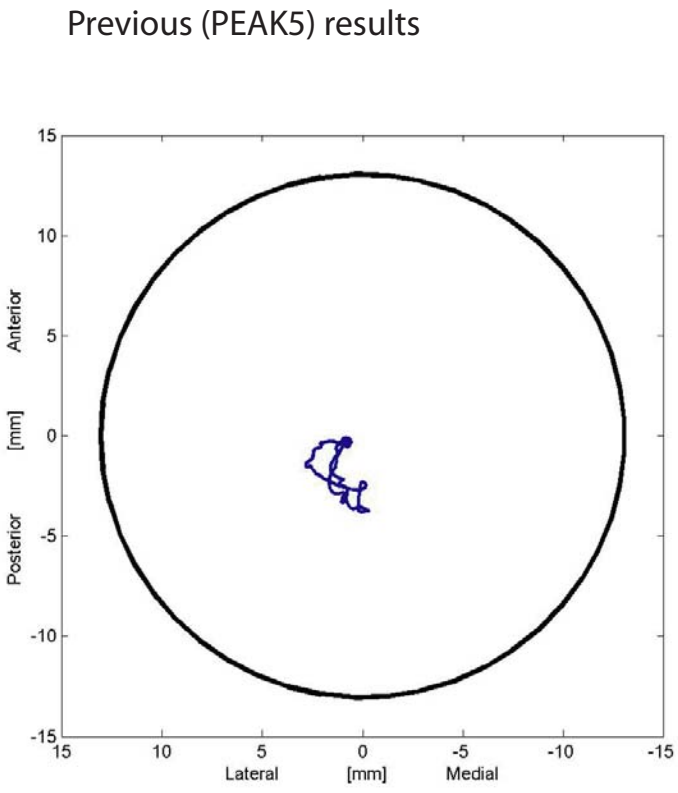
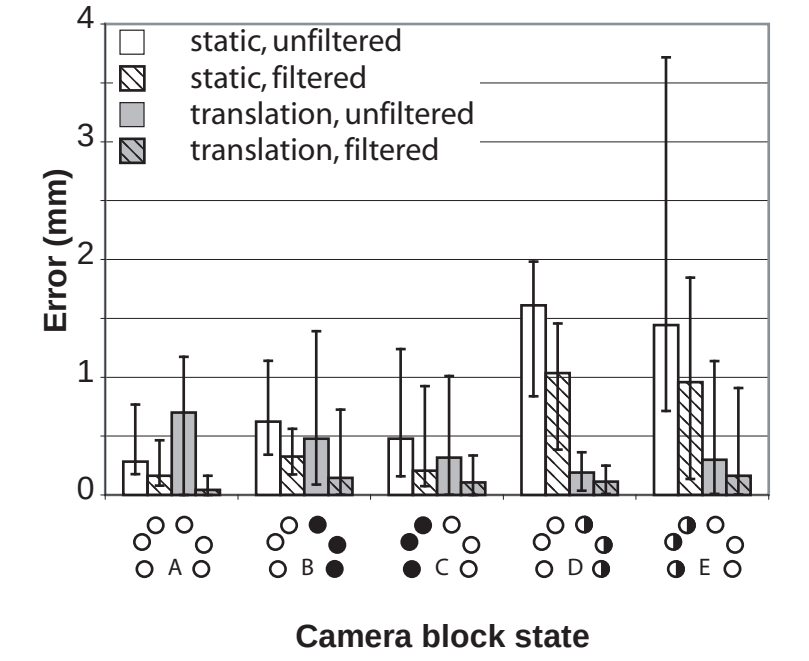


Figure 25: Radial head travel from Vicon and PEAK5 systems.

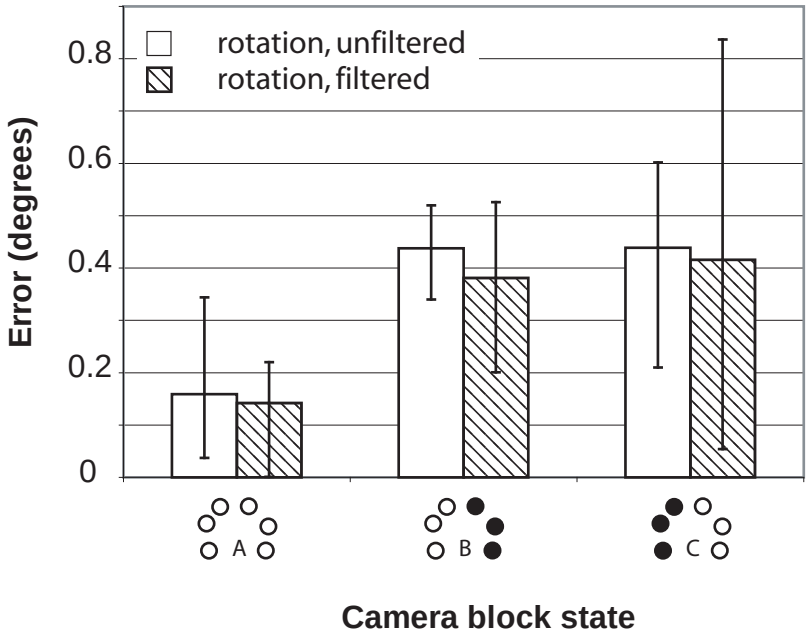
noise, as illustrated in Figure 24, is profound. Figure 26 shows the difference between the actual and measured position and gives both the average and range of values for each case. Note that the values depend on which cameras were blocked. The statistical results show that the difference between the true magnitude of movement, the measured (computed) value, and the filtered values is always statistically significant at the $\alpha=0.05$ level. Post-hoc contrasts on the main effects showed that, for the translation and rotation trials, the camera block state always has a significant influence on the results ($p<0.05$). For the static trials, the camera block state can distinguish between the differences for all cases except between camera states A and E ($p=0.056$). The contrast for these static trials was significant at $p<0.001$, with an observed power of 1.000.

Figure 27 shows an example of the position coordinates of one marker and indicates the timing of the camera blocking and restoration to view for the static trials. Although no motion actually occurred, the Continuity Chart confirmed that camera-switching occurred at the same time as perceived motion. This controlled alteration of view simulates markers going in and out of view of some cameras, as may occur during typical biomechanical studies.

An example taken from the static data shows the large effect of camera switching as measured in data collected as preliminary tests for the cadaver studies (Figure 28). Note that the patterns in the insets which show calculated radial head travel occur at the same time, and have the same shape, as the pattern in the RMS error. The sharp changes in RMS error correspond to the changes in camera selection by the software during the reconstruction, as verified with the Continuity Chart in the Workstation software (Figure 29). The Continuity Chart is truncated to show only the region of interest. The colored segments (the first, third, and sixth groups of lines) correspond to three of the four radial head markers. (The fourth did not experience camera-



(a)



(b)

Figure 26: Error between measured and actual translation (a) and rotation (b) motion.

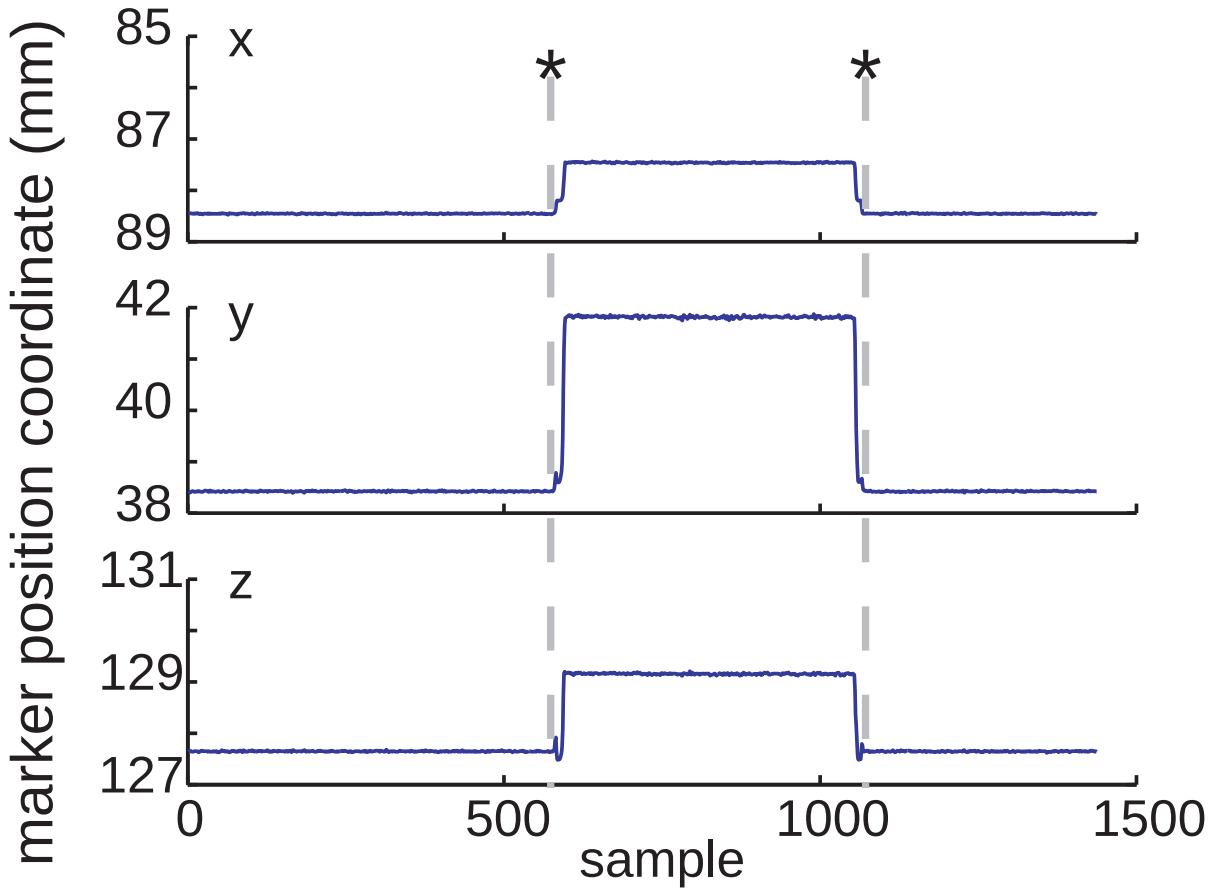


Figure 27: Marker coordinate components from a trial in which some cameras were obscured and restored to view. The asterisks indicate the onset and release of the obstruction.

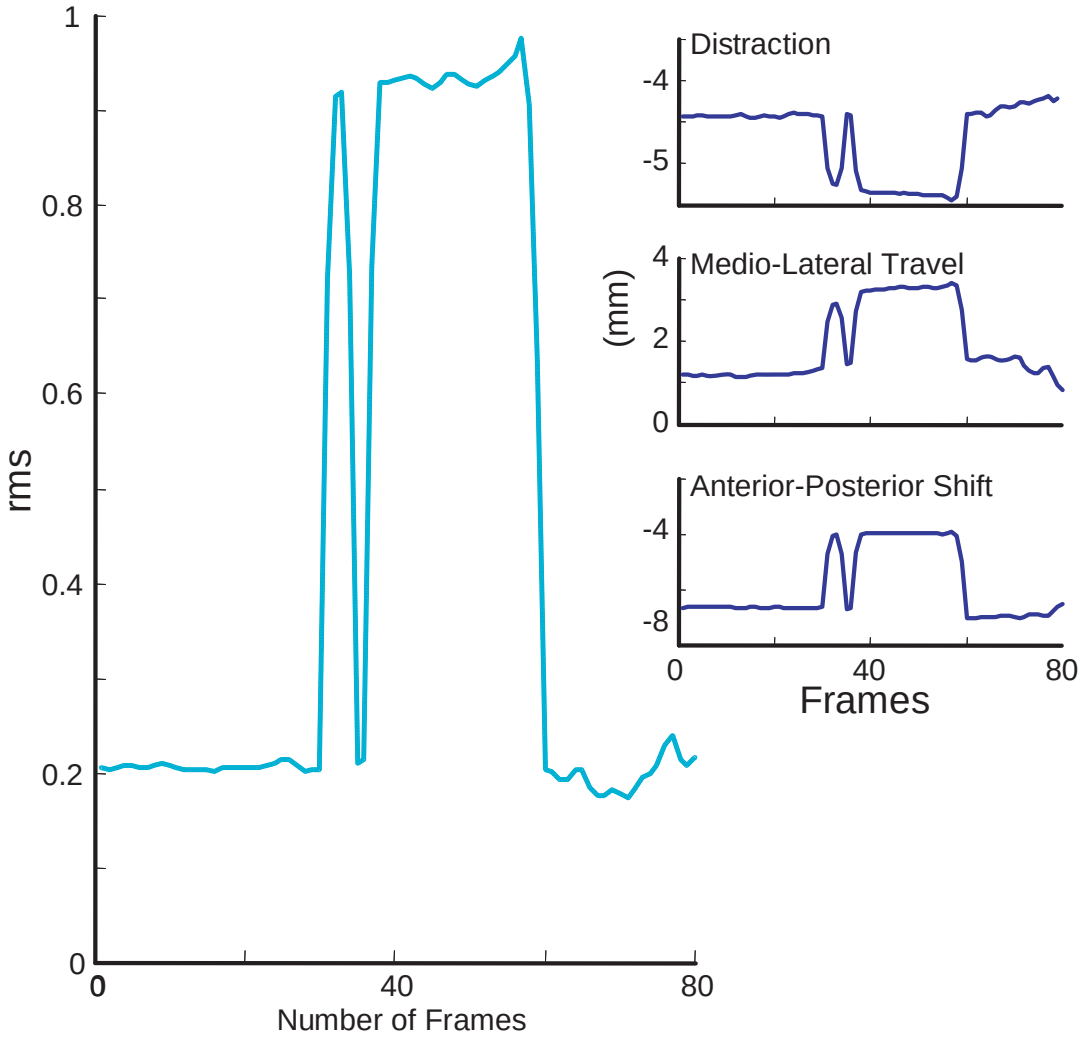


Figure 28: RMS error of radial head markers during movement and the corresponding calculated radial head travel (insets).

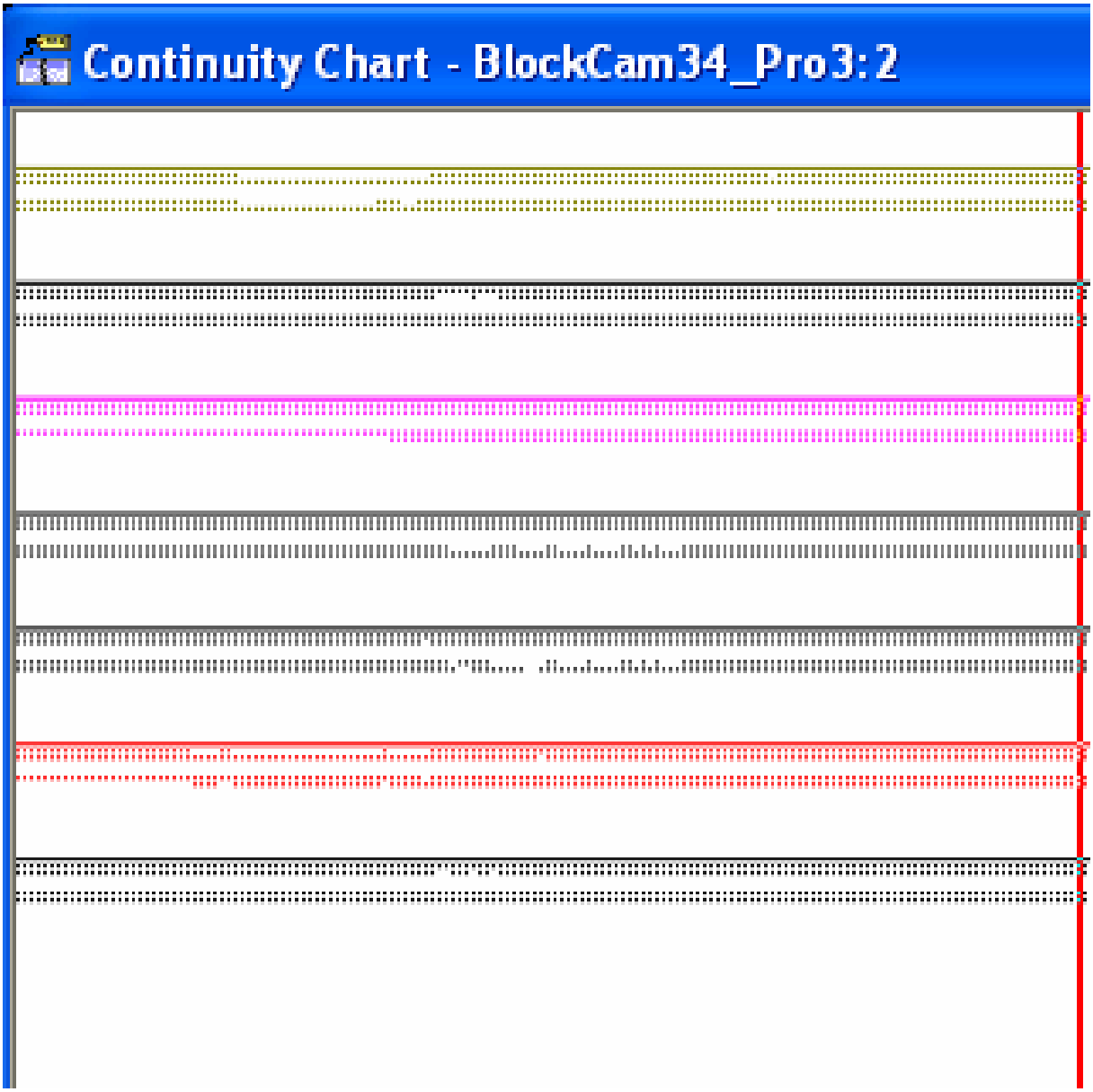


Figure 29: Continuity chart excerpt.

switching during this time, and was displayed too far down in the Continuity Chart to be captured on the same screen as the others.)

7.3 DISCUSSION

The results indicate the statistically significant effects of camera-switching on both real and simulated data. The effect of noise on the simulated data is extreme. Ehara *et al.* reported maximum errors of up to 28.23 mm in a volume calibrated for gait analysis when comparing the accuracy of multiple motion analysis systems. In that study, the Vicon systems exhibited up to 8.57 mm of noise in a volume of 10'x 6'(3.048 m x 1.8288 m) and undisclosed height. [77] The resultant path with no noise indicates that the postprocessing in Matlab is not to blame. This is only confirmed with the success of the postprocessing using data collected from the Peak5 system. The results from the controlled camera-blocking experiments show that large variations of up to 3.7 mm in perceived marker position can come from changing the combination of cameras viewing the markers, which de facto changes the cameras used during the reconstruction process. For example, in observations of some trials, the Continuity Chart often indicated that two alternating sets of cameras would contribute to a marker's construction, alternating across successive time instants. While this obviously reflects the software's attempt to ensure the most accurate results, it creates this new problem which introduces error into the final product. This problem is unique to systems which have more than two cameras.

The association between the camera-switching and increased RMS error is striking. This is a clear indicator of the fact that, when different cameras are used to compute a marker's trajectory in space, the outcome calculation of motion can be altered. While such small errors

(<4 mm) may not affect large-scale biomechanics work, such as gait analysis, the impact on the tracking of small motions is large. It is worth mentioning that this problem could be related to the focal length of the lenses used. Though factory-supplied 12.5 mm lenses were used, a set of 50 mm lenses was also acquired. From casual experimental observation, the different lenses did not mitigate the camera-switching problem, nor did they permit the accurate tracking of very small (<2 mm) markers.

In general, the magnitude of the difference between the measured and actual motions is reduced with filtering, as shown in Figure 26. While this result is not surprising because the differences are still nonzero, it indicates that filtering alone is not a sufficient solution to account for these accuracy problems. Given that the restoration to view of blocked cameras returns the marker position coordinates to their original path (Figure), and the small variation in measured position when all cameras could view all markers, one can conclude that this may affect the position results when a marker is obscured from, or returns to, a camera's view. This is of particular concern in any biomechanical study which requires measurement of very fine motions. In addition, it is noteworthy that the translation errors are in general smaller than the static errors, which suggests that an automated tracking or interpolation algorithm may be in use.

One solution might be to offer the end user the opportunity to participate in the marker reconstruction and trajectory calculation process so that the user could determine which cameras to use at a given time instant. Another would be to create calibration that considered all possible combinations of camera during the trajectory reconstruction process.

This is the first study to report on the effects of camera-switching on the tracking of small motions. Previous studies [77, 78] have reported on large motions that span multiple camera views and showed that the effects were minimal for large-volume applications. The effects on

small motions presented here illustrate that this issue cannot be ignored during the study of radial head motions, and that automated tracking software may not be appropriate for use when studying the nuances of radial head motion.

8.0 ELBOW MOTION TRACKING: ALGORITHM DEVELOPMENT

A method to track the motion of the radial head is developed and presented here. There are differences in methodology between the approach used to track the motion of the native radial head and the replaced head. A method to track the finite helical axis of the elbow during pronation/supination motion is also presented. The data used to validate all tracking methods is described here and the results are presented in Chapter 9.0 .

8.1 RADIAL HEAD TRACKING

The motivation behind tracking the motion of the radial head was to develop the hardware and software tools necessary to compare the kinematics of different radial head implants in cadaver specimens actuated by the AGH Elbow Simulator. The method used could not impede the motion of the radial head. This requirement dictated that any physical markers added to the native head (or implant) must be small in both mass and in size. Additionally, due to the ligamentous constraints of the joint such as the annular ligament, markers placed on the radial head (native or implant) are not guaranteed to remain visible during p/s motion. That is, some markers will disappear during extreme pronation and supination.

The overall approach to tracking the native head is rooted in the fact that, during the motions, the radial head is rigidly attached to the distal radius. Therefore, by tracking the

movements of the distal radius, the movement of the radial head itself can be computed. However, this approach cannot be used with the implants because some implants are designed to permit rotation of the radial head component with respect to the stem. In some cases (see Chapter 2.2.3) the stem itself can rotate with respect to the distal radius. Thus the implant motion must be tracked directly. Since Challis [47] showed that the greatest increase in accuracy with increasing numbers of markers per rigid-body segment occurs with the increase from three to four markers, four markers each per segment were used. The desired accuracy of the motion tracking system is to track the radial head position within 0.1 mm.

8.1.1 Tracking the radial head implant

Tracking the motion of radial head implants requires direct observation of markers on the implant. The implants of interest for future evaluation are the Wright *Evolve*, the KMI *Katalyst*, and the Avanta *rHEAD*. Each of these has a circular cross section. Circular divots ("dots") of 2 mm diameter were precision-machined to be equally-spaced around the circumference of the radial head implant (Figure 30). An indexer was used to ensure that these divots were all in the same plane. Then, so long as any three dots are visible to the cameras, the center of the circle can be computed and will be coincident with the center of the radial head implant. This can be projected onto the plane of the capitellum so long as rigid markers attached to the humerus are also visible to the cameras during data collection and the position of the capitellum with respect to these humerus markers is digitized following motion data collection.

Figure 30: Radial head implant with machined divots.

8.1.1.1 Circular tracking algorithm

Given three points on a shared plane, a circle can be fit to these points. Since there is not a direct solution to this problem, an optimization of some sort must be used. Simply doing a 2D

projection on to the best-fit plane to the points loses information in the projection, so a more robust solution is desired. [60, 86] The strategy is to first do an in initial fit to serve as a starting point for the optimization. This initial fit is generated by finding the plane that best-fits the visible points, project the points to this best-fit plane, and then fitting a circle to the projected points. With this as the starting point, the optimization problem is then to find the center of the circle that minimizes the Euclidian distance from the center to all points.

To successfully execute this initial fit, six parameters are necessary: three for the best-fit plane, and three to determine the circle in that plane. Executing the subsequent optimization requires the definition of a two-parameter three-dimensional rotation matrix that maps planes to planes ($\boldsymbol{M}$), a three-parameter center of a three-dimensional circle (c), and the radius of that circle (R). The final optimization equation is shown below (Equation 4), and a more detailed

description of the process, with equations, is given in Appendix L. The algorithm is implemented in the program *center_circle_implants.m* in Appendix N.

$$F = \sum_{i=1}^{n} \left[\left\| M(p_i - c) \times (0,0,1) \right\|^2 - R^2 \right]^2 \quad \textbf{(4)}$$

8.1.1.2 Postprocessing code

Once the circle is fit to the visible dots on the implant to determine the center of the radial head for each instant in time, this must be transformed onto a plane of the capitellum to produce a clinically-meaningful result. Custom MATLAB code (*LaurelsProgram.m*) was created for this purpose, and uses the circle-fitting techniques described above. A schematic of the code is shown in Appendix M, and the code itself is given in Appendix N. An outline of the method is presented below. Line numbers refer to the code as printed in Appendix N.

To successfully use this code to track the motion of radial head implants, it is necessary that: 1) at least three of the dots on the implant are visible to the cameras at all times, though the visible dots can change throughout the motion so long as any three are visible at any given timepoint; and 2) four markers must be placed on the humerus and visible to the cameras throughout the motion. Following the data collection, these markers must be digitized with respect to the capitellum and other humeral landmarks, including the trochlea and the center of the humeral shaft. Landmarks for all bones are shown in Figure 31. Optionally, a potentiometer may collect p/s angle data so that the relationship between radial head position and p/s angle can be explored.

The desired outcome is radial head position in an anatomically-meaningful reference frame, such that the z-direction points anteriorly, the x-direction points laterally on right elbows

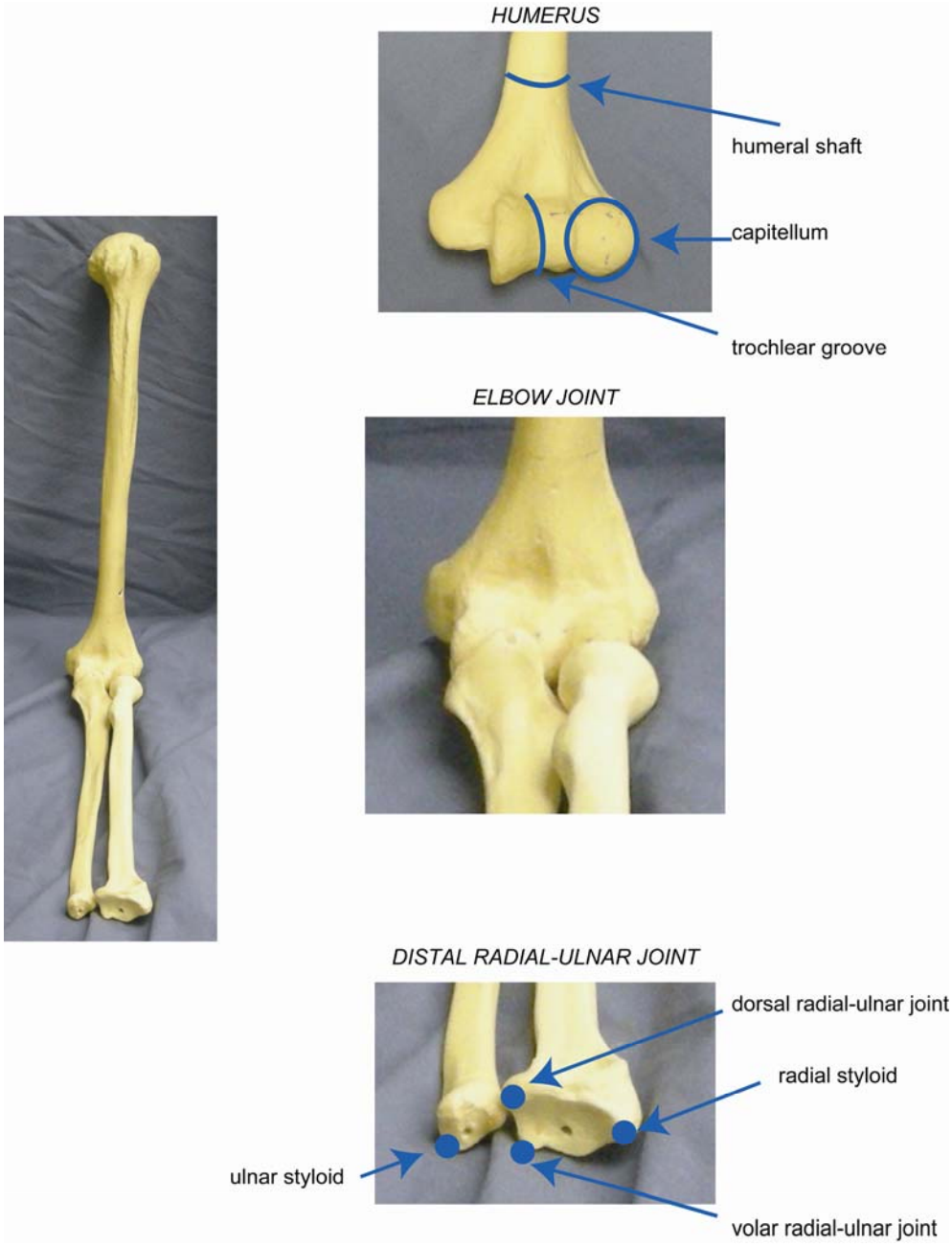


Figure 31: Elbow joint landmarks.

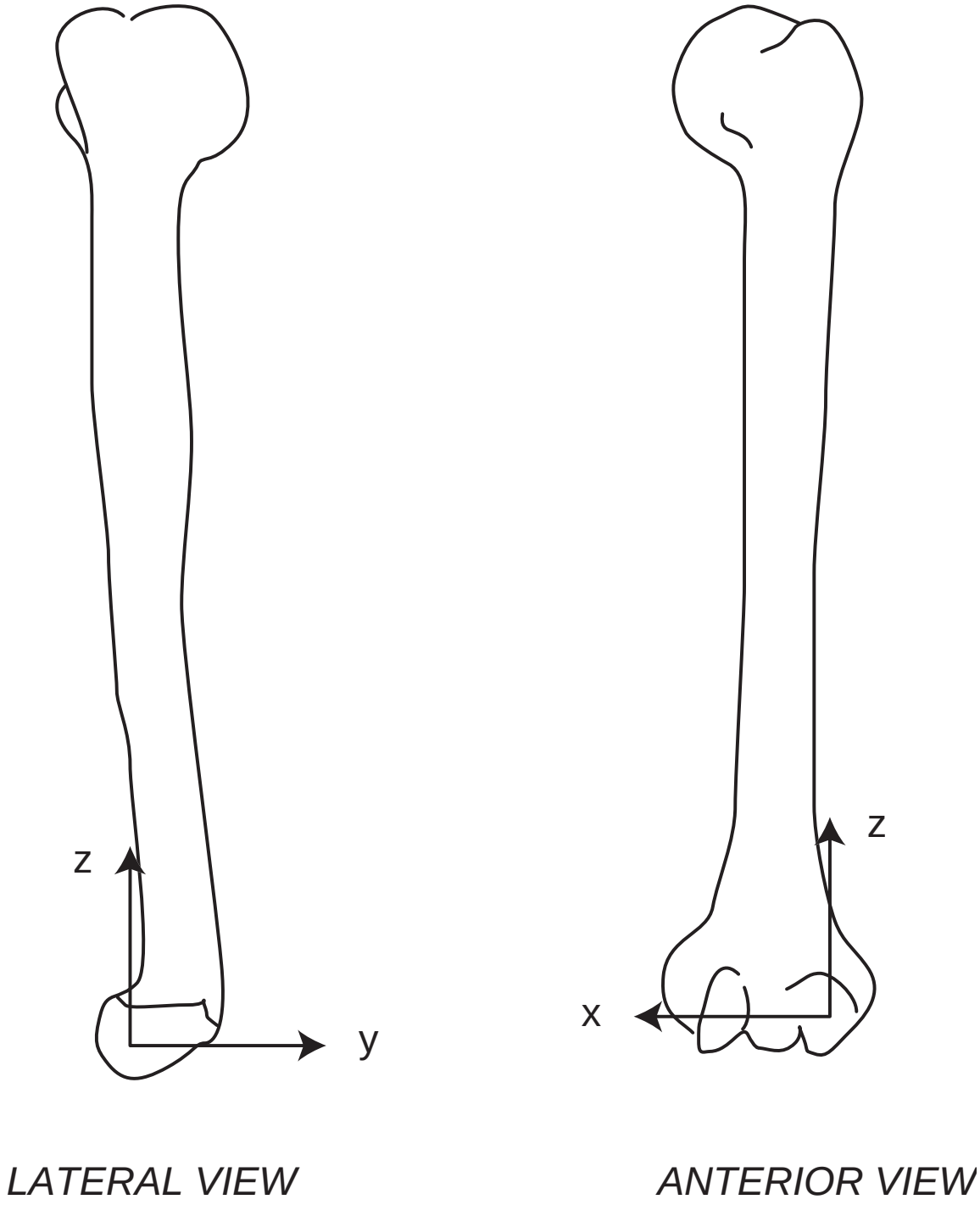


Figure 32: Elbow coordinate system (left elbow shown)

(and medially for left elbows), and the y-direction points ventrally, as shown in Figure 32. To achieve this, several transformation matrices are necessary. Three transformation matrices are necessary for the following (*lines 355-357 in LaurelsProgram.m*) to give the radial head position in the capitellum-centered coordinate system:

$$[X\_RadHdOnCap] = [T\_feang][Tac\_hum][Tcg\_hum][CircleCenter] \quad (5)$$

where *X_RadHdOnCap* is the position of the radial head in the capitellum-centered coordinate system, *CircleCenter* is the center of the circle of the implant in the global coordinate system as computed with *center_circle_implant.m* (Appendix L), and *T_feang*, *Tac_hum*, and *Tcg_hum* are as described below.

The anatomic coordinate system (ACS) is established from the digitizing on the Coordinate Measuring Machine: the x-direction is defined as that from the center of the trochlea to the center of the capitellum; the y-direction is computed as the cross-product between the x-direction and a vector from the center of the capitellum to the center of the humeral shaft, and the z-direction is then computed as the cross-product of the x- and y-directions. This anatomic coordinate system (ACS) transformation matrix, *Tac_hum*, is one of the three transformation matrices necessary to transform the center of the radial head implant, in the global motion analysis coordinates, to the ACS centered at the capitellum. It does not vary with time and is computed in line 224 of *LaurelsProgram.m.* The other two matrices are described below.

To transform from the global motion analysis coordinate system (GCS) to the CMM coordinate system (CCS), the matrix *Tcg_hum* is created (*line 241*). This uses the four small

markers, which are tracked by the motion analysis system and measured on the CMM. A rigid-body is assumed and is fit to the position coordinates as described by Soederqvist [226] and also by Challis. [47] Briefly, the marker points are expressed relative to the centroid of all visible humerus markers. A singular-value decomposition algorithm computes the rotation matrix between these two coordinate systems, and then the translation is computed. The result is a 4x4 transformation matrix that includes both the rotation and translation of points in the GCS to the CCS. This transformation matrix is constant with respect to time for a given motion.

The ACS is oriented as if the elbow is flexed 90°. At other flexion angles, the desired result is achieved by projection of the radial head travel onto a plane that passes through the center of the capitellum, but is perpendicular to the long-axis of the ulna. The required final transformation, *T_feang (notated as "R_feang" in the MATLAB code)*, required is that which maps the results from the coronal (vertical) plane to this desired plane. It is simply a rotation about the anatomic x-axis and is computed in *line 342*. Given that the intent is to test p/s motions at a constant f/e angle, this matrix will be constant with respect to time for each motion collected.

8.1.2 Tracking the native radial head

The general approach to tracking the native radial head is slightly different than that for tracking the implant. The shape of the native radial head has been described as elliptical [43] and with the annular ligament intact, the surface of the radial head is not readily visible. However, unlike the implant case, the native radial head is a part of the radius bone. Thus tracking the movement of the radius is sufficient to compute the location of the radial head, which is the general approach used here. This requires measurement of the center of the radial head relative to the

radius markers, which is complicated by the fact that, in the planned future experiments, the radial head will be resected to allow implant insertion. Thus the relative position of the entire radial head relative to the radius bone markers must be established.

To successfully track the motion of the native radial head, the postprocessing MATLAB code (*LaurelsProgram.m*) assumes that during motion data collection, the distal radius markers and humerus markers are visible (four markers on each bone). Optionally, a potentiometer may be used to monitor p/s angle so that the relationship between radial head position and p/s angle can be studied. Additionally, four small markers must be placed on the portion of the radial head that will be excised. These markers need not be visible by the motion capture system during the motion collection. However, a static image showing both the distal radius and small radial head markers is required in the analysis described below in order to compute the matrix *Tgt_gst* as described below. Following excision, the radial head must be measured on the CMM to determine the position of the center of the radial head with respect to the small markers. Similar to the implant case, the humerus must be digitized to measure the relative position of the centers of the humeral shaft, capitellum, and trochlear groove to the markers visible during the motion capture of the elbow with the native head.

Five transformation matrices are required. *T_feang* is computed as described above for the implant case, as is *Tac_hum*. The method to compute *Tcg_hum* is the same, but the humerus markers visible during the motion capture for the native head case need not necessarily be the same as those for the implant case. For example, they could be bigger markers or markers in a different location. Two additional transformations are required: *Tgc_head* and *Tgt_gst*. A schematic is given in Appendix M.

The final computation of the center of the radial head in the desired plane passing through the center of the capitellum is as follows (*lines 383-389*):

$$\begin{aligned}&[X\_RadHdOnCap]\\&=[T\_feang][Tac\_hum][Tcg\_hum][Tgt\_gst][Tgc\_head][x\_head\_CMM]\end{aligned} \qquad (6)$$

where *X_RadHdOnCap* is the position of the radial head in the capitellum-centered coordinate system, *x_head_CMM* is the center of the native head as measured on the CMM, and *Tgt_gst,* and *Tgc_head* are as described below.

Tgc_head (*line 328*) transforms from the CCS when measuring the radial head to the GCS of the static data file. It is computed in a manner similar to that to determine *Tcg_hum* above. A local coordinate system is established on the radial head, and the digitized markers in the CCS are used in the same algorithm to compute *Tcg_hum*. The matrix *Tgc_head* is critical because the CCS is the only coordinate system in which the center of the radial head is known.

The remaining transformation matrix, *Tgt_gst* (*line 312*), relates the radial head position to the dynamic position of the radius. It is the transformation from the static to the dynamic motion trial. This is computed for each timepoint and is based on the position of the markers on the radius bone in both the static and dynamic data sets. Since the native radial head is a part of the same rigid body as the radius during motion collection, *Tgt_gst* as computed by using the radius markers is identical to that which would be computed using the small markers on the radial head if they were indeed visible throughout the motion.

For both the native head and implant cases, the MATLAB program offers optional filtering of the marker position data. Each of the three-dimensional coordinates for each point is filtered independently of each other using a fourth-order Butterworth filter with a user-specified cutoff frequency.

8.1.3 Validation of tracking methods

To validate the MATLAB code described above, two approaches were used: simulated data and collected data. The simulated data consisted of a set of fictitious points expressed in three different coordinate systems to replicate the three different coordinate systems (ACS, GCS, and CCS) that will occur with real data. Table 20 shows the points used in one coordinate system. These data simulated motion capture of a static subject. Results were computed for six different combinations of coordinate systems: a) all coordinate systems were the same; b) the CCS was different (the ACS and GCS were the same); c) The static trial GCS was different; d) The dynamic trial GCS was different from all other coordinate systems; e) the dynamic GCS, static GCS, and CCS were all different from each other, and f) the CCS for the radial head was

Table 20: Input coordinates for simulated data.

	x	y	z
center of circle	0	0	0
center of capitellum	-2	0	0
center of trochlear groove	-2	2	0
center of shaft (humeral)	-2	0	2
humeral ball 1	0	0	4
humeral ball 2	0	4	0
humeral ball 3	0	-4	0
humeral ball 4	4	0	0
center on the backside of the "implant"	1	0	0
radius ball 1	5	0	0
radius ball 2	5	2	0
radius ball 3	5	2	2
radius ball 4	5	-2	2
radial head ball 1	0	-3.0303	2.611
radial head ball 2	0	2.4646	3.1505
radial head ball 3	0	0.92929	-3.8906
radial head ball 4	0	-2.8687	-2.7876
any point on rim	0	4	0

different from the CCS for the humerus. (Cases c and f only had practical interpretation for the native head case as the static trial and the radial head CCS do not apply to the tracking of the implant.) The expectation is that the anterior/posterior (a/p) and medial/lateral (m/l) position would be of the center of the radial head, and the third component (distraction) would be at -2, by design of the simulated data points. The results are shown in Chapter 9.0 .

8.1.3.1 Validation with Spicatek system

A more realistic model was created and actual data were collected using a three-camera Spicatek motion capture system in the Shoulder Laboratory of the Musculoskeletal Research Center. The cameras were Adimec 100m, with 1004x1004 pixels each and were capable of capturing up to 50 frames per second. Data were collected at 30 Hz and the marker positions were tracked using the DMAS™ (version 5) software. This system tracks black markers on a white background. A four-degree-of-freedom micrometer table (Model 147-351, Mitutoyo America, Aurora, IL) was outfitted with markers to simulate both the native head and the implant case. The stage was covered with white felt to optimize the contrast with the black markers.

To represent the implant-tracking case, a metallic radial head implant with precision-machined circular divots around its rim was painted white. (Industrial Eco-Guard latex enamel, Krylon Products group) After drying, the white paint was carefully removed from the divots only, which were then filled with a black paint (Black Washable Poster Paint, Palmer Paint Products, Troy, MI). A set of four larger (9.5 mm-diameter) acrylic spheres were spray-painted black (Industrial Eco-Guard latex enamel, Krylon Products group) and affixed via threaded rods to the side of the micrometer stage to simulate the humerus markers visible during the tracking of the implant. Figure 33 shows the entire setup and Table 21 shows the markers required to be tracked for each file type.

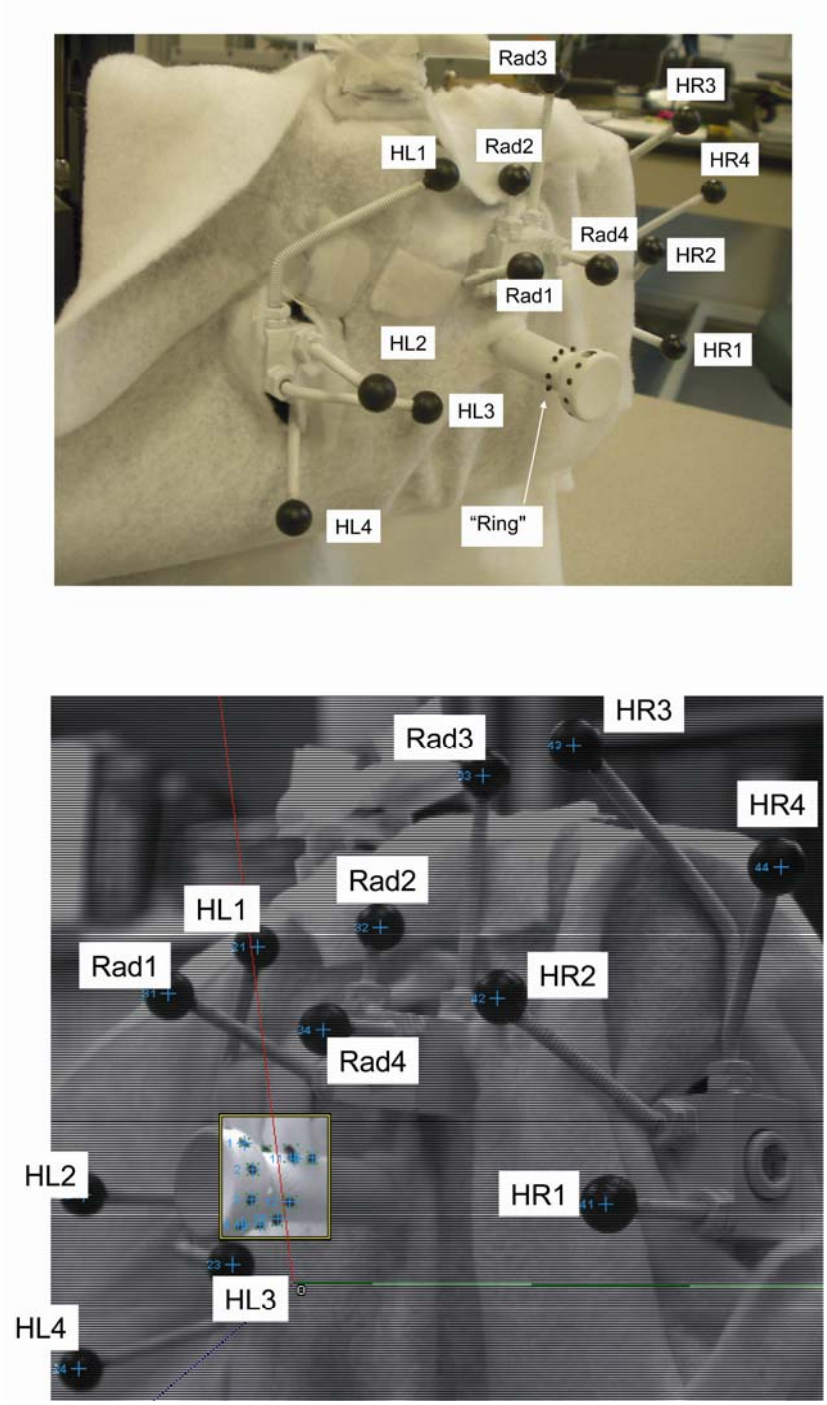
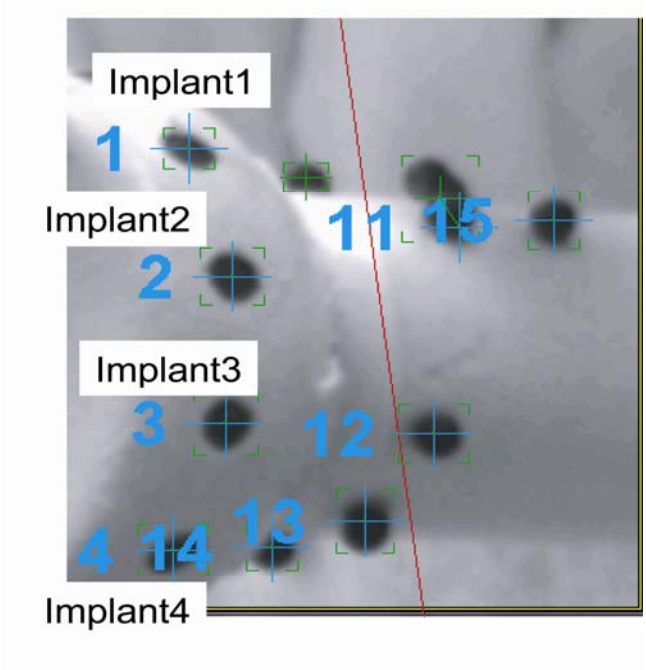


Figure 33: Markers used with Spicatek System.



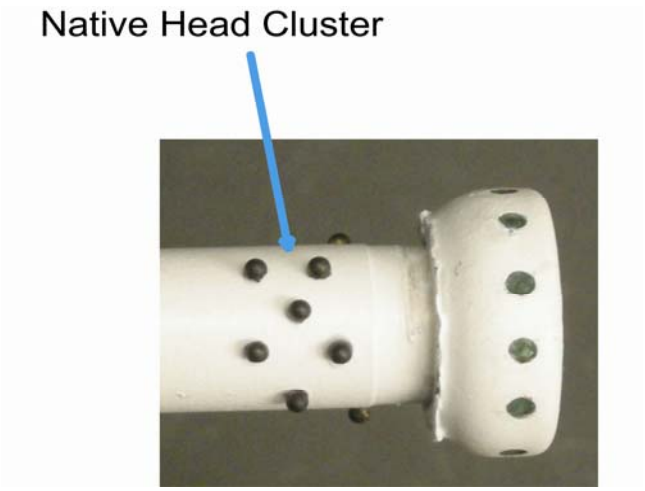


Figure 34: Markers for simulated native head and implant.

Table 21: Markers that must be tracked for each case.

	Static	**Dynamic**
Implant tracking	Implant "dots"; Humerus markers	
Native head tracking	Radial head markers; Distal radius markers	Distal radius markers; Humerus markers

To simulate the native head case, six 1.58 mm diameter brass balls were spray-painted black and fixed in a cluster formation (Figure 33, and Figure 34 in detail) to a circular rod with cyanoacrylate adhesive (Super Glue, Super Glue Corporation, Rancho Cucamonga, CA) to simulate the small markers that will be put on the cadaveric radial heads. Four larger (9.5 mm diameter) markers were affixed to the nonrotating part of the stage to simulate the humerus markers that will be visible during cadaver specimen studies, and another four of the larger markers were affixed to the rotational part of the micrometer table. As a final test, four of the small markers were affixed to the same brass rod in a circular fashion around its perimeter ("ring", seen in Figure 33, top), which may more accurately represent what is feasible to do in cadaver specimens.

Static, translation, and rotation data trials were collected. Table 22 indicates how many trials of each case were collected. (Collecting equal numbers of all trial types was not feasible

Table 22: Number of trials collected with Spicatek system.

	static	**rotation**	**translation**
Implant	3	2	0
Native head (cluster)	3	2	1
Native head (ring)	2	1	0

due to time constraints.) For the static trials, no motion occurred. In the rotation trials, the radius, radial head, and implant markers rotated. For the translation trials, the entire assembly was translated in one direction. Note that for the translation case, the relative positions between markers did not change. The DMAS software was used to track the position of the markers. The system was able to automatically track the small dots on the implant and the small balls that simulated the radial head. However, the position of the larger markers, which simulated the radius and humerus bones, could not be tracked automatically. For these cases, manual tracking was used. The manual labor required reduced the effective sampling frequency to about 2 Hz. The dynamic trials varied from 2 seconds to 6 seconds in length, and the maximum motions were 2.26 mm and 12°30'. The marker position coordinates were saved and exported via Microsoft Excel prior to use by the custom Matlab program described above. Each trial was processed with the raw data, and again with the fourth-order Butterworth filter with a cutoff frequency of 1 Hz. The results are presented and discussed in Chapter 9.0 .

8.2 FINITE HELICAL AXIS TRACKING

The concept of a helical axis of rotation between two bodies is rooted in the fact that any finite rigid body transformation can be expressed as a rotation and translation along a single axis. This is the helical axis. [230] The tracking of the finite helical axis (FHA) of rotation during p/s motion is accomplished by measuring the motion of the ulna, radius, and humerus bones. Arrays of four markers each must be attached to each of the three bones of interest and the position of these markers must be tracked.

The computations necessary to observe the FHA are executed in MATLAB code (*Peak_CMM3_LK3_FHA_final.m*). All MATLAB code is included in Appendix P and an extended schematic of this code is shown in Appendix O. The method described by Spoor and Veldpaus [233] is employed. To ultimately compute the FHA vector, along with the translation and rotation about that axis, several preliminary calculations must be performed. It is important to note that the preexisting MATLAB code was originally, as designed by Karol Galik, intended to be used for both radial head tracking and FHA calculation. To preserve the original structure of the programs, modifications were kept to a minimum to give a functional end result. For example, given that no radial head markers are necessary for the computation of the FHA, it is not necessary to collect data from radial head markers. However, the program requires data to serve as placeholders for the radial head data, and the radius bone marker data will serve this purpose. Thus intermediate results produced by the code, such as radial head position, will not have physical meaning. To this end, outputs pertaining to this information, such as plots, have been suppressed.

The actual computation of the FHA is done once the transformation matrix, including both rotation and translation, from the global to the anatomic coordinate system is determined. This 4x4 matrix, **T** (an input to the subfunction *screw_axis.m*)**,** is computed through a series of steps. First, the marker coordinates are transferred from the CCS and the GCS to an anatomical coordinate system by methods similar to those described above to track the motion of implants. Separate coordinate systems are created for the humero-radial, ulnar, humero-ulnar, and humero-radial coordinates. Note that not all of these are required to compute the FHA, though they are required for the current program to run without error. The same is then performed for the radial head, though the results lack physical interpretation since there are no actual radial head markers.

The data are filtered with a fourth-order Butterworth Filter at the user-specified cutoff frequency (*lines 363-384 of Peak_CMM3_LK3_FHA_final.m*) and, using the *soder.m* algorithm described above in conjunction with an algorithm based on the work of Grood and Suntay [107] various rotations are computed. The radial head coordinates on the capitellum are computed, but again have no meaning since there are no actual radial head markers. Finally, the FHA can be computed.

In general, FHA computation can be formulated as the Eigenvalue problem,

$$\begin{aligned} \mathbf{A}\hat{\mathbf{n}}_m &= \lambda_m \hat{\mathbf{n}} \\ m &= 1....M \end{aligned} \tag{7}$$

where **A** is the rotation matrix between the two bodies, $\hat{\mathbf{n}}$ is the eigenvector (a unit vector pointing in the direction of the helical axis), and λ is the eigenvalue. [105] As shown in Spoor and Veldpaus [233] the rotation angle, ϕ, about the helical axis can be computed using:

$$\phi = \sin^{-1}\left(\frac{1}{2}\sqrt{(R_{32}-R_{23})^2+(R_{13}-R_{31})^2+(R_{21}-R_{12})^2}\right) \tag{8}$$

or

$$\phi = \cos^{-1}\left(\frac{1}{2}(R_{11}+R_{22}+R_{33}-1)\right) \tag{9}$$

where R_{ij} are the components of the 3x3 rotation matrix which is itself an interior matrix of the transformation matrix **T**. (*These equations are implemented in lines 41-42 of screw_axis.m*) Numerically, Equation 8 is recommended when $\sin\phi \le \sqrt{2}/2$ and Equation 9 otherwise. The direction of the axis is given by the vector

$$\hat{\mathbf{n}} = \frac{\begin{bmatrix} R_{32} - R_{23} \\ R_{13} - R_{31} \\ R_{21} - R_{12} \end{bmatrix}}{\left\| \begin{bmatrix} R_{32} - R_{23} \\ R_{13} - R_{31} \\ R_{21} - R_{12} \end{bmatrix} \right\|} \quad \text{if } \phi < \pi/4 \tag{10}$$

$$\hat{\mathbf{n}} = \pm \frac{\mathbf{b}_i}{\sqrt{\max(\mathbf{b}_1^T \mathbf{b}_1, \mathbf{b}_2^T \mathbf{b}_2, \mathbf{b}_3^T \mathbf{b}_3)}} \quad \text{.....}$$

where $\mathbf{b}_i$ is the ith column of the following matrix:

$$\frac{1}{2}\left(\mathbf{R} + \mathbf{R}^T\right) - \cos\phi \mathbf{I} \tag{11}$$

(*Equations 10 and 11 are implemented on lines 36 and 55 of screw_axis.m.*) The amount of translation, *t*, along the helical axis can then be computed as:

$$t = \hat{\mathbf{n}}^T \begin{bmatrix} T_{21} & T_{31} & T_{41} \end{bmatrix} \tag{12}$$

or

$$t = \sqrt{\begin{bmatrix} T_{21} & T_{31} & T_{41} \end{bmatrix} \begin{bmatrix} T_{21} & T_{31} & T_{41} \end{bmatrix}^T} \tag{13}$$

if ϕ is exactly equal to zero. (*See lines 59 and 69 of screw_axis.m.*) In Equations 12 and 13, the T_{ij} are the elements of the original transformation matrix, **T**, which represent its translation components.

8.2.1 Validation of FHA tracking method

To validate the MATLAB routine modified to only compute the FHA, a portion of the same data that was created for the Vicon accuracy study was used. Specifically, as described in Chapter 6.1.1, a static image of the mock elbow outfitted with markers was captured using the Vicon motion analysis system. The data points from one frame were collected and then rotated about an axis passing through the center of the capitellum and parallel to the ulna. This was used as the kinematic input. The expected result is a constant helical axis which is equivalent to the axis of rotation used to generate the simulated data.

9.0 ELBOW MOTION TRACKING: ALGORITHM RESULTS

In this chapter, the results to validate the MATLAB code as presented in Chapter 8.0 are presented and discussed. These results demonstrate the functionality of the radial head tracking code with simulated and collected data, as well as the utility of the finite helical axis code.

9.1 RADIAL HEAD TRACKING

Using the radial head tracking code with simulated data serves to illustrate that the results computed by the MATLAB routine are as predicted. The results from tracking both the implant and native head case indicate that tracking the radial head using the mathematical methods described may be feasible with the Spicatek system.

9.1.1 Tracking simulated data

The results from the simulated data described in Chapter 8.1.1 are as expected for cases a) through f). The center of the radial head remains exactly in the center of the capitellum as expected. An example of the results, from Case F, is shown below for both the implant (Figure 35) and native head (Figure 36) case. In both figures, the top image shows the capitellum (large circle) with the radial head travel plotted in the interior.

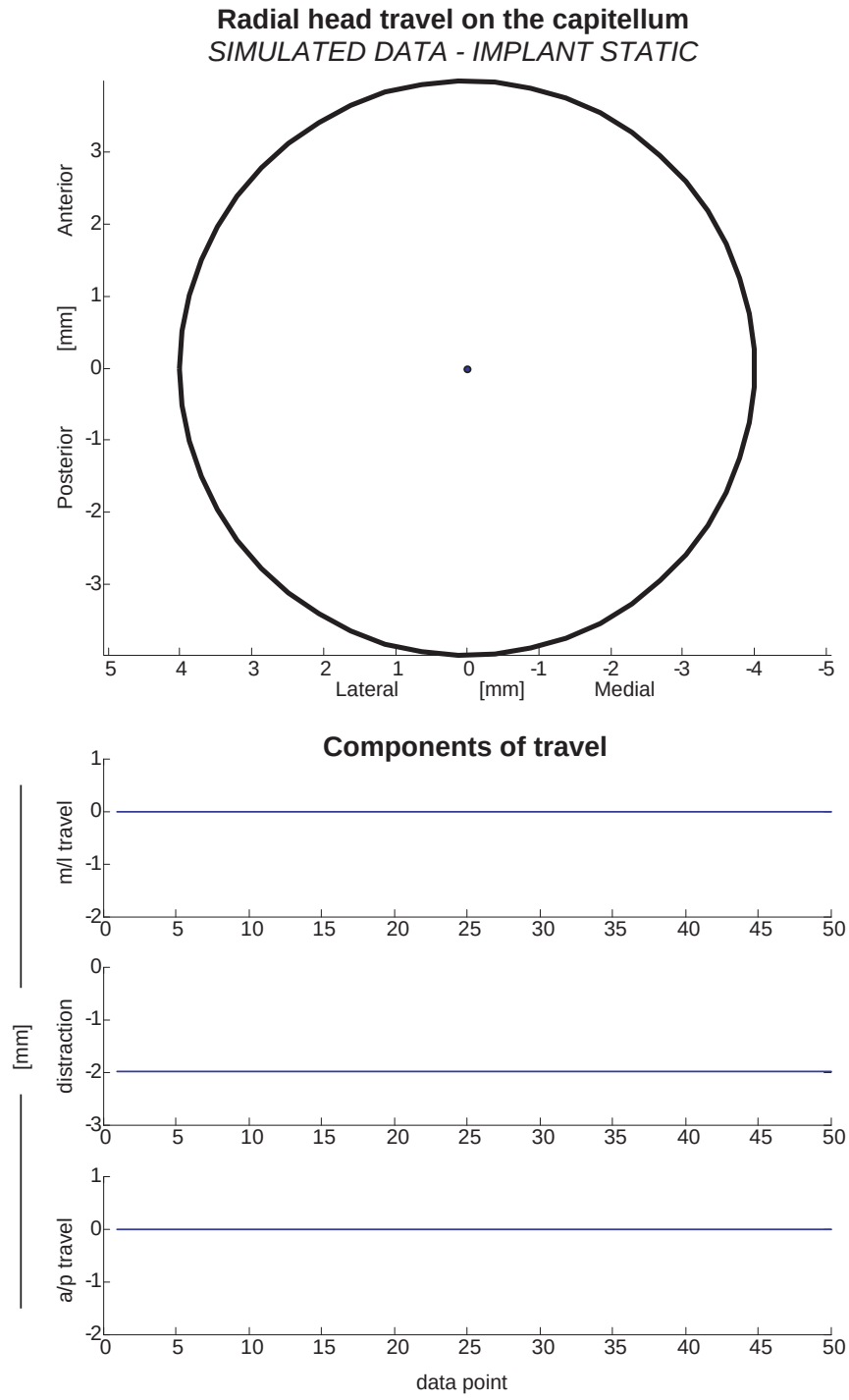


Figure 35: Implant travel from simulated (generated) data.

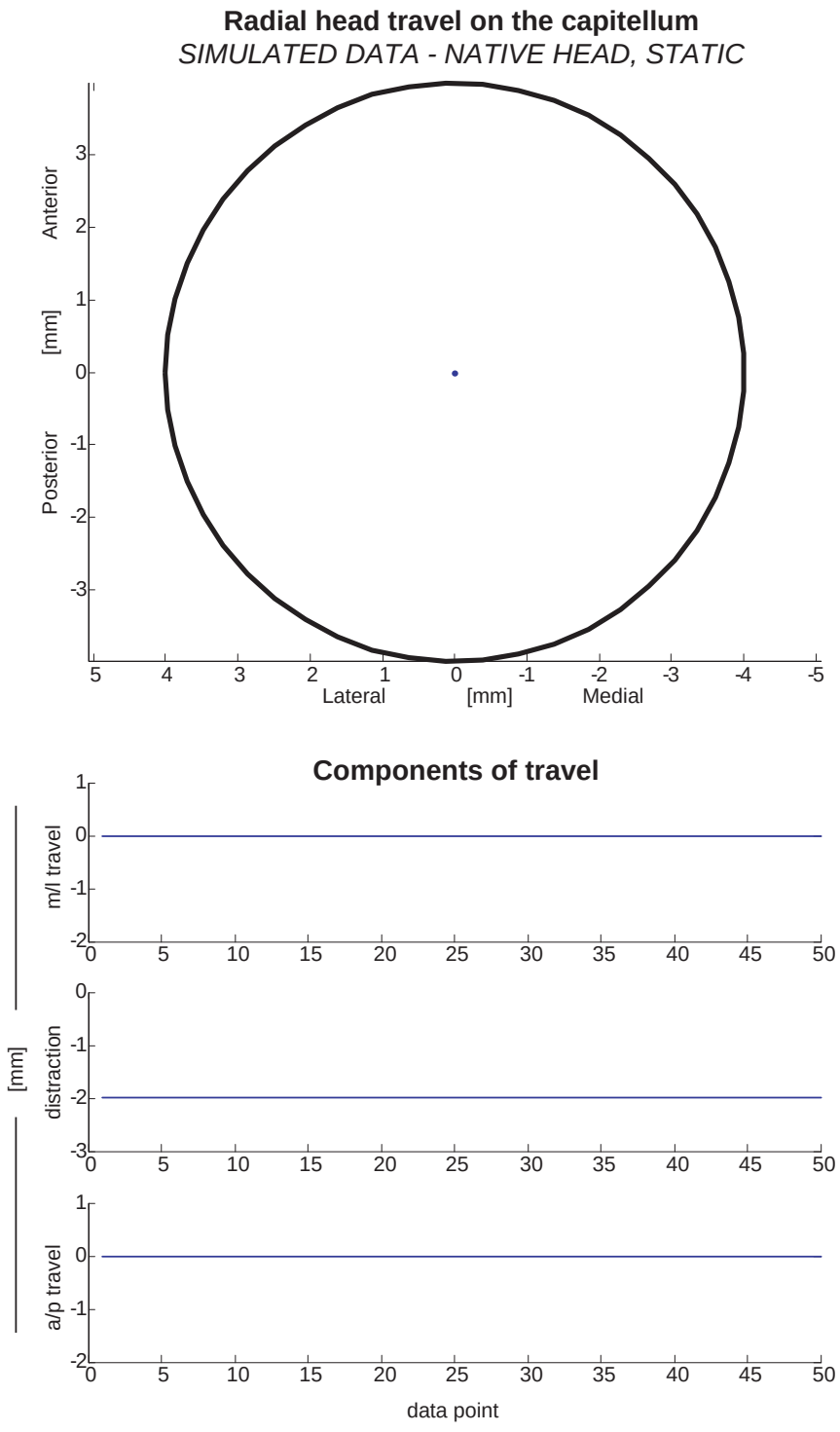


Figure 36: Native head travel from simulated data.

In these figures, the bottom images show the components of radial head position. As expected, the anterior/posterior (a/p) and medial-lateral (m/l) positions remain at the center of the radial head. The distraction, or out-of-plane, component is nearly 2. The slight difference from 2 is a result of conversions between units (mm and inches) that occur throughout the course of the processing. The success of these results indicates that the postprocessing code functions as expected. The results of all other cases of coordinate system permutations, as described in Chapter 8.1.3., were identical to those shown here.

9.1.2 Tracking implants using the Spicatek system

The first notable result from tracking the implant markers in the Spicatek system is the success of the circle-fitting algorithm, as described in Chapter 8.1.1.1. The center of the circle fitted to the visible implant markers (dots) during a static trial is shown in Figure 37. These are computed with raw (unfiltered) data. The range of values is extremely small – less than 0.1 mm in each direction, which indicates good precision of the system and robustness of the circle-fitting algorithm.

The next notable result from the experiments designed to simulate the implant case is the precision of the calculated results of radial head position projected onto the capitellum. This is shown in Figure 38. As with Figure 35 and Figure 36, the large black circle in the top image represents the capitellum and the radial head travel is plotted in its interior. The bottom portion of the figure shows the components of radial head travel. The a/p and m/l components are expected to be at the exact center of the capitellum, and the distraction component is expected to be a constant some distance away, approximately 52 mm. The a/p component is very close to the expected value of zero. The m/l component is incorrect by about 1 mm. This could be due to

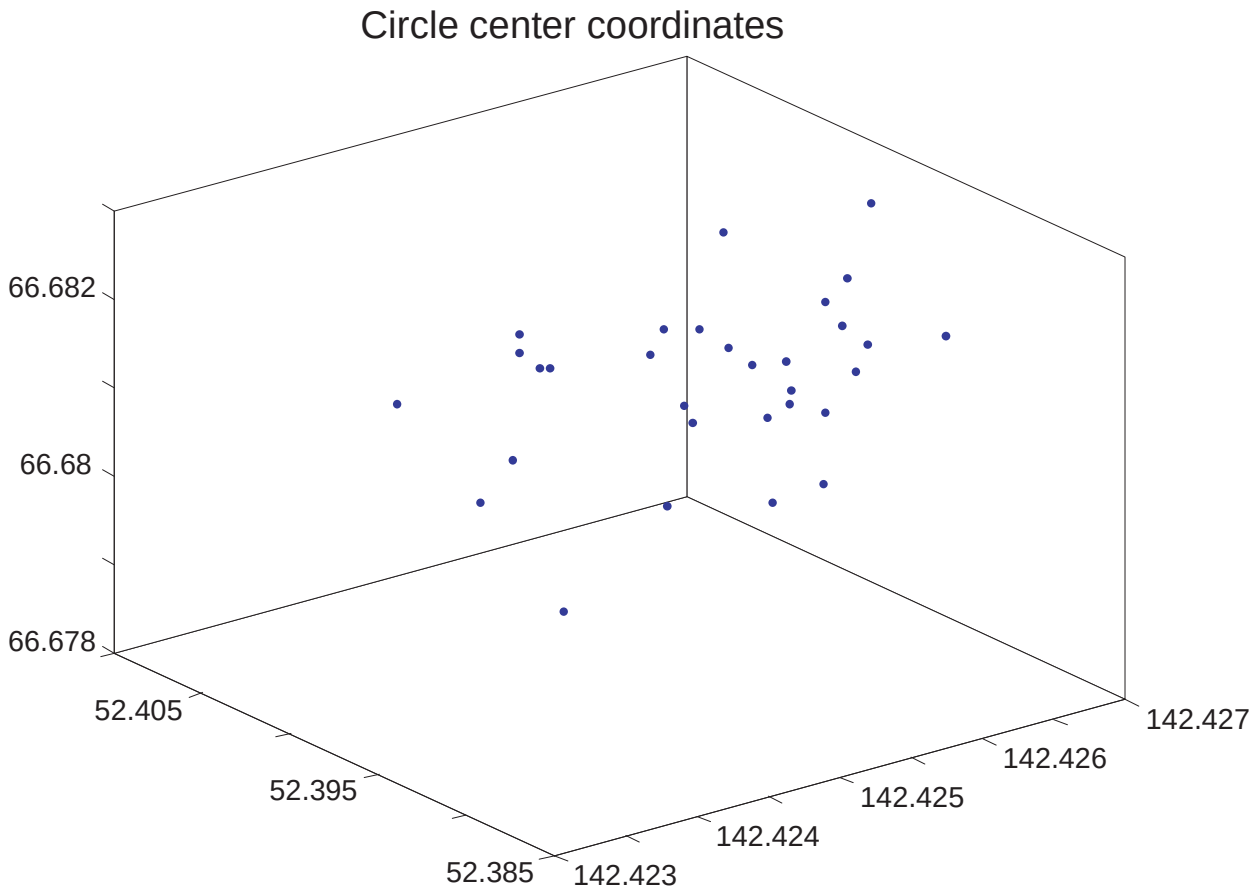


Figure 37: Fitted circle center coordinates (units are mm) of an implant as collected by the Spicatek system.

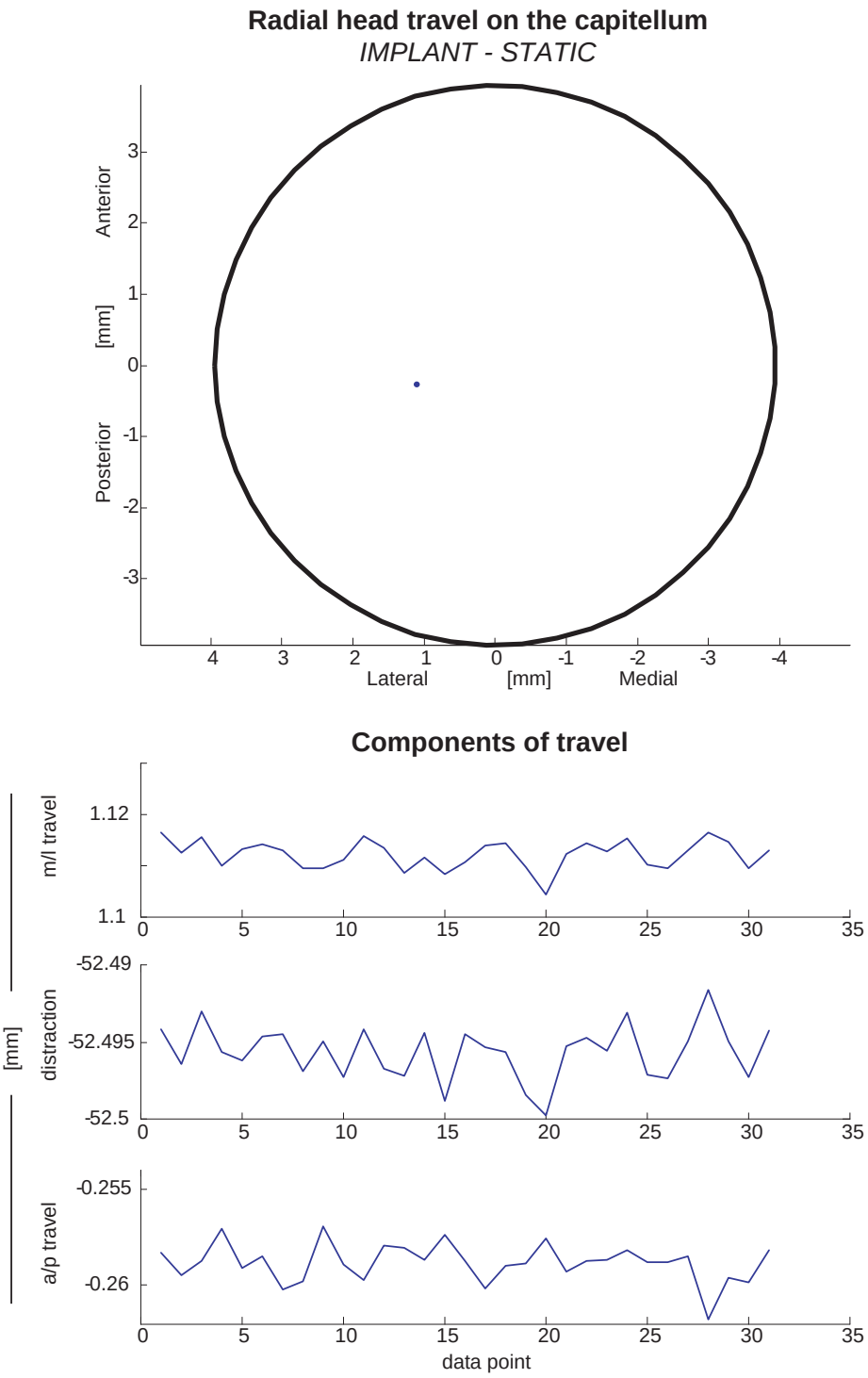


Figure 38: Position of an implant tracked using Spicatek system.

computational errors within the Spicatek system, or may have been influenced by the sub-optimal lighting or background conditions during the data collection. Future studies can investigate these topics. It is also important to recall that the center of the capitellum relative to the larger "humerus" markers was computed from the positions of the large markers. These markers could not be automatically tracked by the Spicatek system due to their size and other issues discussed below. Since manual tracking was used, which involves the user determining each marker's center, it is likely that errors in marker position estimation occurred. That said, the overall results are encouraging: the center of the implant, based on automatically-tracked markers, did not incur large spurious motion, and the projection of this center on to the capitellum, while not exactly as expected, was close.

Figure 39 shows the results from an implant rotation trial. The expected result is for the path of the radial head to be exactly a point in the center of the capitellum. While the unfiltered results are scattered, the filtered results are nearly as expected in variability, even if the position is not perfectly at the center of the radial head.

9.1.3 Tracking the native radial head with the Spicatek system

The tracking of the native radial head case was similarly successful. Figure 40 shows a representative example of the end results using both the raw and filtered data collected during rotation. Note that, for the tracking of the dynamic markers for both the cluster and the ring of "native head" markers, the observable markers are all large-diameter and were thus manually tracked throughout the motion. The left-hand images in Figure 40 show the results processed from the raw data from the cluster of markers. There is some dispersion in the radial head

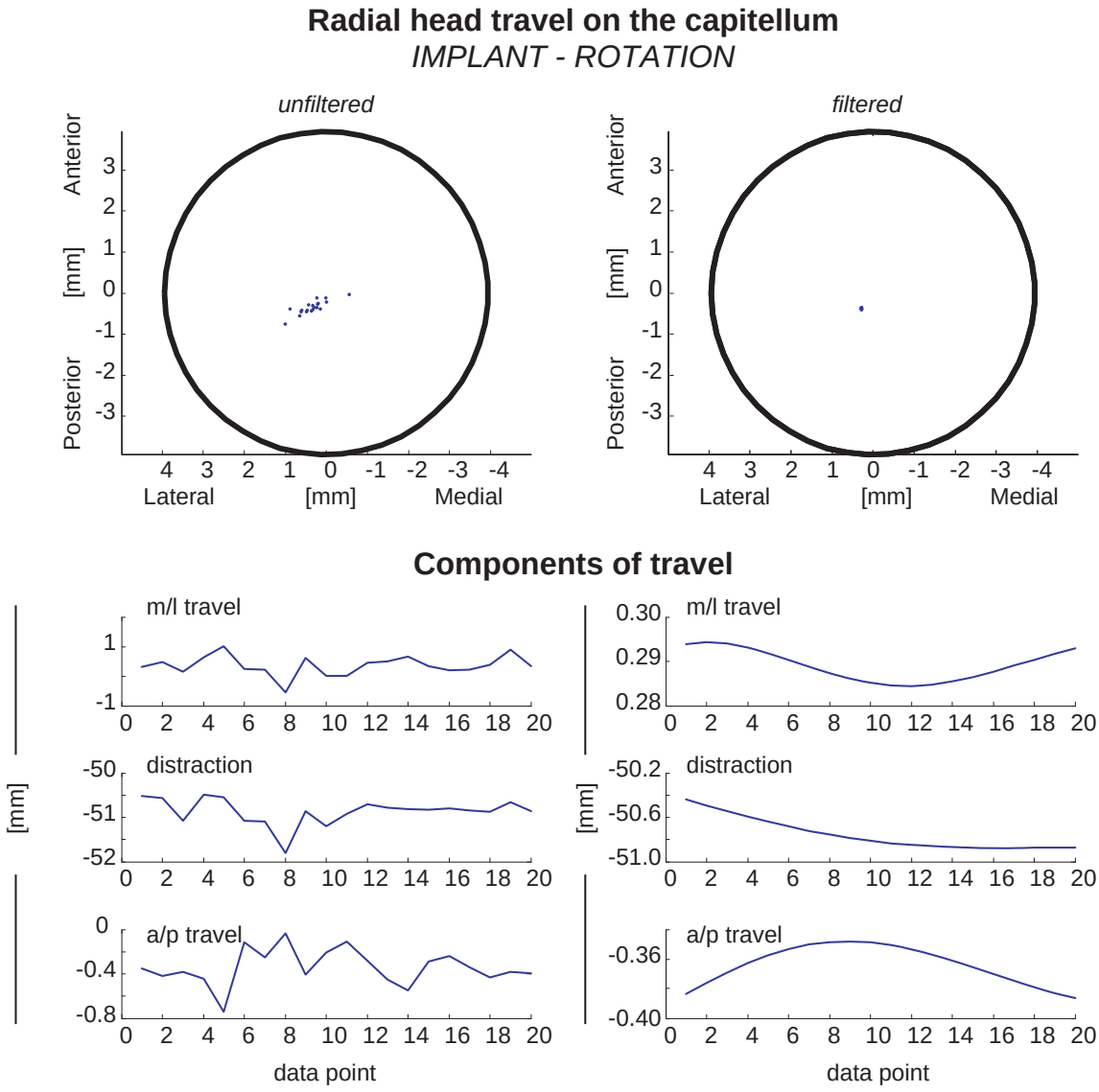


Figure 39: Implant position during rotation as tracked using the Spicatek system.

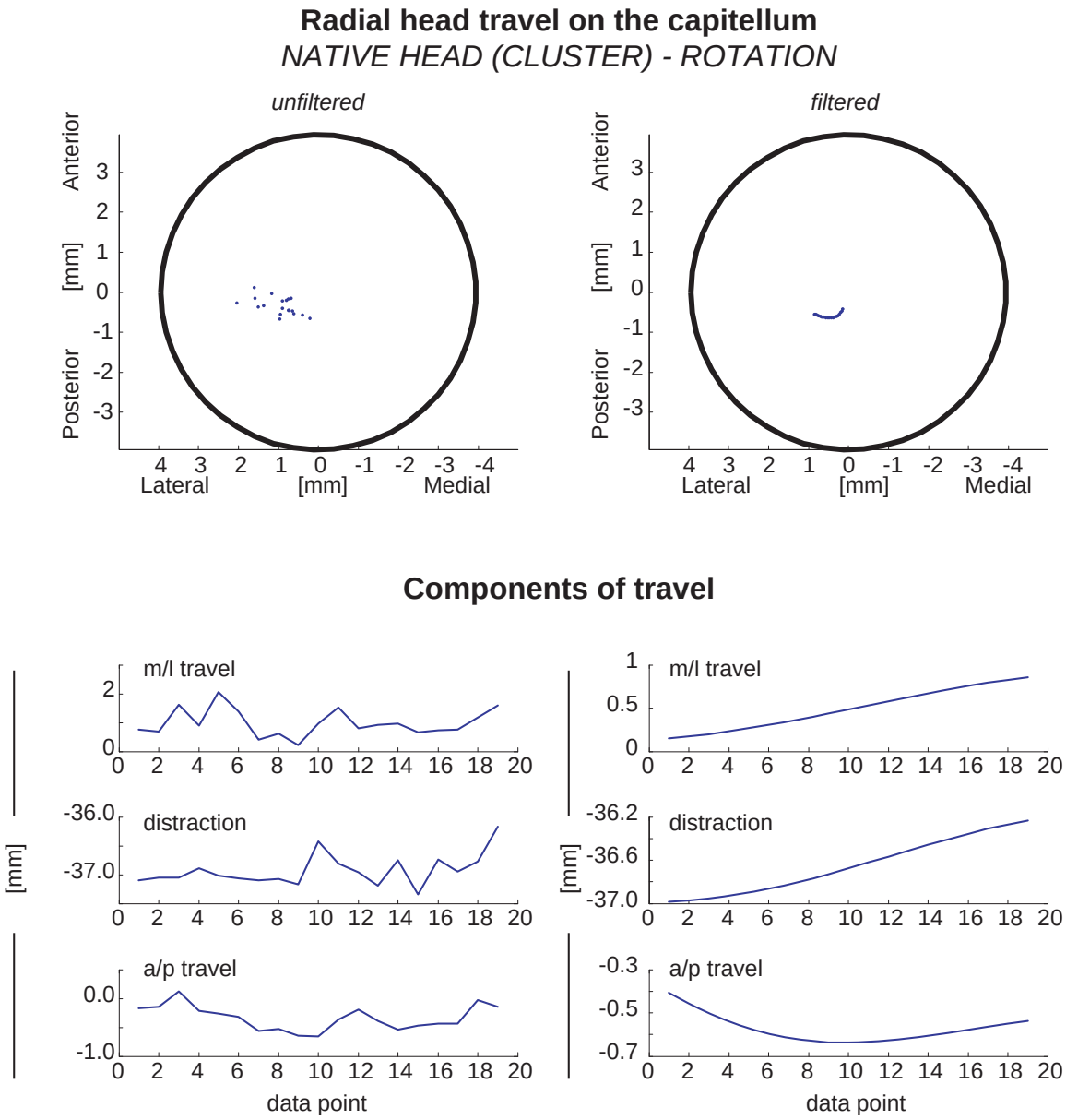


Figure 40: Native head cluster position during rotation as tracked using Spicatek system.

position. Again, this may be induced by the manual tracking of the larger markers. Also as above, the expectation is that the m/l and a/p components of the radial head position to always be at the center of the capitellum. However, the variation in radial head position is dramatically reduced with the filtering, as indicated by the plots on the right-hand side of Figure 40. The arc of motion is of the shape expected if the center of rotation was not aligned perfectly with both the center of the capitellum and the center of the radial head. It is also possible that, due to the low effective sampling frequency, the true motion is masked, though this is unlikely given the low movement frequencies, as mentioned in Chapter 8.1.3.1. Again, the manual tracking of the pertinent markers undoubtedly contributes to the disagreement between the expected and actual results. Figure 41 shows the results from the raw data collected during a translation. Recall that during translation, the entire assembly moved and thus the radial head position is expected to remain in the center of the capitellum. These data could not be successfully filtered due to the limited number of data points that were manually tracked.

The results using the "ring" of markers to simulate the native radial head are similar to those with the cluster of markers and are shown in Figure 42. The expectation was that the radial head position would be conincident with the center of the capitellum in the a/p and m/l plane. Again, there is dispersion in radial head position when computed using the raw data (on the left) that is reduced with filtering (on the right). As with the cluster of markers, it is possible that the center of the radial head was not perfectly aligned with the center of the capitellum, which was the center of rotation of the micrometer table. The manual tracking of the larger markers is the likeliest source of this error.

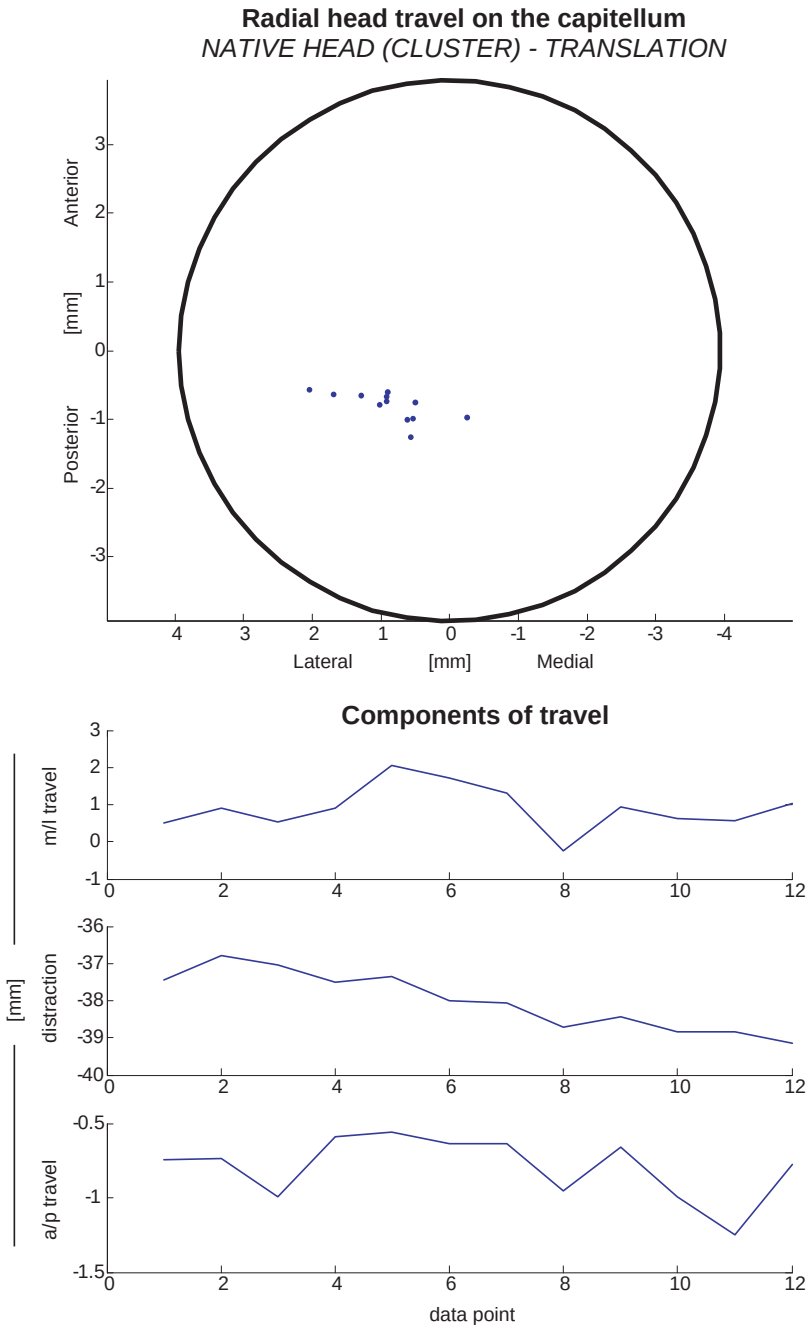


Figure 41: Native head position during translation as tracked using the Spicatek system.

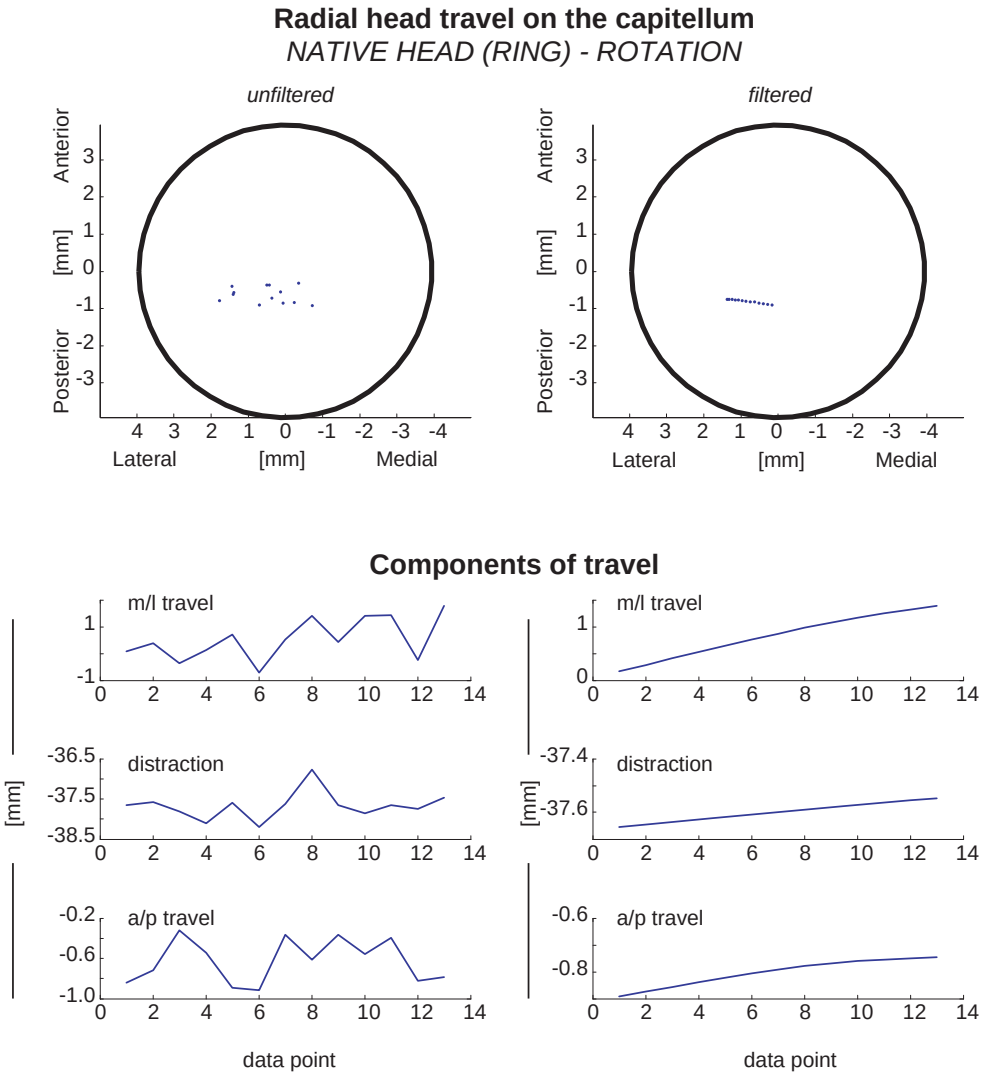


Figure 42: Native head position, computed from using a ring of markers, during rotation as tracked using Spicatek system.

9.1.4 Noise of the Spicatek system

One concern is the amount of noise in the estimation of marker positions. To assess this for the small markers, which were the only ones that were automatically tracked during these tests, the standard deviation of each marker's 3D position coordinates was computed. The results are shown in Table 23. The largest is 0.1138 mm in the x-direction of marker #3 during the trial "Native Head Static 1". Note that the x, y, and z designations are in Spicatek global coordinates.

Table 23: Standard deviation of marker position coordinates. Units are mm.

		marker					
		1	*2*	*3*	*4*	*5*	*6*
Native Head (cluster)	*x*	0.0077	0.0006	0.1138	0.0007	0.0005	0.0881
	y	0.0003	0.0003	0.0142	0.0003	0.0003	0.0334
	z	0.0374	0.0003	0.0364	0.0003	0.0002	0.0562
Native Head (cluster)	*x*	0.0535	0.0068	0.0288	0.0006	0.0032	0.0008
	y	0.0371	0.0010	0.0198	0.0003	0.0034	0.0006
	z	0.0209	0.0387	0.0091	0.0003	0.0382	0.0003
Native Head (cluster)	*x*	0.0011	0.1122	0.0040	0.0006	0.0009	0.1093
	y	0.0004	0.0149	0.0005	0.0003	0.0006	0.0129
	z	0.0004	0.0306	0.0221	0.0003	0.0004	0.0298
Implant Static 1	*x*	0.0014	0.0008	0.0008	0.0007		
	y	0.0011	0.0006	0.0006	0.0005		
	z	0.0007	0.0004	0.0003	0.0004		
Implant Static 2	*x*	0.0335	0.0548	0.0009	0.0118		
	y	0.0359	0.0599	0.0008	0.0034		
	z	0.0062	0.0140	0.0004	0.0526		
Implant Static 3	*x*	0.0080	0.0064	0.0264	0.0594		
	y	0.0039	0.0023	0.0291	0.0660		
	z	0.0021	0.0315	0.0075	0.0182		
Native Head (ring)	*x*	0.1059	0.0013	0.0013	0.0706		
	y	0.0136	0.0011	0.0011	0.0493		
	z	0.0270	0.0004	0.0004	0.0203		
Native Head (Ring)	*x*	0.0691	0.0012	0.0324	0.0007		
	y	0.0089	0.0010	0.0223	0.0005		
	z	0.0177	0.0004	0.0087	0.0003		

9.1.5 Discussion of results from simulated data and Spicatek system

In general, the method for tracking the radial head implant, including the implementation of the circle-fitting algorithm, is a feasible way to track the motion of radial head implants. The precision of the center of the circle fit to the implant data during the static trial is particularly noteworthy in that the . Additionally, the method employed to track the native radial head by tracking the motion of the distal radius relative to the humerus, and using a static image to relate the distal humerus coordinates to small markers on the radial head itself, is also feasible. The Spicatek motion capture system was not sufficient to automatically track the larger markers employed in this study. The accuracy in the experiments presented here is limited to the manual tracking of the big markers, which was itself limited the data to a slower effective sampling frequency due to the costs of the manual labor. However, the automatic tracking of the smaller markers shows promise for use in radial head tracking experiments.

In consultation with a representative from the Spicatek company, it has been recommended that future tests use smaller markers (4 mm) in lieu of the larger 9 mm markers employed herein. In the conducted experiments, the large markers had two detrimental effects: 1) they were larger than the size threshold permitted by the tracking software and 2) the reflections from the ambient light in the room created white pixels in the middle of some markers despite the flat black paint used (the pixels should have been black). The size threshold imposed by the software is set to maintain sufficient accuracy when using the system. Additionally, one camera was slightly out of focus which may have affected both the calibration of the system and the automatic tracking results. It is also recommended that future studies employ a white-colored backdrop behind the entire experimental setup. Such a backdrop was not included in the collected experiments on the advice of the Shoulder Lab personnel who deemed the ambient

background sufficient. Despite these difficulties with the experiment presented here, the Spicatek system has the potential to be used successfully to track radial head motion.

9.2 FINITE HELICAL AXIS TRACKING

The implementation of the MATLAB program to compute the FHA is similarly successful. The results from the computation from simulated data are shown in Figure 43. The image shows the insertion of the FHA on to the capitellum, and Figure 44 shows a sketch of the axis itself in relation to some anatomic landmarks. In that image, the three stars connected by green lines in the rear of the image indicate (from left to right) the center of the capitellum, the center of the humeral shaft, and the center of the trochlear groove, which are used to define the anatomical coordinate system. The black circle represents the capitellum. The blue dot at the front right-hand side of the image represents the tip of the ulna and the nearby green triangle represents the distal radius. The pink (thick) line is the average FHA across all timepoints, and the blue (thin) lines show the FHA computed for each timepoint. Note that in this case, all of the thin blue lines coincide.

9.2.1 Discussion

This simple data shows that the FHA computation method works as expected when implemented by the MATLAB code. This is not surprising given that the structure was heavily based on previously-generated code and minimal modifications were made. However, this fact also contributes to the excessive length and inefficiency of the code itself. Given that the program is

not computationally intensive and can complete a run in just a few minutes, it is not prudent to invest additional time into further modifications.

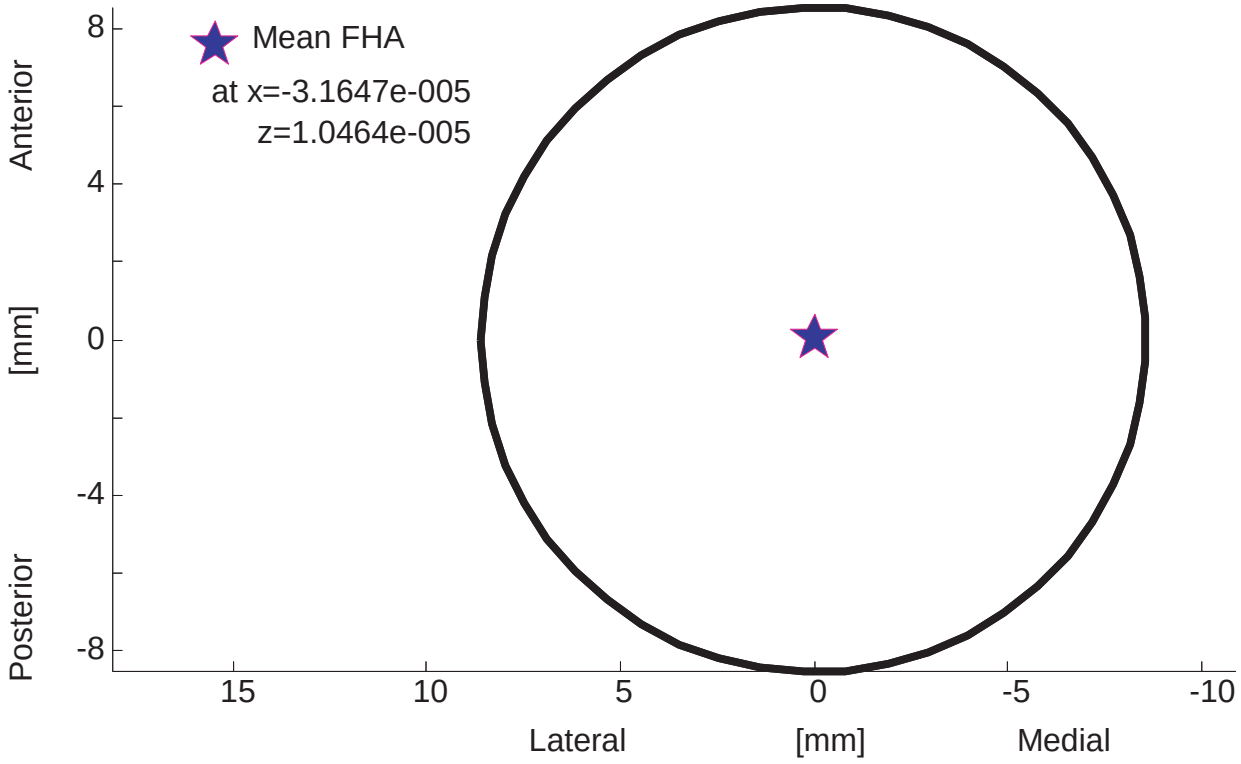


Figure 43: FHA intersection with the capitellum.

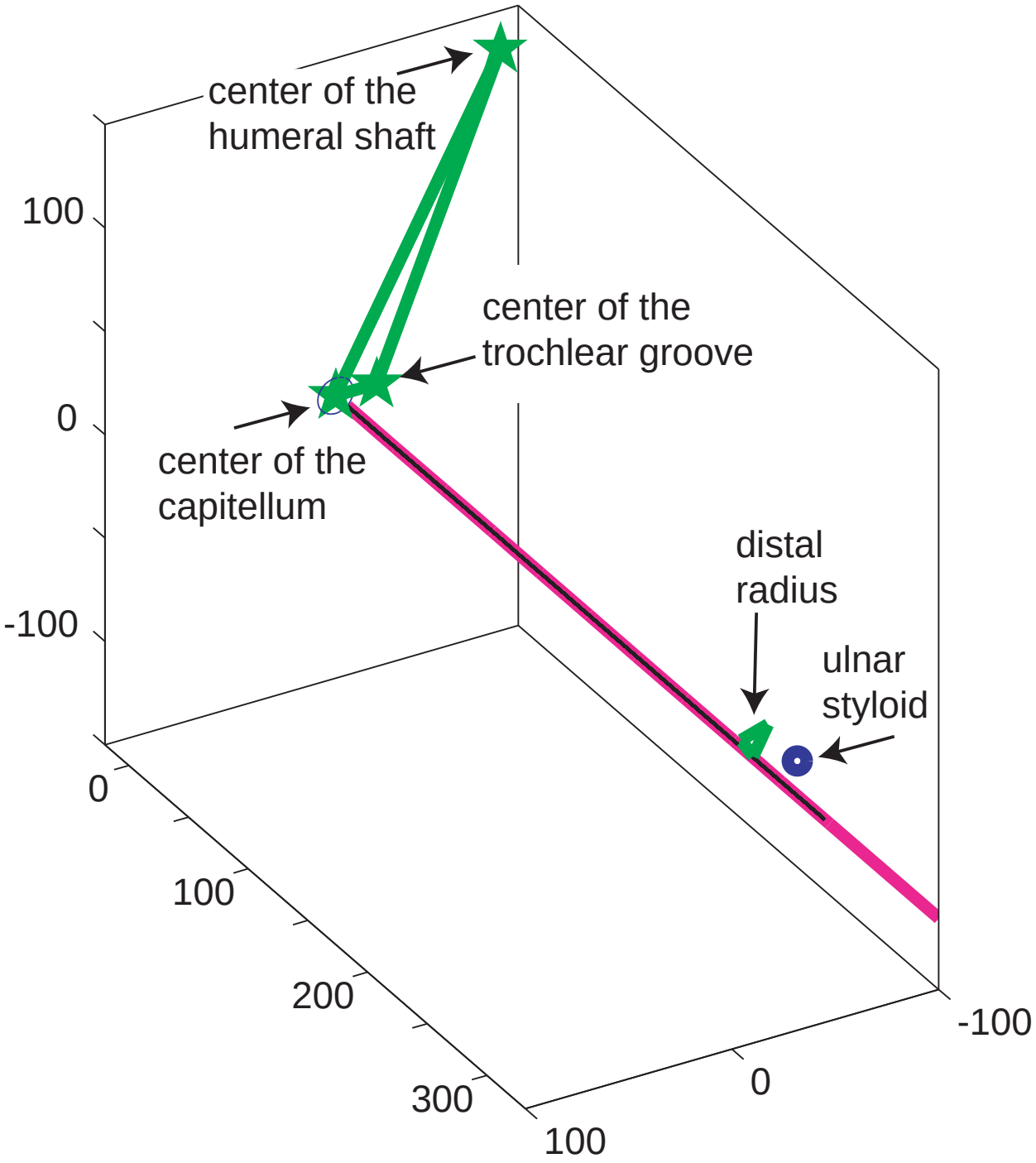


Figure 44: FHA with respect to anatomic landmarks.

10.0 DISCUSSION AND FUTURE WORK

This document has presented work in three distinct steps in the development of the AGH Elbow Simulator: i) simulator validation via the measurement of moment arms in cadaver specimens (Chapters 4.0 and 5.0); ii) motion analysis (Chapters 6.0 and 7.0 , also used in Chapters 8.0 and 9.0); and iii) the development and validation of algorithms which can be used to track elbow motion (Chapters 8.0 and 9.0). Each of these components was a necessary step towards the long-term goal of tracking radial head movements.

10.1 DISCUSSION

The work presented with respect to the Vicon motion analysis system accuracy represents a significant contribution to the biomechanics literature. Scientific questions of interest drive technological innovation. In this case, the study was inspired by the desire to track the radial head in a way that did not impede its *in vivo* motion. Previous work [91, 176-178] used an array of large reflective markers to do the tracking. This physically limited the range of feasible motion for both the native head and the implants. Additionally, for implants in which the radial head is designed to move independently of the shaft, the weight of an array of large markers was sufficient to impede the implant's motion. Figure 45 shows an image of a KMI Katalyst implant with an array of large markers as used previously. Note the large size of the marker array

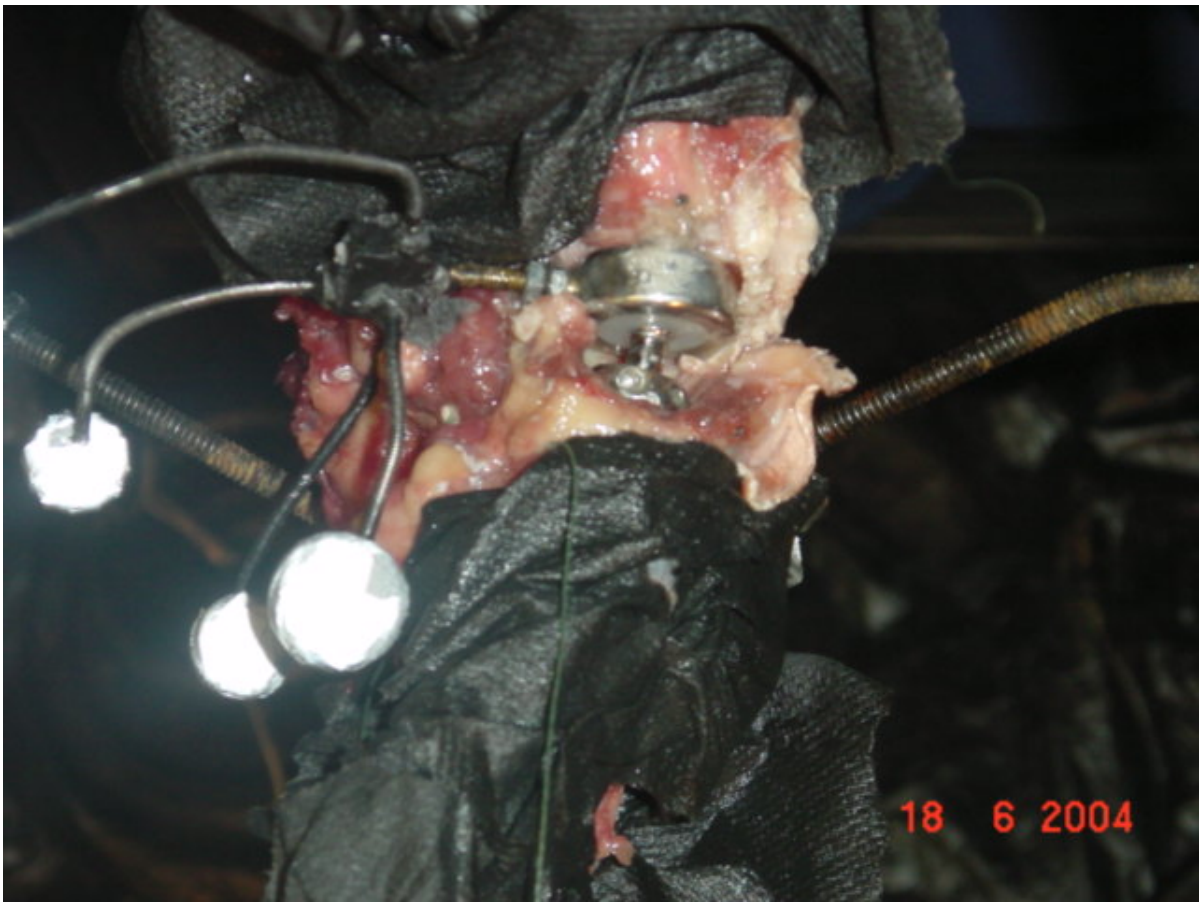


Figure 45: KMI implant with array of large markers.

relative to the size of the implant. A method to track extremely small markers positioned on the radial head, which would not interfere with its motion, was needed. The *camera-switching* phenomena is unavoidable with the packaged Vicon software and thus makes it unsuitable for the tracking of extremely small markers. It is also worth noting that wetness affects the reflectivity of all markers, [210] and this effect is debilitating with small markers. That is, when the small reflective markers get wet, as they will when used in a cadaver specimen, the cameras cannot see them. As discussed by Rainis, [210] tracking dark (black) markers on a light-colored background is a compelling reason to use a Spicatek motion analysis system. The use of

computer vision algorithms, such as those used in the DMAS software, is a significant advantage in that it avoids any problems caused by changes in reflectivity. More relevant to the camera-switching phenomena as discussed in this work, the tracking algorithms used in the DMAS software permit user interaction and thus camera-switching can be avoided. Based on the prevalence and uncontrollability of camera-switching in the Vicon system, having a motion analysis system with the ability to control which cameras are used across different timepoints is critical to the goal of tracking radial head movement and the future development of the AGH Elbow Simulator. The work relevant to camera-switching, presented here in Chapters 6.0 and 7.0 , has been provisionally accepted to the ASME Journal of Biomechanical Engineering, pending successful revision of the text.

The measurement of moment arms is a significant contribution to the understanding of the biomechanics of the elbow as the variation of f/e moment arms with p/s position, and vice versa, has rarely been reported in the same specimens among and between muscles. While the number of specimens tested is small with only three specimens, distinct trends emerge. More importantly, the good agreement of the measured moment arms with those that have been previously-reported validates one of the design intents of the AGH Elbow Simulator. When published, this work will emphasize the intent to validate the simulator design goal of replicating physiologically-accurate moment arms across specimens.

The development of algorithms to track the radial head motion and the FHA furthers the preparations to use the AGH Elbow Simulator in the evaluation and comparison of radial head replacements. Without accurate motion tracking, the study of radial head implants cannot proceed. The validation of the moment arm aspects of the simulator removes uncertainty about

the realism of the joint actuation from the results that will be gathered when radial head motion is investigated.

10.2 FUTURE WORK

The work presented here sets the stage for successful experiments to track radial head implants in cadaver specimens. Once an accurate motion tracking system is acquired and validated, experiments can begin with minimal additional preparation. Uncertainties remain with regard to the specific mechanism of attaching the markers to the cadaver specimen. However, it is expected that using traditional arrays of large markers, such as those used in the previous work mentioned above, will suffice for markers on the humerus, the distal radius, and the ulna. The array on the ulna is required for the computation of the FHA. Affixing small black balls to nail- or pin-heads should permit easy insertion into the humerus closer to the radial head, as required to track the motion of radial head implants. This technique has been successfully used with reflective markers on the knee and hip in the laboratory. Inserting small nails with the heads painted black into the radial head just distal to the annular ligament should be sufficient for the native head markers. The robustness of the painting technique used to paint the radial head implant in this study may require further development. In particular, the black paint used to fill in the divots may smear when subjected to the damp environment of a cadaveric elbow. If this is indeed the case, alternate black paints will be employed. In-house options include using automobile touch-up paint, or dimensional fabric paint, which are both robust against dampness. The painting technique itself is time- and labor-intensive. However, it is anticipated that, with sufficient practice, the process will be more efficient. Additionally, if treated with care, each

implant should only need to be painted once as the paint should not need to be removed between specimen tests. If the in-house options are not sufficient, other techniques may be explored, such as coatings.

Another logical extension of this work is in simulation and modeling. Simulation of the two-degree of freedom elbow system with physiologically correct inertial, muscular and neural parameters can test theories and use the physical elbow simulator for validation or modification. A computer model would allow for realistic simulation of orthopaedic injuries and could be developed into a useful clinical tool. Work is currently beginning in the laboratory to create a model of the medial ulnar-collateral ligament as a clinically-relevant finite element model.

In addition, this work has many applications to the control of prosthetic and paralyzed limbs. Functional Electrical Stimulation (FES) has been extensively explored as a means to control paralyzed limbs. Many control strategies have been tried, including PID and Feedback Error Learning. [150] The results of the control work completed in the lab [218] could also be applied to the control of neuroprosthetic devices. There is ample room for future expansion of the controller programs to more accurately simulate the control strategies that live humans use. Various researchers seek to map the brain's command signals at the cortical level to gross motor outputs of limbs. [83, 93, 134, 242] The algorithms developed and knowledge gained in future studies using the AGH Elbow Simulator will contribute to more successful results in this arena. The application to neuroprosthetic devices could be extended to include other forms of manipulation aids, such as feeding devices (MySpoon, SECOM Co., Ltd., Tokyo, Japan) and wheelchair-mounted robotic arms, or the control of more human-like robots..

Another extension of this work includes augmentation of the hardware. Integrating Programmable Intelligent Computers (PICs) into the simulator will permit the ACR-8020

controller card to serve as the brain, delegating the force- or position-control for each actuator to its own PIC. This will improve the controller's speed and accuracy, as the same technique did with a foot simulator. [222] Another long-term goal is the use of the motion-analysis system in real time to monitor the position of the elbow specimen and the feedback of this information in real time as feedback to the controller. This may be possible with the use of a Spicatek system. A productive loop of theory and experiment should result from this future development with limitless possibilities for future research in the fields of prosthesis design, elbow control, and elbow kinematics, with contributions to surgical and rehabilitation protocols.

APPENDIX A

PHYSICAL ADJUSTMENTS TO THE FRAME TO ACCOMMODATE BOTH RIGHT AND LEFT SPECIMENS, FOR USE WITH FOUR MUSCLES

Tools required:

- Allen wrenches:

    6mm (for motors)

    5/32” (for pulleys)

    1/8” for top-level countersink screw.

- Adjustable wrench

- Small screwdriver

Before beginning, be sure that the power to the system is off, and disconnect the muscle cables from the ends of cylinders X and Y. To convert the frame between right and left elbows, four adjustments are required:

1) Switch biceps and brachialis actuators.

2) Move pronator actuator.

3) Biceps, brachialis, and pronator top level pulleys.

4) Pronator pulley assembly installation.

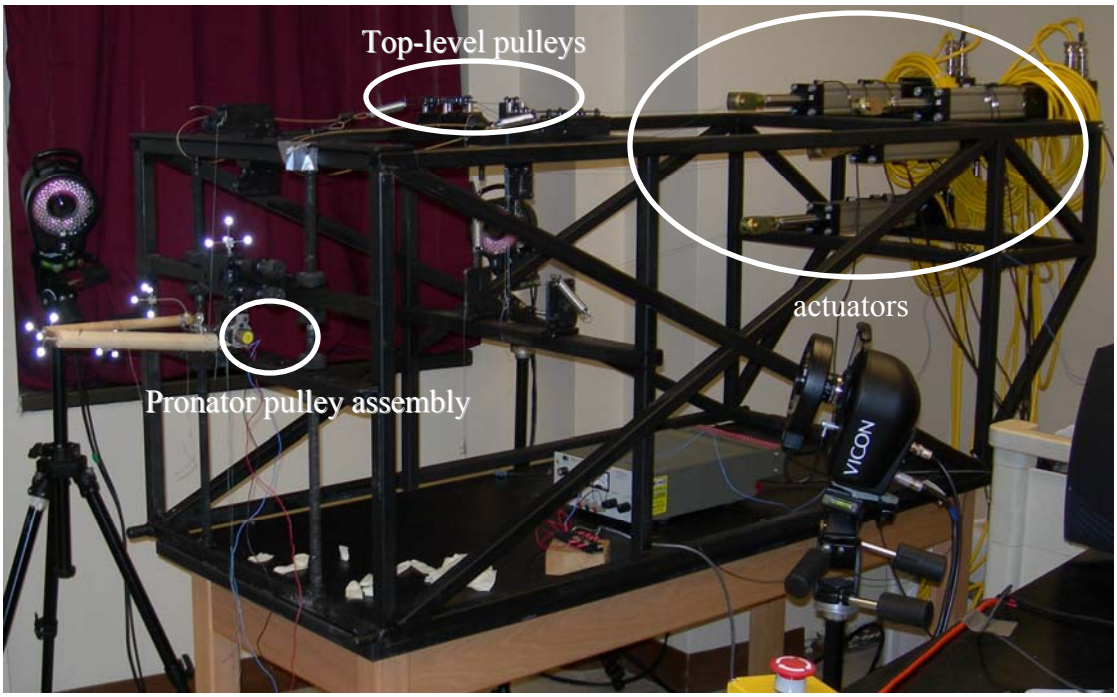


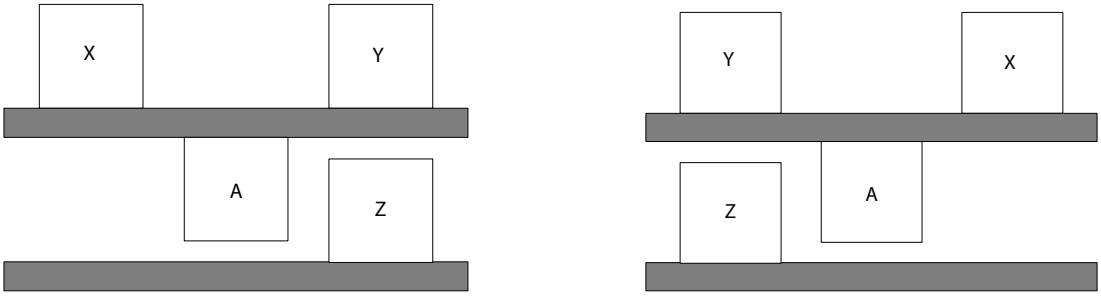
Figure 46: Overall view of the elbow simulator frame, configured for a right elbow.

1) Switch biceps and brachialis actuators. (Two people are preferred.)

- Using the 6mm Allen wrench, unbolt the cylinders from the frame and switch their positions. (See Figure 46 for final positions.)
- Reattach the muscle cables (or springs, if required) to the cylinders.

2) Move pronator actuator:

- As above, loosen the bolts that hold the pronator (Z) cylinder to the frame. Slide it over to the correct side, and bolt it down.



Physical adjustments to the frame to accommodate both right and left specimens, for use

3) Biceps, brachialis, and pronator top level pulleys: Adjust the top-level pulleys such that the brachialis pulleys (actuator Y) are the taller set. See Figure 49.

- Remove the four screws on both the brachialis and biceps pulleys.
- Move the silver extensions, and longer screws, to the correct side.
- Tighten all removed screws.
- Remove the pulley attached to the pronator (Figure 46) and install it on the correct side.

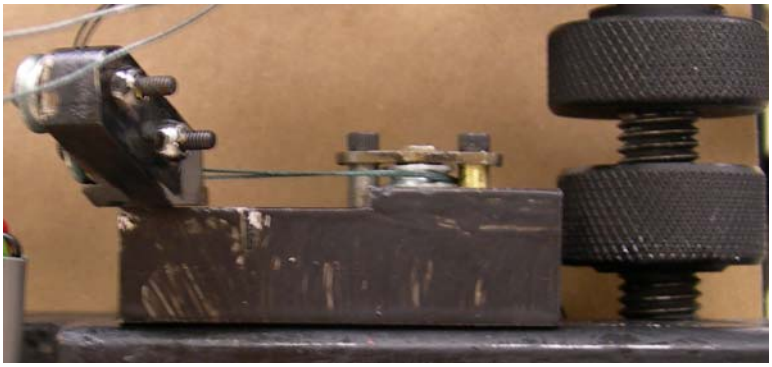
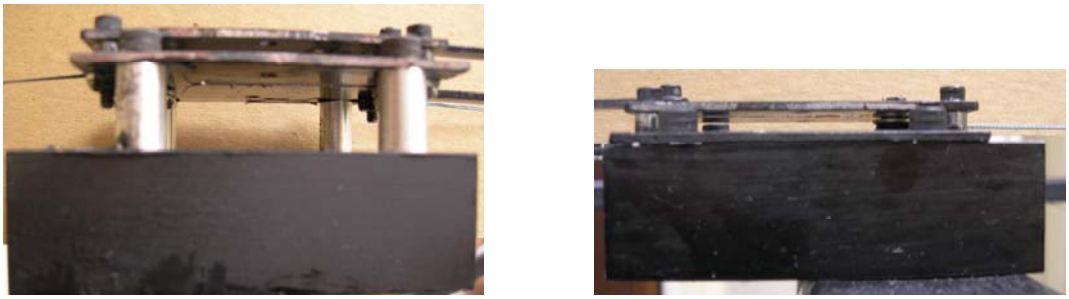


Figure 48: Photo of installed pronator pulley assembly for a right elbow.

4) Pronator pulley assembly installation: Remove the current pulley assembly using the allen wrench and the adjustable wrench. Install the correct one with the same tools. (See Figure 48.)



(a) **(b)**

Figure 49: Top level pulleys (a) tall and (b) short.

APPENDIX B

ELBOW SIMULATOR WIRING INSTRUCTIONS

An overview of the simulator's wiring is shown at the end of this appendix in Figure 51.

In addition, there are other components that contribute to the system. Figure 50 shows the required connections to integrate two potentiometers in to the system. For use with elbow experiments similar to those proposed in this thesis, the potentiometer which measures flexion-extension is "pot. 1" and that which measures pronation-supination is "pot. 2".

It is also important to note that there are two power strips that control the power to the system. Table 24 lists which items are connected to each. Also note that the two power strips, and all other electrical components, share a common ground via a star-grounding strategy implemented with a series of banana-plug cables.

Table 24: Power strip configuration.

Power strip 1 (“Computer”)	Power Strip 2 (“Motors”)
Computer Monitor 24V power supply (for ACR8020 card and limit switches) 4V power supply (for potentiometers, as needed) Power strip 2 Banana plugs for star-grounding Power strip for the load cell conditioners	Plugs to 4 motor drives

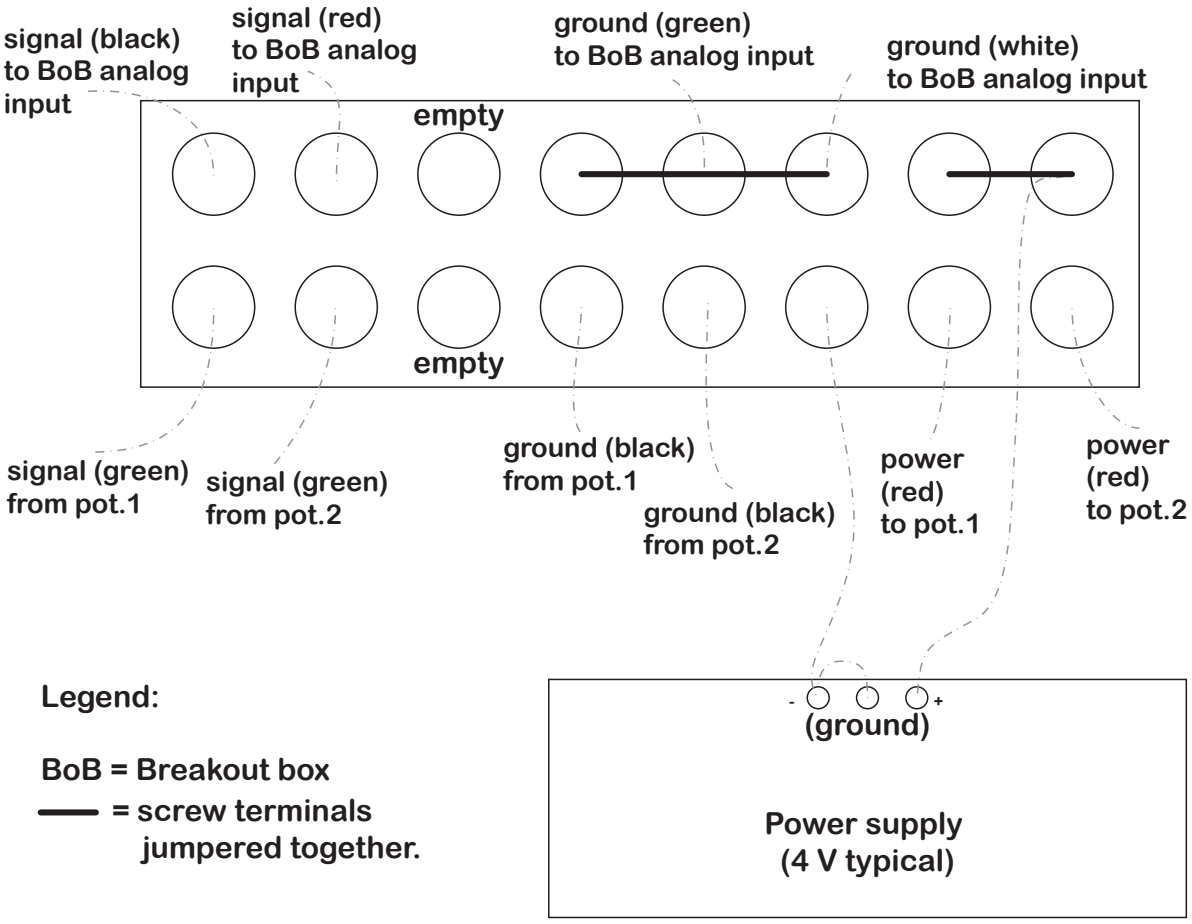


Figure 50: Terminal strip connections to integrate two potentiometers into the system.

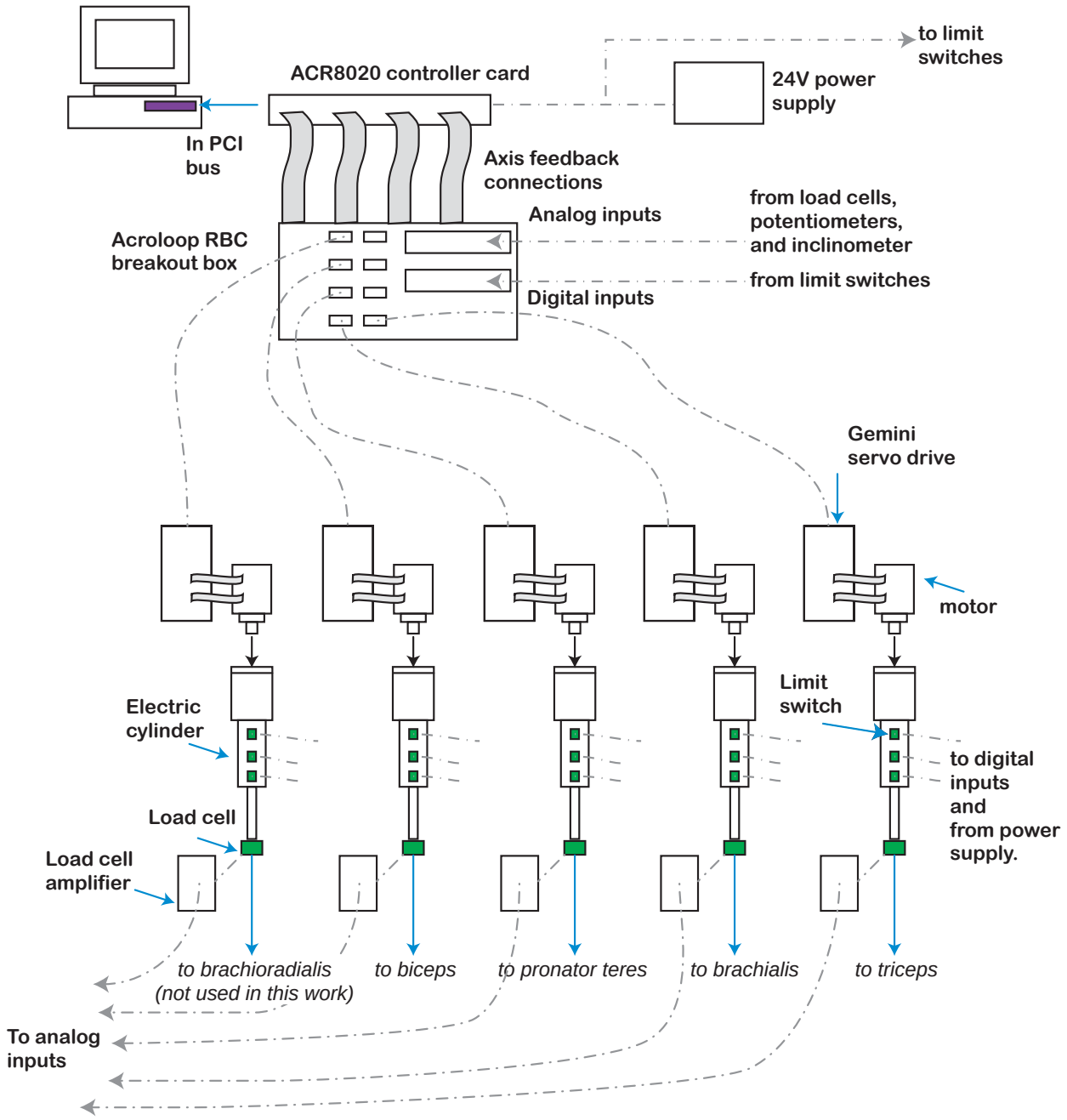


Figure 51: Overall simulator wiring.

APPENDIX C

ACR-VIEW SOFTWARE OPERATION

This document gives an overview of the ACR-View software operation, including the proper order to power up the elbow simulator system.

1) Turn on computer power strip. IMPORTANT: Do NOT turn on the motors' power strip until after the project has been downloaded to the controller.

2) Turn on the computer and log in.

3) Start the ACR-View program

Start ->ACR-View program.

4) Choose an existing project to open OR name a new project at the screen shown below.

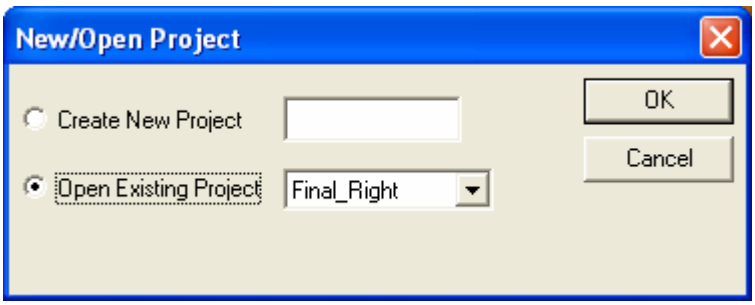


Figure 52: Open Project dialog box.

5) In ACR-View, click “connect”

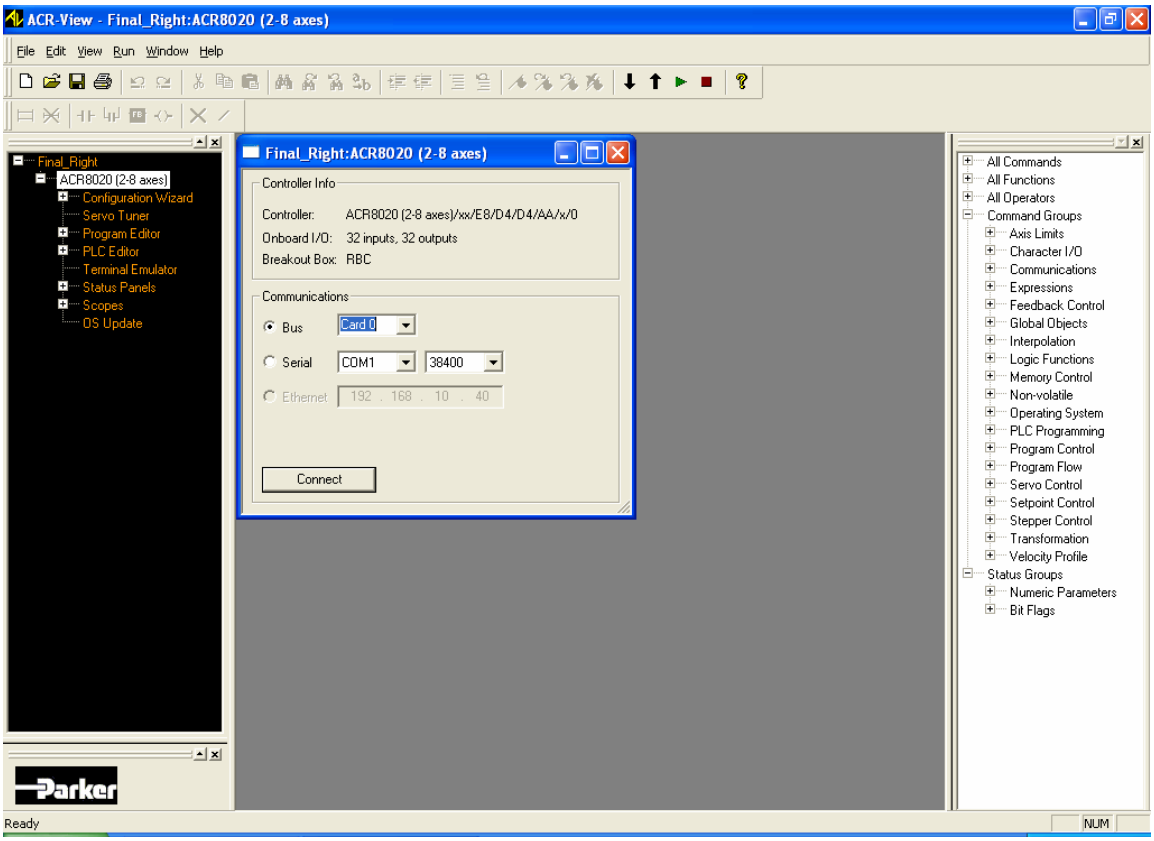


Figure 53: ACR-View initial screen.

6) If using an old project, simply download the configuration and programs.

To do this, click on the down arrow ("Download Project")

To download the configuration, choose:

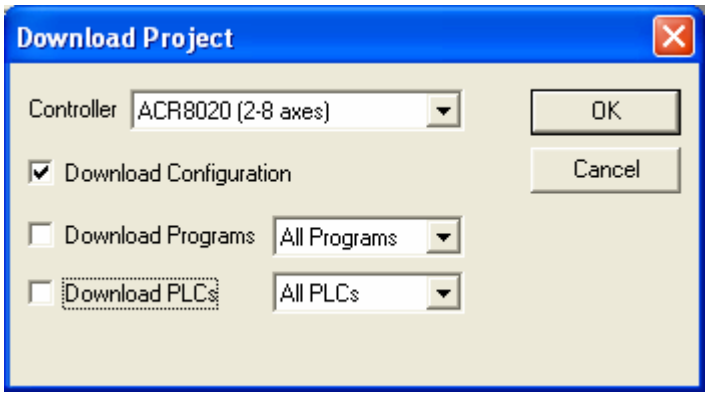


Figure 54: Download configuration dialog box.

To download the programs choose:

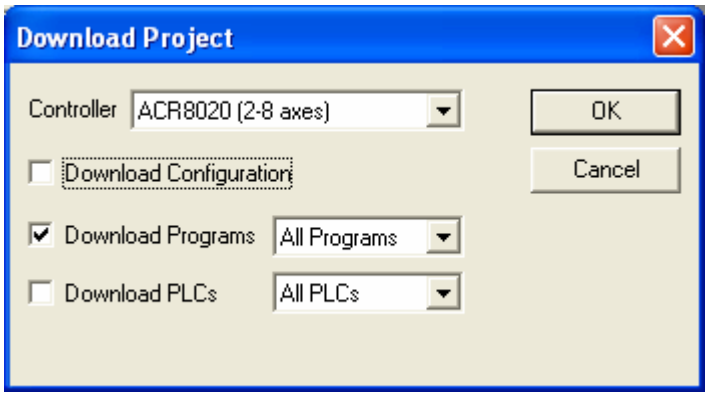


Figure 55: Download programs dialog box.

NOTE: It is not recommended to download both the configuration and the programs simultaneously – this can make ACR-View crash, or give an error about not detecting the controller.

7) If starting a new project, click on the “Configuration Wizard” in the left-hand side black box. (If using an old project, skip to step #13.) The following figures show the appropriate settings.

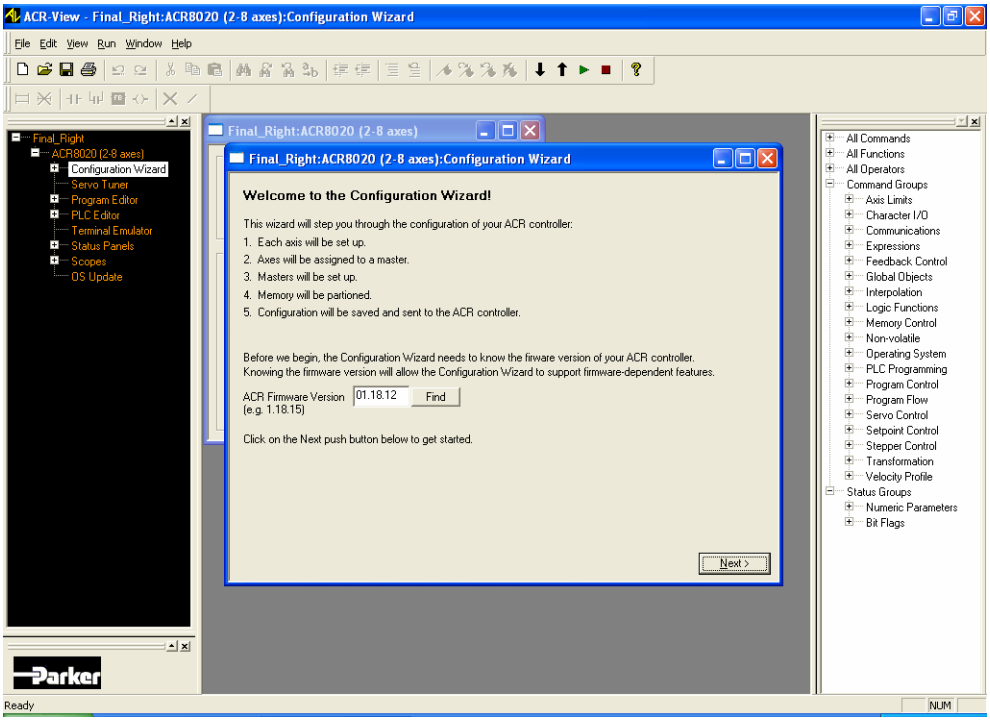


Figure 56: Configuration Wizard initial screen.

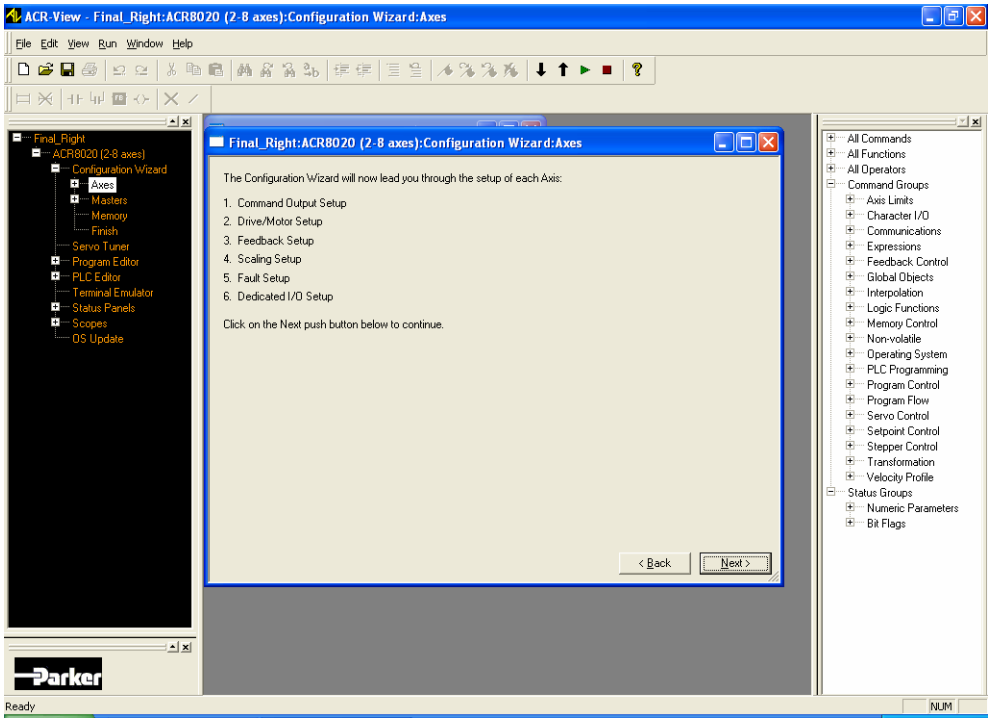


Figure 57: ACR-View Configuration Wizard: Screen 2

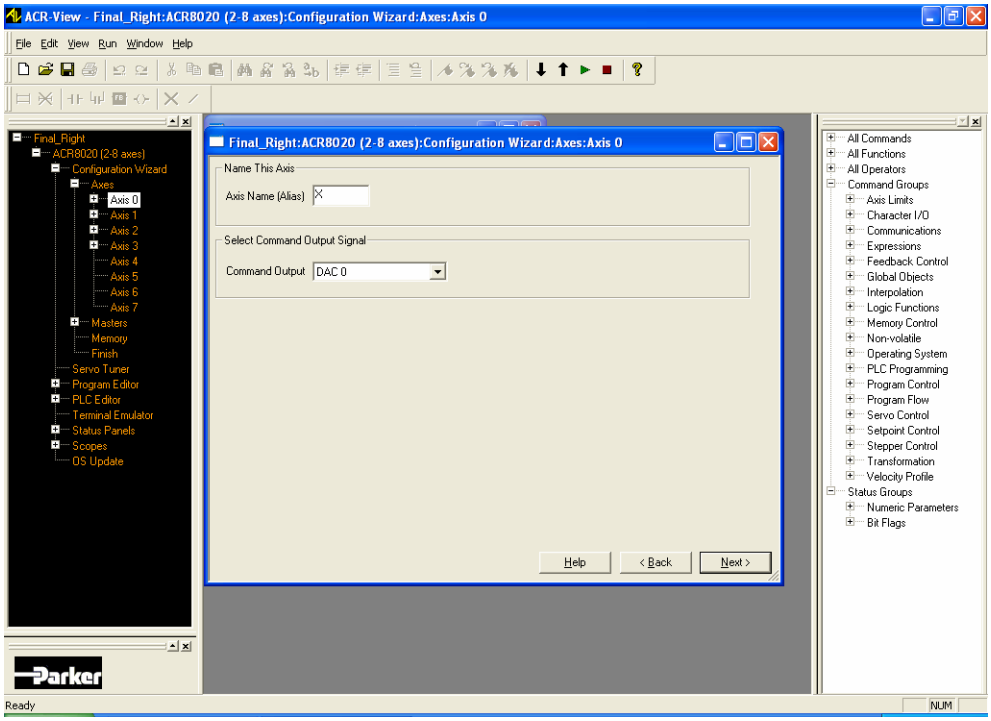


Figure 58: ACR-View Configuration Wizard: Screen 3

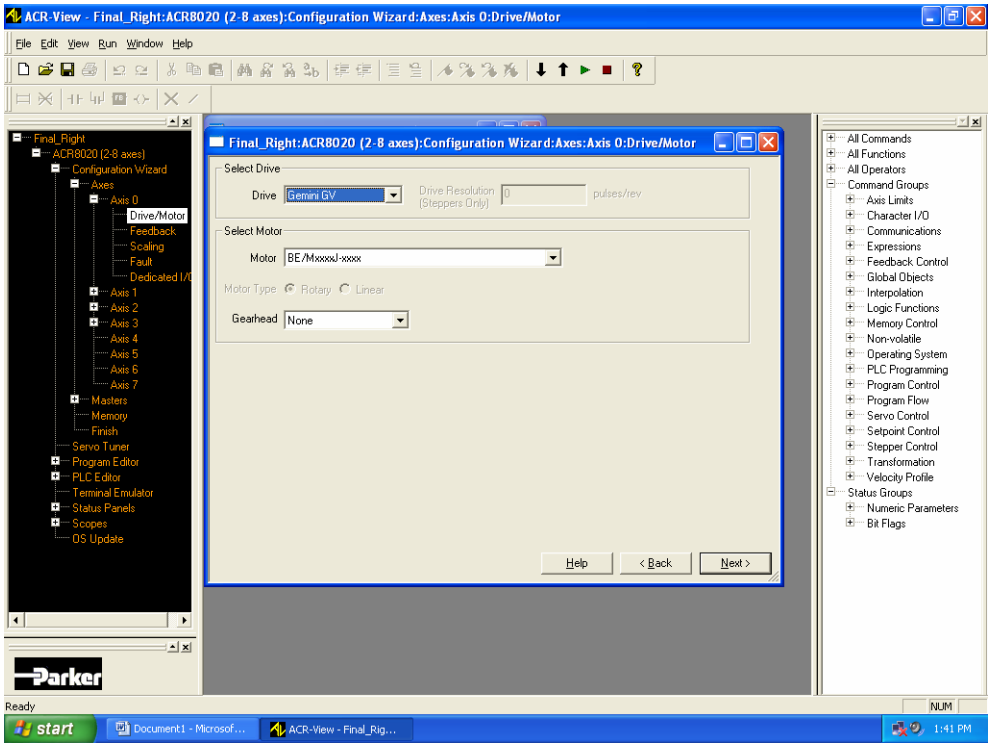


Figure 59: ACR-View Configuration Wizard: Screen 4

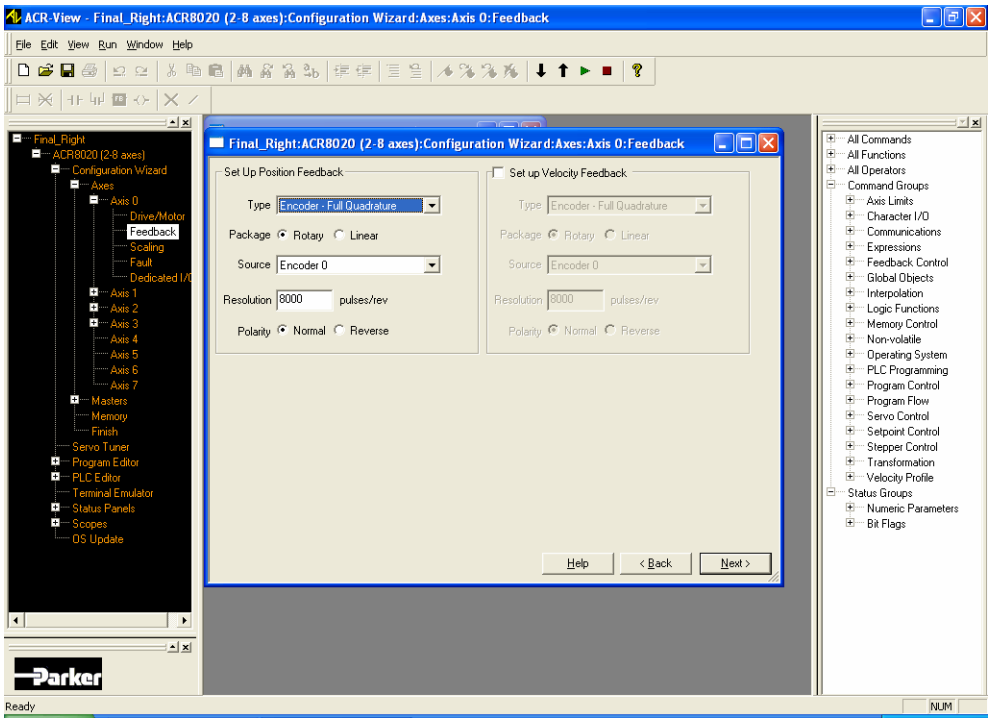


Figure 60: ACR-View Configuration Wizard: Screen 5

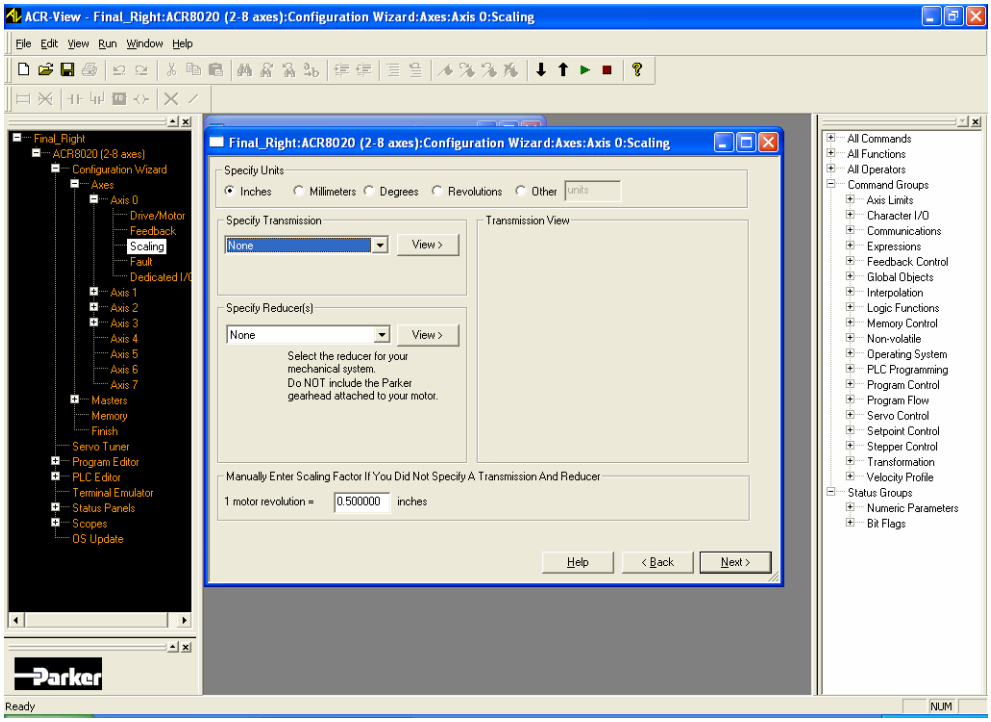


Figure 61: ACR-View Configuration Wizard: Screen 6

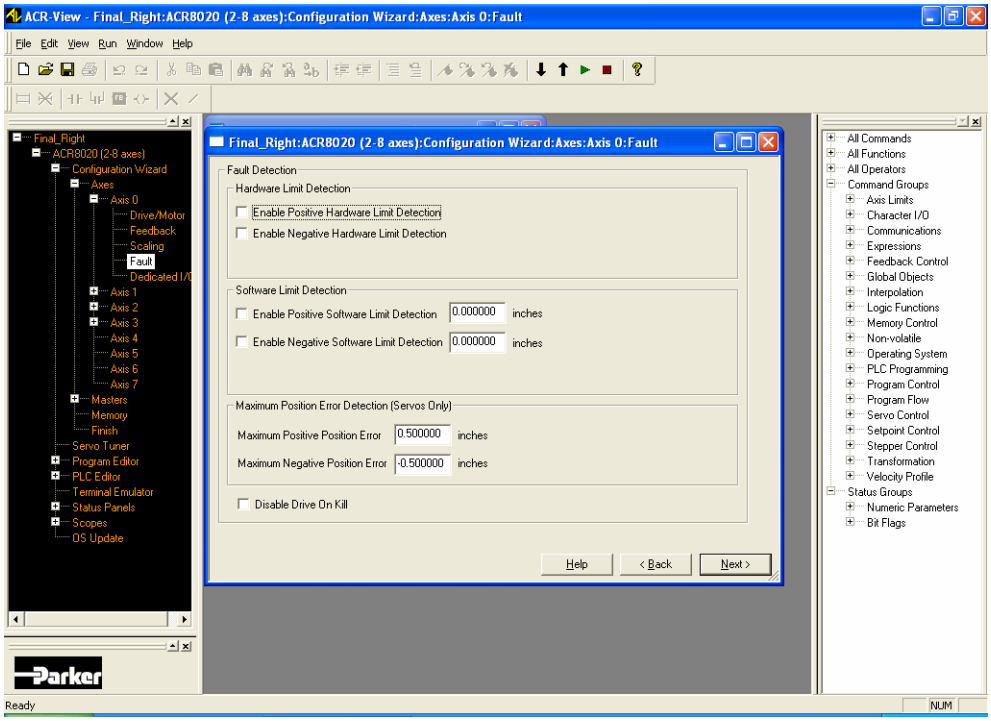


Figure 62: ACR-View Configuration Wizard: Screen 7

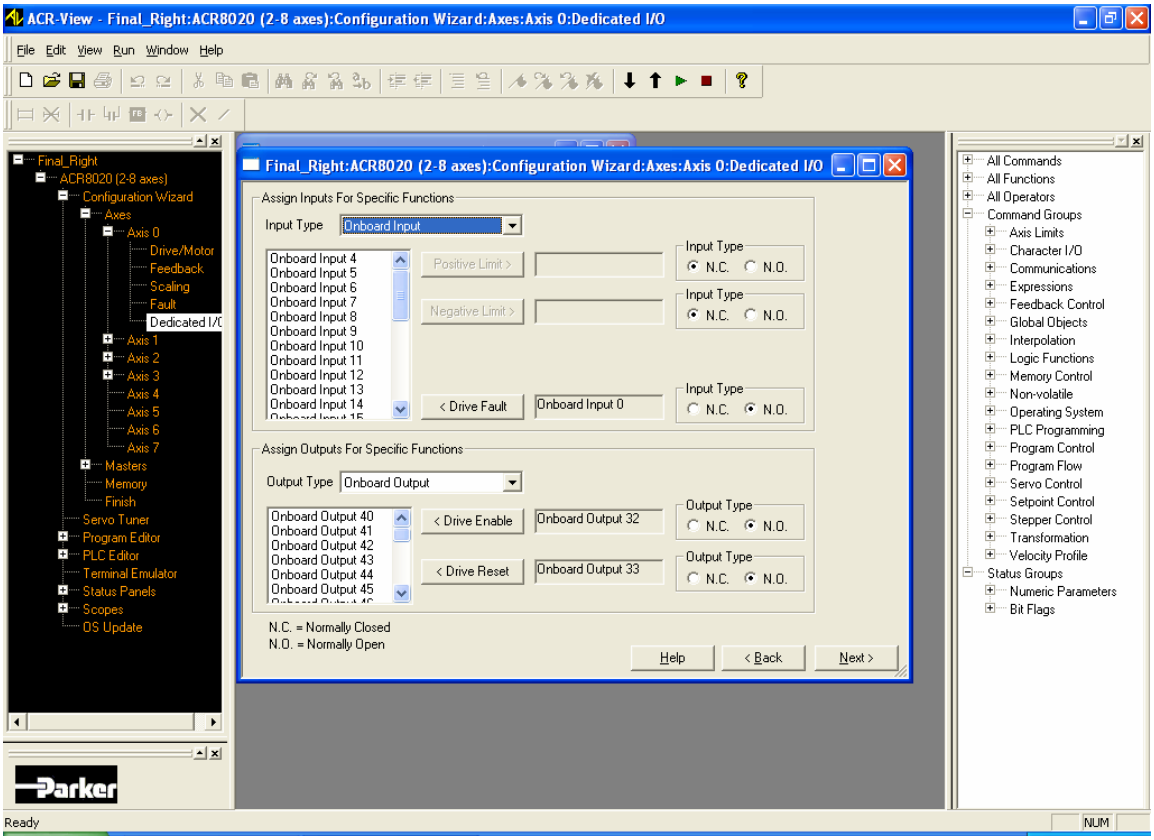


Figure 63: ACR-View Configuration Wizard: Screen 8

8) Screens 3-8 (Figure 58 to Figure 63) are repeated for each of the remaining axes (1-3, to be named Y, Z, and A respectively) in turn.

9) Move all axes to Master0, as shown in the figure below.

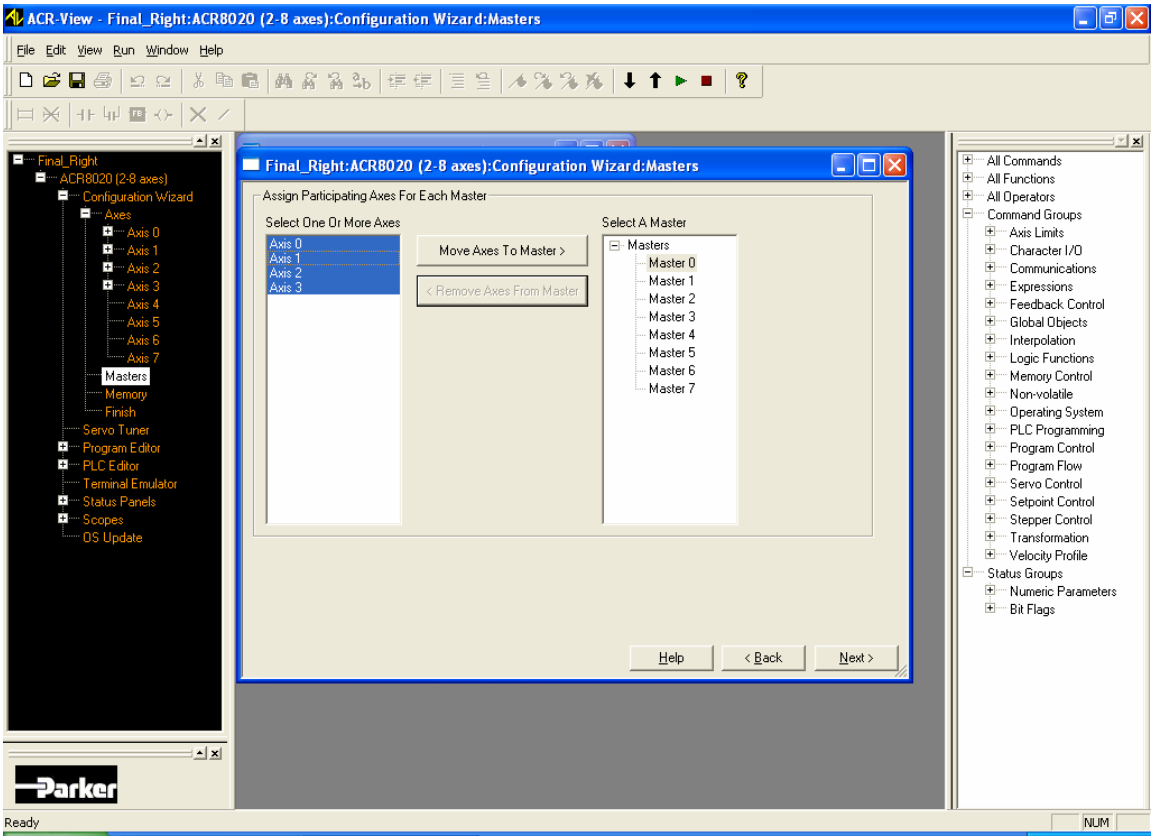


Figure 64: Configuration Wizard: Attach Axes to Master screen.

10) Set the speeds of each axes, as shown in the screen below.

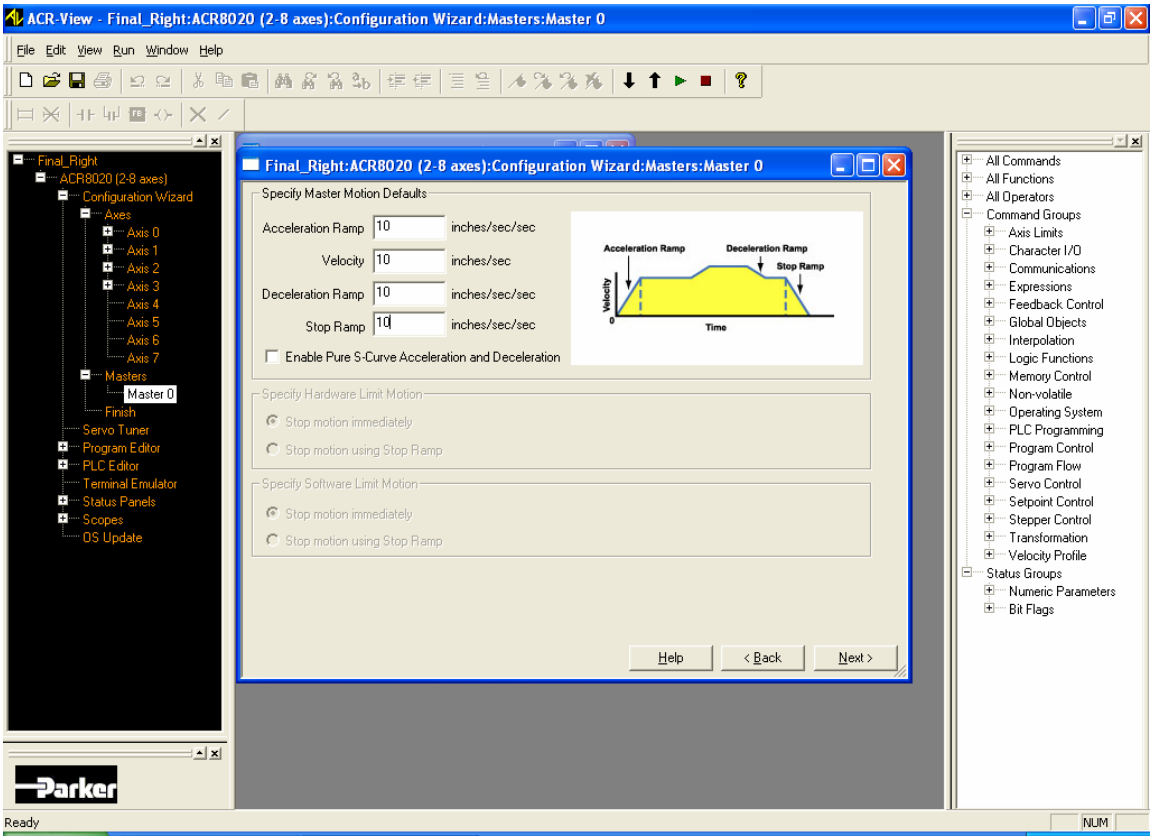


Figure 65: Configuration Wizard - movement settings.

11) Allocate memory to each program. Recommended values are shown in the figure below.

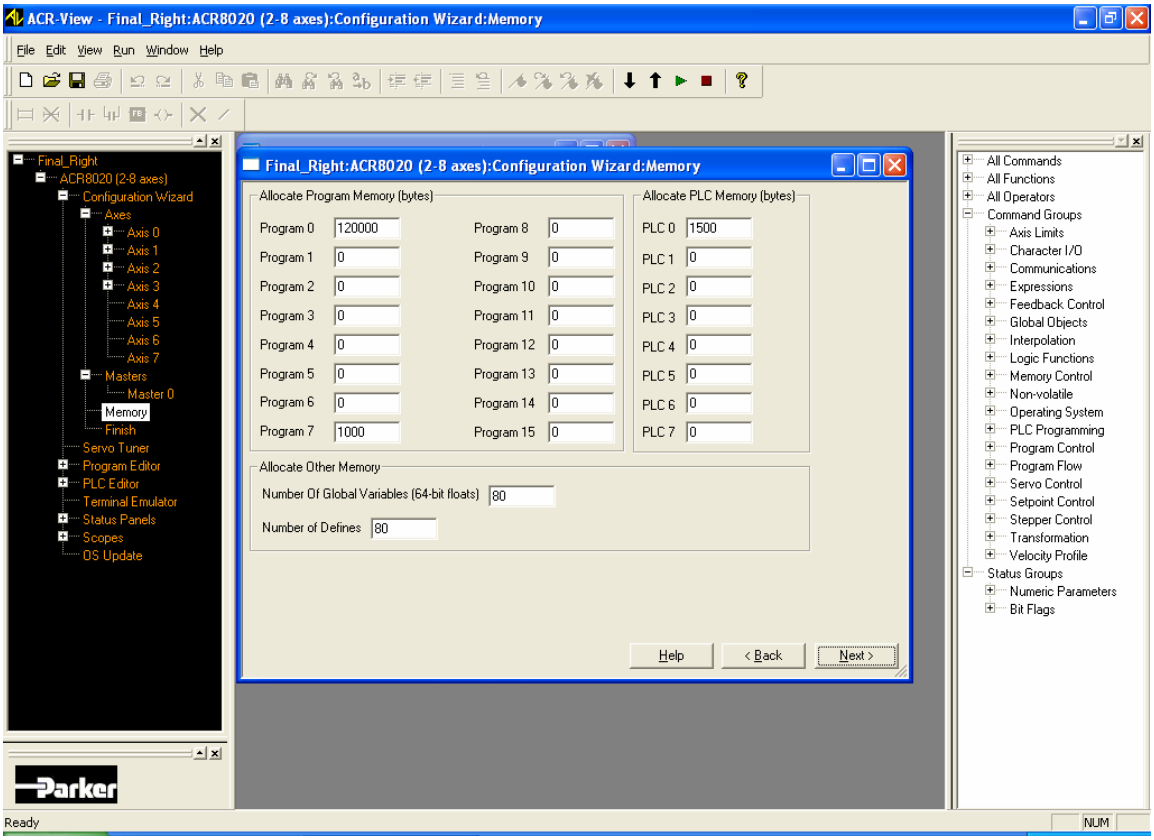


Figure 66: Configuration Wizard - Memory Allocation screen.

12) The final screen of the Configuration Wizard is shown below. Click "Download".

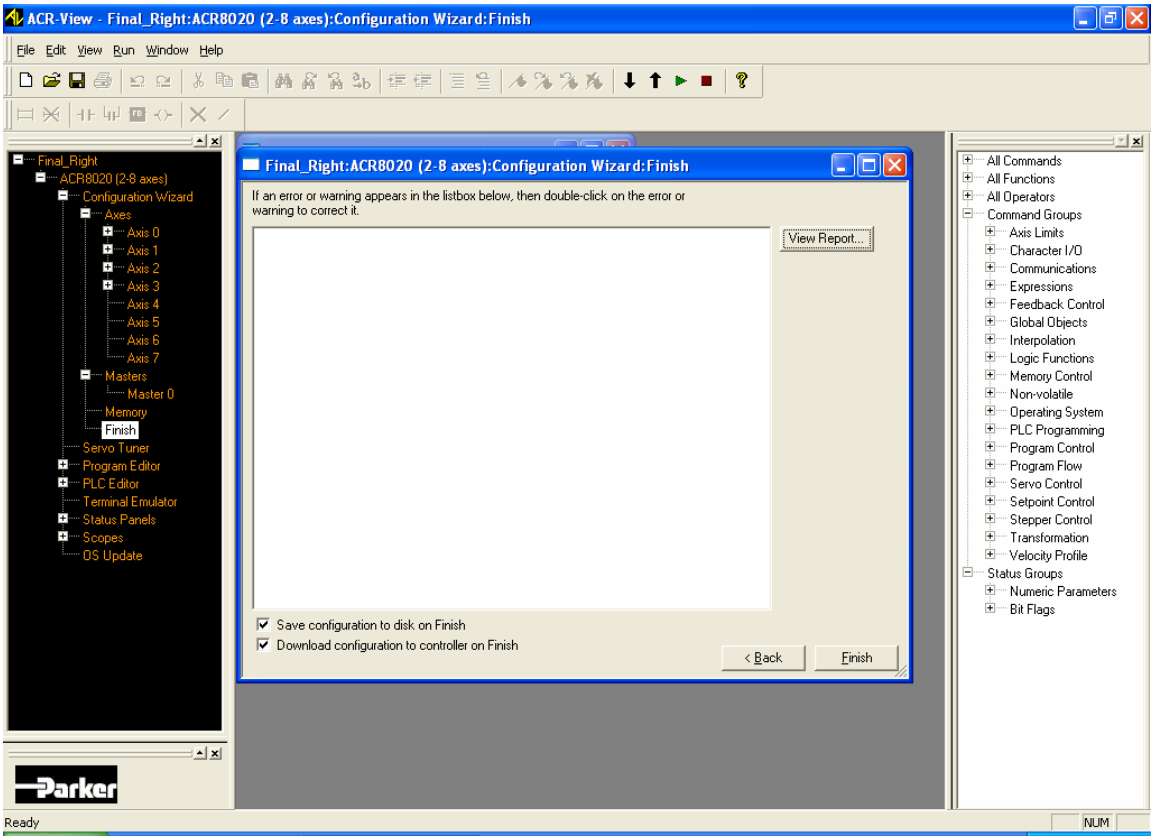


Figure 67: Configuration Wizard final screen.

13) After successfully downloading the configuration, download the programs as needed, as instructed in step #6.

14) After a successful download, turn on the motors' power strip.

15) In the TERMINAL EMULATOR, which is accessed via the left-hand side of the ACR-View program, type the following:

>> CLR 8648

>> SET 32

>> CLR 8500

>> SET 34

>> CLR 8532

>> SET 36

>> CLR 8564

>> SET 38

To RUN a program: In the TERMINAL EMULATOR type “lrun” (no quotation marks) at the “P00” prompt after downloading the program.

Note: If the prompt does not say “P00”, type “prog0” to get the right prompt.

To RUN a PLC: From any command prompt, type “run PLC0” (no quotation marks, where “0” is the number of the PLC that you want to run.

To STOP a program:

In the TERMINAL EMULATOR, hit <Esc>, then type “halt all” (no quotation marks.)

If the power to the motors has been interrupted (e.g. if a situation required the use of the big red emergency-stop button), repeat step #15 at the command prompt BEFORE turning the power back on.

APPENDIX D

MOCK ELBOW

The mock elbow is designed to accommodate two potentiometers: one each to measure flexion/extension and pronation/supination. The original version is shown in Figure 68, as designed and manufactured by students Justin Culagar, Herb Hewitt, and Tim Pournaras at the University of Pittsburgh.

Figure 68: Photo of original mock elbow.

.

The mock elbow has since been modified to ensure realistic pronation-supination moment arms. The modification is that a new radius has been manufactured to interface with

the existing hardware. An updated figure is shown below (Figure 69), with the muscle insertion points indicated.

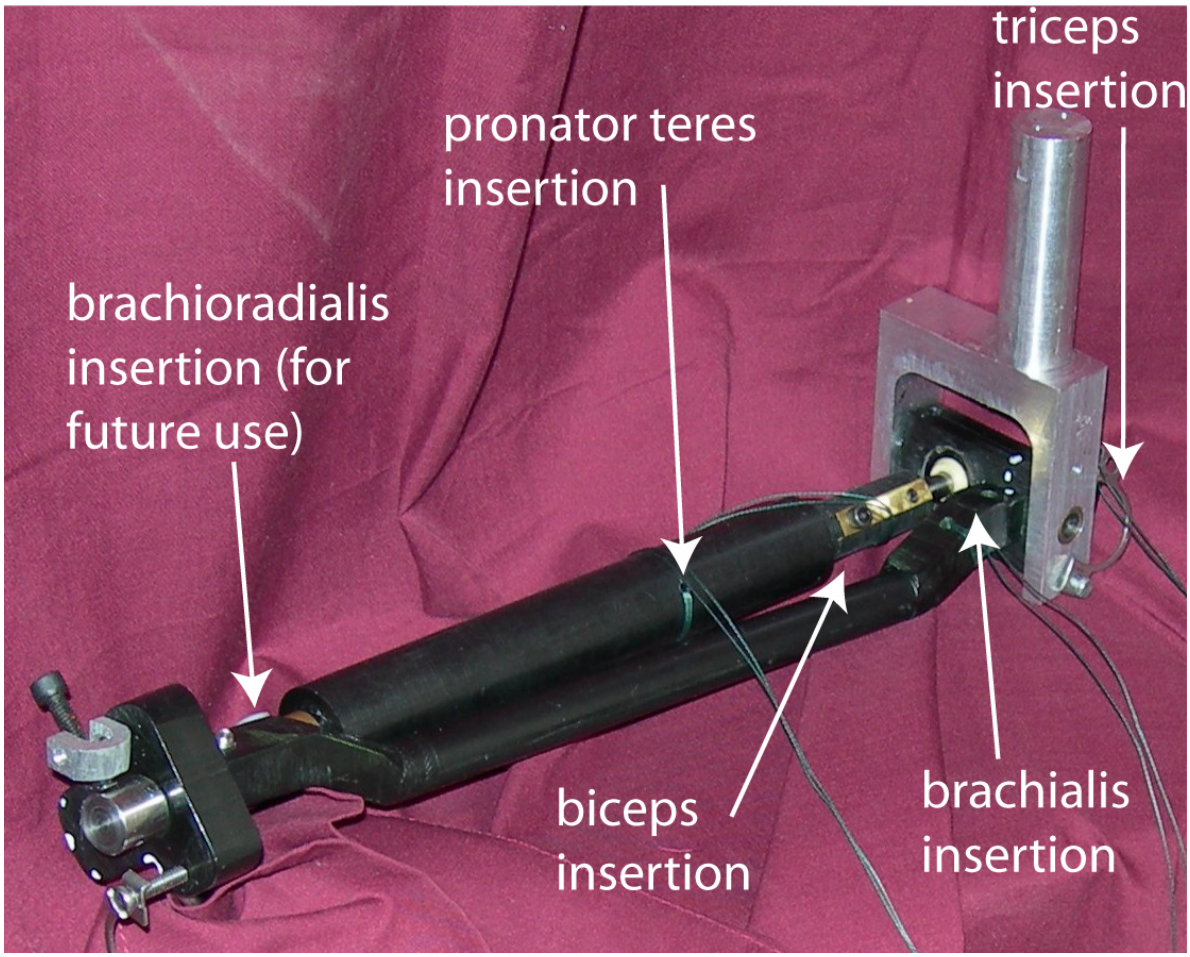


Figure 69. Photo of modified mock elbow, with muscle insertion sites labeled.

APPENDIX E

COORDINATE MEASURING MACHINE OPERATION

This appendix contains information about the operation of the Coordinate Measuring Machine (CMM) and specific details related to the measurement of elbow specimens as intended by the work presented in this dissertation.

E.1 GENERAL CMM OPERATION

Before beginning, turn on the air to the CMM and start up the computer. The U-SOFT software will load automatically. The general procedure is:

1. Choose the "element" you want to use for the measurement
2. Measure the object and store the results. (Also write down the results.)
3. Save the data file.

When the U-SOFT software starts:

-Hit F1 –*Measure*

- Hit ESC three times till you see:

MAIN MENU (only used menus are shown)

1. *Elements*
2. -------
3. -------
4. ------
5. -------
6. *Special Tasks*
7. --------
8. --------
9. *Options*

For example, to measure a round ball:

-Hit 1 (*Elements*)

-Hit 5 (*Spheres*) Touch 3 points at one level as if pointing on a circle. Then touch fourth point that gives the altitude. Pick more arbitrary points to improve accuracy.

-Hit F1, finish pointing, manually write down the results to a piece of paper.

-Hit F8, this will store the data into memory

Another useful feature is using the "live display", which is useful for determining the x, y, and z directions. From the *MAIN MENU*

-Hit 6 (*Special Tasks*)

-Hit 1 (*Live Display*). The coordinates of the probe tip will be displayed.

<u>Saving data on the disk</u>

From the *MAIN MENU:*

- > 9 (*Options*)

->8 (*Element Export*)

- > 1 *ASCII*

Type “a:<filename>”, where <filename> is the desired filename. There is an 8-character limit.

Then hit F1 (All) and then it will save the data to the floppy.

Note: U-Soft keeps adding data into memory. To clear the memory (as in: if you want to measure a different object), from the *MAIN MENU, choose* > 9 (*Options*), then >9 (*Exit*), then *>F1 (YES)*.

E.2 MEASUREMENTS REQUIRED FOR RADIAL HEAD PROCESSING

The following things need to be measured for the radial head processing. The element in parentheses indicates the choice from the CMM “Elements” menu.

E.2.1 Humerus

- Center of Capitellum *(sphere)*
- Center of trochlear groove *(circle)*
- 4 balls on the array *(sphere)*
- 4 small balls near the radial head *(sphere)*
- Center of the humeral shaft *(circle, or just use the top ball from the array if it is positioned as such.)*

E.2.2 Radial head

- Center of the radial head *(circle)*

- Center on the backside *(circle)*
- Position of the landmark "dots" *(circles or spheres if possible)*
- Any point on the rim of the radial head. *(point)*

E.2.3 Sample data sheets (for recording)

The table below shows a sample data sheet for the "HumBig" pointing file that is needed by the Matlab processing code. "NN" indicates that these items need not be measured on the CMM, but values must be inputted in when making the .dat file so that the Matlab code will run properly. It is strongly recommended to manually record the values on paper to keep track of which numbers correspond to which anatomic points.

Table 25: Sample humerus (big markers) record-keeping table.

HUMERUS (Hbig)		X (in)	Y(in)	Z(in)	Radius(in)
1.Lateral Epicondyle (NN)					
2.Medial Epicondyle (NN)					
3.Center of the Shaft					
4.Trochlear Groove					
Humeral Marker Balls (big)	5.Big1				
	6.Big2				
	7.Big3				
	8.Big4				
9.Capitulum					

The next table shows a sample data sheet for the "HumSmall" pointing file that is needed by the Matlab processing code.

Table 26: Sample humerus (small) record-keeping table.

HUMERUS (Hsmall)		X (in)	Y(in)	Z(in)	Radius(in)
1.Lateral Epicondyle (NN)					
2.Medial Epicondyle (NN)					
3.Center of the Shaft					
4.Trochlear Groove					
Humeral Marker Balls (small)	5.Small1				
	6. Small 2				
	7. Small 3				
	8. Small 4				
9.Capitulum					

The next table shows a sample data sheet for the "HorI_CMM" pointing file that is needed by the Matlab processing code.

Table 27: Radial head or implant recording table.

Implant ( )		X (in)	Y(in)	Z(in)	Radius(in)
1. Center on the contact surface					
Implant Marker Balls	2.Dot1				
	3.Dot2				
	4.Dot3				
	5.Dot4				
6. Any point on the rim.					
7. Center on the back side					

Finally, for the FHA, the ulna and distal radius markers will also need to be measured. The tables below can be used.

Table 28: Ulna recording table.

ULNA (U)		X (in)	Y(in)	Z(in)	Radius(in)
1.Styloid					
2. Trochlea (NN)					
Ulnar Marker Balls	3. Ball 1				
	4. Ball 2				
	5. Ball 3				
	6. Ball 4				

Table 29: Distal radius recording table.

Distal RADIUS (R)		X (in)	Y(in)	Z(in)	Radius(in)
1.Styloid					
2.Volar Radioulnar Joint					
3.Dorsal Radioulnar Joint					
Radial Marker Balls	4. Ball 1				
	5. Ball 2				
	6. Ball 3				
	7. Ball 4				

It is important to note that, for all markers for a given segment, the order of the CMM measurement must be equivalent to the order in which the marker position data appears in the dynamic data-collection files.

APPENDIX F

SENSOR CALIBRATION

Six states are monitored on the elbow simulator. Sensors measure loads in the four tendons and angular displacements of the two degrees of freedom. All of the sensors are calibrated in nearly the same way. All six sensors provide a voltage change proportional to a state change. Therefore, the calibration of these sensors depends on the determination of two fundamental parameters of a line: a slope and an offset. The slope corresponds to a sensitivity relating the physical value being measured to the output voltage of the sensor. The offset accounts for any sort of bias voltage present. Equation 14 illustrates the relationship between the physical quantity being measured and the input voltage read by the ACR 8020.

$$PhysicalValue = ElectromechanicalGain * InputVoltage + Offset \quad (14)$$

Conveniently, the Acroloop software provides access to gains and offsets relating to the analog inputs. Thus, the ability to modify the analog inputs exists and values approximately the same as the physical values can be used as variables within the control code making its programming more intuitive. Table 30 provides the necessary parameter values for all eight

axes. Prior to calibrating the sensors it is important to set all of the gains to a value of one and all of the offsets to a value of zero.

Table 30: Parameter values for obtaining physical values from analog input voltages

Axis	Current Value	Gain	Offset
0	P6408	P6410	P6411
1	P6424	P6426	P6427
2	P6440	P6442	P6443
3	P6456	P6458	P6459
4	P6472	P6474	P6475
5	P6488	P6490	P6491
6	P6504	P6506	P6507
7	P6520	P6522	P6523

The gain and offset are determined in generally the same way for all of the sensors. The gain is determined by applying two known physical values to the sensor and recording the voltage input to the ACR 8020 for each physical value. The voltage input, corresponding to the current values in the table above, can be read from the numeric status panel within Acroloop. The gain is then determined from Equation 15. It is important to enter the correct gain parameter value into Acroloop before moving on to determining the offsets.

$$Gain = \frac{PhysicalValue1 - PhysicalValue2}{Voltage1 - Voltage2} \qquad (15)$$

Next, the offset can be determined by considering what the current level is compared to what it should be. The level the ACR 8020 should be reporting is based on the current state of the system which should be known. The offset can be found using Equation 16.

$$Offset = DesiredValue - CurrentValue \quad (16)$$

F.1.1 Inclinometer Calibration Specifics

An inclinometer is used to measure the f/e angle of the arm. The inclinometer is connected to a conditioning box which provides both power to the inclinometer and analog voltage proportional to angular displacement change. The inclinometer is effective to +/- ~105° degrees above and below the horizon. To determine the value necessary for the gain, record the voltage in the ACR 8020 numeric status panel under the correct axis parameter value with the arm at two different angles. Two possibilities exist for setting the offset. The equation above can be used if the current angle of the arm is known, which can be found from a measurement or estimation if accuracy is not critical. The second option uses a button on the side of the conditioning box to set the voltage output at that angle to zero. If the button is pushed while the arm is horizontal, the offset can be set to zero, and the resulting reading in the numeric status panel will correspond to the angle above or below the horizon, with the units being determined by the physical values used in setting the gain.

F.1.2 Potentiometer Calibration Specifics

A potentiometer is used to measure the supination / pronation angle of the arm. The calibration of the potentiometer is the same as that of the inclinometer minus the possibility of zeroing the voltage with a reset button. One important note for the potentiometer is its effective range. There is a deadzone in the potentiometer output, it is critical that this deadzone is not within the usable range of motion of the arm. When the potentiometer is first attached, move the arm through the full range of motion while viewing the analog input to the controller through the numeric status panel or with a multimeter. As long as the change in voltage is continuous, the positioning of the potentiometer is acceptable. Otherwise, rotate the potentiometer until the voltage output changes in a continuous manner over the desired range of motion of the arm.

F.1.3 Load Cell Calibration Specifics

The load applied to the arm through the tendons is determined by load cells strung in series with the actuators. The determination of their gain and offset follows exactly from the equations listed above. Dead weights of known value should be hung from the pulleys to provide accurate gains. After determining and setting the gain parameter value, a simplified procedure exists for setting the offset. With no weight hanging from the load cell, the offset can be set simply to the negative of the current value.

APPENDIX G

HOW TO COLLECT MOMENT ARM DATA

1) In ACR-View, open up the project MomArm_XZ.

2) Download the configuration only. (See Appendix C for instructions.)

3) In the Terminal Emulator, type “adc on”.

4) Set the Gains and Offset for the f/e inclinometer and p/s potentiometer. After calculating the values (see Appendix F) input them into ACR-View.

5) Calibrate the load cells (see Appendix F)

6) Open up the Visual Studio project ChalkBoard1 (Figure 70).

Hit F5 to run it.

In the new window that pops up, click on the “Save” tab. (Figure 71)

Figure 70: ChalkBoard1 initial screen.

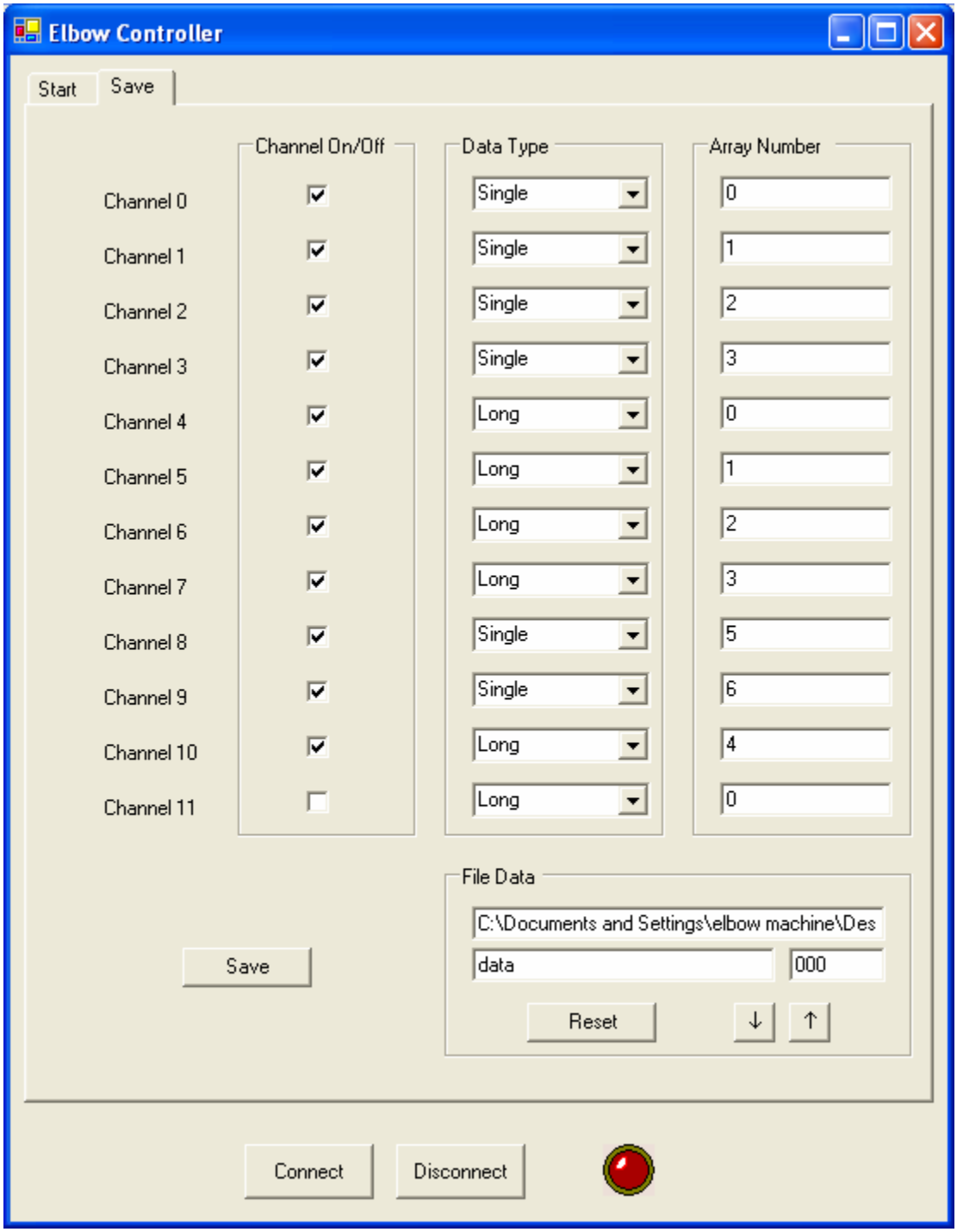


Figure 71: ChalkBoard1 "Save" screen.

Check channels 0->11.

In the two right-hand columns, set them to:

Single 0

Single 1

Single 2

Single 3

Long 0

Long 1

Long 2

Long 3

Single 5

Single 6

Long 4

Type in the filename prefix to the box, and set the counter to 1.

Filename convention:

<XZ or YA>_<fe or ps motion>_<ps or fe constant><Neu, Pro, Sup, 30_, 60_, 90_ >

Example: When recording f/e moment arms for biceps (actuator x) and pronator (actuator z) at full supination, the name would be "XZ_fe_psSup" and the counter will append the trial number to that.

7) In ACR-View, move the cylinders until there is force on each muscle until all are between 3 and 10 lbs. (Monitor the values in the Numeric Status Panel in ACR-View. The forces are on ADC's 4 through 7.) Record these values (approximately) in a notebook.

8) Go in to the Program Editor, choose Program 0, and set the reference force values to the numbers that you just wrote down.

9) Download Program 0 to the controller. (See Appendix C for instructions.)

10) At the command prompt, type "lrun". Once someone hits enter, then manually move the elbow in the desired motion. At the end of the motion, stop and wait for the ACR-View program to finish.

- Prior to the collection of the first trial, run the program once for the purpose of moving the elbow to the starting position (full flexion, or full supination.)

11) Once data collection has ended, in the ChalkBoard1 program, hit "Save". The counter will increment when the file has been successfully saved.

Typically, flexion and extension are collected in different trials. Pronation and supination are fast enough to be collected within the same trial.

APPENDIX H

MOMENT ARM ACR-VIEW CODE

H.1 MOMARMS_XZ

```
PROGRAM
'Program 0
'TODO: edit your program here

'from Pat's MomArmsXZA program.
' This one will be set up to do X and Z.  (So axes 1,3 , or Inputs 4, 6,

'Variables
Dim SA(8)
'Dim SA(9)
Dim SA0(750)
Dim SA1(750)
Dim SA2(750)
Dim SA3(750)
Dim SA4(2)

Dim SA5(750)
Dim SA6(750)
Dim SA7(750)
```

```
' For encoder positions.
Dim LA(5)
Dim LA0(750)
Dim LA1(750)
Dim LA2(750)
Dim LA3(750)
Dim LA4(750)



Dim SV(13)

'ADC
ADC ON

'ADC Gains - ADCs 4-7
'P6474 = 400
'P6490 = 400
'P6506 = 1600
'P6522 = 400

'ADC Offsets
'P6475 = -0.3
'P6491 = -0.34
'P6507 = -2.2
'P6523 = -0.13

'Sampling Variable Initializations
Samp Clear
' source = ADC4
Samp0 SRC P6472
Samp0 Base SA0
```

```
'source = ADC5
'Samp1 SRC P6488
'Samp1 Base SA1

'source = ADC6
Samp2 SRC P6504
Samp2 Base SA2

'source = ADC7
'Samp3 SRC P6520
'Samp3 Base SA3

' even though this is weird, the encoder positions will be on Samps 1, 4, and 5
Samp1 SRC P6144
Samp1 Base LA0

Samp4 SRC P6176
Samp4 Base LA2

'Samp5 SRC P6192
'Samp5 Base LA3

'potentiometers
' f/e pot
Samp6 SRC P6456
Samp6 Base SA5

' p/s pot
Samp5 SRC P6424
Samp5 Base SA6

'sync signal
```

```
Samp7 SRC P4097
Samp7 Base LA4

'Sampling Period (Milliseconds)
P6915 = 50

'Reference Level (LBS)
'for X
SV0 = 4.5

' for Y:
SV10=8.5

'for Z
SV4=3

'for A
SV5=4.5

'Control Level (LBS)
SV1 = SV0
SV11=SV10
SV6=SV4
SV7=SV5

'Error Level
SV2 = 0
SV12=0
SV8=0
SV9=0

'Counter
SV3 = 1
```

```
'Start Time
SA4(0) = P6916

'Sampled Program
Set 105

dwl 1
set bit 56

'Turn Jog On
'Z
P12860=0
Set 861

'Y
P12604=0
Set 829

'X
P12348=0
Set 797

'A
P13116 = 0
Set 893

'Control Loop
While (SV3 < 2000)

'SV1=ADC4 value
SV1 = P6472
'SV2=current value - target value
```

```
SV2 = SV1 - SV0

SV11=P6488
SV12=SV11-SV10

SV6=P6504
SV8= SV6 - SV4

SV7=P6520
SV9= SV7-SV5

'X
If (SV2 < 0)
' jog in reverse
        Set 797
        P12348 = Absf(0.15*SV2)
Else
' jog forward
        Set 796
        P12348 = Absf(0.15*SV2)
EndIf

'Y
If (SV12 < 0)
' jog in reverse
        Set 829
        P12604 = Absf(0.1*SV12)
Else
' jog forward
        Set 828
        P12604 = Absf(0.1*SV12)
EndIf
```

```
'Z
If (SV8 < 0)
' jog in reverse
        Set 861
        P12860 = Absf(0.1*SV8)
Else
' jog forward
        Set 860
        P12860 = Absf(0.1*SV8)
EndIf

'A
If (SV9 < 0)
' jog in reverse
        Set 893
        P13116 = Absf(0.15*SV9)
Else
' jog forward
        Set 892
        P13116 = Absf(0.15*SV9)
EndIf

SV3 = SV3 + 1

Wend

Clr 56

'Stop Movement
P13116 = 0
P12860 = 0
P12348 = 0
P12860=0
```

```
Clr 892
Clr 893
Clr 860
Clr 861
Clr 796
Clr 797
Clr 828
Clr 829


'Stop Sampling
INH -105


'Stop Time
SA4(1) = P6916


ENDP
```

H.2 MOMARMS_YA

```
PROGRAM
'Program 0
'TODO: edit your program here

'from MomArms_YZA
' This one will be set up to do Y and A.  (So axes 2 and 4, or Inputs 5 and 7
```

```
'Variables
Dim SA(7)
'Dim SA(9)
Dim SA0(750)
Dim SA1(750)
Dim SA2(750)
Dim SA3(750)
Dim SA4(2)

Dim SA5(750)
Dim SA6(750)

' For encoder positions.
Dim LA(5)
Dim LA0(750)
Dim LA1(750)
Dim LA2(750)
Dim LA3(750)
Dim LA4(750)



Dim SV(13)

'ADC
ADC ON

'ADC Gains - ADCs 4-7
'P6474 = 400
'P6490 = 400
'P6506 = 1600
'P6522 = 400

'ADC Offsets
'P6475 = -0.3
'P6491 = -0.34
'P6507 = -2.2
'P6523 = -0.13

'Sampling Variable Initializations
Samp Clear

'x
' source = ADC4
'Samp0 SRC P6488
```

```
'Samp0 Base SA0

'source = ADC5; Y
Samp0 SRC P6488
Samp0 Base SA1

'source = ADC6
'Samp2 SRC P6504
'Samp2 Base SA2

'source = ADC7
Samp2 SRC P6520
Samp2 Base SA3

' even though this is weird, the encoder positions will be on Samps 1, 4, and 5
Samp1 SRC P6160
Samp1 Base LA1


Samp4 SRC P6192
Samp4 Base LA3

'potentiometers

Samp6 SRC P6456
Samp6 Base SA5

'p/s one:

Samp5 SRC P6424
Samp5 Base SA6


'Synch

Samp7 SRC P4097
Samp7 Base LA4




'Sampling Period (Milliseconds)
P6915 = 50

'Reference Level (LBS)
' x
```

```
SV10=4.5

'for Y
SV0 = 8.5

'for Z
SV4= 3

'for A
SV5= 4.5

'Control Level (LBS)
SV11=SV10
SV1 = SV0
SV6=SV4
SV7=SV5

'Error Level
SV12=0
SV2 = 0
SV8=0
SV9=0

'Counter
SV3 = 1

'Start Time
SA4(0) = P6916

'Sampled Program
Set 105

'Synch Signal
dwl 1
set bit 56

'Turn Jog On

P12348=0
Set 797

P12860=0
Set 861

P12604=0
Set 829
```

```
P13116 = 0
Set 893

'Control Loop
While (SV3 < 2000)
' For X:
SV11=P6472
SV12=SV11-SV10


' For Y
'SV1=ADC4 value
SV1 = P6488
'SV2=current value - target value
SV2 = SV1 - SV0

SV6=P6504
SV8= SV6 - SV4

SV7=P6520
SV9= SV7-SV5
' For X
If (SV12 < 0)
' jog in reverse
        Set 797
        P12348 = Absf(0.15*SV12)
Else
' jog forward
        Set 796
        P12348 = Absf(0.15*SV12)
EndIf


' For Y
If (SV2 < 0)
' jog in reverse
        Set 829
        P12604 = Absf(0.1*SV2)
Else
' jog forward
        Set 828
        P12604 = Absf(0.1*SV2)
EndIf
```

```
'Z
If (SV8 < 0)
' jog in reverse
        Set 861
        P12860 = Absf(0.1*SV8)
Else
' jog forward
        Set 860
        P12860 = Absf(0.1*SV8)
EndIf

'A
If (SV9 < 0)
' jog in reverse
        Set 893
        P13116 = Absf(0.15*SV9)
Else
' jog forward
        Set 892
        P13116 = Absf(0.15*SV9)
EndIf

SV3 = SV3 + 1

Wend

'Clear Synch
Clr 56

'Stop Movement
P13116 = 0
P12860 = 0
P12604=0
P12348=0

Clr 892
Clr 893
Clr 860
Clr 861
Clr 828
Clr 829
clr 796
clr 797
```

```
'Stop Sampling
INH -105


'Stop Time
SA4(1) = P6916

ENDP
```

APPENDIX I

SIMULATOR MOMENT ARMS CALCULATIONS CODE

I.1 MOMENTS_LK3.M

```
%moments_LK3.m
% Laurel Kuxhaus, lck4@pitt.edu
% Allegheny General Hospital
% Orthopaedic Biomechanics Research Laboratory
% 26 July 2005
% modified from moments.m, from Josh Magnusen's thesis.
% key modifications include turning the script in to a function that does
% not require manual user inputs, because that is annoying.

% revised today to keep everything in the elbow coordinate system.

%Calculate moment arms through 140 deg of flexion

% clear all;
% close all;

function momentout=moments_LK3(orig_x, orig_y, orig_z, ins_x, ins_y,
 ins_z)
% inputs must be in centimeters, relative to the humeral coordinate
 system.
%
%disp('Enter units in centimeters');

%Prompt for muscle origin
% orig_x = input('Muscle Origin x: ');
% orig_y = input('Muscle Origin y: ');
% orig_z = input('Muscle Origin z: ');

%make a vector of the muscle origin, and move it to the elbow's coordinate
%system.
%orig = [orig_x+0, orig_y-2, orig_z+30.5];
orig = [orig_x+0, orig_y, orig_z];
% comment: check this with the simulator "origins"....I suspect they may
% already be in the elbow coordinate system....
```

```
33
34
35 %Prompt for muscle insertion
36 % ins_x = input('Muscle Insertion x: ');
37 % ins_y = input('Muscle Insertion y: ');
38 % ins_z = input('Muscle Insertion z: ');
39 %make a vector of the muscle insertion, and move it to the elbow's
    coordinate
40 %system.
41 %ins = [ins_x+0, ins_y-2, ins_z+30.5];
42 ins = [ins_x, ins_y-2, ins_z+30.5];
43
44
45
46 %Vector with angles from 0 to 140 deg. in multiples of 5
47 angle = linspace(0, 140, 29);
48
49 %Calculate moment arms throughout motion
50
51 for i = 1:29
52
53 %run 'rot.m' function
54 ins_new = rot_LK3(ins, angle(i)) ;
55 %run 'perp.m' function
56 moment(i) = 10*(perp_LK2(orig, ins_new)); % changed to perp_LK 7/6/05
57
58 end
59 figure;
60 plot(angle, moment); % think the 10 makes it be in mm, not cm.
61 xlabel('Flexion Angle (deg)');
62 ylabel('Moment Arm (mm)');
63
64
65 momentout=moment;
```

I.2 PERP_LK2.M

```
%perp_LK2.m
% Laurel Kuxhaus, lck4@pitt.edu
% Allegheny General Hospital
% Orthopaedic Biomechanics Research Laboratory
% 26 July 2005
% modified from perp.m, from Josh Magnusen's thesis.
% key modifications include using a documented technique that I understand
% to calculate the mysterious distance.
% method from: http://mathworld.wolfram.com/Line-LineDistance.html


%function perp.m

%Calculate perpindicular distance from muscle to center of %rotation

%Input variables
%% 'orig': muscle's origin coordinates, in elbow coordinate system
%% 'ins': muscle's insertion coordinates, in elbow coordinate system

%Output variables
%% 'dist': calculated moment arm, in mm? cm? not sure.

function [dist] = perp_LK(orig, ins);


x1 = orig(1); y1 = orig(2); z1 = orig(3);
x2 = ins(1); y2 = ins(2); z2 = ins(3);
x3=0; y3=10; z3=0;
x4=0; y4=-10; z4=0;

pt1=[x1; y1; z1];
pt2=[x2; y2; z2];
pt3=[x3; y3; z3];
pt4=[x4; y4; z4];

a=pt2-pt1;
b=pt4-pt3;
c=pt3-pt2;

dist=(norm(dot(c,cross(a,b))))/(norm(cross(a,b)));
```

I.3 ROT_LK3.M

```
%function rot_LK3.m
% Laurel Kuxhaus, lck4@pitt.edu
```

```
3 % Allegheny General Hospital
4 % Orthopaedic Biomechanics Research Laboratory
5 % 26 July 2005
6 % This is code that should determine new insertion after flexion.
7 % "inspired" by rot.m from Josh Magnusen's thesis.
89
10 %Input variables
11 %% 'ins': insertion points (in elbow coordinate system)
12 %% 'angle': angle of rotation
13
14 %Output variables
15 %% 'ins_new': new insertion coordinates after rotation
16
17 function [ins_new] = rot_LK3(ins, angle);
18
19 %Convert degrees to radians
20 angle = angle*pi/180;
21
22
23
24 %Rotation matrix
25 Ry = [cos(angle) 0 -sin(angle) ; 0 1 0 ; sin(angle) 0 cos(angle) ];
26 Rz=[cos(pi/2) sin(pi/2) 0; -sin(pi/2) cos(pi/2) 0; 0 0 1];
27 %Calculate new insertions
28 ins_new =Ry*Rz*[ins]';
29 %ins_new = [ins_new(1) ins_new(2) ins_new(3)];
```

I.4 RUN_MOMENTS_LK2ADJ.M

```
1
2 % run_moments.m
3 % Laurel Kuxhaus, lck4@pitt.edu
4 % Allegheny General Hospital
5 % Orthopaedic Biomechanics Research Laboratory
6 % 10 July 2005
7 % this is a simple file that will run the moments_LK for each muscle, using
8 % the numbers supposedly used in Josh's thesis.
9
10 % modified 11 July 2005 to make one plot at the end that shows all figs.
11
12 clear all; close all;
13
14 % biceps
15 biceps=moments_LK3(5.9, .8, 31.4, -.8, 2.8, -36);
16 biceps_hi=moments_LK3(5.9+2.5, .8, 31.4, -.8, 2.8, -36);
17 biceps_lo=moments_LK3(5.9-2.5, .8, 31.4, -.8, 2.8, -36);
18 title('biceps')
19
20 % brachialis
21 brach=moments_LK2(5.5, 0, 33.3, 0, 1.2, -33.5);
22 brach_hi=moments_LK2(5.5+2.5, 0, 33.3, 0, 1.2, -33.5);
23 brach_lo=moments_LK2(5.5-2.5, 0, 33.3, 0, 1.2, -33.5);
```

```
24 title('brachialis')
25
26 % triceps
27 triceps=moments_LK2(-3, 0, 25.1, -1.5, -2, -28);
28 title('triceps')
29
30 % brachioradialis
31 % brd=moments_LK2(2.7, -5, 2.5, 0,0, -54.5);
32 brd=moments_LK3(0, -5, 3.5, 0,3, -54.5);
33 % numbers from choosing the best from BrachProAdjust.m
34 brd_hi=moments_LK3(0, -5, 3.5, 0,3+2.5, -54.5);
35 brd_lo=moments_LK3(0, -5, 3.5, 0,3-2.5, -54.5);
36 title('brachioradialis')
37
38 % pronator teres
39 % pronator=moments_LK2(2.7, 5, 2.5, .5, -2, -43.5);
40 pronator=moments_LK3(.5, 5, 2, .5, 1, -43.5);
41 % numbers from choosing the best from BrachProAdjust.m
42 pronator_hi=moments_LK3(.5, 5, 2, .5, 1+2.5, -43.5);
43 pronator_lo=moments_LK3(.5, 5, 2, .5, 1-2.5, -43.5);
44 title('pronator teres')
45
46 figure;
47 hold on;
48 title('Laurel Results');
49 subplot(3,2,1)
50 angle = linspace(0, 140, 29);
51 plot(angle, biceps)
52 xlabel('biceps')
53
54 subplot(3,2,2)
55 plot(angle, triceps)
56 xlabel('triceps')
57
58 subplot(3,2,3)
59 plot(angle, brach)
60 xlabel('brachialis')
61
62 subplot(3,2,4)
63 plot(angle, brd)
64 xlabel('brachioradialis')
65
66 subplot(3,2,5)
67 plot(angle, pronator)
68 xlabel('pronator')
```

APPENDIX J

HOW TO PROCESS MOMENT ARM DATA

1) Put all data in one folder. Filenames should be of the scheme:

<XZ or YA>_<fe or ps motion>_<ps or fe constant position><descriptor: Pro, Sup, Neu, fe30_, fe60_, fe90_><trial number>.txt

Example: YA_ps_fe90_7.txt was the 7^{th} trial to record from the Y and A actuators during pronation/supination motion while flexion/extension was held constant at 90 degrees.

*If you have previously processed some data in that folder, make sure to delete (or move) any generated .ppt files and the saved Matlab workspace.

2) Open the most recent "MA_processall_from<date>_curvefits.m" file. Save a copy as a new file that includes today's date in the filename.

3) Comment out the last 30 pronation/supination trials (this should be trials 6-10 for each of the 6 cases.)

4) Go through the code and set the start/stop limits for each trial to be 0 and 749, respectively.

5) Run this m-file. Do not do anything else on the computer while it is running. If you have done a lot of things in Matlab since starting it, it may be smart to re-start it before beginning.

6) When done, open up the generated .ppt files. For each file, look at where the f/e (or p/s, whichever is on the first slide for each) angle levels out. Write down approximately where. (I usually round to the nearest 25.) Also, look at the beginning – if the angle does not begin changing right away, make a note of approximately where it starts changing as well.

7) In Matlab, run findpsmins. This will output a 30-number long vector. These are the mins for each of the 30 pronation/supination trials. Write those down. The order is:

XZ_ps_fe30_(1 through 5)

XZ_ps_fe60_(1 through 5)

XZ_ps_fe90_(1 through 5)

YA_ps_fe30_(1 through 5)

YA_ps_fe60_(1 through 5)

YA_ps_fe90_(1 through 5)

** if by chance supination was negative, use findpsmaxes instead. THEN, in the run file, change the "MA_process_withpot_curvefit_6" to "MA_process_withpot_curvefit_6psneg" EVERY TIME it's called.

8) Now, take all of the powerpoint files (and the matlab workspace) that have been saved, and either delete them or move them to a separate folder. (This is important – otherwise the new ppt files get appended to the old, and the workspace gets confused.)

9) In the MA_processall_from<today's date>_curvefits.m file:

For each trial, change the "start" and "stop" inputs.

For flexion/extension, the start will be 0 (unless you noted that motion does not begin until later, in which case, it will be that number), and the "stop" will be whatever

you wrote down when looking to see when motion stopped, minus 1 (because the function starts counting at 0.)

For pronation/supination trials: For the items listed as trials 1-5, the start will be 0 (unless you noted that motion does not begin until later, in which case it will be that number) and the stop will be "min-1", where "min" is the corresponding value from "findpsmins"' output.

For pronation/supination trials listed as 6-10, the start will be the min. value as outputted from findpsmins, and the end will be the ending point that you noted above, minus 1.

10) Then, run the MA_processall_from<today's date>_curvefits.m file again. (Again, do not do anything else on the computer while it is running.)

11) After that is done, you need to run collatemomarms_<date>_curves.m (This one is probably ok to not save-as each date, just use the most recent variation.)

12) To do the averaging and compute the standard deviations, the first step is to run "RunMomArmsForAveraging.m". Modify as needed to accommodate additional specimens.

13) Make an Excel file for each muscle and angle combination, following the model of Biceps_FE.xls. Manually truncate each sheet's data to eliminate unrealistic end-effects. Create the "SummarySheet", which will do the averaging. (It is done in Excel because Excel neglects "NaN" from the averaging, whereas MATLAB includes it.)

14) In MATLAB, Run "PostprocessMomArmAverages_5deg". You will need to change line 7 to load the correct workspace for the specimen you want to analyze.

15) Run PlotMomArmStDevEtc.m . This will make the plots with the standard deviations shown.

16) Run collate_by_type_5deg.m.

APPENDIX K

MATLAB PROCESSING CODE FOR COLLECTED MOMENT ARM DATA

K.1 COLLATE_BY_TYPE_5DEG.M

```
1 function moment_outputs=collate_by_type_5deg(fe_or_ps, measured_angles_rad,
    moment_arms, muscle, specimenID, offaxisposition)
2
3% This is teh same as collate_by_type, but does it in increments of 5
4 % degrees.
5
6% Laurel Kuxhaus, lck4@pitt.edu, laurel.k@gmail.com
7 % Allegheny General Hospital
8 % 16 August 2007
9 % This is a piece of code that will be used to make a matrix that will look
10 % like:
11
12 % [ (angle) (trial 1 moment arm) (trial 2) (trial3) (trial4) (trial 5)]
13 % It will be run four times for each specimen: for flexion, extension,
14 % pronation, and supination.
15 % The "empty" values will be filled with NaN's.
16 % Thie idea is that then these will be opened in Excel, and the actual
17 % averaging computed there so that the NaN's will be neglected.
18 % I anticipate having to "edit" out some of the edge effects manually in
19 % Excel as well.
20 % WIll then be able to (still in Excel)
21
22
23 % measured_angles will be a cell array, as will moment_arms.
24
25
26
27 % convert to degrees:
28 for mm=1:length(measured_angles_rad)
29 measured_angles{mm}=measured_angles_rad{mm}*180/pi;
30 end % end mm for
31
32 Angle_inc=5;
33
34 if fe_or_ps=='f'
```

```
35 angle_list=linspace(0, 140, 29);
36 elseif fe_or_ps=='e'
37 angle_list=linspace(0, 140, 29);
38 else
39 angle_list=linspace(-60, 60, 25);
40 end % end if.
41
42 moment_outputs=NaN*ones(length(angle_list), length(moment_arms));
43
44 moment_outputs=[angle_list', moment_outputs]; % put angle numbers in the
    first column.
45
46 num_times=length(measured_angles);
47
48 for kk=1:num_times
49 % first = Beta_PS(1)
50 %Laurel made this change, 7/11/06
51 first = round(measured_angles{kk}(1)/Angle_inc)*Angle_inc;
52 rounded_angle{kk}(1)=first; % see if that helps.
53
54 remember_index{kk}(1) = 1;
55 j = 1;
56 for i = 1:length(measured_angles{kk}) %number of rows
57 diff = abs(measured_angles{kk}(i)-first);
58 if diff >= Angle_inc
59 j = j+1;
60 remember_index{kk}(j) = i;
61 if j>1
62 first = round(measured_angles{kk}(i)/Angle_inc)*Angle_inc;
63 end
64 rounded_angle{kk}(j) = first;
65 end
66 end
67 % try this:
68 % end % end kk for
69
70 % p = size(remember_index{kk},2); %Number of columns
71 % p = length(remember_index{kk}); %Number of rows
72 p = size(remember_index{kk},2); %Number of cols?
73
74 % for qq=1:num_times
75 % qq
76 % LK - this part is going to be different.
77
78 for i = 1:p
79
80 % want the values to go in the remember_index's value row, and the
81 % kk+1th column.
82
83 if fe_or_ps=='f'|fe_or_ps=='e'
84 if round(measured_angles{kk}(remember_index{kk}(i)))>0 &&
    round(measured_angles{kk}(remember_index{kk}(i)))<=141
85
86 moment_outputs(round(measured_angles{kk}(remember_index{kk}(i))),
    kk+1)=moment_arms{kk}(remember_index{kk}(i));
87
88 end % end round if.
```

```
89 else % for pronation/supination, have to add 60 to the row number.
90 if round(measured_angles{kk}(remember_index{kk}(i)))>-61 &&
    round(measured_angles{kk}(remember_index{kk}(i)))<=60
91
92 moment_outputs(round(measured_angles{kk}(remember_index{kk}(i)))+13,
    kk+1)=moment_arms{kk}(remember_index{kk}(i));
93
94 end
95 end
96
97
98 end % end i=1:p for
99
100
101
102 end %old kk for
103
104
105 output_file = strcat(muscle, specimenID, '_', fe_or_ps,'_',
    offaxisposition, '.out'); %name of the output file
106 %will be the name of the master file.out
107
108 % try just using dlmwrite
109
110 dlmwrite(output_file, moment_outputs, '\t');
```

K.2 FROM COLLATEMOMARMS_070720_CURVE.M

For brevity, the excerpts below show representative example sections from the program collatemomarms_070720_curve.m.

```
1 % Laurel Kuxhaus
2 % collatemomarms_070328curve.m
3 % 02 April 2007
4
5% This is a short code to make nic epowerpoints of things by type.
6
7load momentarms_cf
8
9% f/e limits
10 brach_ylim=([-50 30]);
11 bic_ylim=([-70 0]);
12 tri_ylim=([0 30]);
13 pro_ylim=([-30 10]);
14
15 % p/s limits
16 brach_ylimps=([-20 20]);
17 bic_ylimps=([-20 20]);
18 tri_ylimps=([-10 10]);
19 pro_ylimps=([-30 20]);
20
21 % Brachialis, Neutral, as individual plots:
22
```

```
23 for ii=1:(length(biceps_momarm_xz_fe_psNeu))
24 figure;
25 plot(feang_xz_fe_psNeu{ii}*180/pi, biceps_momarm_xz_fe_psNeu{ii});
26 xlabel('flexion angle in degrees')
27 ylabel('biceps moment arm, in mm');
28 title(strcat(num2str(ii), ' p/s Neutral'));
29 ylim(bic_ylim);
30 end % end for
31
32 for ii=1:(length(biceps_momarm_xz_fe_psPro))
33 figure;
34 plot(feang_xz_fe_psPro{ii}*180/pi, biceps_momarm_xz_fe_psPro{ii});
35 xlabel('flexion angle in degrees')
36 ylabel('biceps moment arm, in mm');
37 title(strcat(num2str(ii), ' p/s Pronated'));
38 ylim(bic_ylim);
39 end % end for
….
89
90 alltoppt('triceps_fe_all.ppt');
91
92 close all;
93
94
95 % Biceps as individual plots:
96
97 for ii=1:(length(brach_momarm_ya_fe_psNeu))
98 figure;
99 plot(feang_ya_fe_psNeu{ii}*180/pi, brach_momarm_ya_fe_psNeu{ii});
100 xlabel('flexion angle in degrees')
101 ylabel('brachialis moment arm, in mm');
102 title(strcat(num2str(ii), ' p/s Neutral'));
103 ylim(brach_ylim);
104 end % end for
105
106 for ii=1:(length(brach_momarm_ya_fe_psPro))
107 figure;
108 plot(feang_ya_fe_psPro{ii}*180/pi, brach_momarm_ya_fe_psPro{ii});
109 xlabel('flexion angle in degrees')
110 ylabel('brachialis moment arm, in mm');
111 title(strcat(num2str(ii), ' p/s Pronated'));
112 ylim(brach_ylim);
113 end % end for
…
158 alltoppt('pronator_fe_all.ppt');
159
160 close all;
161
162 % set up a colors cell
163 mystyles={'b-', 'g-','r-','c-','m-','y-','k-', 'b:',
    'g:','r:','c:','m:','y:','k:'};
164
165 % set up one to do pairs.
166 mystyles_pairs={'b-', 'b:', 'g-', 'g:', 'r-', 'r:', 'c-', 'c:', 'm-',
    'm:', 'k-','k:', 'y-', 'y:'};
167
168
```

```
169 % Now do Brachialis, with 3 plots.
170 figure;
171 hold on;
172 for ii=1:(length(biceps_momarm_xz_fe_psNeu))
173 hold on;
174 plot(feang_xz_fe_psNeu{ii}*180/pi, biceps_momarm_xz_fe_psNeu{ii},
    mystyles_pairs{ii});
175 xlabel('flexion angle in degrees')
176 ylabel('biceps moment arm, in mm');
177 title(strcat(num2str(ii), ' p/s Neutral'));
178 ylim(bic_ylim);
179 end % end for
180
181
182 figure;
183 hold on;
184 for ii=1:(length(biceps_momarm_xz_fe_psPro))
185 hold on;
186 plot(feang_xz_fe_psPro{ii}*180/pi,
    biceps_momarm_xz_fe_psPro{ii},mystyles_pairs{ii});
187 xlabel('flexion angle in degrees')
188 ylabel('biceps moment arm, in mm');
189 title(strcat(num2str(ii), ' p/s Pronated'));
190 ylim(bic_ylim);
191 end % end for
…
242 alltoppt('triceps_fe_bytype.ppt');
243
244 close all;
245
246 % Biceps by type:
247 figure; hold on;
248 for ii=1:(length(brach_momarm_ya_fe_psNeu))
249 hold on;
250 plot(feang_ya_fe_psNeu{ii}*180/pi, brach_momarm_ya_fe_psNeu{ii},
    mystyles_pairs{ii});
251 xlabel('flexion angle in degrees')
252 ylabel('brachialis moment arm, in mm');
253 title(strcat(num2str(ii), ' p/s Neutral'));
254 ylim(brach_ylim);
255 end % end for
256
257
258 figure; hold on;
259 for ii=1:(length(brach_momarm_ya_fe_psPro))
260 hold on;
261 plot(feang_ya_fe_psPro{ii}*180/pi, brach_momarm_ya_fe_psPro{ii},
    mystyles_pairs{ii});
262 xlabel('flexion angle in degrees')
263 ylabel('brachialis moment arm, in mm');
264 title(strcat(num2str(ii), ' p/s Pronated'));
265 ylim(brach_ylim);
266 end % end for
…
314 alltoppt('pronator_fe_bytype.ppt');
315 close all;
316
```

```
317 % Now do Brachialis, with 1 plot.
318 figure;
319 hold on;
320 for ii=1:(length(biceps_momarm_xz_fe_psNeu))
321 hold on;
322 plot(feang_xz_fe_psNeu{ii}*180/pi, biceps_momarm_xz_fe_psNeu{ii},
    mystyles{1});
323 end % end for
324
325 for ii=1:(length(biceps_momarm_xz_fe_psPro))
326 hold on;
327 plot(feang_xz_fe_psPro{ii}*180/pi,
    biceps_momarm_xz_fe_psPro{ii},mystyles{2});
328 end % end for
329
330 for ii=1:(length(biceps_momarm_xz_fe_psSup))
331 hold on;
332 plot(feang_xz_fe_psSup{ii}*180/pi,
    biceps_momarm_xz_fe_psSup{ii},mystyles{3});
333 end % end for
334 xlabel('flexion angle in degrees')
335 ylabel('biceps moment arm, in mm');
336 title(strcat(num2str(ii), ' B=Neu; G=Pro; R=Sup'));
337 ylim(bic_ylim);
338 alltoppt('biceps_fe_families.ppt');
....
```

K.3 CURVEFITANDPEAKSOFAVERAGES.M

For brevity, selections from CurveFitAndPeaksofAverages.m are shown below. The remainder of the program repeats the same content for different muscles.

```
1 % CurveFitAndPeaksOfAverages.m
2 % Laurel Kuxhaus, lck4@pitt.edu, laurel.k@gmail.com
3 % 08 September 2007
4 % Allegheny General Hospital and the University of Pittsburgh
5 % This code will read in the every-5-degree average values, pick out the
6 % peak values, and also will fit curves to each.
7
8
9 % Every5thPoint_ does this!
10 % bic_avg_fe=xlsread('Biceps_FE.xls','SummarySheet');
11 % brach_avg_fe=xlsread('Brachialis_FE.xls','SummarySheet');
12 % tri_avg_fe=xlsread('Triceps_FE.xls','SummarySheet');
13 % pron_avg_fe=xlsread('Pronator_FE.xls','SummarySheet');
14 %
15 % bic_avg_ps=xlsread('Biceps_PS.xls','SummarySheet');
16 % brach_avg_ps=xlsread('Brachialis_PS.xls','SummarySheet');
17 % tri_avg_ps=xlsread('Triceps_PS.xls','SummarySheet');
18 % pron_avg_ps=xlsread('Pronator_PS.xls','SummarySheet');
```

```
19
20 % bic_avg_ps=bic_avg_ps(7:end,:);
21 % brach_avg_ps=brach_avg_ps(7:end,:);
22 % tri_avg_ps=tri_avg_ps(7:end,:);
23 % pron_avg_ps=pron_avg_ps(7:end,:);
24
25 % REmember to take only every 5th point.
26 Every5thPoint_CF;
27
28 bic_avg_fe=bic_avg_fe5;
29 brach_avg_fe=brach_avg_fe5;
30 tri_avg_fe=tri_avg_fe5;
31 pron_avg_fe=pron_avg_fe5;
32
33 bic_avg_ps=bic_avg_ps5;
34 brach_avg_ps=brach_avg_ps5;
35 tri_avg_ps=tri_avg_ps5;
36 pron_avg_ps=pron_avg_ps5;
37
38
39
40 % Do biceps, FE, first.
41 % Recall, there are 3 cases - Pro, Sup, and Neu.
42 [s1_bic_fe_pro_max, s1_bic_fe_pro_max_place] =max(abs(bic_avg_fe(4:24,
    8)));
43 s1_bic_fe_pro_maxang=bic_avg_fe(s1_bic_fe_pro_max_place,1);
44
45 [s2_bic_fe_pro_max, s2_bic_fe_pro_max_place] =max(abs(bic_avg_fe(4:24,
    17)));
46 s2_bic_fe_pro_maxang=bic_avg_fe(s2_bic_fe_pro_max_place,1);
47
48 [s3_bic_fe_pro_max, s3_bic_fe_pro_max_place] =max(abs(bic_avg_fe(4:24,
    26)));
49 s3_bic_fe_pro_maxang=bic_avg_fe(s3_bic_fe_pro_max_place, 1);

…
71
72 % write these to excel;
73
74 bic_fe_maxmat=[s1_bic_fe_pro_maxang, s1_bic_fe_pro_max,
    s1_bic_fe_neu_maxang, s1_bic_fe_neu_max, s1_bic_fe_sup_maxang, s1_bic_fe_s
up_max;
75 s2_bic_fe_pro_maxang, s2_bic_fe_pro_max, s2_bic_fe_neu_maxang,
    s2_bic_fe_neu_max, s2_bic_fe_sup_maxang, s2_bic_fe_sup_max;
76 s3_bic_fe_pro_maxang, s3_bic_fe_pro_max, s3_bic_fe_neu_maxang,
    s3_bic_fe_neu_max, s3_bic_fe_sup_maxang, s3_bic_fe_sup_max];
77
78 dlmwrite('Biceps_FE_maxes.xls',bic_fe_maxmat, '\t')
79
80 % now deal with curvefits.
81
82 %first, MUST REMOVE ALL NaNs.
83 bic_avg_fe8=[];
84 bic_ang_fe8=[];
85 bic_avg_fe9=[];
86 bic_ang_fe9=[];
…
```

```
for ii=1:length(bic_avg_fe(:,1))
if isnan(bic_avg_fe(ii,8))==1
bic_avg_fe8=bic_avg_fe8;
bic_ang_fe8=bic_ang_fe8;
else
bic_avg_fe8=[bic_avg_fe8; bic_avg_fe(ii,8)];
bic_ang_fe8=[bic_ang_fe8; bic_avg_fe(ii,1)];
end % end if

…

coeffs_bic_fe{1}=polyfit(bic_ang_fe8, bic_avg_fe8,6);
coeffs_bic_fe{2}=polyfit(bic_ang_fe17, bic_avg_fe17,6);
coeffs_bic_fe{3}=polyfit(bic_ang_fe26, bic_avg_fe26,6);
coeffs_bic_fe{4}=polyfit(bic_ang_fe9, bic_avg_fe9,6);
coeffs_bic_fe{5}=polyfit(bic_ang_fe18, bic_avg_fe18,6);
coeffs_bic_fe{6}=polyfit(bic_ang_fe27, bic_avg_fe27,6);
coeffs_bic_fe{7}=polyfit(bic_ang_fe10, bic_avg_fe10,6);
coeffs_bic_fe{8}=polyfit(bic_ang_fe19, bic_avg_fe19,6);
coeffs_bic_fe{9}=polyfit(bic_ang_fe28, bic_avg_fe28,6);

for jj=1:9
if length(coeffs_bic_fe{jj})<6
for ii=1:6-length(coeffs_bic_fe{jj})
coeffs_bic_fe{jj}=[0 coeffs_bic_fe{jj}];
end % end ii for
end % end if
end % end jj for.

% make one big matrix.
Bic_Coeffs=[coeffs_bic_fe{1};
coeffs_bic_fe{2};
coeffs_bic_fe{3};
coeffs_bic_fe{4};
coeffs_bic_fe{5};
coeffs_bic_fe{6};
coeffs_bic_fe{7};
coeffs_bic_fe{8};
coeffs_bic_fe{9}];

% write to Excel
dlmwrite('Biceps_FE_Coeffs.xls',Bic_Coeffs, '\t')


…
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%NOW, do
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%PRONATI
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%ON _
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%SUPINATIO
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%N.

% Do biceps, FE, first.
% Recall, there are 3 cases - Pro, Sup, and Neu.
```

```
722 [s1_bic_ps_pro_max, s1_bic_ps_pro_max_place] =max(abs(bic_avg_ps(5:19,
    8)));
723 s1_bic_ps_pro_maxang=bic_avg_ps(s1_bic_ps_pro_max_place,1);
724
725 [s2_bic_ps_pro_max, s2_bic_ps_pro_max_place] =max(abs(bic_avg_ps(5:19,
    17)));
726 s2_bic_ps_pro_maxang=bic_avg_ps(s2_bic_ps_pro_max_place,1);
727
728 [s3_bic_ps_pro_max, s3_bic_ps_pro_max_place] =max(abs(bic_avg_ps(5:19,
    26)));
729 s3_bic_ps_pro_maxang=bic_avg_ps(s3_bic_ps_pro_max_place, 1);
730
731
…
751
752 % write these to excel;
753
754 bic_ps_maxmat=[s1_bic_ps_pro_maxang, s1_bic_ps_pro_max,
    s1_bic_ps_neu_maxang, s1_bic_ps_neu_max, s1_bic_ps_sup_maxang, s1_bic_p
s_sup_max;
755 s2_bic_ps_pro_maxang, s2_bic_ps_pro_max, s2_bic_ps_neu_maxang,
    s2_bic_ps_neu_max, s2_bic_ps_sup_maxang, s2_bic_ps_sup_m
ax;
756 s3_bic_ps_pro_maxang, s3_bic_ps_pro_max, s3_bic_ps_neu_maxang,
    s3_bic_ps_neu_max, s3_bic_ps_sup_maxang, s3_bic_ps_sup_m
ax];
757
758 dlmwrite('Biceps_ps_maxes.xls',bic_ps_maxmat, '\t')
759
760 % now deal with curvefits.
761
762 %first, MUST REMOVE ALL NaNs.
763 bic_avg_ps8=[];
764 bic_ang_ps8=[];
765 bic_avg_ps9=[];
766 bic_ang_ps9=[];
767 bic_avg_ps10=[];
768 bic_ang_ps10=[];
769 bic_avg_ps17=[];
770 bic_ang_ps17=[];
771 bic_avg_ps18=[];
772 bic_ang_ps18=[];
773 bic_avg_ps19=[];
774 bic_ang_ps19=[];
775 bic_avg_ps26=[];
776 bic_ang_ps26=[];
777 bic_avg_ps27=[];
778 bic_ang_ps27=[];
779 bic_avg_ps28=[];
780 bic_ang_ps28=[];
781 for ii=1:length(bic_avg_ps(:,1))
782 if isnan(bic_avg_ps(ii,8))==1
783 bic_avg_ps8=bic_avg_ps8;
784 bic_ang_ps8=bic_ang_ps8;
785 else
786 bic_avg_ps8=[bic_avg_ps8; bic_avg_ps(ii,8)];
787 bic_ang_ps8=[bic_ang_ps8; bic_avg_ps(ii,1)];
```

```
788 end % end if
789
790 if isnan(bic_avg_ps(ii,9))==1
791 bic_avg_ps9=bic_avg_ps9;
792 bic_ang_ps9=bic_ang_ps9;
793 else
794 bic_avg_ps9=[bic_avg_ps9; bic_avg_ps(ii,9)];
795 bic_ang_ps9=[bic_ang_ps9; bic_avg_ps(ii,1)];
796 end % end if
797
798 if isnan(bic_avg_ps(ii,10))==1
799 bic_avg_ps10=bic_avg_ps10;
800 bic_ang_ps10=bic_ang_ps10;
801 else
802 bic_avg_ps10=[bic_avg_ps10; bic_avg_ps(ii,10)];
803 bic_ang_ps10=[bic_ang_ps10; bic_avg_ps(ii,1)];
804 end % end if
805
806 if isnan(bic_avg_ps(ii,26))==1
807 bic_avg_ps26=bic_avg_ps26;
808 bic_ang_ps26=bic_ang_ps26;
809 else
810 bic_avg_ps26=[bic_avg_ps26; bic_avg_ps(ii,26)];
811 bic_ang_ps26=[bic_ang_ps26; bic_avg_ps(ii,1)];
812 end % end if
813
814 if isnan(bic_avg_ps(ii,27))==1
815 bic_avg_ps27=bic_avg_ps27;
816 bic_ang_ps27=bic_ang_ps27;
817 else
818 bic_avg_ps27=[bic_avg_ps27; bic_avg_ps(ii,27)];
819 bic_ang_ps27=[bic_ang_ps27; bic_avg_ps(ii,1)];
820 end % end if
821
822 if isnan(bic_avg_ps(ii,28))==1
823 bic_avg_ps28=bic_avg_ps28;
824 bic_ang_ps28=bic_ang_ps28;
825 else
826 bic_avg_ps28=[bic_avg_ps28; bic_avg_ps(ii,28)];
827 bic_ang_ps28=[bic_ang_ps28; bic_avg_ps(ii,1)];
828 end % end if
829
830 if isnan(bic_avg_ps(ii,17))==1
831 bic_avg_ps17=bic_avg_ps17;
832 bic_ang_ps17=bic_ang_ps17;
833 else
834 bic_avg_ps17=[bic_avg_ps17; bic_avg_ps(ii,17)];
835 bic_ang_ps17=[bic_ang_ps17; bic_avg_ps(ii,1)];
836 end % end if
837
838 if isnan(bic_avg_ps(ii,18))==1
839 bic_avg_ps18=bic_avg_ps18;
840 bic_ang_ps18=bic_ang_ps18;
841 else
842 bic_avg_ps18=[bic_avg_ps18; bic_avg_ps(ii,18)];
843 bic_ang_ps18=[bic_ang_ps18; bic_avg_ps(ii,1)];
844 end % end if
```

```
845
846 if isnan(bic_avg_ps(ii,19))==1
847 bic_avg_ps19=bic_avg_ps19;
848 bic_ang_ps19=bic_ang_ps19;
849 else
850 bic_avg_ps19=[bic_avg_ps19; bic_avg_ps(ii,19)];
851 bic_ang_ps19=[bic_ang_ps19; bic_avg_ps(ii,1)];
852 end % end if
853 end % end for
854
855 coeffs_bic_ps{1}=polyfit(bic_ang_ps8, bic_avg_ps8,6);
856 coeffs_bic_ps{2}=polyfit(bic_ang_ps17, bic_avg_ps17,6);
857 coeffs_bic_ps{3}=polyfit(bic_ang_ps26, bic_avg_ps26,6);
858 coeffs_bic_ps{4}=polyfit(bic_ang_ps9, bic_avg_ps9,6);
859 coeffs_bic_ps{5}=polyfit(bic_ang_ps18, bic_avg_ps18,6);
860 coeffs_bic_ps{6}=polyfit(bic_ang_ps27, bic_avg_ps27,6);
861 coeffs_bic_ps{7}=polyfit(bic_ang_ps10, bic_avg_ps10,6);
862 coeffs_bic_ps{8}=polyfit(bic_ang_ps19, bic_avg_ps19,6);
863 coeffs_bic_ps{9}=polyfit(bic_ang_ps28, bic_avg_ps28,6);
864
865 for jj=1:9
866 if length(coeffs_bic_ps{jj})<6
867 for ii=1:6-length(coeffs_bic_ps{jj})
868 coeffs_bic_ps{jj}=[0 coeffs_bic_ps{jj}];
869 end % end ii for
870 end % end if
871 end % end jj for.
872
873 % make one big matrix.
874 Bic_Coeffs_ps=[coeffs_bic_ps{1};
875 coeffs_bic_ps{2};
876 coeffs_bic_ps{3};
877 coeffs_bic_ps{4};
878 coeffs_bic_ps{5};
879 coeffs_bic_ps{6};
880 coeffs_bic_ps{7};
881 coeffs_bic_ps{8};
882 coeffs_bic_ps{9}];
883
884 % write to Excel
885 dlmwrite('Biceps_ps_Coeffs.xls',Bic_Coeffs_ps, '\t')
886
887
…
```

K.4 EVERY5THPIONT_CF.M

```
1 % Every5thPoint.m
2 % Laurel Kuxhaus
3 % This is a quick file that will read in the sheets of averages, and keep
4 % only every 5th point.
5
```

```
6
7 bic_avg_fe=xlsread('Biceps_FE.xls','SummarySheet');
8 brach_avg_fe=xlsread('Brachialis_FE.xls','SummarySheet');
9 tri_avg_fe=xlsread('Triceps_FE.xls','SummarySheet');
10 pron_avg_fe=xlsread('Pronator_FE.xls','SummarySheet');
11
12 bic_avg_ps=xlsread('Biceps_PS.xls','SummarySheet');
13 brach_avg_ps=xlsread('Brachialis_PS.xls','SummarySheet');
14 tri_avg_ps=xlsread('Triceps_PS.xls','SummarySheet');
15 pron_avg_ps=xlsread('Pronator_PS.xls','SummarySheet');
16
17 % bic_avg_ps(:,1)
18
19 bic_avg_ps=bic_avg_ps(7:end,:);
20 brach_avg_ps=brach_avg_ps(7:end,:);
21 tri_avg_ps=tri_avg_ps(7:end,:);
22 % pron_avg_ps=pron_avg_ps(7:end,:); %for some reason, pronator doesn't
    need
23 % this.
24
25 % bic_avg_ps(:,1)
26 for ii=1:28
27 bic_avg_fe5(ii,:)=bic_avg_fe(5*ii,:);
28 brach_avg_fe5(ii,:)=brach_avg_fe(5*ii,:);
29 tri_avg_fe5(ii,:)=tri_avg_fe(5*ii,:);
30 pron_avg_fe5(ii,:)=pron_avg_fe(5*ii,:);
31 end
32 clear ii;
33
34 for ii=1:24
35 bic_avg_ps5(ii,:)=bic_avg_ps(5*ii,:);
36 brach_avg_ps5(ii,:)=brach_avg_ps(5*ii,:);
37 tri_avg_ps5(ii,:)=tri_avg_ps(5*ii,:);
38 pron_avg_ps5(ii,:)=pron_avg_ps(5*ii,:);
39 end
```

K.5 EVERY5THPOINT_INCLSTDEV.M

```
1 % Every5thPoint.m
2 % Laurel Kuxhaus
3 % This is a quick file that will read in the sheets of averages, and keep
4 % only every 5th point.
5
6
7 bic_avg_fe=xlsread('Biceps_FE.xls','SummarySheet');
8 brach_avg_fe=xlsread('Brachialis_FE.xls','SummarySheet');
9 tri_avg_fe=xlsread('Triceps_FE.xls','SummarySheet');
10 pron_avg_fe=xlsread('Pronator_FE.xls','SummarySheet');
11
12 bic_avg_ps=xlsread('Biceps_PS.xls','SummarySheet');
13 brach_avg_ps=xlsread('Brachialis_PS.xls','SummarySheet');
14 tri_avg_ps=xlsread('Triceps_PS.xls','SummarySheet');
```

```
pron_avg_ps=xlsread('Pronator_PS.xls','SummarySheet');

bic_std_fe=xlsread('Biceps_FE.xls','StDev');
brach_std_fe=xlsread('Brachialis_FE.xls','StDev');
tri_std_fe=xlsread('Triceps_FE.xls','StDev');
pron_std_fe=xlsread('Pronator_FE.xls','StDev');

bic_std_ps=xlsread('Biceps_PS.xls','StDev');
brach_std_ps=xlsread('Brachialis_PS.xls','StDev');
tri_std_ps=xlsread('Triceps_PS.xls','StDev');
pron_std_ps=xlsread('Pronator_PS.xls','StDev');

bic_cov_fe=xlsread('Biceps_FE.xls','CoV');
brach_cov_fe=xlsread('Brachialis_FE.xls','CoV');
tri_cov_fe=xlsread('Triceps_FE.xls','CoV');
pron_cov_fe=xlsread('Pronator_FE.xls','CoV');

bic_cov_ps=xlsread('Biceps_PS.xls','CoV');
brach_cov_ps=xlsread('Brachialis_PS.xls','CoV');
tri_cov_ps=xlsread('Triceps_PS.xls','CoV');
pron_cov_ps=xlsread('Pronator_PS.xls','CoV');


for ii=1:28
bic_avg_fe5(ii,:)=bic_avg_fe(5*ii,:);
brach_avg_fe5(ii,:)=brach_avg_fe(5*ii,:);
tri_avg_fe5(ii,:)=tri_avg_fe(5*ii,:);
pron_avg_fe5(ii,:)=pron_avg_fe(5*ii,:);

bic_std_fe5(ii,:)=bic_std_fe(5*ii,:);
brach_std_fe5(ii,:)=brach_std_fe(5*ii,:);
tri_std_fe5(ii,:)=tri_std_fe(5*ii,:);
pron_std_fe5(ii,:)=pron_std_fe(5*ii,:);

bic_cov_fe5(ii,:)=bic_cov_fe(5*ii,:);
brach_cov_fe5(ii,:)=brach_cov_fe(5*ii,:);
tri_cov_fe5(ii,:)=tri_cov_fe(5*ii,:);
pron_cov_fe5(ii,:)=pron_cov_fe(5*ii,:);
end
clear ii;

for ii=1:24
bic_avg_ps5(ii,:)=bic_avg_ps(5*ii,:);
brach_avg_ps5(ii,:)=brach_avg_ps(5*ii,:);
tri_avg_ps5(ii,:)=tri_avg_ps(5*ii,:);
pron_avg_ps5(ii,:)=pron_avg_ps(5*ii,:);

bic_std_ps5(ii,:)=bic_std_ps(5*ii,:);
brach_std_ps5(ii,:)=brach_std_ps(5*ii,:);
tri_std_ps5(ii,:)=tri_std_ps(5*ii,:);
pron_std_ps5(ii,:)=pron_std_ps(5*ii,:);

bic_cov_ps5(ii,:)=bic_cov_ps(5*ii,:);
brach_cov_ps5(ii,:)=brach_cov_ps(5*ii,:);
tri_cov_ps5(ii,:)=tri_cov_ps(5*ii,:);
pron_cov_ps5(ii,:)=pron_cov_ps(5*ii,:);
end
```

K.6 FITANDDERIVE_6.M

```
% fitandderive.m
% Laurel Kuxhaus, lck4@pitt.edu, laurel.k@gmail.com
% 05 April 2007
%
% This code will be used to fit the radial head data, and take the
% derivative.

function [coeffs, Rsq, deriv_vals]=fitandderive(xvals, yvals)

[coeffs svals]=polyfit(xvals,yvals, 6)

% the next few lines are from
% http://mathstat.carleton.ca/~help/matlab/help211/fitting.html , to get
% the r-squared values.
ypred = polyval(coeffs,xvals); % predictions
dev = yvals - mean(yvals); % deviations - measure of spread
SST = sum(dev.^2); % total variation to be accounted for
resid = yvals - ypred; % residuals - measure of mismatch
SSE = sum(resid.^2); % variation NOT accounted for
Rsq = 1 - SSE/SST; % percent of error explained


der_coeffs=[5*coeffs(1), 4*coeffs(2), 3*coeffs(3), 2*coeffs(4),
   coeffs(5)];

deriv_vals=polyval(der_coeffs, xvals);


% afterwards, to plot, do somethign like this:
% figure(1); hold on;
% >> plot(feang_xz_fe_psNeu{1}*180/pi,brach_momarm_xz_fe_psNeu{1},'r')
% >> ylim([-50,0])
```

K.7 MA_PLOTS_POT_WITHFITS.M

```

% Laurel Kuxhaus, lck4@pitt.edu, laurel.k@gmail.com
% Orthopaedic Biomechanics Laboratory, University of Pittsburgh and
% Allegheny General Hospital
% 28 January 2007

% This code will make some plots for the moment arms.

function outputs=MA_plots_pot_withfits(ang_fe, ang_ps, encod_x, encod_z,
   drdth_fe_x, drdth_fe_z, feps, xory, RorL, coeffstoplot)
% encod_x and encod_z are just names - they can hold the data for encoders
% y and a just as easily.
outputs=3; % just because.
fe_ang=ang_fe*180/pi;
```

```
14 ps_ang=ang_ps*180/pi;
15
16 %%% NEED to change labels depending on if it's fe or ps, xza or yza
17
18 % for now (1/28/07), let's say that feps will always be fe. So if we want
19 % to use the ps part later, we'll have to change the code then.
20
21 % figure(10)
22 % subplot(2,1,1)
23 % % plot(ang_fe_raw)
24 % ylabel('ang_fe')
25 % % hold on;
26 % plot(ang_fe,'r');
27 % subplot(2,1,2)
28 % % plot(ang_ps_raw)
29 % % hold on;
30 % plot(ang_ps,'r');
31 % ylabel('ang_ps')
32 %
33 % figure(11)
34 % subplot(3,1,1)
35 % % plot(encod_x_raw)
36 % % hold on;
37 % plot(encod_x,'r')
38 % ylabel('encoder x')
39 % subplot(3,1,2)
40 % plot(encod_z_raw)
41 % % hold on;
42 % plot(encod_z,'r')
43 % ylabel('encoder z')
44 % subplot(3,1,3)
45 % % plot(encod_a_raw)
46 % % hold on;
47 % plot(encod_a,'r')
48 % ylabel('encoder a')
49
50 %
51 % figure;
52 % % plot(encod_x_raw)
53 % % hold on
54 % plot(encod_x,'r')
55
56 % Set up dummy x values to plot against.
57 for ii=1:length(fe_ang)
58 fe_xvalues(ii)=ii;
59 end
60 for ii=1:length(ps_ang)
61 ps_xvalues(ii)=ii;
62 end
63
64 if xory=='x'
65 figure;
66 % subplot(2,1,1)
67 if feps=='fe'
68 plot(fe_ang);
69 hold on;
70 plot(polyval(coeffstoplot{3},fe_xvalues)*180/pi,'r')
```

```
71 ylabel('flexion-extension angle');
72 else
73 plot(ps_ang);
74 hold on;
75 plot(polyval(coeffstoplot{3},ps_xvalues)*180/pi,'r')
76 ylabel('pronation-supination angle');
77 end % end feps if.
78
79 % if feps=='fe'
80 figure;
81 plot(encod_x);
82 hold on;
83 plot(polyval(coeffstoplot{1},fe_xvalues),'r')
84 if RorL=='R'
85 ylabel('biceps encoder position');
86 else
87 ylabel('brachialis encoder position');
88 end % end RorL if.
89
90 figure;
91 plot(encod_z);
92 hold on;
93 plot(polyval(coeffstoplot{2},fe_xvalues),'r')
94
95 ylabel('pronator teres encoder position');
96
97
98 % else
99 % plot(ps_ang);
100 % hold on;
101 % plot(polyval(coeffstoplot{1},ps_xvalues)*180/pi,'r')
102 % ylabel('encoder position');
103 % end % end feps if.
104 % hold on;
105 % subplot(2,1,2)
106 % plot(ps_ang);
107 % ylabel('pronation-supination angle');
108
109
110
111 figure;
112 % plot(fe_ang, drdth_fe_xf);
113 if feps=='fe'
114 plot(fe_ang, drdth_fe_x');
115 xlabel('flexion-extension angle')
116 else
117 plot(ps_ang, drdth_fe_x')
118 xlabel('pronation-supination angle')
119 end % end feps if.
120
121
122 if RorL=='L'
123 ylabel('brachialis moment arm in mm')
124 else
125 ylabel('biceps moment arm in mm')
126 end % end if.
127 ylim([-50 50]);
```

```
128
129 figure;
130 % plot(fe_ang, drdth_fe_xf);
131 if feps=='fe'
132 plot(fe_ang, encod_x);
133
134 else
135 plot(ps_ang, encod_x);
136
137 end % end feps if.
138 title('excursion vs. angle')
139 xlabel('angle (degrees)')
140 if RorL=='L'
141 ylabel('brachialis excursion (mm)')
142 else
143 ylabel('biceps excursion (mm)')
144 end % end if.
145
146
147 figure;
148 % plot(fe_ang, drdth_fe_xf);
149 if feps=='fe'
150 plot(fe_ang, drdth_fe_z'); % 600 is for test file only!
151 xlabel('flexion-extension angle')
152 else
153 plot(ps_ang, drdth_fe_z')
154 xlabel('pronation-supination angle')
155 end % end feps if.
156
157 % if xory=='x'
158 ylabel('pronator teres moment arm in mm')
159 % else
160 % % ylabel('triceps moment arm in mm')
161 % end % end if
162 ylim([-50 50]);
163
164 figure;
165 % plot(fe_ang, drdth_fe_xf);
166 if feps=='fe'
167 plot(fe_ang, encod_z);
168
169 else
170 plot(ps_ang, encod_z);
171
172 end % end feps if.
173 title('excursion vs. angle')
174 xlabel('angle (degrees)')
175 ylabel('pronator teres excursion (mm)')
176
177
178 % figure(4);
179 % % plot(fe_ang, drdth_fe_xf);
180 % plot(fe_ang, drdth_fe_a');
181 % xlabel('flexion-extension angle')
182 % ylabel('triceps moment arm in mm')
183 % ylim([-100 100]);
184
```

```
185 else
186 figure;
187 % subplot(2,1,1)
188 if feps=='fe'
189 plot(fe_ang);
190 hold on;
191 plot(polyval(coeffstoplot{3},fe_xvalues)*180/pi,'r')
192 ylabel('flexion-extension angle');
193 else
194 plot(ps_ang);
195 hold on;
196 plot(polyval(coeffstoplot{3},ps_xvalues)*180/pi,'r')
197 ylabel('pronation-supination angle');
198 end % end feps if.
199 % hold on;
200 % subplot(2,1,2)
201 % plot(ps_ang);
202 % ylabel('pronation-supination angle');
203
204 % if feps=='fe'
205 figure;
206 plot(encod_x);
207 hold on;
208 plot(polyval(coeffstoplot{1},fe_xvalues),'r')
209 if RorL=='R'
210 ylabel('brachialis encoder position');
211 else
212 ylabel('biceps encoder position');
213 end % end RorL if.
214
215 figure;
216 plot(encod_z);
217 hold on;
218 plot(polyval(coeffstoplot{2},fe_xvalues),'r')
219
220 ylabel('triceps encoder position');
221
222
223
224
225
226 figure;
227 % plot(fe_ang, drdth_fe_xf);
228 if feps=='fe'
229 plot(fe_ang, drdth_fe_x');
230 xlabel('flexion-extension angle')
231 else
232 plot(ps_ang, drdth_fe_x');
233 xlabel('pronation-supination angle');
234 end % end feps if.
235
236 if RorL=='L'
237 ylabel('biceps moment arm in mm')
238 else
239 ylabel('brachialis moment arm in mm')
240 end % end if
241 ylim([-50 50]);
```

```
242
243 figure;
244 % plot(fe_ang, drdth_fe_xf);
245 if feps=='fe'
246 plot(fe_ang, encod_x);
247 else
248 plot(ps_ang, encod_x);
249 end
250
251 title('excursion vs. angle')
252 xlabel('angle (degrees)')
253 if RorL=='L'
254 ylabel('biceps excursion (mm)')
255 else
256 ylabel('brachialis excursion (mm)')
257 end % end if.
258
259 figure;
260 % plot(fe_ang, drdth_fe_xf);
261 if feps=='fe'
262 plot(fe_ang, drdth_fe_z'); % 600 is for test file only!
263 xlabel('flexion-extension angle')
264 else
265 plot(ps_ang, drdth_fe_z');
266 xlabel('pronation-supination angle');
267 end % end feps if.
268
269 % % if xory=='x'
270 ylabel('pronator teres moment arm in mm')
271 % else
272 % ylabel('triceps moment arm in mm')
273 % end % end xory if
274
275 ylim([-50 50]);
276
277 figure;
278 % plot(fe_ang, drdth_fe_xf);
279 if feps=='fe'
280 plot(fe_ang, encod_z);
281
282 else
283 plot(ps_ang, encod_z);
284
285 end % end feps if.
286
287 title('excursion vs. angle')
288 xlabel('angle (degrees)')
289
290 % if xory=='x'
291 % ylabel('pronator teres excursion (mm)')
292 % else
293 ylabel('triceps excursion (mm)')
294 % end % end xory if.
295
296
297 % figure(4);
298 % % plot(fe_ang, drdth_fe_xf);
```

```
299 % plot(fe_ang, drdth_fe_a');
300 % xlabel('flexion-extension angle')
301 % ylabel('triceps moment arm in mm')
302 % ylim([-100 100]);
303
304 end % end if
```

K.8 MA_PROCESS_WITHPOT_CURVEFIT_6.M

```
1 % MA_process_withpot.m
2 % from
3 % MA_process.m
4 % Laurel Kuxhaus, lck4@pitt.edu, laurel.k@gmail.com
5 % Orthopaedic Biomechanics Laboratory, Allegheny General Hospital and
6 % University of Pittsburgh.
7
8 % This modification of the code is to process using potentiometer data
9 % to get the elbow angles (ie: not using Vicon.)
10
11 % This is the shell program to run for everything.
12
13 function [acro_data, fe_ang, ps_ang, momentarms, Rsqs, coeffs_first,
   coeffs_second, coeffs_th]=MA_process_withpot_curvefit_6(acro_file, xory,
   fep
s, RorL, start, stop)
14
15
16 start = start;
17 stop=stop;
18
19
20
21 acro_data=dlmread(acro_file,'\t',[start 0 stop 10]); % add in the range
   here so that we can stop futzing around with editing it in Excel.
22
23 fe_ang=acro_data(:,9)*pi/180; % I think that's the correct column. in
   radians.
24 ps_ang=acro_data(:,10)*pi/180; % I think that's the correct column.
25
26 encod_x=acro_data(:,5)*25.4/16000; % convert to mm.
27 encod_y=acro_data(:,6)*25.4/16000;
28 encod_z=acro_data(:,7)*25.4/16000;
29 encod_a=acro_data(:,8)*25.4/16000;
30
31
32 % calculate moment arms
33
34 for ii=1:length(fe_ang)
35 xvalues(ii)=ii;
36 end
37
38 if feps=='fe'
```

```
39
40 [coeffs_th, Rsq_th, deriv_vals_th]=fitandderive_6(xvalues', fe_ang); % the
    angle is in radians, whew!
41 else
42
43
44 [coeffs_th, Rsq_th, deriv_vals_th]=fitandderive_6(xvalues', ps_ang); % the
    angle is in radians, whew!
45 end % end feps if.
46
47 if xory=='x'
48
49 [coeffs_x, Rsq_x, deriv_vals_x]=fitandderive_6(xvalues', encod_x);
50 [coeffs_z, Rsq_z, deriv_vals_z]=fitandderive_6(xvalues', encod_z);
51
52
53 dxdth=deriv_vals_x./deriv_vals_th;
54 dzdth=deriv_vals_z./deriv_vals_th;
55
56 momentarms=[dxdth, dzdth];
57 Rsqs=[Rsq_x, Rsq_z, Rsq_th];
58 coeffstoplot={coeffs_x, coeffs_z, coeffs_th};
59 coeffs_first=coeffs_x;
60 coeffs_second = coeffs_z;
61
62 else % xory if
63 [coeffs_y, Rsq_y, deriv_vals_y]=fitandderive_6(xvalues', encod_y);
64 [coeffs_a, Rsq_a, deriv_vals_a]=fitandderive_6(xvalues', encod_a);
65
66
67 dydth=deriv_vals_y./deriv_vals_th;
68 dadth=deriv_vals_a./deriv_vals_th;
69
70 momentarms=[dydth, dadth];
71 Rsqs=[Rsq_y, Rsq_a, Rsq_th];
72
73 coeffstoplot={coeffs_y, coeffs_a, coeffs_th};
74 coeffs_first=coeffs_y;
75 coeffs_second = coeffs_a;
76
77 end % end xory if
78
79
80
81
82 % else
83 % momentarms=MomArms_YZA_vicon(acro_final, fe_ang, ps_ang, h, feps);
84 % end % end xory if.
85
86
87
88 % make plots.
89 if xory=='x'
90 MA_plots_pot_withfits(fe_ang, ps_ang, encod_x, encod_z, momentarms(:,1),
    momentarms(:,2), feps, xory, RorL, coeffstoplot); % THIS TOO!
91 'hello'
92 else
```

```
93 MA_plots_pot_withfits(fe_ang, ps_ang, encod_y, encod_a, momentarms(:,1),
    momentarms(:,2), feps, xory, RorL, coeffstoplot); % THIS TOO!
94 end % end if.
95
96
97 % once this works to here, should probably save the workspace, and maybe
98 % export the results to a file of some sort.
99 %***** If saving everything, be sure to save the x's as y's for YZA!
100
```

K.9 MA_PROCESSALL.M

The following is an excerpt from the MA_ProcessAll.m program. The remainder of the program iterates this portion of the code for the different files of collected data.

```
1 % MA_processall_from070720_curvefits_6thorder.m
2 % Laurel Kuxhaus, lck4@pitt.edu, laurel.k@gmail.com
3 % Orthopaedic Biomechanics Laboratory, Allegheny General Hospital and
4 % University of Pittsburgh
5 % 29 March 2007 - original; 05 April 2007 - modify to work with curvefits.
6 %
7 % This code will process the moment arm files from yesterday's elbow.
89
% The data is:
10 % flexion/extension at 3 kinds of p/s (Sup, Pro, Neu);
11 % pronation/supination at 3 different f/e angles (30, 60, 90)
12
13 [acro_data, fe_ang, ps_ang, momentarms, Rsqs, coeffs_x, coeffs_z,
    coeffs_th]=MA_process_withpot_curvefit_6psneg('XZ_fe_psNeu1.txt', 'x', '
fe', 'R', 30, 374) ;
14 acro_data_xz_fe_psNeu{1}=acro_data;
15
16 feang_xz_fe_psNeu{1}=fe_ang;
17 ps_ang_xz_fe_psNeu{1}=ps_ang;
18 biceps_momarm_xz_fe_psNeu{1}=momentarms(:,1);
19 pronator_momarm_xz_fe_psNeu{1}=momentarms(:,2);
20 Rsqs_xz_fe_psNeu{1}=Rsqs;
21 biceps_xcoeffs_xz_fe_psNeu{1}=coeffs_x;
22 pronator_coeffs_xz_fe_psNeu{1}=coeffs_z;
23 ang_coeffs_xz_fe_psNeu{1}=coeffs_th;
24 save 'momentarms_cf.mat'
25
26 alltoppt('XZ_fe_psNeu1_cfit.ppt')
27
28 clear all; close all;
29 load momentarms_cf;
30
31 [acro_data, fe_ang, ps_ang, momentarms, Rsqs, coeffs_x, coeffs_z,
    coeffs_th]=MA_process_withpot_curvefit_6psneg('XZ_fe_psNeu2.txt', 'x', '
fe', 'R', 45, 349) ;
```

```
32 acro_data_xz_fe_psNeu{2}=acro_data;
33
34 feang_xz_fe_psNeu{2}=fe_ang;
35 ps_ang_xz_fe_psNeu{2}=ps_ang;
36 biceps_momarm_xz_fe_psNeu{2}=momentarms(:,1);
37 pronator_momarm_xz_fe_psNeu{2}=momentarms(:,2);
38 Rsqs_xz_fe_psNeu{2}=Rsqs;
39 biceps_xcoeffs_xz_fe_psNeu{2}=coeffs_x;
40 pronator_coeffs_xz_fe_psNeu{2}=coeffs_z;
41 ang_coeffs_xz_fe_psNeu{2}=coeffs_th;
42 save 'momentarms_cf.mat'
43
44 alltoppt('XZ_fe_psNeu2_cfit.ppt')
45
46 clear all; close all;
47 load momentarms_cf;
```

K.10 PLOTMOMARMSTDEVETC.M

The following are excerpts form the MATLAB program PlotMomArmsStdevEtc.m . The remainder of the code (not shown here)

```
1 % PlotMomArmStDevEtc.m
2 % Laurel Kuxhaus, lck4@pitt.edu and laurel.k@gmail.com
3 % 20 August 2007
4 % Allegheny General Hospital and the University of Pittsburgh
5
6 % This will read in the averages and stdev's, and will make some plots.
7 bic_avg_fe=xlsread('Biceps_FE.xls','SummarySheet');
8 brach_avg_fe=xlsread('Brachialis_FE.xls','SummarySheet');
9 tri_avg_fe=xlsread('Triceps_FE.xls','SummarySheet');
10 pron_avg_fe=xlsread('Pronator_FE.xls','SummarySheet');
11
12 bic_avg_ps=xlsread('Biceps_PS.xls','SummarySheet');
13 brach_avg_ps=xlsread('Brachialis_PS.xls','SummarySheet');
14 tri_avg_ps=xlsread('Triceps_PS.xls','SummarySheet');
15 pron_avg_ps=xlsread('Pronator_PS.xls','SummarySheet');
16
17 Every5thPoint_inclStdev; % this will make the ones that end in "5".
18
19 % f/e limits
20 brach_ylim=([0 60]);
21 bic_ylim=([0 60]);
22 tri_ylim=([-40 0]);
23 pro_ylim=([0 40]);
24
25 % p/s limits
26 brach_ylimps=([-20 20]);
27 bic_ylimps=([-20 20]);
28 tri_ylimps=([-10 10]);
```

```
29 pro_ylimps=([-20 30]);
30
31 xlimit=([0 140]);
32 xlimitps=([-60 60]);
33
34
35
36 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
37 % now do just the combined averages
38 close all; % get rid of previous figures.
39
40 % biceps, flexion/extension;
41 figure;
42 hold on;
43 plot(bic_avg_fe5(:,1), -bic_avg_fe5(:,8),'g-', 'LineWidth', 4);
44 plot(bic_avg_fe5(:,1), -bic_avg_fe5(:,9),'r-', 'LineWidth', 4);
45 plot(bic_avg_fe5(:,1), -bic_avg_fe5(:,10),'b-', 'LineWidth', 4);
46
47 plot(bic_avg_fe5(:,1), -bic_avg_fe5(:,8)+bic_std_fe5(:,8),'g-',
    'LineWidth', 1);
48 plot(bic_avg_fe5(:,1), -bic_avg_fe5(:,9)+bic_std_fe5(:,9),'r-',
    'LineWidth', 1);
49 plot(bic_avg_fe5(:,1), -bic_avg_fe5(:,10)+bic_std_fe5(:,10),'b-',
    'LineWidth', 1);
50
51 plot(bic_avg_fe5(:,1), -bic_avg_fe5(:,8)-bic_std_fe5(:,8),'g-',
    'LineWidth', 1);
52 plot(bic_avg_fe5(:,1), -bic_avg_fe5(:,9)-bic_std_fe5(:,9),'r-',
    'LineWidth', 1);
53 plot(bic_avg_fe5(:,1), -bic_avg_fe5(:,10)-bic_std_fe5(:,10),'b-',
    'LineWidth', 1);
54
55 title('Specimen 1, Flexion-extension moment arms. GRB=Pro,Sup, Neu');
56 xlabel('f/e angle in degrees')
57 ylabel('biceps moment arm, in mm');
58 ylim(bic_ylim);
59 xlim(xlimit);
60
61
…
303 %!!!!!!!!!!!!!!!!!Pronation/supination now.
304
305 % biceps, flexion/extension;
306 figure;
307 hold on;
308
309 plot(bic_avg_ps5(:,1), bic_avg_ps5(:,8),'b-', 'LineWidth', 4);
310 plot(bic_avg_ps5(:,1), bic_avg_ps5(:,9),'g-', 'LineWidth', 4);
311 plot(bic_avg_ps5(:,1), bic_avg_ps5(:,10),'r-', 'LineWidth', 4);
312
313 plot(bic_avg_ps5(:,1), bic_avg_ps5(:,8)+bic_std_ps5(:,8),'b-',
    'LineWidth', 1);
314 plot(bic_avg_ps5(:,1), bic_avg_ps5(:,9)+bic_std_ps5(:,9),'g-',
    'LineWidth', 1);
315 plot(bic_avg_ps5(:,1), bic_avg_ps5(:,10)+bic_std_ps5(:,10),'r-',
    'LineWidth', 1);
316
```

```
317 plot(bic_avg_ps5(:,1), bic_avg_ps5(:,8)-bic_std_ps5(:,8),'b-',
    'LineWidth', 1);
318 plot(bic_avg_ps5(:,1), bic_avg_ps5(:,9)-bic_std_ps5(:,9),'g-',
    'LineWidth', 1);
319 plot(bic_avg_ps5(:,1), bic_avg_ps5(:,10)-bic_std_ps5(:,10),'r-',
    'LineWidth', 1);
320
321
322 title('Specimen 1, p/s moment arms. BGR=30,60.90;');
323 xlabel('f/e angle in degrees')
324 ylabel('biceps moment arm, in mm');
325 ylim(bic_ylimps);
326 xlim(xlimitps);
327
…
572 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
573 % now do just the combined averages
574 close all; % get rid of previous figures.
575
576 % biceps, flexion/extension;
577 figure;
578 hold on;
579 plot(bic_avg_fe5(:,1), -bic_avg_fe5(:,8),'g-', 'LineWidth', 4);
580 plot(bic_avg_fe5(:,1), -bic_avg_fe5(:,9),'r-', 'LineWidth', 4);
581 plot(bic_avg_fe5(:,1), -bic_avg_fe5(:,10),'b-', 'LineWidth', 4);
582
583 plot(bic_avg_fe5(:,1), -bic_avg_fe5(:,8)+bic_cov_fe5(:,8),'g-',
    'LineWidth', 1);
584 plot(bic_avg_fe5(:,1), -bic_avg_fe5(:,9)+bic_cov_fe5(:,9),'r-',
    'LineWidth', 1);
585 plot(bic_avg_fe5(:,1), -bic_avg_fe5(:,10)+bic_cov_fe5(:,10),'b-',
    'LineWidth', 1);
586
587 plot(bic_avg_fe5(:,1), -bic_avg_fe5(:,8)-bic_cov_fe5(:,8),'g-',
    'LineWidth', 1);
588 plot(bic_avg_fe5(:,1), -bic_avg_fe5(:,9)-bic_cov_fe5(:,9),'r-',
    'LineWidth', 1);
589 plot(bic_avg_fe5(:,1), -bic_avg_fe5(:,10)-bic_cov_fe5(:,10),'b-',
    'LineWidth', 1);
590
591 title('Specimen 1, Flexion-extension moment arms. GRB=Pro,Sup, Neu');
592 xlabel('f/e angle in degrees')
593 ylabel('biceps moment arm, in mm');
594 ylim(bic_ylim);
595 xlim(xlimit);
596
597


…
839 %!!!!!!!!!!!!!!!!!Pronation/supination now.
840
841 % biceps, flexion/extension;
842 figure;
843 hold on;
844
845 plot(bic_avg_ps5(:,1), bic_avg_ps5(:,8),'b-', 'LineWidth', 4);
846 plot(bic_avg_ps5(:,1), bic_avg_ps5(:,9),'g-', 'LineWidth', 4);
```

```
847 plot(bic_avg_ps5(:,1), bic_avg_ps5(:,10),'r-', 'LineWidth', 4);
848
849 plot(bic_avg_ps5(:,1), bic_avg_ps5(:,8)+bic_cov_ps5(:,8),'b-',
    'LineWidth', 1);
850 plot(bic_avg_ps5(:,1), bic_avg_ps5(:,9)+bic_cov_ps5(:,9),'g-',
    'LineWidth', 1);
851 plot(bic_avg_ps5(:,1), bic_avg_ps5(:,10)+bic_cov_ps5(:,10),'r-',
    'LineWidth', 1);
852
853 plot(bic_avg_ps5(:,1), bic_avg_ps5(:,8)-bic_cov_ps5(:,8),'b-',
    'LineWidth', 1);
854 plot(bic_avg_ps5(:,1), bic_avg_ps5(:,9)-bic_cov_ps5(:,9),'g-',
    'LineWidth', 1);
855 plot(bic_avg_ps5(:,1), bic_avg_ps5(:,10)-bic_cov_ps5(:,10),'r-',
    'LineWidth', 1);
856
857
858 title('Specimen 1, p/s moment arms. BGR=30,60.90;');
859 xlabel('f/e angle in degrees')
860 ylabel('biceps moment arm, in mm');
861 ylim(bic_ylimps);
862 xlim(xlimitps);
863
864
…
1112 % maybe this will work for finding the peaks.
1113 [bic_psmax, bic_ps_maxang]=min(bic_avg_ps5);
1114 [brach_psmax, brach_ps_maxang]=max(brach_avg_ps5);
1115 [tri_psmax, tri_ps_maxang]=max(tri_avg_ps5);
1116 [pron_psmax, pron_ps_maxang]=max(pron_avg_ps5);
1117
1118
1119 peaks_ps=[bic_psmax; brach_psmax; tri_psmax; pron_psmax];
1120 peaks_psang=[bic_avg_ps5(bic_ps_maxang,1)';
    brach_avg_ps5(brach_ps_maxang,1)'; tri_avg_ps5(tri_ps_maxang,1)';
    pron_avg_ps5(pron_ps_m
axang,1)'];
1121
1122 dlmwrite('Every5deg_PSpeaks.out', peaks_ps, '\t')
1123 dlmwrite('Every5deg_PSpeaksangs.out', peaks_psang, '\t')
1124
1125 [bic_femax, bic_fe_maxang]=max(-bic_avg_fe5);
1126 [brach_femax, brach_fe_maxang]=max(-brach_avg_fe5);
1127 [tri_femax, tri_fe_maxang]=min(-tri_avg_fe5);
1128 [pron_femax, pron_fe_maxang]=max(-pron_avg_fe5);
1129
1130
1131 peaks_fe=[bic_femax; brach_femax; tri_femax; pron_femax];
1132 peaks_feang=[bic_avg_fe5(bic_fe_maxang,1)';
    brach_avg_fe5(brach_fe_maxang,1)'; tri_avg_fe5(tri_fe_maxang,1)';
    pron_avg_fe5(pron_fe_maxang
,1)'];
1133
1134 dlmwrite('Every5deg_FEpeaks.out', peaks_fe, '\t')
1135
1136
1137 [bic_psmax_std, bic_ps_maxang_std]=max(bic_std_ps5);
```

```
1138 [brach_psmax_std, brach_ps_maxang_std]=max(brach_std_ps5);
1139 [tri_psmax_std, tri_ps_maxang_std]=max(tri_std_ps5);
1140 [pron_psmax_std, pron_ps_maxang_std]=max(pron_std_ps5);
1141
1142
1143 [bic_femax_std, bic_fe_maxang_std]=max(bic_std_fe5);
1144 [brach_femax_std, brach_fe_maxang_std]=max(brach_std_fe5);
1145 [tri_femax_std, tri_fe_maxang_std]=max(tri_std_fe5);
1146 [pron_femax_std, pron_fe_maxang_std]=max(pron_std_fe5);
1147
1148 [bic_psmax_cov, bic_ps_maxang_cov]=max(bic_cov_ps5);
1149 [brach_psmax_cov, brach_ps_maxang_cov]=max(brach_cov_ps5);
1150 [tri_psmax_cov, tri_ps_maxang_cov]=max(tri_cov_ps5);
1151 [pron_psmax_cov, pron_ps_maxang_cov]=max(pron_cov_ps5);
1152
1153
1154 [bic_femax_cov, bic_fe_maxang_cov]=max(bic_cov_fe5);
1155 [brach_femax_cov, brach_fe_maxang_cov]=max(brach_cov_fe5);
1156 [tri_femax_cov, tri_fe_maxang_cov]=max(tri_cov_fe5);
1157 [pron_femax_cov, pron_fe_maxang_cov]=max(pron_cov_fe5);
1158
1159 std_fe_maxes=[bic_femax_std; brach_femax_std; tri_femax_std;
    pron_femax_std];
1160 std_ps_maxes=[bic_psmax_std; brach_psmax_std; tri_psmax_std;
    pron_psmax_std];
1161
1162 cov_fe_maxes=[bic_femax_cov; brach_femax_cov; tri_femax_cov;
    pron_femax_cov];
1163 cov_ps_maxes=[bic_psmax_cov; brach_psmax_cov; tri_psmax_cov;
    pron_psmax_cov];
1164
1165 dlmwrite('Every5deg_FE_std_maxes.out', std_fe_maxes, '\t')
1166 dlmwrite('Every5deg_PS_std_maxes.out', std_ps_maxes, '\t')
1167 dlmwrite('Every5deg_FE_cov_maxes.out', cov_fe_maxes, '\t')
1168 dlmwrite('Every5deg_PS_cov_maxes.out', cov_ps_maxes, '\t')
```

K.11 POSTPROCESSMOMARMAVERAGES_5DEG.M

The following shows an excerpt from the code. The remainder iterates through all desired plotting combinations.

```
1 % PostprocessMomArmAverages_5deg.m
2 % This is to be run after RunMomArmsFOrAveraging, and after manually
3 % copying/pasting everything to the Excel sheets.
4
5 % load workspace
6
7load MA_specimen3
8 bic_avg_fe=xlsread('Biceps_FE.xls','SummarySheet');
9 brach_avg_fe=xlsread('Brachialis_FE.xls','SummarySheet');
```

```
tri_avg_fe=xlsread('Triceps_FE.xls','SummarySheet');
pron_avg_fe=xlsread('Pronator_FE.xls','SummarySheet');

bic_avg_ps=xlsread('Biceps_PS.xls','SummarySheet');
brach_avg_ps=xlsread('Brachialis_PS.xls','SummarySheet');
tri_avg_ps=xlsread('Triceps_PS.xls','SummarySheet');
pron_avg_ps=xlsread('Pronator_PS.xls','SummarySheet');

Every5thPoint; % this will make the ones that end in "5".

% f/e limits
brach_ylim=([0 60]);
bic_ylim=([0 60]);
tri_ylim=([-40 0]);
pro_ylim=([0 40]);

% p/s limits
brach_ylimps=([-20 20]);
bic_ylimps=([-20 20]);
tri_ylimps=([-10 10]);
pro_ylimps=([-20 30]);

xlimit=([0 140]);
xlimitps=([-60 60]);

%%%%%%%%%%%%%%%%%%%%%%%%%%%
mystyles={'b-', 'g:','r-.','c-','m-','y-','k-', 'b:',
 'g:','r:','c:','m:','y:','k:'};

% set up one to do pairs.
mystyles_pairs={'b-', 'b:', 'g-', 'g:', 'r-', 'r:', 'c-', 'c:', 'm-',
 'm:', 'k-','k:', 'y-', 'y:'};



%%%%%%%%%%%Flexion-extension
% Now do Brachialis, with 1 plot.
figure;
hold on;
for ii=1:(length(biceps_momarm_xz_fe_psNeu))
hold on;
plot(feang_xz_fe_psNeu{ii}*180/pi, -biceps_momarm_xz_fe_psNeu{ii},
 mystyles{1});
end % end for

for ii=1:(length(biceps_momarm_xz_fe_psPro))
hold on;
plot(feang_xz_fe_psPro{ii}*180/pi, -
 biceps_momarm_xz_fe_psPro{ii},mystyles{2});
end % end for

for ii=1:(length(biceps_momarm_xz_fe_psSup))
hold on;
plot(feang_xz_fe_psSup{ii}*180/pi, -
 biceps_momarm_xz_fe_psSup{ii},mystyles{3});
end % end for
xlabel('flexion angle in degrees')
```

```
62 ylabel('biceps moment arm, in mm');
63 title(strcat(num2str(ii), ' B=Neu; G=Pro; R=Sup'));
64 ylim(bic_ylim);
65 xlim(xlimit);
66
67 plot(bic_avg_fe5(:,1), -bic_avg_fe5(:,26),'g-', 'LineWidth', 4);
68 plot(bic_avg_fe5(:,1), -bic_avg_fe5(:,27),'r-', 'LineWidth', 4);
69 plot(bic_avg_fe5(:,1), -bic_avg_fe5(:,28),'b-', 'LineWidth', 4);
```

K.12 RUNMOMARMSFORAVERAGING.M

```
1 % RunMomArmsForAveraging.m
2 % Laurel Kuxhaus, lck4@pitt.edu, laurel.k@gmail.com
3 % Allegheny General Hospital - Orthopaedic Biomechanics Research Laboratory
4 %
5 % This is a run file that will set up things to run collate_by_type.m .
6 % The end result will be a bunch of files ready to go to Excel for
7 % averaging of moment arms.
8 %
9 % Each specimen will generate 12 files: flexion, extension, pronation,
10 % and supination, at each of three different cases.
11 %
12 % In order to add additional specimens, after processing is complete, you
13 % will have to copy the saved workspace (.mat) into this folder, give it a
14 % sensible name, and copy/paste/modify a unit of code here.
15
16 % Step 1: load file.
17
18 load MA_specimen3
19
20 % Now, run collate_by_type 24 times per specimen.
21
22 % step 1: separate flexions from extensions.
23
24 for ii=1:5
25 extension_angles_yaPro{ii}=feang_ya_fe_psPro{2*ii-1};
26 flexion_angles_yaPro{ii}=feang_ya_fe_psPro{2*ii};
27 extension_angles_yaSup{ii}=feang_ya_fe_psSup{2*ii-1};
28 flexion_angles_yaSup{ii}=feang_ya_fe_psSup{2*ii};
29 extension_angles_yaNeu{ii}=feang_ya_fe_psNeu{2*ii-1};
30 flexion_angles_yaNeu{ii}=feang_ya_fe_psNeu{2*ii};
31
32 brach_ext_momarms_psPro{ii}=brach_momarm_ya_fe_psPro{2*ii-1};
33 brach_flex_momarms_psPro{ii}=brach_momarm_ya_fe_psPro{2*ii};
34 brach_ext_momarms_psSup{ii}=brach_momarm_ya_fe_psSup{2*ii-1};
35 brach_flex_momarms_psSup{ii}=brach_momarm_ya_fe_psSup{2*ii};
36 brach_ext_momarms_psNeu{ii}=brach_momarm_ya_fe_psNeu{2*ii-1};
37 brach_flex_momarms_psNeu{ii}=brach_momarm_ya_fe_psNeu{2*ii};
38
39 tri_ext_momarms_psPro{ii}=triceps_momarm_ya_fe_psPro{2*ii-1};
```

```
tri_flex_momarms_psPro{ii}=triceps_momarm_ya_fe_psPro{2*ii};
tri_ext_momarms_psSup{ii}=triceps_momarm_ya_fe_psSup{2*ii-1};
tri_flex_momarms_psSup{ii}=triceps_momarm_ya_fe_psSup{2*ii};
tri_ext_momarms_psNeu{ii}=triceps_momarm_ya_fe_psNeu{2*ii-1};
tri_flex_momarms_psNeu{ii}=triceps_momarm_ya_fe_psNeu{2*ii};

extension_angles_xzPro{ii}=feang_xz_fe_psPro{2*ii-1};
flexion_angles_xzPro{ii}=feang_xz_fe_psPro{2*ii};
extension_angles_xzSup{ii}=feang_xz_fe_psSup{2*ii-1};
flexion_angles_xzSup{ii}=feang_xz_fe_psSup{2*ii};
extension_angles_xzNeu{ii}=feang_xz_fe_psNeu{2*ii-1};
flexion_angles_xzNeu{ii}=feang_xz_fe_psNeu{2*ii};

biceps_ext_momarms_psPro{ii}=biceps_momarm_xz_fe_psPro{2*ii-1};
biceps_flex_momarms_psPro{ii}=biceps_momarm_xz_fe_psPro{2*ii};
biceps_ext_momarms_psSup{ii}=biceps_momarm_xz_fe_psSup{2*ii-1};
biceps_flex_momarms_psSup{ii}=biceps_momarm_xz_fe_psSup{2*ii};
biceps_ext_momarms_psNeu{ii}=biceps_momarm_xz_fe_psNeu{2*ii-1};
biceps_flex_momarms_psNeu{ii}=biceps_momarm_xz_fe_psNeu{2*ii};

pronator_ext_momarms_psPro{ii}=pronator_momarm_xz_fe_psPro{2*ii-1};
pronator_flex_momarms_psPro{ii}=pronator_momarm_xz_fe_psPro{2*ii};
pronator_ext_momarms_psSup{ii}=pronator_momarm_xz_fe_psSup{2*ii-1};
pronator_flex_momarms_psSup{ii}=pronator_momarm_xz_fe_psSup{2*ii};
pronator_ext_momarms_psNeu{ii}=pronator_momarm_xz_fe_psNeu{2*ii-1};
pronator_flex_momarms_psNeu{ii}=pronator_momarm_xz_fe_psNeu{2*ii};
end

% now, set up for pronation/supination!

for ii=1:5
pronation_angles_ya90{ii}=ps_ang_ya_ps_fe90{ii};
supination_angles_ya90{ii}=ps_ang_ya_ps_fe90{ii+5};
pronation_angles_ya60{ii}=ps_ang_ya_ps_fe60{ii};
supination_angles_ya60{ii}=ps_ang_ya_ps_fe60{ii+5};
pronation_angles_ya30{ii}=ps_ang_ya_ps_fe30{ii};
supination_angles_ya30{ii}=ps_ang_ya_ps_fe30{ii+5};

brach_pro_momarms_fe90{ii}=brach_momarm_ya_ps_fe90{ii};
brach_sup_momarms_fe90{ii}=brach_momarm_ya_ps_fe90{ii+5};
brach_pro_momarms_fe60{ii}=brach_momarm_ya_ps_fe60{ii};
brach_sup_momarms_fe60{ii}=brach_momarm_ya_ps_fe60{ii+5};
brach_pro_momarms_fe30{ii}=brach_momarm_ya_ps_fe30{ii};
brach_sup_momarms_fe30{ii}=brach_momarm_ya_ps_fe30{ii+5};

tri_pro_momarms_fe90{ii}=triceps_momarm_ya_ps_fe90{ii};
tri_sup_momarms_fe90{ii}=triceps_momarm_ya_ps_fe90{ii+5};
tri_pro_momarms_fe60{ii}=triceps_momarm_ya_ps_fe60{ii};
tri_sup_momarms_fe60{ii}=triceps_momarm_ya_ps_fe60{ii+5};
tri_pro_momarms_fe30{ii}=triceps_momarm_ya_ps_fe30{ii};
tri_sup_momarms_fe30{ii}=triceps_momarm_ya_ps_fe30{ii+5};

pronation_angles_xz90{ii}=ps_ang_xz_ps_fe90{ii};
supination_angles_xz90{ii}=ps_ang_xz_ps_fe90{ii+5};
pronation_angles_xz60{ii}=ps_ang_xz_ps_fe60{ii};
supination_angles_xz60{ii}=ps_ang_xz_ps_fe60{ii+5};
pronation_angles_xz30{ii}=ps_ang_xz_ps_fe30{ii};
```

```
97 supination_angles_xz30{ii}=ps_ang_xz_ps_fe30{ii+5};
98
99 biceps_pro_momarms_fe90{ii}=biceps_momarm_xz_ps_fe90{ii};
100 biceps_sup_momarms_fe90{ii}=biceps_momarm_xz_ps_fe90{ii+5};
101 biceps_pro_momarms_fe60{ii}=biceps_momarm_xz_ps_fe60{ii};
102 biceps_sup_momarms_fe60{ii}=biceps_momarm_xz_ps_fe60{ii+5};
103 biceps_pro_momarms_fe30{ii}=biceps_momarm_xz_ps_fe30{ii};
104 biceps_sup_momarms_fe30{ii}=biceps_momarm_xz_ps_fe30{ii+5};
105
106 pronator_pro_momarms_fe90{ii}=pronator_momarm_xz_ps_fe90{ii};
107 pronator_sup_momarms_fe90{ii}=pronator_momarm_xz_ps_fe90{ii+5};
108 pronator_pro_momarms_fe60{ii}=pronator_momarm_xz_ps_fe60{ii};
109 pronator_sup_momarms_fe60{ii}=pronator_momarm_xz_ps_fe60{ii+5};
110 pronator_pro_momarms_fe30{ii}=pronator_momarm_xz_ps_fe30{ii};
111 pronator_sup_momarms_fe30{ii}=pronator_momarm_xz_ps_fe30{ii+5};
112 end
113
114
115 % Now, run the file.
116
117 % f, Pro
118 s3_brach_momout_f_psPro=collate_by_type('f', flexion_angles_yaPro,
    brach_flex_momarms_psPro, 'brach', 's3', 'psPro');
119 s3_triceps_momout_f_psPro=collate_by_type('f', flexion_angles_yaPro,
    tri_flex_momarms_psPro, 'triceps', 's3', 'psPro');
120 s3_briceps_momout_f_psPro=collate_by_type('f', flexion_angles_xzPro,
    biceps_flex_momarms_psPro, 'biceps', 's3', 'psPro');
121 s3_pron_momout_f_psPro=collate_by_type('f', flexion_angles_xzPro,
    pronator_flex_momarms_psPro, 'pronator', 's3', 'psPro');
122
123 % f, Sup
124 s3_brach_momout_f_psSup=collate_by_type('f', flexion_angles_yaSup,
    brach_flex_momarms_psSup, 'brach', 's3', 'psSup');
125 s3_triceps_momout_f_psSup=collate_by_type('f', flexion_angles_yaSup,
    tri_flex_momarms_psSup, 'triceps', 's3', 'psSup');
126 s3_briceps_momout_f_psSup=collate_by_type('f', flexion_angles_xzSup,
    biceps_flex_momarms_psSup, 'biceps', 's3', 'psSup');
127 s3_pron_momout_f_psSup=collate_by_type('f', flexion_angles_xzSup,
    pronator_flex_momarms_psSup, 'pronator', 's3', 'psSup');
128
129 % f, Neu
130 s3_brach_momout_f_psNeu=collate_by_type('f', flexion_angles_yaNeu,
    brach_flex_momarms_psNeu, 'brach', 's3', 'psNeu');
131 s3_triceps_momout_f_psNeu=collate_by_type('f', flexion_angles_yaNeu,
    tri_flex_momarms_psNeu, 'triceps', 's3', 'psNeu');
132 s3_briceps_momout_f_psNeu=collate_by_type('f', flexion_angles_xzNeu,
    biceps_flex_momarms_psNeu, 'biceps', 's3', 'psNeu');
133 s3_pron_momout_f_psNeu=collate_by_type('f', flexion_angles_xzNeu,
    pronator_flex_momarms_psNeu, 'pronator', 's3', 'psNeu');
134
135 % e, Pro
136 s3_brach_momout_e_psPro=collate_by_type('e', extension_angles_yaPro,
    brach_ext_momarms_psPro, 'brach', 's3', 'psPro');
137 s3_triceps_momout_e_psPro=collate_by_type('e', extension_angles_yaPro,
    tri_ext_momarms_psPro, 'triceps', 's3', 'psPro');
138 s3_briceps_momout_e_psPro=collate_by_type('e', extension_angles_xzPro,
    biceps_ext_momarms_psPro, 'biceps', 's3', 'psPro');
```

```
139 s3_pron_momout_e_psPro=collate_by_type('e', extension_angles_xzPro,
    pronator_ext_momarms_psPro, 'pronator', 's3', 'psPro');
140
141 % e, Sup
142 s3_brach_momout_e_psSup=collate_by_type('e', extension_angles_yaSup,
    brach_ext_momarms_psSup, 'brach', 's3', 'psSup');
143 s3_triceps_momout_e_psSup=collate_by_type('e', extension_angles_yaSup,
    tri_ext_momarms_psSup, 'triceps', 's3', 'psSup');
144 s3_briceps_momout_e_psSup=collate_by_type('e', extension_angles_xzSup,
    biceps_ext_momarms_psSup, 'biceps', 's3', 'psSup');
145 s3_pron_momout_e_psSup=collate_by_type('e', extension_angles_xzSup,
    pronator_ext_momarms_psSup, 'pronator', 's3', 'psSup');
146
147 % e, Neu
148 s3_brach_momout_e_psNeu=collate_by_type('e', extension_angles_yaNeu,
    brach_ext_momarms_psNeu, 'brach', 's3', 'psNeu');
149 s3_triceps_momout_e_psNeu=collate_by_type('e', extension_angles_yaNeu,
    tri_ext_momarms_psNeu, 'triceps', 's3', 'psNeu');
150 s3_briceps_momout_e_psNeu=collate_by_type('e', extension_angles_xzNeu,
    biceps_ext_momarms_psNeu, 'biceps', 's3', 'psNeu');
151 s3_pron_momout_e_psNeu=collate_by_type('e', extension_angles_xzNeu,
    pronator_ext_momarms_psNeu, 'pronator', 's3', 'psNeu');
152
153 % p, 90
154 s3_brach_momout_p_fe90=collate_by_type('p', pronation_angles_ya90,
    brach_pro_momarms_fe90, 'brach', 's3', 'fe90');
155 s3_triceps_momout_p_fe90=collate_by_type('p', pronation_angles_ya90,
    tri_pro_momarms_fe90, 'triceps', 's3', 'fe90');
156 s3_briceps_momout_p_fe90=collate_by_type('p', pronation_angles_xz90,
    biceps_pro_momarms_fe90, 'biceps', 's3', 'fe90');
157 s3_pron_momout_p_fe90=collate_by_type('p', pronation_angles_xz90,
    pronator_pro_momarms_fe90, 'pronator', 's3', 'fe90');
158
159 % p, 60
160 s3_brach_momout_p_fe60=collate_by_type('p', pronation_angles_ya60,
    brach_pro_momarms_fe60, 'brach', 's3', 'fe60');
161 s3_triceps_momout_p_fe60=collate_by_type('p', pronation_angles_ya60,
    tri_pro_momarms_fe60, 'triceps', 's3', 'fe60');
162 s3_briceps_momout_p_fe60=collate_by_type('p', pronation_angles_xz60,
    biceps_pro_momarms_fe60, 'biceps', 's3', 'fe60');
163 s3_pron_momout_p_fe60=collate_by_type('p', pronation_angles_xz60,
    pronator_pro_momarms_fe60, 'pronator', 's3', 'fe60');
164
165 % p, 30
166 s3_brach_momout_p_fe30=collate_by_type('p', pronation_angles_ya30,
    brach_pro_momarms_fe30, 'brach', 's3', 'fe30');
167 s3_triceps_momout_p_fe30=collate_by_type('p', pronation_angles_ya30,
    tri_pro_momarms_fe30, 'triceps', 's3', 'fe30');
168 s3_briceps_momout_p_fe30=collate_by_type('p', pronation_angles_xz30,
    biceps_pro_momarms_fe30, 'biceps', 's3', 'fe30');
169 s3_pron_momout_p_fe30=collate_by_type('p', pronation_angles_xz30,
    pronator_pro_momarms_fe30, 'pronator', 's3', 'fe30');
170
171 % s, 90
172 s3_brach_momout_s_fe90=collate_by_type('s', supination_angles_ya90,
    brach_sup_momarms_fe90, 'brach', 's3', 'fe90');
```

```
s3_ticeps_momout_s_fe90=collate_by_type('s', supination_angles_ya90, tri_sup_momarms_fe90, 'triceps', 's3', 'fe90');
s3_briceps_momout_s_fe90=collate_by_type('s', supination_angles_xz90, biceps_sup_momarms_fe90, 'biceps', 's3', 'fe90');
s3_pron_momout_s_fe90=collate_by_type('s', supination_angles_xz90, pronator_sup_momarms_fe90, 'pronator', 's3', 'fe90');

% s, 60
s3_brach_momout_s_fe60=collate_by_type('s', supination_angles_ya60, brach_sup_momarms_fe60, 'brach', 's3', 'fe60');
s3_ticeps_momout_s_fe60=collate_by_type('s', supination_angles_ya60, tri_sup_momarms_fe60, 'triceps', 's3', 'fe60');
s3_briceps_momout_s_fe60=collate_by_type('s', supination_angles_xz60, biceps_sup_momarms_fe60, 'biceps', 's3', 'fe60');
s3_pron_momout_s_fe60=collate_by_type('s', supination_angles_xz60, pronator_sup_momarms_fe60, 'pronator', 's3', 'fe60');

% s, 30
s3_brach_momout_s_fe30=collate_by_type('s', supination_angles_ya30, brach_sup_momarms_fe30, 'brach', 's3', 'fe30');
s3_ticeps_momout_s_fe30=collate_by_type('s', supination_angles_ya30, tri_sup_momarms_fe30, 'triceps', 's3', 'fe30');
s3_briceps_momout_s_fe30=collate_by_type('s', supination_angles_xz30, biceps_sup_momarms_fe30, 'biceps', 's3', 'fe30');
s3_pron_momout_s_fe30=collate_by_type('s', supination_angles_xz30, pronator_sup_momarms_fe30, 'pronator', 's3', 'fe30');

save sepc3_foravg.mat
clear all;
'Specimen 3 done!'
% then repeat.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%SPECIMEN2%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%2%%%%%%%%%%

% Step 1: load file.

load MA_specimen2

% Now, run collate_by_type 24 times per specimen.

% step 1: separate flexions from extensions.

for ii=1:5
% angles should be the same here, just feed them to collate_by_type
% separately.
extension_angles_yaPro{ii}=feang_ya_fe_psPro{2*ii-1};
flexion_angles_yaPro{ii}=feang_ya_fe_psPro{2*ii};
extension_angles_yaSup{ii}=feang_ya_fe_psSup{2*ii-1};
flexion_angles_yaSup{ii}=feang_ya_fe_psSup{2*ii};
extension_angles_yaNeu{ii}=feang_ya_fe_psNeu{2*ii-1};
flexion_angles_yaNeu{ii}=feang_ya_fe_psNeu{2*ii};

brach_ext_momarms_psPro{ii}=brach_momarm_xz_fe_psPro{2*ii-1};
brach_flex_momarms_psPro{ii}=brach_momarm_xz_fe_psPro{2*ii};
```

```
216 brach_ext_momarms_psSup{ii}=brach_momarm_xz_fe_psSup{2*ii-1};
217 brach_flex_momarms_psSup{ii}=brach_momarm_xz_fe_psSup{2*ii};
218 brach_ext_momarms_psNeu{ii}=brach_momarm_xz_fe_psNeu{2*ii-1};
219 brach_flex_momarms_psNeu{ii}=brach_momarm_xz_fe_psNeu{2*ii};
220
221 tri_ext_momarms_psPro{ii}=triceps_momarm_ya_fe_psPro{2*ii-1};
222 tri_flex_momarms_psPro{ii}=triceps_momarm_ya_fe_psPro{2*ii};
223 tri_ext_momarms_psSup{ii}=triceps_momarm_ya_fe_psSup{2*ii-1};
224 tri_flex_momarms_psSup{ii}=triceps_momarm_ya_fe_psSup{2*ii};
225 tri_ext_momarms_psNeu{ii}=triceps_momarm_ya_fe_psNeu{2*ii-1};
226 tri_flex_momarms_psNeu{ii}=triceps_momarm_ya_fe_psNeu{2*ii};
227
228 extension_angles_xzPro{ii}=feang_xz_fe_psPro{2*ii-1};
229 flexion_angles_xzPro{ii}=feang_xz_fe_psPro{2*ii};
230 extension_angles_xzSup{ii}=feang_xz_fe_psSup{2*ii-1};
231 flexion_angles_xzSup{ii}=feang_xz_fe_psSup{2*ii};
232 extension_angles_xzNeu{ii}=feang_xz_fe_psNeu{2*ii-1};
233 flexion_angles_xzNeu{ii}=feang_xz_fe_psNeu{2*ii};
234
235 biceps_ext_momarms_psPro{ii}=biceps_momarm_ya_fe_psPro{2*ii-1};
236 biceps_flex_momarms_psPro{ii}=biceps_momarm_ya_fe_psPro{2*ii};
237 biceps_ext_momarms_psSup{ii}=biceps_momarm_ya_fe_psSup{2*ii-1};
238 biceps_flex_momarms_psSup{ii}=biceps_momarm_ya_fe_psSup{2*ii};
239 biceps_ext_momarms_psNeu{ii}=biceps_momarm_ya_fe_psNeu{2*ii-1};
240 biceps_flex_momarms_psNeu{ii}=biceps_momarm_ya_fe_psNeu{2*ii};
241
242 pronator_ext_momarms_psPro{ii}=pronator_momarm_xz_fe_psPro{2*ii-1};
243 pronator_flex_momarms_psPro{ii}=pronator_momarm_xz_fe_psPro{2*ii};
244 pronator_ext_momarms_psSup{ii}=pronator_momarm_xz_fe_psSup{2*ii-1};
245 pronator_flex_momarms_psSup{ii}=pronator_momarm_xz_fe_psSup{2*ii};
246 pronator_ext_momarms_psNeu{ii}=pronator_momarm_xz_fe_psNeu{2*ii-1};
247 pronator_flex_momarms_psNeu{ii}=pronator_momarm_xz_fe_psNeu{2*ii};
248 end
249
250 % now, set up for pronation/supination!
251
252 for ii=1:5
253 pronation_angles_ya90{ii}=ps_ang_ya_ps_fe90{ii};
254 supination_angles_ya90{ii}=ps_ang_ya_ps_fe90{ii+5};
255 pronation_angles_ya60{ii}=ps_ang_ya_ps_fe60{ii};
256 supination_angles_ya60{ii}=ps_ang_ya_ps_fe60{ii+5};
257 pronation_angles_ya30{ii}=ps_ang_ya_ps_fe30{ii};
258 supination_angles_ya30{ii}=ps_ang_ya_ps_fe30{ii+5};
259
260 brach_pro_momarms_fe90{ii}=brach_momarm_xz_ps_fe90{ii};
261 brach_sup_momarms_fe90{ii}=brach_momarm_xz_ps_fe90{ii+5};
262 brach_pro_momarms_fe60{ii}=brach_momarm_xz_ps_fe60{ii};
263 brach_sup_momarms_fe60{ii}=brach_momarm_xz_ps_fe60{ii+5};
264 brach_pro_momarms_fe30{ii}=brach_momarm_xz_ps_fe30{ii};
265 brach_sup_momarms_fe30{ii}=brach_momarm_xz_ps_fe30{ii+5};
266
267 tri_pro_momarms_fe90{ii}=triceps_momarm_ya_ps_fe90{ii};
268 tri_sup_momarms_fe90{ii}=triceps_momarm_ya_ps_fe90{ii+5};
269 tri_pro_momarms_fe60{ii}=triceps_momarm_ya_ps_fe60{ii};
270 tri_sup_momarms_fe60{ii}=triceps_momarm_ya_ps_fe60{ii+5};
271 tri_pro_momarms_fe30{ii}=triceps_momarm_ya_ps_fe30{ii};
272 tri_sup_momarms_fe30{ii}=triceps_momarm_ya_ps_fe30{ii+5};
```

```
273
274 pronation_angles_xz90{ii}=ps_ang_xz_ps_fe90{ii};
275 supination_angles_xz90{ii}=ps_ang_xz_ps_fe90{ii+5};
276 pronation_angles_xz60{ii}=ps_ang_xz_ps_fe60{ii};
277 supination_angles_xz60{ii}=ps_ang_xz_ps_fe60{ii+5};
278 pronation_angles_xz30{ii}=ps_ang_xz_ps_fe30{ii};
279 supination_angles_xz30{ii}=ps_ang_xz_ps_fe30{ii+5};
280
281 biceps_pro_momarms_fe90{ii}=biceps_momarm_ya_ps_fe90{ii};
282 biceps_sup_momarms_fe90{ii}=biceps_momarm_ya_ps_fe90{ii+5};
283 biceps_pro_momarms_fe60{ii}=biceps_momarm_ya_ps_fe60{ii};
284 biceps_sup_momarms_fe60{ii}=biceps_momarm_ya_ps_fe60{ii+5};
285 biceps_pro_momarms_fe30{ii}=biceps_momarm_ya_ps_fe30{ii};
286 biceps_sup_momarms_fe30{ii}=biceps_momarm_ya_ps_fe30{ii+5};
287
288 pronator_pro_momarms_fe90{ii}=pronator_momarm_xz_ps_fe90{ii};
289 pronator_sup_momarms_fe90{ii}=pronator_momarm_xz_ps_fe90{ii+5};
290 pronator_pro_momarms_fe60{ii}=pronator_momarm_xz_ps_fe60{ii};
291 pronator_sup_momarms_fe60{ii}=pronator_momarm_xz_ps_fe60{ii+5};
292 pronator_pro_momarms_fe30{ii}=pronator_momarm_xz_ps_fe30{ii};
293 pronator_sup_momarms_fe30{ii}=pronator_momarm_xz_ps_fe30{ii+5};
294 end
295
296
297 % Now, run the file.
298
299 % f, Pro
300 s2_brach_momout_f_psPro=collate_by_type('f', flexion_angles_xzPro,
    brach_flex_momarms_psPro, 'brach', 's2', 'psPro');
301 s2_triceps_momout_f_psPro=collate_by_type('f', flexion_angles_yaPro,
    tri_flex_momarms_psPro, 'triceps', 's2', 'psPro');
302 s2_briceps_momout_f_psPro=collate_by_type('f', flexion_angles_yaPro,
    biceps_flex_momarms_psPro, 'biceps', 's2', 'psPro');
303 s2_pron_momout_f_psPro=collate_by_type('f', flexion_angles_xzPro,
    pronator_flex_momarms_psPro, 'pronator', 's2', 'psPro');
304
305 % f, Sup
306 s2_brach_momout_f_psSup=collate_by_type('f', flexion_angles_xzSup,
    brach_flex_momarms_psSup, 'brach', 's2', 'psSup');
307 s2_triceps_momout_f_psSup=collate_by_type('f', flexion_angles_yaSup,
    tri_flex_momarms_psSup, 'triceps', 's2', 'psSup');
308 s2_briceps_momout_f_psSup=collate_by_type('f', flexion_angles_yaSup,
    biceps_flex_momarms_psSup, 'biceps', 's2', 'psSup');
309 s2_pron_momout_f_psSup=collate_by_type('f', flexion_angles_xzSup,
    pronator_flex_momarms_psSup, 'pronator', 's2', 'psSup');
310
311 % f, Neu
312 s2_brach_momout_f_psNeu=collate_by_type('f', flexion_angles_xzNeu,
    brach_flex_momarms_psNeu, 'brach', 's2', 'psNeu');
313 s2_triceps_momout_f_psNeu=collate_by_type('f', flexion_angles_yaNeu,
    tri_flex_momarms_psNeu, 'triceps', 's2', 'psNeu');
314 s2_briceps_momout_f_psNeu=collate_by_type('f', flexion_angles_yaNeu,
    biceps_flex_momarms_psNeu, 'biceps', 's2', 'psNeu');
315 s2_pron_momout_f_psNeu=collate_by_type('f', flexion_angles_xzNeu,
    pronator_flex_momarms_psNeu, 'pronator', 's2', 'psNeu');
316
317 % e, Pro
```

```
318 s2_brach_momout_e_psPro=collate_by_type('e', extension_angles_xzPro,
    brach_ext_momarms_psPro, 'brach', 's2', 'psPro');
319 s2_ticeps_momout_e_psPro=collate_by_type('e', extension_angles_yaPro,
    tri_ext_momarms_psPro, 'triceps', 's2', 'psPro');
320 s2_briceps_momout_e_psPro=collate_by_type('e', extension_angles_yaPro,
    biceps_ext_momarms_psPro, 'biceps', 's2', 'psPro');
321 s2_pron_momout_e_psPro=collate_by_type('e', extension_angles_xzPro,
    pronator_ext_momarms_psPro, 'pronator', 's2', 'psPro');
322
323 % e, Sup
324 s2_brach_momout_e_psSup=collate_by_type('e', extension_angles_xzSup,
    brach_ext_momarms_psSup, 'brach', 's2', 'psSup');
325 s2_ticeps_momout_e_psSup=collate_by_type('e', extension_angles_yaSup,
    tri_ext_momarms_psSup, 'triceps', 's2', 'psSup');
326 s2_briceps_momout_e_psSup=collate_by_type('e', extension_angles_yaSup,
    biceps_ext_momarms_psSup, 'biceps', 's2', 'psSup');
327 s2_pron_momout_e_psSup=collate_by_type('e', extension_angles_xzSup,
    pronator_ext_momarms_psSup, 'pronator', 's2', 'psSup');
328
329 % e, Neu
330 s2_brach_momout_e_psNeu=collate_by_type('e', extension_angles_xzNeu,
    brach_ext_momarms_psNeu, 'brach', 's2', 'psNeu');
331 s2_ticeps_momout_e_psNeu=collate_by_type('e', extension_angles_yaNeu,
    tri_ext_momarms_psNeu, 'triceps', 's2', 'psNeu');
332 s2_briceps_momout_e_psNeu=collate_by_type('e', extension_angles_yaNeu,
    biceps_ext_momarms_psNeu, 'biceps', 's2', 'psNeu');
333 s2_pron_momout_e_psNeu=collate_by_type('e', extension_angles_xzNeu,
    pronator_ext_momarms_psNeu, 'pronator', 's2', 'psNeu');
334
335 % p, 90
336 s2_brach_momout_p_fe90=collate_by_type('p', pronation_angles_xz90,
    brach_pro_momarms_fe90, 'brach', 's2', 'fe90');
337 s2_ticeps_momout_p_fe90=collate_by_type('p', pronation_angles_ya90,
    tri_pro_momarms_fe90, 'triceps', 's2', 'fe90');
338 s2_briceps_momout_p_fe90=collate_by_type('p', pronation_angles_ya90,
    biceps_pro_momarms_fe90, 'biceps', 's2', 'fe90');
339 s2_pron_momout_p_fe90=collate_by_type('p', pronation_angles_xz90,
    pronator_pro_momarms_fe90, 'pronator', 's2', 'fe90');
340
341 % p, 60
342 s2_brach_momout_p_fe60=collate_by_type('p', pronation_angles_xz60,
    brach_pro_momarms_fe60, 'brach', 's2', 'fe60');
343 s2_ticeps_momout_p_fe60=collate_by_type('p', pronation_angles_ya60,
    tri_pro_momarms_fe60, 'triceps', 's2', 'fe60');
344 s2_briceps_momout_p_fe60=collate_by_type('p', pronation_angles_ya60,
    biceps_pro_momarms_fe60, 'biceps', 's2', 'fe60');
345 s2_pron_momout_p_fe60=collate_by_type('p', pronation_angles_xz60,
    pronator_pro_momarms_fe60, 'pronator', 's2', 'fe60');
346
347 % p, 30
348 s2_brach_momout_p_fe30=collate_by_type('p', pronation_angles_xz30,
    brach_pro_momarms_fe30, 'brach', 's2', 'fe30');
349 s2_ticeps_momout_p_fe30=collate_by_type('p', pronation_angles_ya30,
    tri_pro_momarms_fe30, 'triceps', 's2', 'fe30');
350 s2_briceps_momout_p_fe30=collate_by_type('p', pronation_angles_ya30,
    biceps_pro_momarms_fe30, 'biceps', 's2', 'fe30');
```

```
s2_pron_momout_p_fe30=collate_by_type('p', pronation_angles_xz30, pronator_pro_momarms_fe30, 'pronator', 's2', 'fe30');

% s, 90
s2_brach_momout_s_fe90=collate_by_type('s', supination_angles_xz90, brach_sup_momarms_fe90, 'brach', 's2', 'fe90');
s2_ticeps_momout_s_fe90=collate_by_type('s', supination_angles_ya90, tri_sup_momarms_fe90, 'triceps', 's2', 'fe90');
s2_briceps_momout_s_fe90=collate_by_type('s', supination_angles_ya90, biceps_sup_momarms_fe90, 'biceps', 's2', 'fe90');
s2_pron_momout_s_fe90=collate_by_type('s', supination_angles_xz90, pronator_sup_momarms_fe90, 'pronator', 's2', 'fe90');

% s, 60
s2_brach_momout_s_fe60=collate_by_type('s', supination_angles_xz60, brach_sup_momarms_fe60, 'brach', 's2', 'fe60');
s2_ticeps_momout_s_fe60=collate_by_type('s', supination_angles_ya60, tri_sup_momarms_fe60, 'triceps', 's2', 'fe60');
s2_briceps_momout_s_fe60=collate_by_type('s', supination_angles_ya60, biceps_sup_momarms_fe60, 'biceps', 's2', 'fe60');
s2_pron_momout_s_fe60=collate_by_type('s', supination_angles_xz60, pronator_sup_momarms_fe60, 'pronator', 's2', 'fe60');

% s, 30
s2_brach_momout_s_fe30=collate_by_type('s', supination_angles_xz30, brach_sup_momarms_fe30, 'brach', 's2', 'fe30');
s2_ticeps_momout_s_fe30=collate_by_type('s', supination_angles_ya30, tri_sup_momarms_fe30, 'triceps', 's2', 'fe30');
s2_briceps_momout_s_fe30=collate_by_type('s', supination_angles_ya30, biceps_sup_momarms_fe30, 'biceps', 's2', 'fe30');
s2_pron_momout_s_fe30=collate_by_type('s', supination_angles_xz30, pronator_sup_momarms_fe30, 'pronator', 's2', 'fe30');

save sepc2_foravg.mat
clear all;
'Specimen 2 done!'


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%SPECIMEN1%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%2%%%%%%%%%
%

% Step 1: load file.

load MA_specimen1

% Now, run collate_by_type 24 times per specimen.

% step 1: separate flexions from extensions.

for ii=1:5
% angles should be the same here, just feed them to collate_by_type
% separately.
extension_angles_yaPro{ii}=feang_ya_fe_psPro{2*ii-1};
flexion_angles_yaPro{ii}=feang_ya_fe_psPro{2*ii};
extension_angles_yaSup{ii}=feang_ya_fe_psSup{2*ii-1};
flexion_angles_yaSup{ii}=feang_ya_fe_psSup{2*ii};
```

```
392 extension_angles_yaNeu{ii}=feang_ya_fe_psNeu{2*ii-1};
393 flexion_angles_yaNeu{ii}=feang_ya_fe_psNeu{2*ii};
394
395 brach_ext_momarms_psPro{ii}=brach_momarm_xz_fe_psPro{2*ii-1};
396 brach_flex_momarms_psPro{ii}=brach_momarm_xz_fe_psPro{2*ii};
397 brach_ext_momarms_psSup{ii}=brach_momarm_xz_fe_psSup{2*ii-1};
398 brach_flex_momarms_psSup{ii}=brach_momarm_xz_fe_psSup{2*ii};
399 brach_ext_momarms_psNeu{ii}=brach_momarm_xz_fe_psNeu{2*ii-1};
400 brach_flex_momarms_psNeu{ii}=brach_momarm_xz_fe_psNeu{2*ii};
401
402 tri_ext_momarms_psPro{ii}=triceps_momarm_ya_fe_psPro{2*ii-1};
403 tri_flex_momarms_psPro{ii}=triceps_momarm_ya_fe_psPro{2*ii};
404 tri_ext_momarms_psSup{ii}=triceps_momarm_ya_fe_psSup{2*ii-1};
405 tri_flex_momarms_psSup{ii}=triceps_momarm_ya_fe_psSup{2*ii};
406 tri_ext_momarms_psNeu{ii}=triceps_momarm_ya_fe_psNeu{2*ii-1};
407 tri_flex_momarms_psNeu{ii}=triceps_momarm_ya_fe_psNeu{2*ii};
408
409 extension_angles_xzPro{ii}=feang_xz_fe_psPro{2*ii-1};
410 flexion_angles_xzPro{ii}=feang_xz_fe_psPro{2*ii};
411 extension_angles_xzSup{ii}=feang_xz_fe_psSup{2*ii-1};
412 flexion_angles_xzSup{ii}=feang_xz_fe_psSup{2*ii};
413 extension_angles_xzNeu{ii}=feang_xz_fe_psNeu{2*ii-1};
414 flexion_angles_xzNeu{ii}=feang_xz_fe_psNeu{2*ii};
415
416 biceps_ext_momarms_psPro{ii}=biceps_momarm_ya_fe_psPro{2*ii-1};
417 biceps_flex_momarms_psPro{ii}=biceps_momarm_ya_fe_psPro{2*ii};
418 biceps_ext_momarms_psSup{ii}=biceps_momarm_ya_fe_psSup{2*ii-1};
419 biceps_flex_momarms_psSup{ii}=biceps_momarm_ya_fe_psSup{2*ii};
420 biceps_ext_momarms_psNeu{ii}=biceps_momarm_ya_fe_psNeu{2*ii-1};
421 biceps_flex_momarms_psNeu{ii}=biceps_momarm_ya_fe_psNeu{2*ii};
422
423 pronator_ext_momarms_psPro{ii}=pronator_momarm_xz_fe_psPro{2*ii-1};
424 pronator_flex_momarms_psPro{ii}=pronator_momarm_xz_fe_psPro{2*ii};
425 pronator_ext_momarms_psSup{ii}=pronator_momarm_xz_fe_psSup{2*ii-1};
426 pronator_flex_momarms_psSup{ii}=pronator_momarm_xz_fe_psSup{2*ii};
427 pronator_ext_momarms_psNeu{ii}=pronator_momarm_xz_fe_psNeu{2*ii-1};
428 pronator_flex_momarms_psNeu{ii}=pronator_momarm_xz_fe_psNeu{2*ii};
429 end
430
431 % now, set up for pronation/supination!
432
433 for ii=1:5
434 pronation_angles_ya90{ii}=ps_ang_ya_ps_fe90{ii};
435 supination_angles_ya90{ii}=ps_ang_ya_ps_fe90{ii+5};
436 pronation_angles_ya60{ii}=ps_ang_ya_ps_fe60{ii};
437 supination_angles_ya60{ii}=ps_ang_ya_ps_fe60{ii+5};
438 pronation_angles_ya30{ii}=ps_ang_ya_ps_fe30{ii};
439 supination_angles_ya30{ii}=ps_ang_ya_ps_fe30{ii+5};
440
441 brach_pro_momarms_fe90{ii}=brach_momarm_xz_ps_fe90{ii};
442 brach_sup_momarms_fe90{ii}=brach_momarm_xz_ps_fe90{ii+5};
443 brach_pro_momarms_fe60{ii}=brach_momarm_xz_ps_fe60{ii};
444 brach_sup_momarms_fe60{ii}=brach_momarm_xz_ps_fe60{ii+5};
445 brach_pro_momarms_fe30{ii}=brach_momarm_xz_ps_fe30{ii};
446 brach_sup_momarms_fe30{ii}=brach_momarm_xz_ps_fe30{ii+5};
447
448 tri_pro_momarms_fe90{ii}=triceps_momarm_ya_ps_fe90{ii};
```

```
449 tri_sup_momarms_fe90{ii}=triceps_momarm_ya_ps_fe90{ii+5};
450 tri_pro_momarms_fe60{ii}=triceps_momarm_ya_ps_fe60{ii};
451 tri_sup_momarms_fe60{ii}=triceps_momarm_ya_ps_fe60{ii+5};
452 tri_pro_momarms_fe30{ii}=triceps_momarm_ya_ps_fe30{ii};
453 tri_sup_momarms_fe30{ii}=triceps_momarm_ya_ps_fe30{ii+5};
454
455 pronation_angles_xz90{ii}=ps_ang_xz_ps_fe90{ii};
456 supination_angles_xz90{ii}=ps_ang_xz_ps_fe90{ii+5};
457 pronation_angles_xz60{ii}=ps_ang_xz_ps_fe60{ii};
458 supination_angles_xz60{ii}=ps_ang_xz_ps_fe60{ii+5};
459 pronation_angles_xz30{ii}=ps_ang_xz_ps_fe30{ii};
460 supination_angles_xz30{ii}=ps_ang_xz_ps_fe30{ii+5};
461
462 biceps_pro_momarms_fe90{ii}=biceps_momarm_ya_ps_fe90{ii};
463 biceps_sup_momarms_fe90{ii}=biceps_momarm_ya_ps_fe90{ii+5};
464 biceps_pro_momarms_fe60{ii}=biceps_momarm_ya_ps_fe60{ii};
465 biceps_sup_momarms_fe60{ii}=biceps_momarm_ya_ps_fe60{ii+5};
466 biceps_pro_momarms_fe30{ii}=biceps_momarm_ya_ps_fe30{ii};
467 biceps_sup_momarms_fe30{ii}=biceps_momarm_ya_ps_fe30{ii+5};
468
469 pronator_pro_momarms_fe90{ii}=pronator_momarm_xz_ps_fe90{ii};
470 pronator_sup_momarms_fe90{ii}=pronator_momarm_xz_ps_fe90{ii+5};
471 pronator_pro_momarms_fe60{ii}=pronator_momarm_xz_ps_fe60{ii};
472 pronator_sup_momarms_fe60{ii}=pronator_momarm_xz_ps_fe60{ii+5};
473 pronator_pro_momarms_fe30{ii}=pronator_momarm_xz_ps_fe30{ii};
474 pronator_sup_momarms_fe30{ii}=pronator_momarm_xz_ps_fe30{ii+5};
475 end
476
477
478 % Now, run the file.
479
480 % f, Pro
481 s1_brach_momout_f_psPro=collate_by_type('f', flexion_angles_xzPro,
    brach_flex_momarms_psPro, 'brach', 's1', 'psPro');
482 s1_triceps_momout_f_psPro=collate_by_type('f', flexion_angles_yaPro,
    tri_flex_momarms_psPro, 'triceps', 's1', 'psPro');
483 s1_briceps_momout_f_psPro=collate_by_type('f', flexion_angles_yaPro,
    biceps_flex_momarms_psPro, 'biceps', 's1', 'psPro');
484 s1_pron_momout_f_psPro=collate_by_type('f', flexion_angles_xzPro,
    pronator_flex_momarms_psPro, 'pronator', 's1', 'psPro');
485
486 % f, Sup
487 s1_brach_momout_f_psSup=collate_by_type('f', flexion_angles_xzSup,
    brach_flex_momarms_psSup, 'brach', 's1', 'psSup');
488 s1_triceps_momout_f_psSup=collate_by_type('f', flexion_angles_yaSup,
    tri_flex_momarms_psSup, 'triceps', 's1', 'psSup');
489 s1_briceps_momout_f_psSup=collate_by_type('f', flexion_angles_yaSup,
    biceps_flex_momarms_psSup, 'biceps', 's1', 'psSup');
490 s1_pron_momout_f_psSup=collate_by_type('f', flexion_angles_xzSup,
    pronator_flex_momarms_psSup, 'pronator', 's1', 'psSup');
491
492 % f, Neu
493 s1_brach_momout_f_psNeu=collate_by_type('f', flexion_angles_xzNeu,
    brach_flex_momarms_psNeu, 'brach', 's1', 'psNeu');
494 s1_triceps_momout_f_psNeu=collate_by_type('f', flexion_angles_yaNeu,
    tri_flex_momarms_psNeu, 'triceps', 's1', 'psNeu');
```

```
495 s1_briceps_momout_f_psNeu=collate_by_type('f', flexion_angles_yaNeu,
    biceps_flex_momarms_psNeu, 'biceps', 's1', 'psNeu');
496 s1_pron_momout_f_psNeu=collate_by_type('f', flexion_angles_xzNeu,
    pronator_flex_momarms_psNeu, 'pronator', 's1', 'psNeu');
497
498 % e, Pro
499 s1_brach_momout_e_psPro=collate_by_type('e', extension_angles_xzPro,
    brach_ext_momarms_psPro, 'brach', 's1', 'psPro');
500 s1_triceps_momout_e_psPro=collate_by_type('e', extension_angles_yaPro,
    tri_ext_momarms_psPro, 'triceps', 's1', 'psPro');
501 s1_briceps_momout_e_psPro=collate_by_type('e', extension_angles_yaPro,
    biceps_ext_momarms_psPro, 'biceps', 's1', 'psPro');
502 s1_pron_momout_e_psPro=collate_by_type('e', extension_angles_xzPro,
    pronator_ext_momarms_psPro, 'pronator', 's1', 'psPro');
503
504 % e, Sup
505 s1_brach_momout_e_psSup=collate_by_type('e', extension_angles_xzSup,
    brach_ext_momarms_psSup, 'brach', 's1', 'psSup');
506 s1_triceps_momout_e_psSup=collate_by_type('e', extension_angles_yaSup,
    tri_ext_momarms_psSup, 'triceps', 's1', 'psSup');
507 s1_briceps_momout_e_psSup=collate_by_type('e', extension_angles_yaSup,
    biceps_ext_momarms_psSup, 'biceps', 's1', 'psSup');
508 s1_pron_momout_e_psSup=collate_by_type('e', extension_angles_xzSup,
    pronator_ext_momarms_psSup, 'pronator', 's1', 'psSup');
509
510 % e, Neu
511 s1_brach_momout_e_psNeu=collate_by_type('e', extension_angles_xzNeu,
    brach_ext_momarms_psNeu, 'brach', 's1', 'psNeu');
512 s1_triceps_momout_e_psNeu=collate_by_type('e', extension_angles_yaNeu,
    tri_ext_momarms_psNeu, 'triceps', 's1', 'psNeu');
513 s1_briceps_momout_e_psNeu=collate_by_type('e', extension_angles_yaNeu,
    biceps_ext_momarms_psNeu, 'biceps', 's1', 'psNeu');
514 s1_pron_momout_e_psNeu=collate_by_type('e', extension_angles_xzNeu,
    pronator_ext_momarms_psNeu, 'pronator', 's1', 'psNeu');
515
516 % p, 90
517 s1_brach_momout_p_fe90=collate_by_type('p', pronation_angles_xz90,
    brach_pro_momarms_fe90, 'brach', 's1', 'fe90');
518 s1_triceps_momout_p_fe90=collate_by_type('p', pronation_angles_ya90,
    tri_pro_momarms_fe90, 'triceps', 's1', 'fe90');
519 s1_briceps_momout_p_fe90=collate_by_type('p', pronation_angles_ya90,
    biceps_pro_momarms_fe90, 'biceps', 's1', 'fe90');
520 s1_pron_momout_p_fe90=collate_by_type('p', pronation_angles_xz90,
    pronator_pro_momarms_fe90, 'pronator', 's1', 'fe90');
521
522 % p, 60
523 s1_brach_momout_p_fe60=collate_by_type('p', pronation_angles_xz60,
    brach_pro_momarms_fe60, 'brach', 's1', 'fe60');
524 s1_triceps_momout_p_fe60=collate_by_type('p', pronation_angles_ya60,
    tri_pro_momarms_fe60, 'triceps', 's1', 'fe60');
525 s1_briceps_momout_p_fe60=collate_by_type('p', pronation_angles_ya60,
    biceps_pro_momarms_fe60, 'biceps', 's1', 'fe60');
526 s1_pron_momout_p_fe60=collate_by_type('p', pronation_angles_xz60,
    pronator_pro_momarms_fe60, 'pronator', 's1', 'fe60');
527
528 % p, 30
```

```
529 s1_brach_momout_p_fe30=collate_by_type('p', pronation_angles_xz30,
    brach_pro_momarms_fe30, 'brach', 's1', 'fe30');
530 s1_ticeps_momout_p_fe30=collate_by_type('p', pronation_angles_ya30,
    tri_pro_momarms_fe30, 'triceps', 's1', 'fe30');
531 s1_briceps_momout_p_fe30=collate_by_type('p', pronation_angles_ya30,
    biceps_pro_momarms_fe30, 'biceps', 's1', 'fe30');
532 s1_pron_momout_p_fe30=collate_by_type('p', pronation_angles_xz30,
    pronator_pro_momarms_fe30, 'pronator', 's1', 'fe30');
533
534 % s, 90
535 s1_brach_momout_s_fe90=collate_by_type('s', supination_angles_xz90,
    brach_sup_momarms_fe90, 'brach', 's1', 'fe90');
536 s1_ticeps_momout_s_fe90=collate_by_type('s', supination_angles_ya90,
    tri_sup_momarms_fe90, 'triceps', 's1', 'fe90');
537 s1_briceps_momout_s_fe90=collate_by_type('s', supination_angles_ya90,
    biceps_sup_momarms_fe90, 'biceps', 's1', 'fe90');
538 s1_pron_momout_s_fe90=collate_by_type('s', supination_angles_xz90,
    pronator_sup_momarms_fe90, 'pronator', 's1', 'fe90');
539
540 % s, 60
541 s1_brach_momout_s_fe60=collate_by_type('s', supination_angles_xz60,
    brach_sup_momarms_fe60, 'brach', 's1', 'fe60');
542 s1_ticeps_momout_s_fe60=collate_by_type('s', supination_angles_ya60,
    tri_sup_momarms_fe60, 'triceps', 's1', 'fe60');
543 s1_briceps_momout_s_fe60=collate_by_type('s', supination_angles_ya60,
    biceps_sup_momarms_fe60, 'biceps', 's1', 'fe60');
544 s1_pron_momout_s_fe60=collate_by_type('s', supination_angles_xz60,
    pronator_sup_momarms_fe60, 'pronator', 's1', 'fe60');
545
546 % s, 30
547 s1_brach_momout_s_fe30=collate_by_type('s', supination_angles_xz30,
    brach_sup_momarms_fe30, 'brach', 's1', 'fe30');
548 s1_ticeps_momout_s_fe30=collate_by_type('s', supination_angles_ya30,
    tri_sup_momarms_fe30, 'triceps', 's1', 'fe30');
549 s1_briceps_momout_s_fe30=collate_by_type('s', supination_angles_ya30,
    biceps_sup_momarms_fe30, 'biceps', 's1', 'fe30');
550 s1_pron_momout_s_fe30=collate_by_type('s', supination_angles_xz30,
    pronator_sup_momarms_fe30, 'pronator', 's1', 'fe30');
551
552 save sepc1_foravg.mat
553 clear all;
554 'Specimen 1 done!'
```

APPENDIX L

CIRCLE-FITTING ALGORITHM

1. Find the optimal plane of the points using least square (normal vector)

The equation of the plane in 3D is

$$\begin{aligned} Ax + By + Cz + D &= 0 \\ A'x + B'y + C'z - 1 &= 0 \end{aligned} \qquad (17)$$

To get the coefficients, minimize the following function.

$$F = \sum_{i=1}^{n} (Ax_i + By_i + Cz_i - 1)^2$$

$$\begin{aligned} \frac{\partial F}{\partial A} &= 2\sum_{i=1}^{n} (Ax_i + By_i + Cz_i - 1)x_i = 0 \\ \frac{\partial F}{\partial B} &= 2\sum_{i=1}^{n} (Ax_i + By_i + Cz_i - 1)y_i = 0 \\ \frac{\partial F}{\partial C} &= 2\sum_{i=1}^{n} (Ax_i + By_i + Cz_i - 1)z_i = 0 \end{aligned} \qquad (18)$$

$$\begin{bmatrix} \sum x_i^2 & \sum x_i y_i & \sum x_i z_i \\ \sum x_i y_i & \sum y_i^2 & \sum y_i z_i \\ \sum x_i z_i & \sum y_i z_i & \sum z_i^2 \end{bmatrix} \begin{bmatrix} A \\ B \\ C \end{bmatrix} = \begin{bmatrix} \sum x_i \\ \sum y_i \\ \sum z_i \end{bmatrix}$$

A, *B* and *C* are components of the normal vector to the optimal plane.

2. Find the rotation matrix M which rotates the optimal plane onto the plane parallel to *XY* plane.

The normal vector to the optimal plane is $\mathbf{n}=[A, B, C]^T$.

The projection vector of the normal vector to the *XY* plane is $[A, B, 0]^T$.

The vector perpendicular to that projection vector is $\mathbf{n_2}=[-B, A, 0]^T$.

The normal vector and $\mathbf{n_2}$ are perpendicular to each other.

The third vector can be calculated by

$= N_3 = (N_2 \times N)/|N_2 \times N|$

$$= \mathbf{[-B, A, 0]^T \times [A, B, C]^T / |[-B, A, 0]^T \times [A, B, C]^T|} \quad (19)$$

These three vectors consist of the rotation matrix $\mathbf{M}=[\mathbf{n_2}\ \mathbf{n_3}\ \mathbf{n}]^T$.

3. Map all the data points using **M** and cross product with (0, 0, 1).

Then the mapped points have only *X* and *Y* coordinates after the cross product. The cross product eliminates only the *Z* component from the points.

4. Find the center and radius of the circle parallel to *XY* plane with these mapped points using least square.

$$F = \sum_{i=1}^{n} \left[\left\| M(p_i - c) \times (0,0,1) \right\|^2 - R^2 \right]^2 = \sum_{i=1}^{n} \left[(x_i - a)^2 + (y_i - b)^2 - R^2 \right]^2$$
$$= \sum_{i=1}^{n} \left[x_i^2 + y_i^2 - 2ax_i - 2ay_i + a^2 + b^2 - R^2 \right]^2 \quad (20)$$

Let $X_i = x_i^2 + y_i^2$, $A = -2a$, $B = -2b$, $C = a^2 + b^2 - R^2$.

$$\frac{\partial F}{\partial A} = 2\sum_{i=1}^{n}\left[X_i + Ax_i + By_i + C\right]x_i = 0$$
$$\frac{\partial F}{\partial B} = 2\sum_{i=1}^{n}\left[X_i + Ax_i + By_i + C\right]y_i = 0$$
$$\frac{\partial F}{\partial C} = 2\sum_{i=1}^{n}\left[X_i + Ax_i + By_i + C\right] = 0 \qquad (21)$$

$$\begin{bmatrix} \sum x_i^2 & \sum x_i y_i & \sum x_i \\ \sum x_i y_i & \sum y_i^2 & \sum y_i \\ \sum x_i & \sum y_i & n \end{bmatrix}\begin{bmatrix} A \\ B \\ C \end{bmatrix} = -\begin{bmatrix} \sum X_i x_i \\ \sum X_i y_i \\ \sum X_i \end{bmatrix}$$

Then, the center in the plane parallel to *XY* plane and radius of the circle are

$$\boldsymbol{A = -A/2,\ B = -B/2,\ R = \mathrm{SQRT}}$$
$$\boldsymbol{(A^2+B^2-C)} \qquad (22)$$

The center in 3D (rotate the center back) is

$$M^{-1}\begin{bmatrix} a \\ b \\ Z \end{bmatrix} \qquad (23)$$

where Z is the average value of z coordinates of the mapped points.

APPENDIX M

DIAGRAMS OF RADIAL HEAD TRACKING CODE

The three diagrams in this appendix show the flow behind the radial head calculation code, as described in Chapter 8.0 and shown in the following Appendix. The first diagram, in Figure 72, shows the components that lead to the ultimate calculation of the implant position in the anatomical coordinate system. In Figure 73 the same is shown for the calculation of the native head position. Figure 74 outlines the code as written.

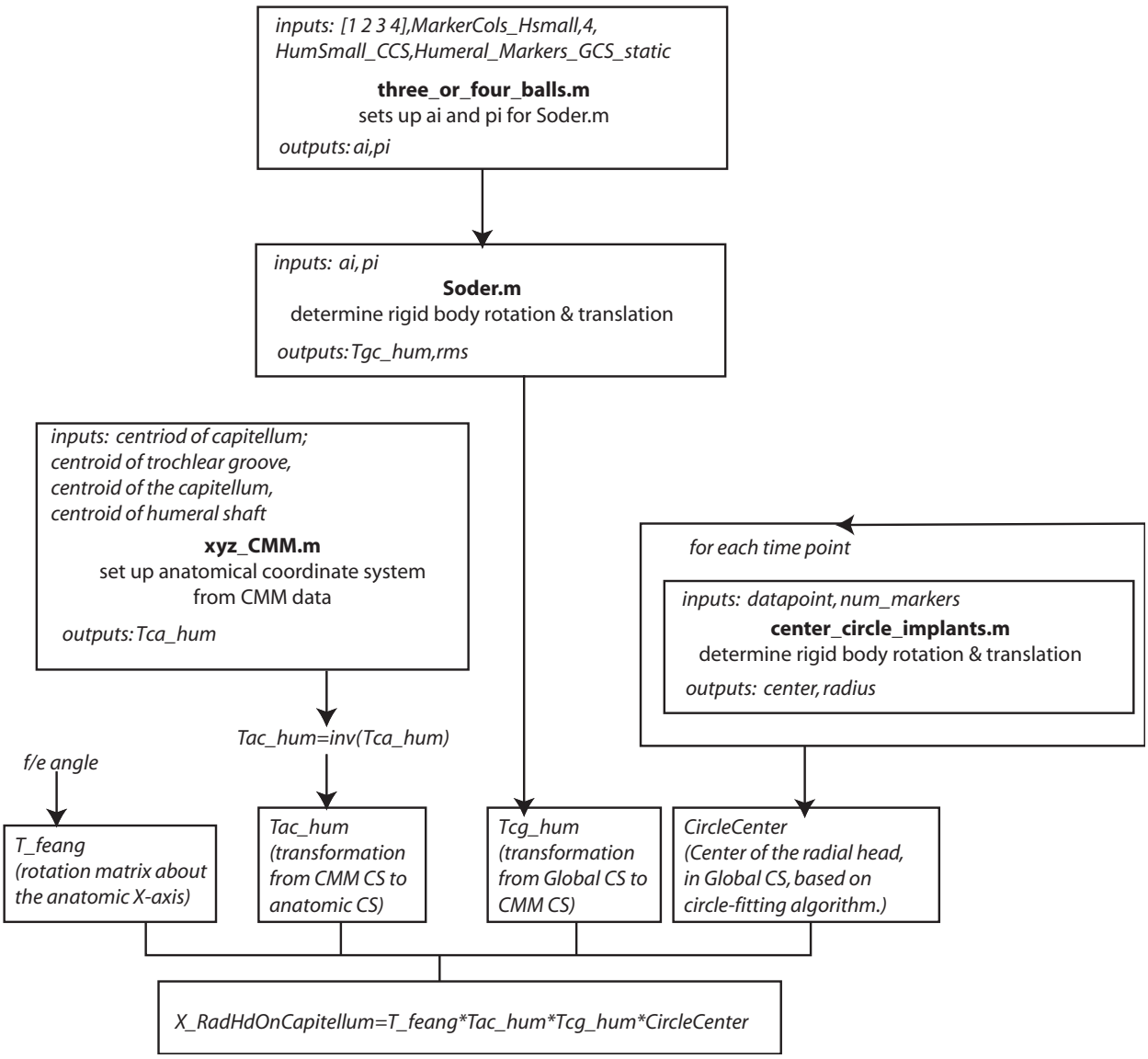


Figure 72: Flowchart to calculation of implant position.

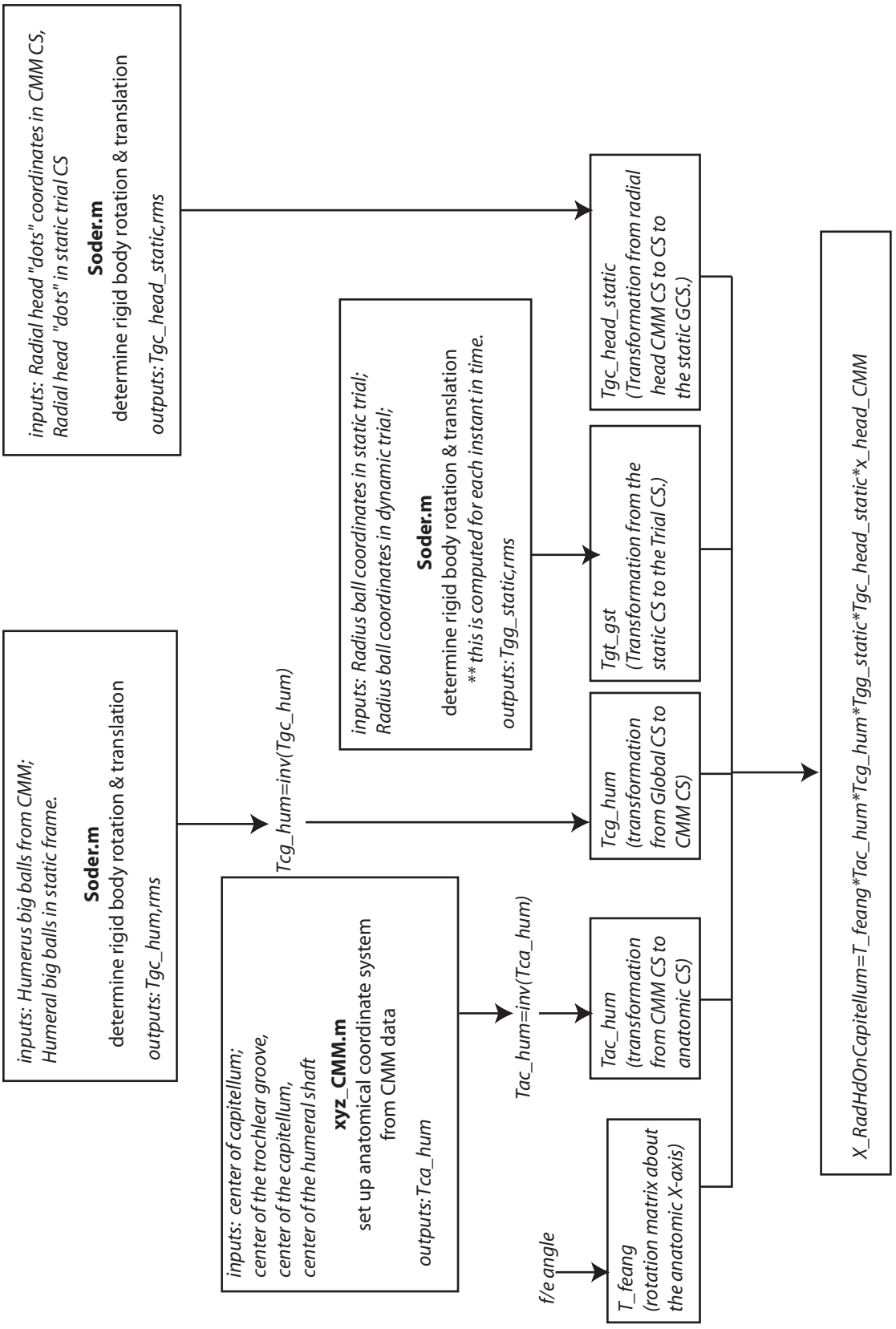
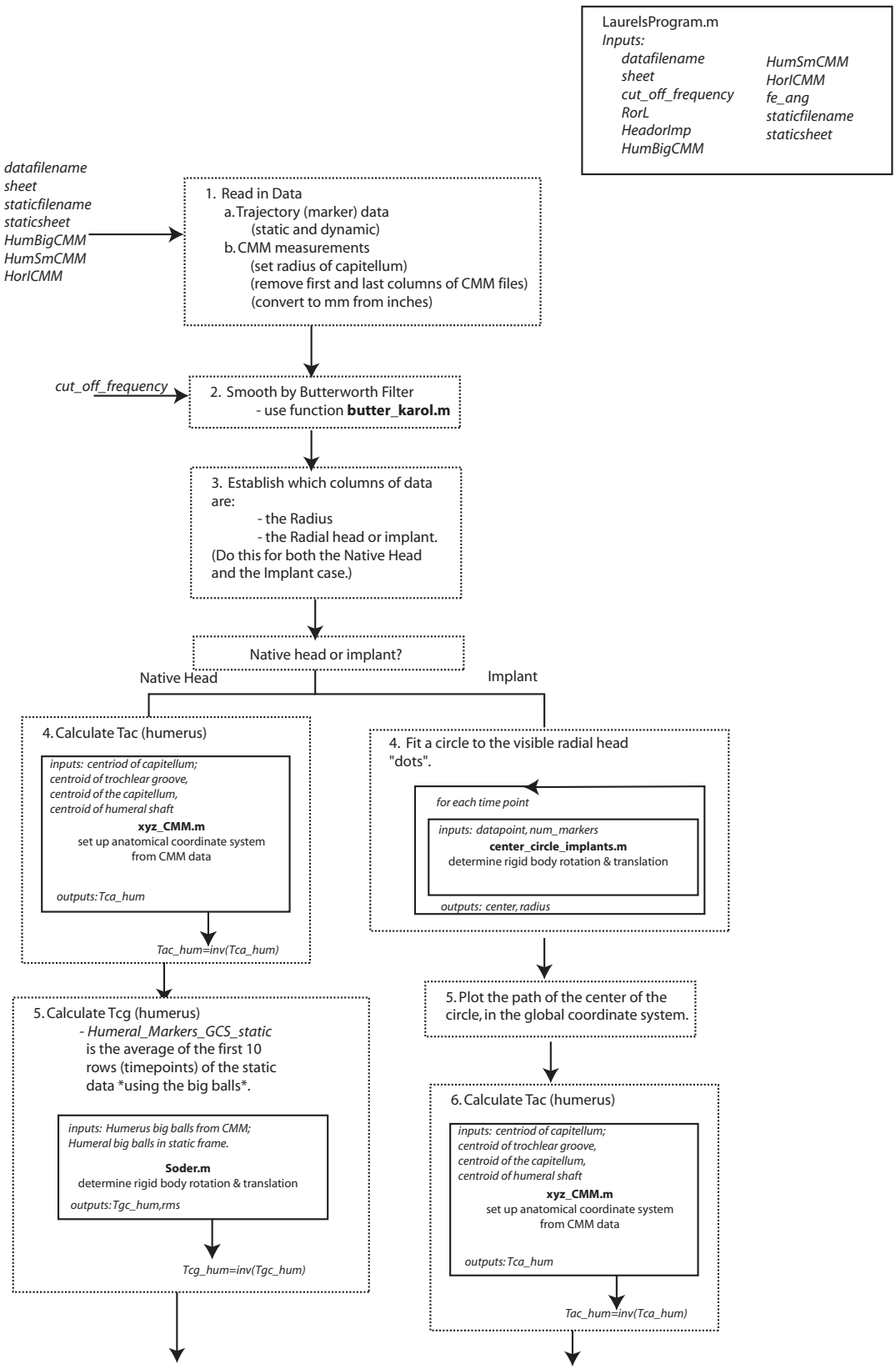
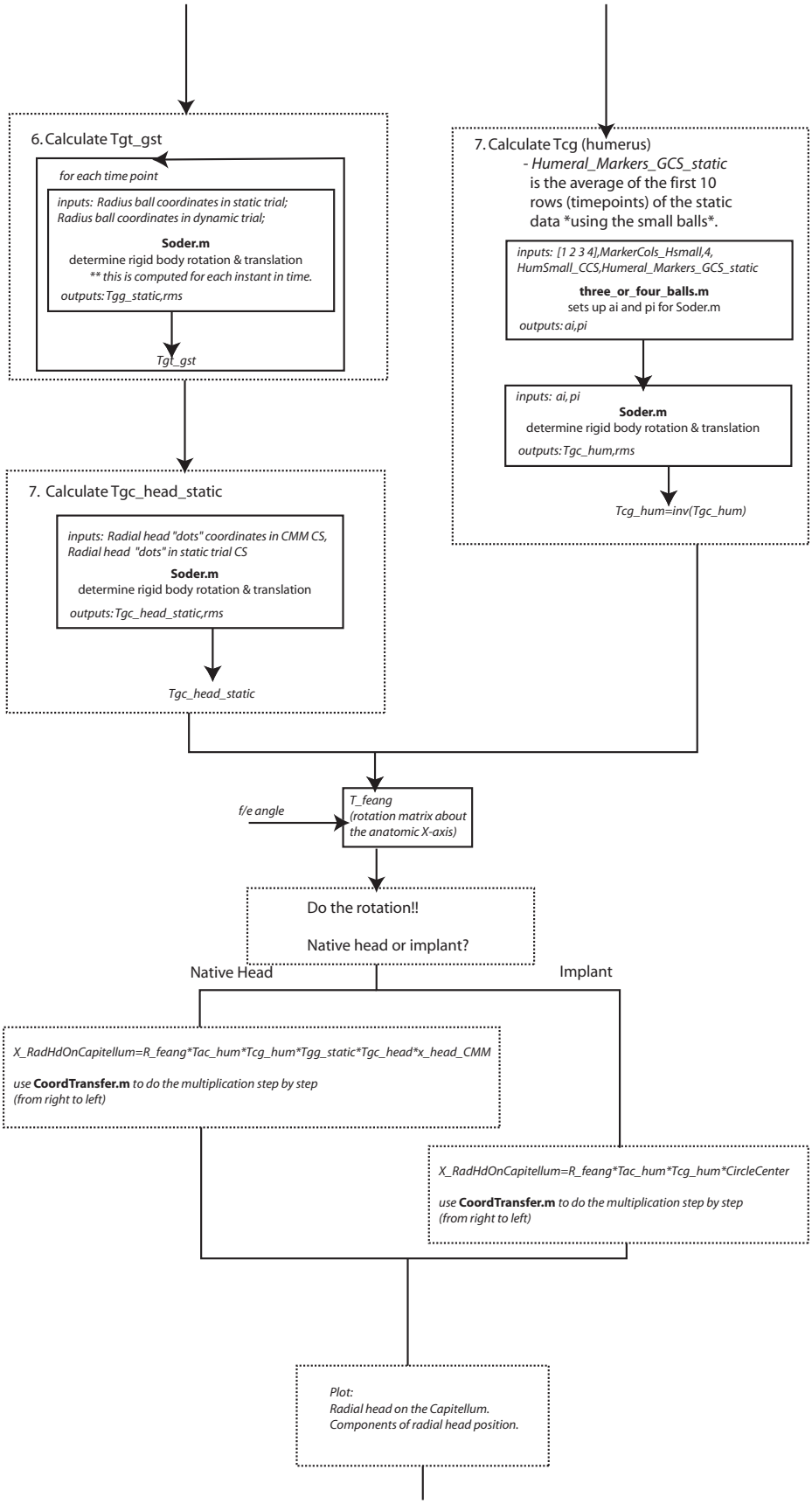


Figure 73: Flowchart leading to the calculation of the native head position.


LaurelsProgram.m
Inputs:
datafilename
sheet
cut_off_frequency
RorL
HeadorImp
HumBigCMM
HumSmCMM
HorlCMM
fe_ang
staticfilename
staticsheet
datafilename
sheet
staticfilename
staticsheet
HumBigCMM
HumSmCMM
HorlCMM
1. Read in Data
a. Trajectory (marker) data
(static and dynamic)
b. CMM measurements
(set radius of capitellum)
(remove first and last columns of CMM files)
(convert to mm from inches)
cut_off_frequency
2. Smooth by Butterworth Filter
- use function butter_karol.m
3. Establish which columns of data are:
- the Radius
- the Radial head or implant.
(Do this for both the Native Head and the Implant case.)
Native head or implant?
Native Head
Implant
4. Calculate Tac (humerus)
inputs: centriod of capitellum;
centroid of trochlear groove,
centroid of the capitellum,
centroid of humeral shaft
xyz_CMM.m
set up anatomical coordinate system from CMM data
outputs: Tca_hum
Tac_hum=inv(Tca_hum)
4. Fit a circle to the visible radial head "dots".
for each time point
inputs: datapoint, num_markers
center_circle_implants.m
determine rigid body rotation & translation
outputs: center, radius
5. Calculate Tcg (humerus)
- Humeral_Markers_GCS_static is the average of the first 10 rows (timepoints) of the static data *using the big balls*.
inputs: Humerus big balls from CMM;
Humeral big balls in static frame.
Soder.m
determine rigid body rotation & translation
outputs: Tgc_hum,rms
Tcg_hum=inv(Tgc_hum)
5. Plot the path of the center of the circle, in the global coordinate system.
6. Calculate Tac (humerus)
inputs: centriod of capitellum;
centroid of trochlear groove,
centroid of the capitellum,
centroid of humeral shaft
xyz_CMM.m
set up anatomical coordinate system from CMM data
outputs: Tca_hum
Tac_hum=inv(Tca_hum)


6. Calculate Tgt_gst
for each time point
inputs: Radius ball coordinates in static trial;
Radius ball coordinates in dynamic trial;
Soder.m
determine rigid body rotation & translation
** this is computed for each instant in time.
outputs: Tgg_static,rms
Tgt_gst
7. Calculate Tcg (humerus)
- Humeral_Markers_GCS_static
is the average of the first 10
rows (timepoints) of the static
data *using the small balls*.
inputs: [1 2 3 4],MarkerCols_Hsmall,4,
HumSmall_CCS,Humeral_Markers_GCS_static
three_or_four_balls.m
sets up ai and pi for Soder.m
outputs: ai,pi
inputs: ai, pi
Soder.m
determine rigid body rotation & translation
outputs: Tgc_hum,rms
Tcg_hum=inv(Tgc_hum)
7. Calculate Tgc_head_static
inputs: Radial head "dots" coordinates in CMM CS,
Radial head "dots" in static trial CS
Soder.m
determine rigid body rotation & translation
outputs: Tgc_head_static,rms
Tgc_head_static
f/e angle
T_feang
(rotation matrix about
the anatomic X-axis)
Do the rotation!!
Native head or implant?
Native Head
Implant
X_RadHdOnCapitellum=R_feang*Tac_hum*Tcg_hum*Tgg_static*Tgc_head*x_head_CMM
use CoordTransfer.m to do the multiplication step by step
(from right to left)
X_RadHdOnCapitellum=R_feang*Tac_hum*Tcg_hum*CircleCenter
use CoordTransfer.m to do the multiplication step by step
(from right to left)
Plot:
Radial head on the Capitellum.
Components of radial head position.

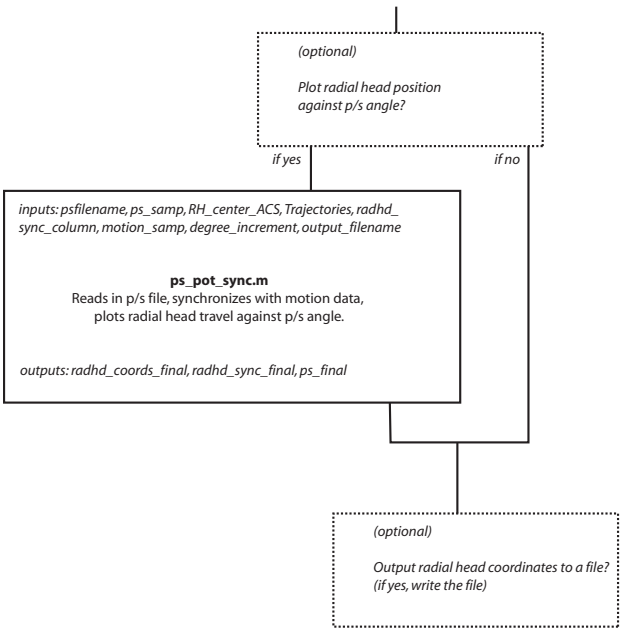


Figure 74: Schematic of MATLAB program.

APPENDIX N

RADIAL HEAD TRACKING MATLAB CODE

This Appendix presents the MATLAB code used for tracking the radial head, as described in Chapter 8.0 . The functions are arranged in alphabetical order by title.

N.1 BUTTER_KAROL.M

```
1 function [num, den, z, p] = butter_karol(n, Wn, varargin)
2
3 % Karol's Galik modification. All functions needed are pasted to the end of
    this
4 % function
5 %BUTTER Butterworth digital and analog filter design.
6 % [B,A] = BUTTER(N,Wn) designs an Nth order lowpass digital
7 % Butterworth filter and returns the filter coefficients in length
8 % N+1 vectors B (numerator) and A (denominator). The coefficients
9 % are listed in descending powers of z. The cut-off frequency
10 % Wn must be 0.0 < Wn < 1.0, with 1.0 corresponding to
11 % half the sample rate.
12 %
13 % If Wn is a two-element vector, Wn = [W1 W2], BUTTER returns an
14 % order 2N bandpass filter with passband W1 < W < W2.
15 % [B,A] = BUTTER(N,Wn,'high') designs a highpass filter.
16 % [B,A] = BUTTER(N,Wn,'stop') is a bandstop filter if Wn = [W1 W2].
17 %
18 % When used with three left-hand arguments, as in
19 % [Z,P,K] = BUTTER(...), the zeros and poles are returned in
20 % length N column vectors Z and P, and the gain in scalar K.
21 %
22 % When used with four left-hand arguments, as in
23 % [A,B,C,D] = BUTTER(...), state-space matrices are returned.
24 %
```

```
25 % BUTTER(N,Wn,'s'), BUTTER(N,Wn,'high','s') and BUTTER(N,Wn,'stop','s')
26 % design analog Butterworth filters. In this case, Wn can be bigger
27 % than 1.0.
28 %
29 % See also BUTTORD, BESSELF, CHEBY1, CHEBY2, ELLIP, FREQZ, FILTER.
30
31 % Author(s): J.N. Little, 1-14-87
32 % J.N. Little, 1-14-88, revised
33 % L. Shure, 4-29-88, revised
34 % T. Krauss, 3-24-93, revised
35 % Copyright (c) 1988-98 by The MathWorks, Inc.
36 % $Revision: 1.1 $ $Date: 1998/06/03 14:42:08 $
37
38 % References:
39 % [1] T. W. Parks and C. S. Burrus, Digital Filter Design,
40 % John Wiley & Sons, 1987, chapter 7, section 7.3.3.
41
42 [btype,analog,errStr] = iirchk(Wn,varargin{:});
43 error(errStr)
44
45 if n>500
46 error('Filter order too large.')
47 end
48
49 % step 1: get analog, pre-warped frequencies
50 if ~analog,
51 fs = 2;
52 u = 2*fs*tan(pi*Wn/fs);
53 else
54 u = Wn;
55 end
56
57 Bw=[];
58 % step 2: convert to low-pass prototype estimate
59 if btype == 1 % lowpass
60 Wn = u;
61 elseif btype == 2 % bandpass
62 Bw = u(2) - u(1);
63 Wn = sqrt(u(1)*u(2)); % center frequency
64 elseif btype == 3 % highpass
65 Wn = u;
66 elseif btype == 4 % bandstop
67 Bw = u(2) - u(1);
68 Wn = sqrt(u(1)*u(2)); % center frequency
69 end
70
71 % step 3: Get N-th order Butterworth analog lowpass prototype
72 [z,p,k] = buttap(n);
73
74 % Transform to state-space
75 [a,b,c,d] = zp2ss(z,p,k);
76
77 % step 4: Transform to lowpass, bandpass, highpass, or bandstop of desired
    Wn
78 if btype == 1 % Lowpass
79 [a,b,c,d] = lp2lp(a,b,c,d,Wn);
80
```

```
81 elseif btype == 2 % Bandpass
82 [a,b,c,d] = lp2bp(a,b,c,d,Wn,Bw);
83
84 elseif btype == 3 % Highpass
85 [a,b,c,d] = lp2hp(a,b,c,d,Wn);
86
87 elseif btype == 4 % Bandstop
88 [a,b,c,d] = lp2bs(a,b,c,d,Wn,Bw);
89 end
90
91 % step 5: Use Bilinear transformation to find discrete equivalent:
92 if ~analog,
93 [a,b,c,d] = bilinear(a,b,c,d,fs);
94 end
95
96 if nargout == 4
97 num = a;
98 den = b;
99 z = c;
100 p = d;
101 else % nargout <= 3
102 % Transform to zero-pole-gain and polynomial forms:
103 if nargout == 3
104 [z,p,k] = ss2zp(a,b,c,d,1);
105 z = buttzeros(btype,n,Wn,Bw,analog);
106 num = z;
107 den = p;
108 z = k;
109 else % nargout <= 2
110 den = poly(a);
111 num = buttnum(btype,n,Wn,Bw,analog,den);
112 % num = poly(a-b*c)+(d-1)*den;
113
114 end
115 end
116
117 %---------------------------------
118 function b = buttnum(btype,n,Wn,Bw,analog,den)
119 % This internal function returns more exact numerator vectors
120 % for the num/den case.
121 % Wn input is two element band edge vector
122 if analog
123 switch btype
124 case 1 % lowpass
125 b = [zeros(1,n) n^(-n)];
126 b = real( b*polyval(den,-j*0)/polyval(b,-j*0) );
127 case 2 % bandpass
128 b = [zeros(1,n) Bw^n zeros(1,n)];
129 b = real( b*polyval(den,-j*Wn)/polyval(b,-j*Wn) );
130 case 3 % highpass
131 b = [1 zeros(1,n)];
132 b = real( b*den(1)/b(1) );
133 case 4 % bandstop
134 r = j*Wn*((-1).^(0:2*n-1)');
135 b = poly(r);
136 b = real( b*polyval(den,-j*0)/polyval(b,-j*0) );
137 end
```

```
138 else
139 Wn = 2*atan2(Wn,4);
140 switch btype
141 case 1 % lowpass
142 r = -ones(n,1);
143 w = 0;
144 case 2 % bandpass
145 r = [ones(n,1); -ones(n,1)];
146 w = Wn;
147 case 3 % highpass
148 r = ones(n,1);
149 w = pi;
150 case 4 % bandstop
151 r = exp(j*Wn*( (-1).^(0:2*n-1)' ));
152 w = 0;
153 end
154 b = poly(r);
155 % now normalize so |H(w)| == 1:
156 kern = exp(-j*w*(0:length(b)-1));
157 b = real(b*(kern*den(:))/(kern*b(:)));
158 end
159
160 function z = buttzeros(btype,n,Wn,Bw,analog)
161 % This internal function returns more exact zeros.
162 % Wn input is two element band edge vector
163 if analog
164 % for lowpass and bandpass, don't include zeros at +Inf or -Inf
165 switch btype
166 case 1 % lowpass
167 z = zeros(0,1);
168 case 2 % bandpass
169 z = zeros(n,1);
170 case 3 % highpass
171 z = zeros(n,1);
172 case 4 % bandstop
173 z = j*Wn*((-1).^(0:2*n-1)');
174 end
175 else
176 Wn = 2*atan2(Wn,4);
177 switch btype
178 case 1 % lowpass
179 z = -ones(n,1);
180 case 2 % bandpass
181 z = [ones(n,1); -ones(n,1)];
182 case 3 % highpass
183 z = ones(n,1);
184 case 4 % bandstop
185 z = exp(j*Wn*( (-1).^(0:2*n-1)' ));
186 end
187 end
188
189 function [z,p,k] = buttap(n)
190 %BUTTAP Butterworth analog lowpass filter prototype.
191 % [Z,P,K] = BUTTAP(N) returns the zeros, poles, and gain
192 % for an N-th order normalized prototype Butterworth analog
193 % lowpass filter. The resulting filter has N poles around
194 % the unit circle in the left half plane, and no zeros.
```

```
195 %
196 % See also BUTTER, CHEB1AP, CHEB2AP, ELLIPAP.
197
198 % Author(s): J.N. Little and J.O. Smith, 1-14-87
199 % L. Shure, 1-13-88, revised
200 % Copyright (c) 1988-98 by The MathWorks, Inc.
201 % $Revision: 1.1 $ $Date: 1998/06/03 14:42:05 $
202
203 % Poles are on the unit circle in the left-half plane.
204 z = [];
205 p = exp(i*(pi*(1:2:n-1)/(2*n) + pi/2));
206 p = [p; conj(p)];
207 p = p(:);
208 if rem(n,2)==1 % n is odd
209 p = [p; -1];
210 end
211 k = real(prod(-p));
212
213
214 function [a,b,c,d] = zp2ss(z,p,k)
215 %ZP2SS Zero-pole to state-space conversion.
216 % [A,B,C,D] = ZP2SS(Z,P,K) calculates a state-space representation:
217 % .
218 % x = Ax + Bu
219 % y = Cx + Du
220 %
221 % for a system given a set of pole locations in column vector P,
222 % a matrix Z with the zero locations in as many columns as there are
223 % outputs, and the gains for each numerator transfer function in
224 % vector K. The A,B,C,D matrices are returned in block diagonal
225 % form.
226 %
227 % This function handles SIMO systems if the Control Toolbox is
228 % present and SISO systems if only the Signal Processing Toolbox
229 % is installed.
230 %
231 % See also SS2ZP.
232
233 % J.N. Little & G.F. Franklin 8-4-87
234 % Revised 12-27-88 JNL, 12-8-89, 11-12-90, 3-22-91, A.Grace, 7-29-96 P.
    Gahinet
235 % Copyright (c) 1984-98 by The MathWorks, Inc.
236 % $Revision: 1.21 $ $Date: 1997/11/21 23:41:23 $
237
238 [mn,nn] = size(z); [md,nd]=size(p);
239 % Put it in column format if its SISO and in row format.
240 if (length(k)==1 & md < 2 & mn < 2) & (nn > 1 | nd > 1)
241 z = z'; p = p';
242 end
243 [m,n] = size(z);
244 if n==0, n=length(k); end % Fix to handle multi-output when z is empty
245 if length(k) ~= n & (~isempty(z))
246 error('Z should be a column vector or K should be SIMO.');
247 end
248 if n > 1
249 % If it's multi-output, we can't use the nice algorithm
250 % that follows, so use the numerically unreliable method
```

```
% of going through polynomial form, and then return.
eval('[num,den] = zp2tf(z,p,k);') % Suppress compile-time diagnostics
[a,b,c,d] = tf2ss(num,den);
return
elseif length(z)>length(p),
% Error out if SISO ZPK model is not proper
error('Denominator must be higher or equal order than numerator.')
end
% Strip infinities and throw away.
p = p(isfinite(p));
z = z(isfinite(z));
% Group into complex pairs
np = length(p);
nz = length(z);
z = cplxpair(z,1e6*nz*norm(z)*eps + eps);
p = cplxpair(p,1e6*np*norm(p)*eps + eps);
% Initialize state-space matrices for running series
a=[]; b=zeros(0,1); c=ones(1,0); d=1;
% If odd number of poles AND zeros, convert the pole and zero
% at the end into state-space.
% H(s) = (s-z1)/(s-p1) = (s + num(2)) / (s + den(2))
if rem(np,2) & rem(nz,2)
a = p(np);
b = 1;
c = p(np) - z(nz);
d = 1;
np = np - 1;
nz = nz - 1;
end
% If odd number of poles only, convert the pole at the
% end into state-space.
% H(s) = 1/(s-p1) = 1/(s + den(2))
if rem(np,2)
a = p(np);
b = 1;
c = 1;
d = 0;
np = np - 1;
end
% If odd number of zeros only, convert the zero at the
% end, along with a pole-pair into state-space.
% H(s) = (s+num(2))/(s^2+den(2)s+den(3))
if rem(nz,2)
num = real(poly(z(nz)));
den = real(poly(p(np-1:np)));
wn = sqrt(prod(abs(p(np-1:np))));
if wn == 0, wn = 1; end
t = diag([1 1/wn]); % Balancing transformation
a = t\[-den(2) -den(3); 1 0]*t;
b = t\[1; 0];
c = [1 num(2)]*t;
d = 0;
nz = nz - 1;
np = np - 2;
end
% Now we have an even number of poles and zeros, although not
% necessarily the same number - there may be more poles.
```

```
% H(s) = (s^2+num(2)s+num(3))/(s^2+den(2)s+den(3))
% Loop thru rest of pairs, connecting in series to build the model.
i = 1;
while i < nz
index = i:i+1;
num = real(poly(z(index)));
den = real(poly(p(index)));
wn = sqrt(prod(abs(p(index))));
if wn == 0, wn = 1; end
t = diag([1 1/wn]); % Balancing transformation
a1 = t\[-den(2) -den(3); 1 0]*t;
b1 = t\[1; 0];
c1 = [num(2)-den(2) num(3)-den(3)]*t;
d1 = 1;
% [a,b,c,d] = series(a,b,c,d,a1,b1,c1,d1);
% Next lines perform series connection
[ma1,na1] = size(a);
[ma2,na2] = size(a1);
a = [a zeros(ma1,na2); b1*c a1];
b = [b; b1*d];
c = [d1*c c1];
d = d1*d;
i = i + 2;
end
% Take care of any left over unmatched pole pairs.
% H(s) = 1/(s^2+den(2)s+den(3))
while i < np
den = real(poly(p(i:i+1)));
wn = sqrt(prod(abs(p(i:i+1))));
if wn == 0, wn = 1; end
t = diag([1 1/wn]); % Balancing transformation
a1 = t\[-den(2) -den(3); 1 0]*t;
b1 = t\[1; 0];
c1 = [0 1]*t;
d1 = 0;
% [a,b,c,d] = series(a,b,c,d,a1,b1,c1,d1);
% Next lines perform series connection
[ma1,na1] = size(a);
[ma2,na2] = size(a1);
a = [a zeros(ma1,na2); b1*c a1];
b = [b; b1*d];
c = [d1*c c1];
d = d1*d;
i = i + 2;
end
% Apply gain k:
c = c*k;
d = d*k;
% End of zp2ss.m function


function [at,bt,ct,dt] = lp2lp(a,b,c,d,wo)
%LP2LP Lowpass to lowpass analog filter transformation.
% [NUMT,DENT] = LP2LP(NUM,DEN,Wo) transforms the lowpass filter
% prototype NUM(s)/DEN(s) with unity cutoff frequency of 1 rad/sec
% to a lowpass filter with cutoff frequency Wo (rad/sec).
% [AT,BT,CT,DT] = LP2LP(A,B,C,D,Wo) does the same when the
```

```
% filter is described in state-space form.
%
% See also BILINEAR, IMPINVAR, LP2BP, LP2BS and LP2HP

% Author(s): J.N. Little and G.F. Franklin, 8-4-87
% Copyright (c) 1988-98 by The MathWorks, Inc.
% $Revision: 1.3 $ $Date: 1998/07/13 19:02:11 $

if nargin == 3 % Transfer function case
% handle column vector inputs: convert to rows
if size(a,2) == 1
a = a(:).';
end
if size(b,2) == 1
b = b(:).';
end
% Transform to state-space
wo = c;
[a,b,c,d] = tf2ss(a,b);
end
error(abcdchk(a,b,c,d));
[ma,nb] = size(b);
[mc,ma] = size(c);
% Transform lowpass to lowpass
at = wo*a;
bt = wo*b;
ct = c;
dt = d;
if nargin == 3 % Transfer function case
% Transform back to transfer function
[z,k] = tzero(at,bt,ct,dt);
num = k * poly(z);
den = poly(at);
at = num;
bt = den;
end

function [at,bt,ct,dt] = lp2bp(a,b,c,d,wo,bw)
%LP2BP Lowpass to bandpass analog filter transformation.
% [NUMT,DENT] = LP2BP(NUM,DEN,Wo,Bw) transforms the lowpass filter
% prototype NUM(s)/DEN(s) with unity cutoff frequency to a
% bandpass filter with center frequency Wo and bandwidth Bw.
% [AT,BT,CT,DT] = LP2BP(A,B,C,D,Wo,Bw) does the same when the


% filter is described in state-space form.
%
% See also BILINEAR, IMPINVAR, LP2LP, LP2BS and LP2HP

% Author(s): J.N. Little and G.F. Franklin, 8-4-87
% Copyright (c) 1988-98 by The MathWorks, Inc.
% $Revision: 1.2 $ $Date: 1998/07/13 19:02:11 $

if nargin == 4 % Transfer function case
% handle column vector inputs: convert to rows
if size(a,2) == 1
a = a(:).';
```

```
422 end
423 if size(b,2) == 1
424 b = b(:).';
425 end
426 % Transform to state-space
427 wo = c;
428 bw = d;
429 [a,b,c,d] = tf2ss(a,b);
430 end
431 error(abcdchk(a,b,c,d));
432 [ma,nb] = size(b);
433 [mc,ma] = size(c);
434 % Transform lowpass to bandpass
435 q = wo/bw;
436 at = wo*[a/q eye(ma); -eye(ma) zeros(ma)];
437 bt = wo*[b/q; zeros(ma,nb)];
438 ct = [c zeros(mc,ma)];
439 dt = d;
440 if nargin == 4 % Transfer function case
441 % Transform back to transfer function
442 [z,k] = tzero(at,bt,ct,dt);
443 num = k * poly(z);
444 den = poly(at);
445 at = num;
446 bt = den;
447 end
448
449 function [at,bt,ct,dt] = lp2hp(a,b,c,d,wo)
450 %LP2HP Lowpass to highpass analog filter transformation.
451 % [NUMT,DENT] = LP2HP(NUM,DEN,Wo) transforms the lowpass filter
452 % prototype NUM(s)/DEN(s) with unity cutoff frequency to a
453 % highpass filter with cutoff frequency Wo.
454 % [AT,BT,CT,DT] = LP2HP(A,B,C,D,Wo) does the same when the
455 % filter is described in state-space form.
456 %
457 % See also BILINEAR, IMPINVAR, LP2BP, LP2BS and LP2LP
458
459 % Author(s): J.N. Little and G.F. Franklin, 8-4-87
460 % Copyright (c) 1988-98 by The MathWorks, Inc.
461 % $Revision: 1.2 $ $Date: 1998/07/13 19:02:12 $
462
463 if nargin == 3 % Transfer function case
464 % handle column vector inputs: convert to rows
465 if size(a,2) == 1
466 a = a(:).';
467 end
468 if size(b,2) == 1
469 b = b(:).';
470 end
471 % Transform to state-space
472 wo = c;
473 [a,b,c,d] = tf2ss(a,b);
474 end
475 error(abcdchk(a,b,c,d));
476 [ma,nb] = size(b);
477 [mc,ma] = size(c);
478
```

```
% Transform lowpass to highpass
at = wo*inv(a);
bt = -wo*(a\b);
ct = c/a;
dt = d - c/a*b;
if nargin == 3 % Transfer function case
% Transform back to transfer function
[z,k] = tzero(at,bt,ct,dt);
num = k * poly(z);
den = poly(at);
at = num;
bt = den;
end

function [at,bt,ct,dt] = lp2bs(a,b,c,d,wo,bw)
%LP2BS Lowpass to bandstop analog filter transformation.
% [NUMT,DENT] = LP2BS(NUM,DEN,Wo,Bw) transforms the lowpass filter
% prototype NUM(s)/DEN(s) with unity cutoff frequency to a
% bandstop filter with center frequency Wo and bandwidth Bw.
% [AT,BT,CT,DT] = LP2BS(A,B,C,D,Wo,Bw) does the same when the
% filter is described in state-space form.
%
% See also BILINEAR, IMPINVAR, LP2BP, LP2LP and LP2HP

% Author(s): J.N. Little and G.F. Franklin, 8-4-87
% Copyright (c) 1988-98 by The MathWorks, Inc.
% $Revision: 1.2 $ $Date: 1998/07/13 19:02:11 $

if nargin == 4 % Transfer function case
% handle column vector inputs: convert to rows
if size(a,2) == 1
a = a(:).';
end
if size(b,2) == 1
b = b(:).';
end
% Transform to state-space
wo = c;
bw = d;
[a,b,c,d] = tf2ss(a,b);
end
error(abcdchk(a,b,c,d));
[ma,nb] = size(b);
[mc,ma] = size(c);
% Transform lowpass to bandstop
q = wo/bw;
at = [wo/q*inv(a) wo*eye(ma); -wo*eye(ma) zeros(ma)];
bt = -[wo/q*(a\b); zeros(ma,nb)];
ct = [c/a zeros(mc,ma)];
dt = d - c/a*b;
if nargin == 4 % Transfer function case
% Transform back to transfer function
[z,k] = tzero(at,bt,ct,dt);
num = k * poly(z);
den = poly(at);
at = num;
bt = den;
```

```
536 end
537
538 function [btype,analog,errStr] = iirchk(Wn,varargin)
539 %IIRCHK Parameter checking for BUTTER, CHEBY1, CHEBY2, and ELLIP.
540 % [btype,analog,errStr] = iirchk(Wn,varargin) returns the
541 % filter type btype (1=lowpass, 2=bandpss, 3=highpass, 4=bandstop)
542 % and analog flag analog (0=digital, 1=analog) given the edge
543 % frequency Wn (either a one or two element vector) and the
544 % optional arguments in varargin. The variable arguments are
545 % either empty, a one element cell, or a two element cell.
546 %
547 % errStr is empty if no errors are detected; otherwise it contains
548 % the error message. If errStr is not empty, btype and analog
549 % are invalid.
550
551 % Copyright (c) 1988-98 by The MathWorks, Inc.
552 % $Revision: 1.1 $
553
554 errStr = '';
555
556 % Define defaults:
557 analog = 0; % 0=digital, 1=analog
558 btype = 1; % 1=lowpass, 2=bandpss, 3=highpass, 4=bandstop
559
560 if length(Wn)==1
561 btype = 1;
562 elseif length(Wn)==2
563 btype = 2;
564 else
565 errStr = 'Wn must be a one or two element vector.';
566 return
567 end
568
569 if length(varargin)>2
570 errStr = 'Too many input arguments.';
571 return
572 end
573
574 % Interpret and strip off trailing 's' or 'z' argument:
575 if length(varargin)>0
576 switch lower(varargin{end})
577 case 's'
578 analog = 1;
579 varargin(end) = [];
580 case 'z'
581 analog = 0;
582 varargin(end) = [];
583 otherwise
584 if length(varargin) > 1
585 errStr = 'Analog flag must be either ''z'' or ''s''.';
586 return
587 end
588 end
589 end
590
591 % At this point, varargin will either be empty, or contain a single
592 % band type flag.
```

```
593
594 if length(varargin)==1 % Interpret filter type argument:
595 switch lower(varargin{1})
596 case 'low'
597 btype = 1;
598 case 'bandpass'
599 btype = 2;
600 case 'high'
601 btype = 3;
602 case 'stop'
603 btype = 4;
604 otherwise
605 if nargin == 2
606 errStr = ['Option string must be one of ''high'', ''stop'',' ...
607 ' ''low'', ''bandpass'', ''z'' or ''s''.'];
608 else % nargin == 3
609 errStr = ['Filter type must be one of ''high'', ''stop'',' ...
610 ' ''low'', or ''bandpass''.'];
611 end
612 return
613 end
614 switch btype
615 case 1
616 if length(Wn)~=1
617 errStr = 'For the ''low'' filter option, Wn must have 1 element.';
618 return
619 end
620 case 2
621 if length(Wn)~=2
622 errStr = 'For the ''bandpass'' filter option, Wn must have 2 elements.';
623 return
624 end
625 case 3
626 if length(Wn)~=1
627 errStr = 'For the ''high'' filter option, Wn must have 1 element.';
628 return
629 end
630 case 4
631 if length(Wn)~=2
632 errStr = 'For the ''stop'' filter option, Wn must have 2 elements.';
633 return
634 end
635 end
636 end
637 %end of "iirchk.m"
638
639 function [zd, pd, kd, dd] = bilinear(z, p, k, fs, fp, fp1)
640 %BILINEAR Bilinear transformation with optional frequency prewarping.
641 % [Zd,Pd,Kd] = BILINEAR(Z,P,K,Fs) converts the s-domain transfer
642 % function specified by Z, P, and K to a z-transform discrete
643 % equivalent obtained from the bilinear transformation:
644 %
645 % H(z) = H(s) |
646 % | s = 2*Fs*(z-1)/(z+1)
647 %
648 % where column vectors Z and P specify the zeros and poles, scalar
649 % K specifies the gain, and Fs is the sample frequency in Hz.
```

```
650 % [NUMd,DENd] = BILINEAR(NUM,DEN,Fs), where NUM and DEN are
651 % row vectors containing numerator and denominator transfer
652 % function coefficients, NUM(s)/DEN(s), in descending powers of
653 % s, transforms to z-transform coefficients NUMd(z)/DENd(z).
654 % [Ad,Bd,Cd,Dd] = BILINEAR(A,B,C,D,Fs) is a state-space version.
655 % Each of the above three forms of BILINEAR accepts an optional
656 % additional input argument that specifies prewarping. For example,
657 % [Zd,Pd,Kd] = BILINEAR(Z,P,K,Fs,Fp) applies prewarping before
658 % the bilinear transformation so that the frequency responses
659 % before and after mapping match exactly at frequency point Fp
660 % (match point Fp is specified in Hz).
661 %
662 % See also IMPINVAR.
663
664 % Author(s): J.N. Little, 4-28-87
665 % J.N. Little, 5-5-87, revised
666 % Copyright (c) 1988-98 by The MathWorks, Inc.
667 % $Revision: 1.1 $ $Date: 1998/06/03 14:41:57 $
668
669 % Gene Franklin, Stanford Univ., motivated the state-space
670 % approach to the bilinear transformation.
671
672 [mn,nn] = size(z);
673 [md,nd] = size(p);
674
675 if (nd == 1 & nn < 2) & nargout ~= 4 % In zero-pole-gain form
676 if mn > md
677 error('Numerator cannot be higher order than denominator.')
678 end
679 if nargin == 5 % Prewarp
680 fp = 2*pi*fp;
681 fs = fp/tan(fp/fs/2);
682 else
683 fs = 2*fs;
684 end
685 z = z(finite(z)); % Strip infinities from zeros
686 pd = (1+p/fs)./(1-p/fs); % Do bilinear transformation
687 zd = (1+z/fs)./(1-z/fs);
688 % real(kd) or just kd?
689 kd = (k*prod(fs-z)./prod(fs-p));
690 zd = [zd;-ones(length(pd)-length(zd),1)]; % Add extra zeros at -1
691
692 elseif (md == 1 & mn == 1) | nargout == 4 %
693 if nargout == 4 % State-space case
694 a = z; b = p; c = k; d = fs; fs = fp;
695 error(abcdchk(a,b,c,d));
696 if nargin == 6 % Prewarp
697 fp = fp1; % Decode arguments
698 fp = 2*pi*fp;
699 fs = fp/tan(fp/fs/2)/2;
700 end
701 else % Transfer function case
702 if nn > nd
703 error('Numerator cannot be higher order than denominator.')
704 end
705 num = z; den = p; % Decode arguments
706 if nargin == 4 % Prewarp
```

```
fp = fs; fs = k; % Decode arguments
fp = 2*pi*fp;
fs = fp/tan(fp/fs/2)/2;
else
fs = k; % Decode arguments
end
% Put num(s)/den(s) in state-space canonical form.
[a,b,c,d] = tf2ss(num,den);
end
% Now do state-space version of bilinear transformation:
t = 1/fs;
r = sqrt(t);
t1 = eye(size(a)) + a*t/2;
t2 = eye(size(a)) - a*t/2;
ad = t2\t1;
bd = t/r*(t2\b);
cd = r*c/t2;
dd = c/t2*b*t/2 + d;
if nargout == 4
zd = ad; pd = bd; kd = cd;
else
% Convert back to transfer function form:
p = poly(ad);
zd = poly(ad-bd*cd)+(dd-1)*p;
pd = p;
end
else
error('First two arguments must have the same orientation.')
end
%end of "bilinear.m"
```

N.2 CENTER_CIRCLE_IMPLANTS.M

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Find the Center of the circle and Radius from points in 3D %%
%% Measured Radius of the circle: 30.61mm %%
%% Cylinder R:1.00507 in, ball R:0.2 in %%
%% By Sunghwan Kim 10/10/2007 %%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Find the optimal plane of the points
% Least square method
% clc;close all;clear all;


function [Circle_center, Mean_Radius]=center_circle_implants(datapoint,
    num_markers)

[mm,nn]=size(datapoint); % it will always be one row, but it will be a
    variable number of columns
px=[];
py=[];
```

```
pz=[];
for ii=1:num_markers
px=[px datapoint(:,ii*3-2)]; % assume data are set up x1, y1, z1, x2, y2,
    z2 (no weird things in between markers.)
py=[py datapoint(:,ii*3-1)];
pz=[pz datapoint(:,ii*3)];

end %% end ii for
size(px);
size(py);
size(pz);


[m,n]=size(px); % m:time step, n:number of points to fit
normal=zeros(m,3); % normal vectors
Circle_center=zeros(m,3); % center of circle
Radius=zeros(m,1); % Radius vector

for k=1:m % time loop

A=zeros(3,3);
B=zeros(3,1);
% Define matrix A
for i=1:n
A(1,1)=A(1,1)+px(k,i)^2;
A(1,2)=A(1,2)+px(k,i)*py(k,i);
A(1,3)=A(1,3)+px(k,i)*pz(k,i);
A(2,2)=A(2,2)+py(k,i)^2;
A(2,3)=A(2,3)+py(k,i)*pz(k,i);
A(3,3)=A(3,3)+pz(k,i)^2;
end
A(2,1)=A(1,2);
A(3,1)=A(1,3);
A(3,2)=A(2,3);

% Define matrix B
for i=1:n
B(1,1)=B(1,1)+px(k,i);
B(2,1)=B(2,1)+py(k,i);
B(3,1)=B(3,1)+pz(k,i);
end

X=inv(A)*B; % solve A*X=B
normal(k,:)=(X/norm(X))';
% Check the result
% p1=[px(1,1),py(1,1),pz(1,1)]';
% p2=[px(2,1),py(2,1),pz(2,1)]';
% p3=[px(3,1),py(3,1),pz(3,1)]';
% normal_1=cross(p1-p2,p3-p2);
% normal_1=normal_1/norm(normal_1)

% Calculate the rotation matrix
if (normal(k,1)<0.00001 & normal(k,2)<0.00001 & abs(normal(k,3))==1)
M=[1 0 0;0 1 0;0 0 1];
else
M=zeros(3,3);
normal_p=[-normal(k,2) normal(k,1) 0]';
```

```
74 normal_p=normal_p/norm(normal_p);
75 n3=cross((normal(k,:))',normal_p);
76 M=[normal_p n3 normal(k,:)'];
77 M=transpose(M);
78 % M*normal(k,:)'; % should be (0 0 1)'
79 end
80
81 % Mapping n points
82 P=M*[px(k,:);py(k,:);pz(k,:)];
83 px_m=P(1,:); % x coordinate of mapped points
84 py_m=P(2,:); % y coordinate of mapped points
85 pz_m=P(3,:); % z coordinate of mapped points
86 Z=mean(pz_m); % average z coordinate
87
88 %% Find the center of the circle in 2D
89 %% Least square fit
90
91 AA=zeros(3,3);
92 BB=zeros(3,1);
93
94 % Define matrix AA
95 for i=1:n
96 AA(1,1)=AA(1,1)+px_m(1,i)^2;
97 AA(1,2)=AA(1,2)+px_m(1,i)*py_m(1,i);
98 AA(2,2)=AA(2,2)+py_m(1,i)^2;
99 end
100
101 AA(1,3)=sum(px_m);
102 AA(2,3)=sum(py_m);
103 AA(3,3)=n;
104 AA(2,1)=AA(1,2);
105 AA(3,1)=AA(1,3);
106 AA(3,2)=AA(2,3);
107
108 % Define matrix BB
109 for i=1:n
110 C=px_m(1,i)^2+py_m(1,i)^2;
111 BB(1,1)=BB(1,1)+C*px_m(1,i);
112 BB(2,1)=BB(2,1)+C*py_m(1,i);
113 BB(3,1)=BB(3,1)+C;
114 end
115
116 XX=inv(AA)*(-BB); % Solve AA*X=-BB
117 a=-XX(1,1)/2; % x coordinate of center of circle in mapped plane
118 b=-XX(2,1)/2; % y coordinate of center of circle in mapped plane
119 Radius(k,1)=sqrt(a^2+b^2-XX(3,1)); % Radius of circle
120 CircleCenter=transpose(M)*[a b Z]';
121 Circle_center(k,:)=CircleCenter'; % Center of circle
122
123 end
124
125 Mean_Radius=mean(Radius);
126 Stand_Dev_Radius=std(Radius);
127 Mean_Center_Circle=mean(Circle_center);
128 Stand_Dev_Center=std(Circle_center);
```

N.3 COORD_TRANSFER.M

```
1 function Coor_Tr = Coord_transfer(point,T)
2
3 %-This function transfers coordinates of a point from
4 % one coordinate system to another
5 %-Basically: {Xgm}=[Tgm]*{Xmg} for example
6 %-Input: "point(ipoint,3)" number of rows corresponds to the number of
    points
7 % "T" transformation matrix (4,4)
8
9 [ipoint,n]=size(point);
10
11 one_point(1)=1; % add one so that T*x can be carried out
12 for i=1:ipoint
13 one_point(2:4)=point(i,:);
14 one_point_transf=T*one_point';
15 Coor_Tr(i,:)=one_point_transf'; %_transf
16 end
17
18 % Coor_Tr=[1,x1,y1,z1
19 % 1,x2,y2,z2
20 % .........
21 % 1,xpoint,ypoint,zpoint]
22 %Ged rid off the first column which contains ones
23
24 Coor_Tr(:,1)=[]; %The matrix will collapse to what remains
downsample_LK.m
1 % downsample_LK
2 % Laurel Kuxhaus, lck4@pitt.edu
3 % a quick downsampling program
4
5 function downsampled_data=downsample_LK(ndown, data)
6
7
8 for ii=1:floor(length(data)/ndown)
9 downsampled_data(ii,:)=data(ii*ndown,:);
10 ii;
11 end
LaurelsProgram.m
1 % LaurelsProgram.m
2 % Laurel Kuxhaus, laurel.k@gmail.com
3 % began 12 October 2007, commented 26 November 2007
4 % This is the master run file that will process the radial head travel on
    the capitellum.
5
6 function outputs=LaurelsProgram(datafilename, sheet, cut_off_frequency,
    RorL, HeadorImp, HumBigCMM, HumSmCMM, HorICMM, fe_ang, stati
cfilename, staticsheet)
7 % examples of command-line entries to debug with the MadeUp Validation data
    are included at the very end of this code.
8
9 %%%%%%%TABLE OF VARIABLES%%%%%%%%%%
10 % Inputs:
```

```
11 % datafilename: string of the file that the data is stored in. Example:
    'MyData.xls'
12 %
13 % sheet: name of the sheet in the datafile Excel spreadsheet. Example:
    'mysheet'
14 %
15 % filterfreq: the cutoff frequency for the filter. Use 'N' for no
    filtering.
16 % fe_ang is the flexion angle. THIS IS W.R.T. the vertical (i.e. full
17 % extention is 0 degrees.) This is in degrees - will be converted to
    radians
18 % later (in the code).
19 %
20 % RorL: Right or Left. ('R' for Right, 'L' for left.)
21 %
22 % HeadOrImp = 'H' for nativehead, 'I' for Implant. This controls the
23 % calculations that get done (i.e. circle-fitting is done for the implant
    only.)
24 %
25 % HumBigCMM, HumSmCMM, HorICMM - these are the filenames of the CMM
26 % pointing files. The extension will be .dat .
27 % The order will be:
28 % HumBigCMM AND HumSmCMM files: (order the same as Karol's)
29 % 1) any numbers (placeholder, artifact from Karol's pointing technique)
30 % 2) any numbers (placeholder, artifact from Karol's pointing technique)
31 % 3) Center of the Shaft (point on CMM as a circle, or use the top big
    ball pointed as a sphere if it's really in the center.)
32 % 4) Trochlear Groove (point on CMM as a circle)
33 % 5, 6, 7, 8) Humeral Markers (point on CMM as a sphere)
34 % 9) Capitellum center (point on CMM as a sphere)
35 % HorICMMfile: (same order as Karol's "implant" file order)
36 % 1) Center on the contact surface ((point on CMM as a circle))
37 % 2) (4 rows) 4 "dot" or "ball" coordinates. (for implants, I think this
    can be any 4 dots.)
38 % 3) Any point on RIM
39 % 4) Center on the backside (for both native head and implants)
40 % The columns for all of the files described above will be: [1] - junk
41 % numbers from CMM; [2], [3],[4] - x, y, z coordinates from CMM; [5] -
42 % radius from CMM (manually fill in for things that doesn't have a
    radius.)
43 %
44 % staticfilename and staticsheet are the .xls filename (as a string) and
45 % the sheet (also a string) within that file that contains the marker
    position coordinates
46 % for the static file that shows the relative positions of the radius
    balls
47 % and the radial head Landmark Dots (LDs) the order of markers will be:
48 % Hbig (3-14), Hsmall(15-26), Radius (27-38), RadialHead (39-end)
49 % RadialHeadCMM is the radial head CMM file. If we are doing an implant
50 % data file, just put in anything for the staticfilename, staticsheet, and
51 % RadialHeadCMM. THey will not get used in calculations FOR THE IMPLANT
52 % PROCESSING.
53
54 %%% THINGS That might need to be changed in the future below:
55 % MarkerCols_Hbig and MarkerCols_Hsmall (depending on how the data from
    the
56 % motion system is structured/output.
```

```
% numcols_beforeradhd (the radial head columns should always be last.)
% in the "Wn" line in the Butterworth Filter section (~ line 135 or so),
% may need to change teh "30" (it should be half of whatever the sampling frequency
% is at the time. Right now it assumes sampling frequency of 60 Hz.)


% Establish which columns of data are for which markers. This may need to
% be changed with future setups.
% The assumptions here are that there are 4 big markers on the humerus; 4
% small markers on the humerus near the radial head. For the native head:
% an array of 4 balls on the distal radius AND small black "landmark dots"
% on the part of the radial head that will come off for the implant. For
% the implants, it is assumed that the implants will have little dots
% precisely placed around the circumference in a perfect (or nearly
% perfect) circle.

MarkerCols_Hbig=[3 4 5; 6 7 8; 9 10 11; 12 13 14]; % for Vicon data, the first 2 columns are time and sample.
MarkerCols_Hsmall=[15 16 17; 18 19 20; 21 22 23; 24 25 26]; % these may be bogus numbers for the native head case. If so, just put in anything t
hat is less than the biggest-numbered column in HumBig.
%The Radius columns are established later below.
numcols_beforeradhd=38; % this is hardwired for now - may need to be changed.

% Read in file.
Trajectories=xlsread(datafilename, sheet);
Trajectories_static=xlsread(staticfilename, staticsheet);
[nrows, ncols]=size(Trajectories);
[nrows_static, ncols_static]=size(Trajectories_static);

% Now read in CMM coordinates.
HumBig_CCS=load(HumBigCMM);
HumSm_CCS=load(HumSmCMM);
HorI_CCS=load(HorICMM);

% Get rid of the first column: (this was the number from the CMM that has
% no physical meaning)
HumBig_CCS(:,1)=[];
HumSm_CCS(:,1)=[];
HorI_CCS(:,1)=[];

% convert to mm (because the CMM is set up to do it in inches)
HumBig_CCS=HumBig_CCS.*25.4;
HumSm_CCS=HumSm_CCS.*25.4;
HorI_CCS=HorI_CCS.*25.4;

if HeadorImp=='H'
Radius_of_Capitellum=HumBig_CCS(9,4);
else
Radius_of_Capitellum=HumSm_CCS(9,4);
end % end if.
% get rid of the radius column; (here, "radius" means radius of the circle or
% sphere from the CMM, not the bone of the same name!)
HumBig_CCS(:,4)=[];
```

```
108 HumSm_CCS(:,4)=[];
109 HorI_CCS(:,4)=[];
110
111 % option to filter. This is straight from Karol Galik's code.
112 % % % % %-----------------------------------------------------------------
    -------------
113 % % % % % Smoothing by Butterworth Filter
114 [m,n] = size(Trajectories);
115 % % % %
116 % % % % %Design the filter butter_karol(N,Wn)
117 % % % % %B Nth order lowpass digital. The cut-off frequency Wn must be
    0.0 < Wn < 1.0,
118 % % % % %with 1.0 corresponding to half the sample rate.
119 % % % % % For example, we tape at 60Hz so 0.5*60=30Hz is half the sample
    rate
120 % % % % % King and Veeger use 1.5 Hz cut-off frequency =>
    Wn=1/(30/1.5)=0.05
121 % % % % %Wn = 1/(30/1.5); ,Cut-off frequency Wn must be 0.0 < Wn < 1.0
122 % % % % %-The fundamental frequency was calculated from looking at the
    pron-sup
123 % % % % %graph and counting the number of frames it took for one part of
    the main
124 % % % % %curve- uphill motion. f_fundamental=1(number of
    frames/60seconds).
125 % % % % %Accoording to Woltring, JB 1994, p.1428, the optimal smoothing
    was around
126 % % % % %for Butterworth filter was for cut-off frequency of 5 times the
    fundamental
127 % % % % %frequency of the motion.
128 % % % %
129 Wn = 1/(15/cut_off_frequency); %%%This will need to change based on
    sampling frequency.
130 % % % % Wn = 1/(60/cut_off_frequency); % this seems to make more sense
    and makes things look much better.
131 N=4; % Fourth-order butterworth filter.
132 [col1,col2] = butter_karol(N,Wn);
133
134 if cut_off_frequency ~= 'N' %if we are *not* doing No filtering
135 for i=1:n %Number of columns of the kinematics file
136 Dirty_Stuff = Trajectories(:,i);
137 Filtered_Stuff = filtfilt(col1,col2,Dirty_Stuff);
138 Trajectories(:,i) = Filtered_Stuff;
139 end % end i for
140 clear Dirty_Stuff Filtered_Stuff
141 end % end cut_off_frequency if.
142
143 % can figure out which columns belong to the radial head now. (this was
144 % moved to after filtering.)
145 if HeadorImp=='H'
146 % for the native case, use the distal radius array.
147 MarkerCols_Radius=[27 28 29; 30 31 32; 33 34 35; 36 37 38]; % this may
    need to be changed in teh future to correspond to which colums are the r
adial head.
148 TrajRadius=Trajectories(MarkerCols_Radius);
149 MarkerCols_RadHd=[39 40 41; 42 43 44; 45 46 47; 48 49 50]; % this may
    need to be changed in teh future to correspond to which colums are the r
adial head.
```

```
150 TrajRadHd=Trajectories(MarkerCols_RadHd);
151
152 else % we have an implant
153 howmanycols_dots=ncols-numcols_beforeradhd; %
154 for ii=1:howmanycols_dots/3
155 MarkerCols_dots(ii,1:3)=[numcols_beforeradhd+3*ii-2,
    numcols_beforeradhd+3*ii-1, numcols_beforeradhd+3*ii];
156 end % end for
157 TrajImplant=Trajectories(MarkerCols_dots);
158 end % end if.
159 TrajHum_big=Trajectories(MarkerCols_Hbig);
160 TrajHum_small=Trajectories(MarkerCols_Hsmall);
161
162
163 % Now, fit the circle to the points (implant) or otherwise track the
    radial
164 % head (native head).
165 % Do the circle-fitting up front here because if we go to anatomic
166 % (humeral) coordinates first, we will end up close to aligned with one
    of
167 % the axes, and thus we'd have a singular matrix and the circle-fitting
168 % algorithm would not work.
169
170 if HeadorImp=='H'
171 % I think we don't need to do anything here for the native head.
172 else % this is if we have an implant
173 % fit a circle to the radial head dots at each point in time.
174 % track the center as it moves in time.
175 [lengthtraj, widthtraj]=size(Trajectories);
176 for ii=1:lengthtraj
177 % for each timepoint: figure out how many dots are in view.
178 point_to_send=[]; % reset every time.
179 for jj=1:howmanycols_dots
180 if isnan(Trajectories(ii,jj))~=1 % if the marker is visible
181 point_to_send=[point_to_send, Trajectories(ii,jj+MarkerCols_dots(1)-1)];
182 end % end isnan if.
183 end % end jj for.
184 [center_coordinates(ii,:),
    radius(ii)]=center_circle_implants(point_to_send, 4); % calculate the
    center of the radial head based on the circle-fitting
algorithm.
185 end % end ii for.
186 end % end big if.
187
188 if HeadorImp=='I'
189 % plot the center of the circle in GCS. This is for debugging.
190 figure;
191 plot3(center_coordinates(:,1), center_coordinates(:,2),
    center_coordinates(:,3),'b.');
192 title('Circle Center Coordinates in GCS')
193 end % end plot center of the circle for implants only.
194
195 if HeadorImp=='I'
196 % Transform center_circle_implants to anatomic coordinates.
197 % This will be the plane of the capitellum that we are interested in.
198 % (i.e. for flexion angles other than 90deg, we'll have already taken
    care
```

```
199 % of the f/e angle.) That is, the y-axis will always be along the
    forearm,
200 % the x-axis always m/l, and the z-axis may be skewed with respect to the
201 % humerus.
202
203 % The strategy is: Global -> CMM; CMM-> Anatomic.
204 % Like Karol's program: GCS = Global Coordinate System; CCS= CMM
    Coordinate System;
205 % ACS = Anatomic coordinate system.
206
207 % Need to get: Tac(humerus); Tcg (humerus); Rotate to correct f/e angle.
208 % (R_feang). Then calculation will be: R_feang*Tac*Tcg*center_coordinates
209 % . Then plot.
210
211 % 1. get Tac(humerus).
212
213 % first, make coordinate system in CMM coordinate system.
214 % This is copy/paste from Karol's program.
215 ww(1,:) = HumSm_CCS(9,:); %Center of the Capitellum (was Trochlear Groove
    for Karol's)
216 ww(2,:) = HumSm_CCS(4,:); %Centroid of the Trochlear Groove (was
    Capitellum for Karol's)
217 ww(3,:) = ww(1,:); %If the Center of the Capitellum groov is the origin
218 ww(4,:) = HumSm_CCS(3,:); %Center of the humeral Shaft
219 Tca_hum = xyz_CMM(1,ww); %X-axis is fixed, Transformation matrix [TCA]
    ACShumerus-->CCS
220 %1- change sign on x-axis
221 if RorL == 'R'
222 Tca_hum = xyz_CMM(0,ww); %X-axis is fixed, Transformation matrix [TCA]
    ACShumerus-->CCS
223 end
224 Tac_hum = inv(Tca_hum); %
225
226
227 % 2. Get Tcg (humerus) (also copied from Karol's code). ASSUME use all 4
228 % balls here.
229 ai=HumSm_CCS(5:8,:);
230
231 % Aha! We want to go from the GLOBAL TRIAL CS to the CMM CS. (NOT the
    static CS!)
232 pi=[Trajectories(1, MarkerCols_Hsmall(1,:));
233 Trajectories(1, MarkerCols_Hsmall(2,:));
234 Trajectories(1, MarkerCols_Hsmall(3,:));
235 Trajectories(1, MarkerCols_Hsmall(4,:))];
236
237 [Tgc_humerus,rms] = Soder(ai,pi);
238 rms_humeral = rms;
239 % figure; plot(rms); title('RMS of Humerus') %% Uncomment this line to
    get
240 % a plot of the humeral RMS. It will just be one point right now.)
241 Tcg_hum=inv(Tgc_humerus); %
242
243 else % if we have the native head case. %%%%%%%%NATIVE HEAD%%%%%%%%%
244 % do calculations just like above, but use big balls.
245 % 1. get Tac(humerus).
246
247 % first, make coordinate system in CMM coordinate system.
```

```
% This is copy/paste from Karol's program.
ww(1,:) = HumBig_CCS(9,:); %Center of the Capitellum (was Trochlear Groove for Karol's)
ww(2,:) = HumBig_CCS(4,:); %Centroid of the Trochlear Groove (was Capitellum for Karol's)
ww(3,:) = ww(1,:); %If the Center of the Capitellum is the origin
ww(4,:) = HumBig_CCS(3,:); %Center of the humeral Shaft
Tca_hum = xyz_CMM(1,ww); %X-axis is fixed, Transformation matrix [TCA] ACShumerus-->CCS
%1- change sign on x-axis
if RorL == 'R'
Tca_hum = xyz_CMM(0,ww); %X-axis is fixed, Transformation matrix [TCA] ACShumerus-->CCS
end
Tac_hum = inv(Tca_hum); %

% 2. Get Tcg (humerus) (also copied from Karol's code). ASSUME use all 4
% balls here.
% Like Karol: Average first 10 rows of humerus markers.

MarkerCols_Hbig(1,:);
ai=HumBig_CCS(5:8,:);

% Aha! We want to go from the GLOBAL TRIAL CS to the CMM CS. (NOT the static CS!)
pi=[Trajectories(1, MarkerCols_Hbig(1,:));
Trajectories(1, MarkerCols_Hbig(2,:));
Trajectories(1, MarkerCols_Hbig(3,:));
Trajectories(1, MarkerCols_Hbig(4,:))];

[Tgc_humerus,rms] = Soder(ai,pi);
rms_humeral = rms;
% figure; plot(rms)
Tcg_hum=inv(Tgc_humerus); %

% Now, need Tgt_gst
% need to calculate:
% Radial_Markers_GCS_static;
% Radius_GCS_trial
% need to do this for each instant in time.

% calculate Radial_Markers_GCS_static here.!!!!!!!
MarkerCols_Radius_static=MarkerCols_Radius; % might need to change this someday.
Radius_Markers_GCS_temp = Trajectories_static;
Radius_Markers_GCS_static = Radius_Markers_GCS_temp;
[N_of_frames,n] = size(Radius_Markers_GCS_static);
if N_of_frames ~= 1 %Because the function "sum" sums rows instead of columns
%if only 1 row is present
Radius_Markers_GCS_static = sum(Radius_Markers_GCS_static(1:10,:))./10; % could increase this to be >10 if needed.
% this is actuall ALL the markers, not just the radius. "Radius" just
% happens to be in its name.
end


```

```
297 temp_cols=MarkerCols_Radius'; % do this so that when we flatten, it works
    out right.
298 trial_LK=[]; static_LK=[]; % just so it doesn't balk on the clear below.
299 for kk=1:nrows % for each timepoint
300
301 trial_LK{kk}=[Trajectories(kk,MarkerCols_Radius(1,:));
302 Trajectories(kk,MarkerCols_Radius(2,:));
303 Trajectories(kk,MarkerCols_Radius(3,:));
304 Trajectories(kk,MarkerCols_Radius(4,:))];
305
306 static_LK=[Radius_Markers_GCS_static(1,MarkerCols_Radius(1,:)); % always
    use the first row, since the static values only have one row of data.
307 Radius_Markers_GCS_static(1,MarkerCols_Radius(2,:));
308 Radius_Markers_GCS_static(1,MarkerCols_Radius(3,:));
309 Radius_Markers_GCS_static(1,MarkerCols_Radius(4,:))];
310
311 size(Radius_Markers_GCS_static);
312 [Tgt_gst{kk},rms(kk)] = Soder(static_LK,trial_LK{kk});
313 end % end kk for.
314
315
316 ai_LK_Tga=[Radius_Markers_GCS_static(1,MarkerCols_RadHd(1,:)); % always
    use the first row, since the static values only have one row of data.
317 Radius_Markers_GCS_static(1,MarkerCols_RadHd(2,:)); % aha! need to use
    radial head here. aaah, confusing! must think about this.
318 Radius_Markers_GCS_static(1,MarkerCols_RadHd(3,:));
319 Radius_Markers_GCS_static(1,MarkerCols_RadHd(4,:))];
320
321 % To find the transformation from the CMM to
322 % the global for the head directly.
323 Head_CMM=HorI_CCS(2:5,:);
324 size(Head_CMM);
325 size(ai_LK_Tga);
326 Head_CMM;
327 ai_LK_Tga;
328 [Tgc_head, rms]=Soder(Head_CMM, ai_LK_Tga);
329 end % end giant IF.
330
331
332 % 3. get R_feang. needed for both
333 % easy!
334 pi=3.14159265358979; % THank yoU Karol.
335
336 if RorL=='R'
337 fe_ang_rad=fe_ang*pi/180;
338 else
339 fe_ang_rad=-fe_ang*pi/180; % do a negative rotation for left elbows
    because their y-axis points the other way.
340 end
341
342 R_feang=[1 0 0 0;
343 0 1 0 0;
344 0 0 cos(fe_ang_rad-pi/2) (-sin(fe_ang_rad-pi/2));
345 0 0 sin(fe_ang_rad-pi/2) cos(fe_ang_rad-pi/2)];
346
347 % 4. Do the rotation!!!
348 if HeadorImp=='I'
```

```
for kk=1:nrows

% want to use Coord_transfer.m
clear temp;
clear temp1;
clear temp2;
temp=Coord_transfer(center_coordinates(kk,:), Tcg_hum);
temp1=Coord_transfer(temp, Tac_hum);
temp2=Coord_transfer(temp1, R_feang);

RH_center_ACS(kk,1:3)=temp2; %
% Try computing this way too - Gives identical results (that is good), so will just
% comment it out.
% temp9=(R_feang*Tac_hum*Tcg_hum*[1 center_coordinates(kk,:)]')';
% RH_center_ACS2(kk, 1:3)=temp9(2:4);
end % end kk for.
Tcg_hum
Tac_hum
R_feang

else % if we have the native head
% put what to do for native head here.
% It will be:
% R_feang*Tac_hum*Tcg_hum*Tgg_static*Tga_head_static*Tac_head*x_head_CMM
% remember to use Coord_transfer!
for mm=1:nrows
% want to use Coord_transfer.m
clear temp;
clear temp1;
clear temp2;
clear temp3;
clear temp4;
clear temp5;
x_head_CMM=[HorI_CCS(1,:)];
temp=Coord_transfer(x_head_CMM, Tgc_head); % this transfers from the CMM to the global static frame.
temp2=Coord_transfer(temp, Tgt_gst{mm}); % from the global static to teh global trial frame
temp3=Coord_transfer(temp2, Tcg_hum); % from the global trial to the CMM_humerus frame
temp4=Coord_transfer(temp3, Tac_hum); % Tac*Tcg=Tag (to line up with notes), from the humerus CMM frame to the anatomic humerus frame.
temp5=Coord_transfer(temp4, R_feang);
size(temp5);
RH_center_ACS(mm,:)=temp5; %
end % end mm for.
% print out matrices (this is for debugging, uncomment if desired)
% Tgc_head
% Tgt_gst{mm}
% Tcg_hum
% Tac_hum
% R_feang
end % end HeadorImp if.

% plotting should be easy. Want to plot:
% Figure 1: x, y, z travel
```

```
401 figure;
402 subplot(3,1,1)
403 plot(RH_center_ACS(:,1));
404 hold on;
405 ylabel('x travel vs. frames');
406
407 subplot(3,1,2)
408 plot(RH_center_ACS(:,2));
409 ylabel('y travel vs. frames');
410
411 subplot(3,1,3)
412 plot(RH_center_ACS(:,3));
413 ylabel('z travel vs. frames');
414
415 % Figure 2: plot on capitellum.
416 figure
417 % Figure #4
418 %Draw a circle representing the Capitellum
419 center=[0,0]; %Origin of the capitellum
420 NOP = 50; %Number of points on the circle
421 pi = 3.14159265358979;
422 THETA=linspace(0,2*pi,NOP);
423 RHO=ones(1,NOP)*Radius_of_Capitellum;
424 [X,Y] = pol2cart(THETA,RHO);
425 X=X+center(1);
426 Y=Y+center(2);
427 H=plot(X,Y,'k-','LineWidth',3);
428 % axis square;
429 axis equal
430 hold on
431 % When looking form the front- anterior view:
432 xlabel('Medial [mm] Lateral ')
433 if RorL == 'R'
434 xlabel(' Lateral [mm] Medial ')
435 % Head_coor_in_Humero_Radial_ACS(:,1) = -
    Head_coor_in_Humero_Radial_ACS(:,1);
436 end
437 title('')
438 % For the medio-lateral motion the Head_coor_in_Humero_Radial_ACS(:,1)
439 % doesn't have to be inverted if the arm is right because the x axis
    always
440 % points left no matter whether the arm is left or right.
441
442 ylabel('Posterior [mm] Anterior')
443 title('Radial head travel on the capitellum')
444 plot(RH_center_ACS(:,1),RH_center_ACS(:,3),'b.','LineWidth',2)
445 set(gca,'Xdir','reverse')
446 hold off
447
448
449 % now do the p/s reading.
450 ps_or_no=input('Do you want to plot against p/s angle? (y/n) ','s'); %
    offer the option of not plotting/exporting
451
452 if ps_or_no=='y'
```

```
453 psfilename=input('p/s filename, including extension (.csv): ','s'); % a
    file that includes the pronation/supination potentiometer data. example:
    ps_te
stfile.csv
454 ps_samp=input('p/s sampling frequency (Hz): '); % number, in Hz. example:
    60
455 radhd_sync_column=input('radial head sync signal is in column: '); %
    number of the column in the Trajectories (dynamic motion) file that has
    the s
ync signal in it. example: 2
456 motion_samp=input('motion analysis sampling frequency (Hz): '); %
    example: 30
457 output_filename=input('output file name will be (include extension): ',
    's'); % output filename. I like to make it an .xls file.
458 degree_increment=input('degree increment for output file?'); % example: 2
459
460 % run ps_pot_sync, which will write the output file and make the plots.
461 [radhd_coords_final, radhd_sync_final, ps_final]=ps_pot_sync(psfilename,
    ps_samp, RH_center_ACS, Trajectories, radhd_sync_column, motion_s
amp, degree_increment, output_filename);
462
463 else
464 file_or_no=input('Do you want to just output the radial head coordinates?
    (y/n) ','s');
465 if file_or_no=='y'
466 filename=input('The filename should be: ','s');
467 Head_kinematics=RH_center_ACS;
468 % Laurel likes to use dlmwrite.
469 dlmwrite(filename, Head_kinematics,'\t')
470 end % end if file_or_no if.
471
472 end % end if.
473
474 % Below are the lines used to validate the code. %%%% THESE WERE DONE
475 % BEFORE ADDING THE PART TO READ IN THE POTENTIOMETER VALUES AND PLOT
476 % AGAINST P/S ANGLE. (Really, that shouldn't matter because the p/s stuff
477 % is put in as an option.)
478 % To really Validate:
479 % 1) Do native head first.
480 % a)static CS = Trial CS = CMM CS;
481 %LaurelsProgram('PointsForValidation.xls', 'CS0', 'N', 'R', 'H',
    'Validate_HumCS0.dat', 'Validate_HumCS0.dat', 'Validate_RadHCS0.dat', 90,
    'PointsFor
Validation.xls', 'CS0')
482 % b) CMM CS is different.
483 % LaurelsProgram('PointsForValidation.xls', 'CS1', 'N', 'R', 'H',
    'Validate_HumCS0.dat', 'Validate_HumCS0.dat', 'Validate_RadHCS0.dat', 90,
    'PointsFo
rValidation.xls', 'CS1')
484 % c) Static CS is different
485 % LaurelsProgram('PointsForValidation.xls', 'CS0', 'N', 'R', 'H',
    'Validate_HumCS0.dat', 'Validate_HumCS0.dat', 'Validate_RadHCS0.dat', 90,
    'PointsFo
rValidation.xls', 'CS1')
486 % d) trial CS is different
```

```
487 % LaurelsProgram('PointsForValidation.xls', 'CS0', 'N', 'R', 'H',
    'Validate_HumCS1.dat', 'Validate_HumCS1.dat', 'Validate_RadHCS1.dat', 90,
    'PointsFo
rValidation.xls', 'CS1')
488
489 % e) All CS are different.
490 % LaurelsProgram('PointsForValidation.xls', 'CS2', 'N', 'R', 'H',
    'Validate_HumCS1.dat', 'Validate_HumCS1.dat', 'Validate_RadHCS1.dat', 90,
    'PointsFo
rValidation.xls', 'CS0')
491 % f) BONUS: Three CS's ( HUM and RadH "CMM" files are different)
492 % LaurelsProgram('PointsForValidation.xls', 'CS2', 'N', 'R', 'H',
    'Validate_HumCS1.dat', 'Validate_HumCS1.dat', 'Validate_RadHCS0.dat', 90,
    'PointsFo
rValidation.xls', 'CS0')
493 %
494 % 2) Then repeat with implant. THIS IS TRICKY BECAUSE THE TRIAL DATA
    CANNOT BE CENTERED AT THE ORIGIN.
495 % (in practical terms, that means that we can't use 'CS0' as the
    data/dynamic file.
496 % a)static CS = Trial CS = CMM CS;
497 %LaurelsProgram('PointsForValidation.xls', 'CS1', 'N', 'R', 'I',
    'Validate_HumCS1.dat', 'Validate_HumCS1.dat', 'Validate_RadHCS1.dat', 90,
    'PointsForV
alidation.xls', 'CS1')
498 % b) CMM CS is different.
499 % LaurelsProgram('PointsForValidation.xls', 'CS1', 'N', 'R', 'I',
    'Validate_HumCS0.dat', 'Validate_HumCS0.dat', 'Validate_RadHCS0.dat', 90,
    'PointsFor
Validation.xls', 'CS1')
500 % c) Static CS is different
501 % LaurelsProgram('PointsForValidation.xls', 'CS1', 'N', 'R', 'I',
    'Validate_HumCS1.dat', 'Validate_HumCS1.dat', 'Validate_RadHCS1.dat', 90,
    'PointsFor
Validation.xls', 'CS0')
502 % d) trial CS is different
503 % LaurelsProgram('PointsForValidation.xls', 'CS1', 'N', 'R', 'I',
    'Validate_HumCS0.dat', 'Validate_HumCS0.dat', 'Validate_RadHCS0.dat', 90,
    'PointsFor
Validation.xls', 'CS0')
504 % e) All CS are different.
505 % LaurelsProgram('PointsForValidation.xls', 'CS2', 'N', 'R', 'I',
    'Validate_HumCS1.dat', 'Validate_HumCS1.dat', 'Validate_RadHCS1.dat', 90,
    'PointsFor
Validation.xls', 'CS0')
506 % f) BONUS: Three CS's ( HUM and RadH "CMM" files are different)
507 % LaurelsProgram('PointsForValidation.xls', 'CS2', 'N', 'R', 'I',
    'Validate_HumCS1.dat', 'Validate_HumCS1.dat', 'Validate_RadHCS0.dat', 90,
    'PointsFor
Validation.xls', 'CS0')
```

N.4 SODER.M

```
1 % soder.m: Matlab function to determine rigid body rotation & translation
2 % From:
3 % I. Soederqvist and P.A. Wedin (1993) Determining the movement of the
    skeleton
4 % using well-configured markers. J. Biomech. 26:1473-1477.
5 % Same algorithm is described in:
6 % J.H. Challis (1995) A prodecure for determining rigid body transformation
7 % parameters, J. Biomech. 28, 733-737.
8 % The latter also includes possibilities for scaling, reflection, and
9 % weighting of marker data.
10 %
11 % Written by Ron Jacobs (R.S. Dow Neurological Institute, Porland OR),
12 % adapted by Ton van den Bogert (University of Calgary).
13 % and by karol Galik in July 2003
14 %
15 % Input:
16 % x: 3-D marker coordinates in position 1 (3 columns, one row for each
    marker)
17 % y: 3-D marker coordinates in position 2 (same format)
18 %
19 % Output:
20 % R: rotation matrix
21 % d: translation vector
22 % rms: the root mean square fit error of the rigid body model
23 %
24 % the rigid body model is: y = R*x + d
25 %
26 function [T,rms]=soder(x,y)
27
28 [nmarkers,ndimensions]=size(x);
29 % we could give an error message if ndimensions is not 3
30
31 mx=mean(x);
32 my=mean(y);
33
34 % construct matrices A and B, subtract the mean so there is only rotation
35 for i=1:nmarkers,
36 A(i,:)=x(i,:)-mx;
37 B(i,:)=y(i,:)-my;
38 end
39 A = A';
40 B = B';
41
42
43 % The singular value decomposition to calculate R with det(R)=1
44 C=B*A';
45 [P,T,Q]=svd(C);
46 R=P*diag([1 1 det(P*Q')])*Q';
47
48 % Calculate the translation vector from the centroid of all markers
49 d=my'-R*mx';
50
51 % calculate RMS value of residuals
```

```
52 sumsq = 0;
53 for i=1:nmarkers
54 ypred = R*x(i,:)' + d;
55 sumsq = sumsq + norm(ypred-y(i,:)')^2;
56 end
57 rms = sqrt(sumsq/3/nmarkers);
58
59 %Transformation matrix (the only change made by KG)
60 T=[1 0 0 0;
61 d(1) R(1,1) R(1,2) R(1,3);
62 d(2) R(2,1) R(2,2) R(2,3);
63 d(3) R(3,1) R(3,2) R(3,3)];
```

N.5 XYZ.M

```
1 function B = xyz(p)
2 % This subroutine creates a coordinate system (characterized
3 % by B) from the coordinates of 4 markers.
4 % Z- axis is fixed (Used for Radius)
5 % p(1,i): coordinates of the origin
6 % p(2,i): coordinates of a point on the z-axis with z>0
7 % p(3,i) & p(4,i):
8
9 z = p(2,:)-p(1,:);
10 w = p(4,:)-p(3,:);
11
12 y = cross(z,w);
13 x = cross(y,z);
14
15 % Normalize the x, y, and z vectors into unit vectors
16 unit_x = x/sqrt(x(1)^2+x(2)^2+x(3)^2);
17 unit_y = y/sqrt(y(1)^2+y(2)^2+y(3)^2);
18 unit_z = z/sqrt(z(1)^2+z(2)^2+z(3)^2);
19
20 % Coordinate system established from three noncolinear points
21 % The first point in the p(i,j) matrix is the origin of the coordinate
   system
22
23 B=[1 0 0 0;
24 p(1,1) unit_x(1) unit_y(1) unit_z(1);
25 p(1,2) unit_x(2) unit_y(2) unit_z(2);
26 p(1,3) unit_x(3) unit_y(3) unit_z(3)];
```

N.6 XYZ_CMM.M

```
1 function B = xyz_CMM(switchX,p)
2 % This subroutine creates a transformation matrix B from coordinates of 3
   points.
```

```
3 % Unlike the xyz.m the sequence here is really X-Y-Z
4 % X is fixed (Flexion-Extension axis)
5 % Used in: Humero Ulnar joint for humerus
6 % : Humero Radial joint for humerus
7 % -p(1,i): coordinates of the origin
8 % -p(2,i): coordinates of a point on the X-axis with x>0
9 %
10 % --SwitchX: Change sign on X depending on right or Left elbow
11 % -HumeroRadial Joint (capitulum origin):
12 % For left elbow, the x-axis is positive to the medial side (from
    capitulum
13 % to the trochlear groove). For right elbow, the x-axis stays in the same
    direction
14 % but this time pointing to the lataral side
15 % -HumeroUlanr Joint trochlear groove origin):
16 % For left elbow, the x-axis is positive to the lateral side. This means
    that the
17 % y axis points backwards, away from the ulna
18 %
19 %
20
21 x = p(2,:)-p(1,:);
22 if switchX == 1
23 x = -x;
24 end
25 w = p(4,:)-p(3,:);
26
27 y = cross(w,x);
28 z = cross(x,y);
29
30 % Normalize the x, y, and z vectors into unit vectors
31 unit_x = x/sqrt(x(1)^2+x(2)^2+x(3)^2); %or x/norm(x)
32 unit_y = y/sqrt(y(1)^2+y(2)^2+y(3)^2);
33 unit_z = z/sqrt(z(1)^2+z(2)^2+z(3)^2);
34
35 % Coordinate system established from three noncolinear points
36 % The first point in the p(i,j) matrix is the origin of the coordinate
    system
37
38 B=[1 0 0 0;
39 p(1,1) unit_x(1) unit_y(1) unit_z(1);
40 p(1,2) unit_x(2) unit_y(2) unit_z(2);
41 p(1,3) unit_x(3) unit_y(3) unit_z(3)];
```

APPENDIX O

FINITE HELICAL AXIS CODE DIAGRAM

This appendix shows a multipage diagram to illustrate the flow of the MATLAB code used to compute the finite helical axis.

START HERE

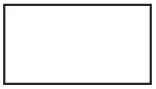
- Establish which rows are which markers;
- Read in information (balls file, CMM files, etc.) from List_of_Experiments_LK
- Read in Humerus kinematics file (can be different from others)
- Write an output file with some information about the CMM coordinates.
- Convert coordinates to mm (CMM does it in inches)

GLOBAL VARIABLES:

balls
cut_off_frequency
File_Name
Humeral_Markers_GCS_static
Humero_Radial_coor_ACS
Kinem_File_dr
Kinem_file_hum
Kinem_File_pu
Kinem_File_rh_i
Kinem_file_u
Left_or_Right
Markers_ACS_Humero_Ulnar
Markers_ACS_Ulna
PronSup
PronSup_static
PronSup_static_RH
Reconstructed_Radius_GCS
rms_humeral
sheet
Tac_hum
Tag_humero_radial
Tag_humero_ulnar
Tgc_Distal_Radius
Tgc_Rad_Head_or_Implant
Ulnar_coor_GCS

LEGEND

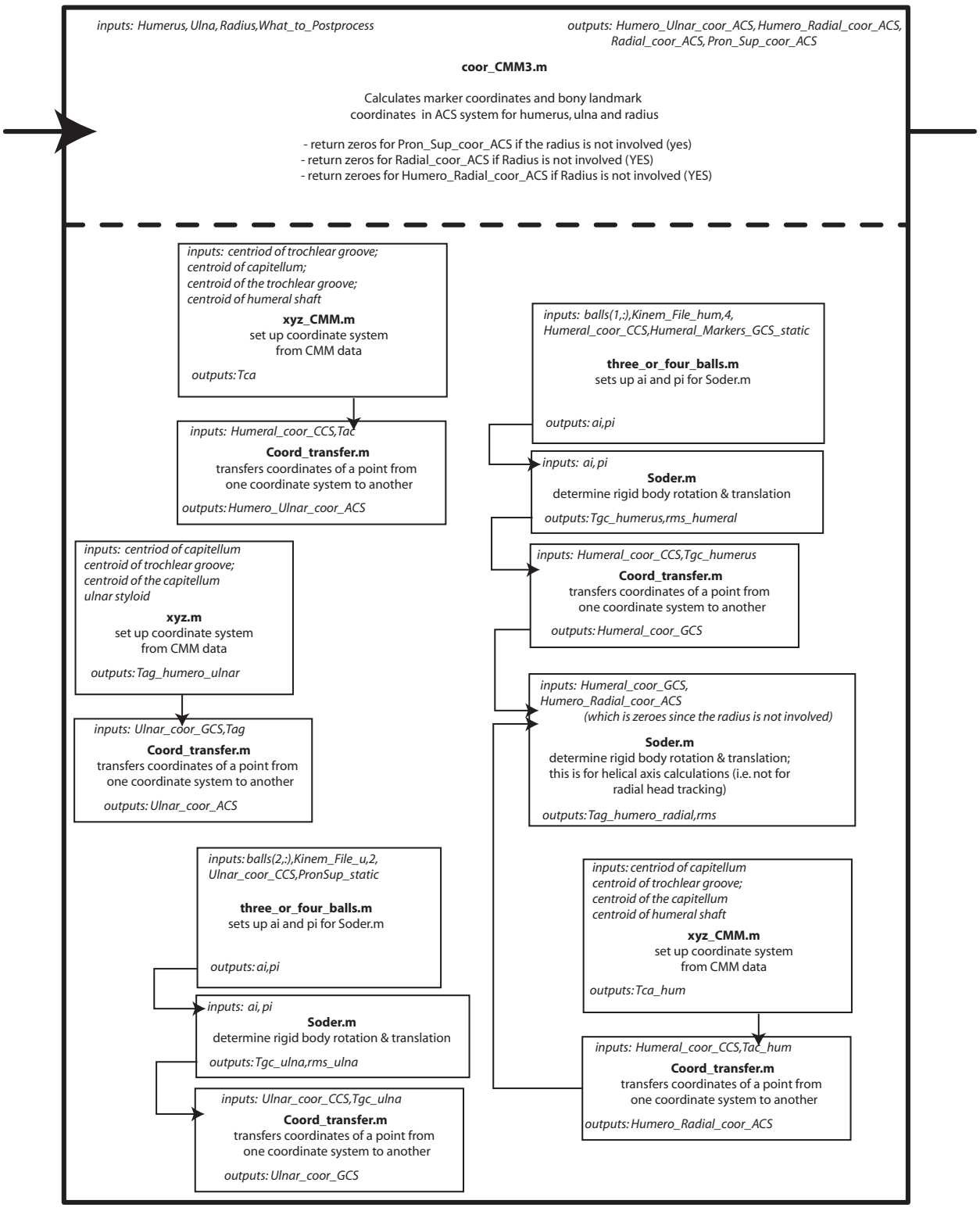
main functions

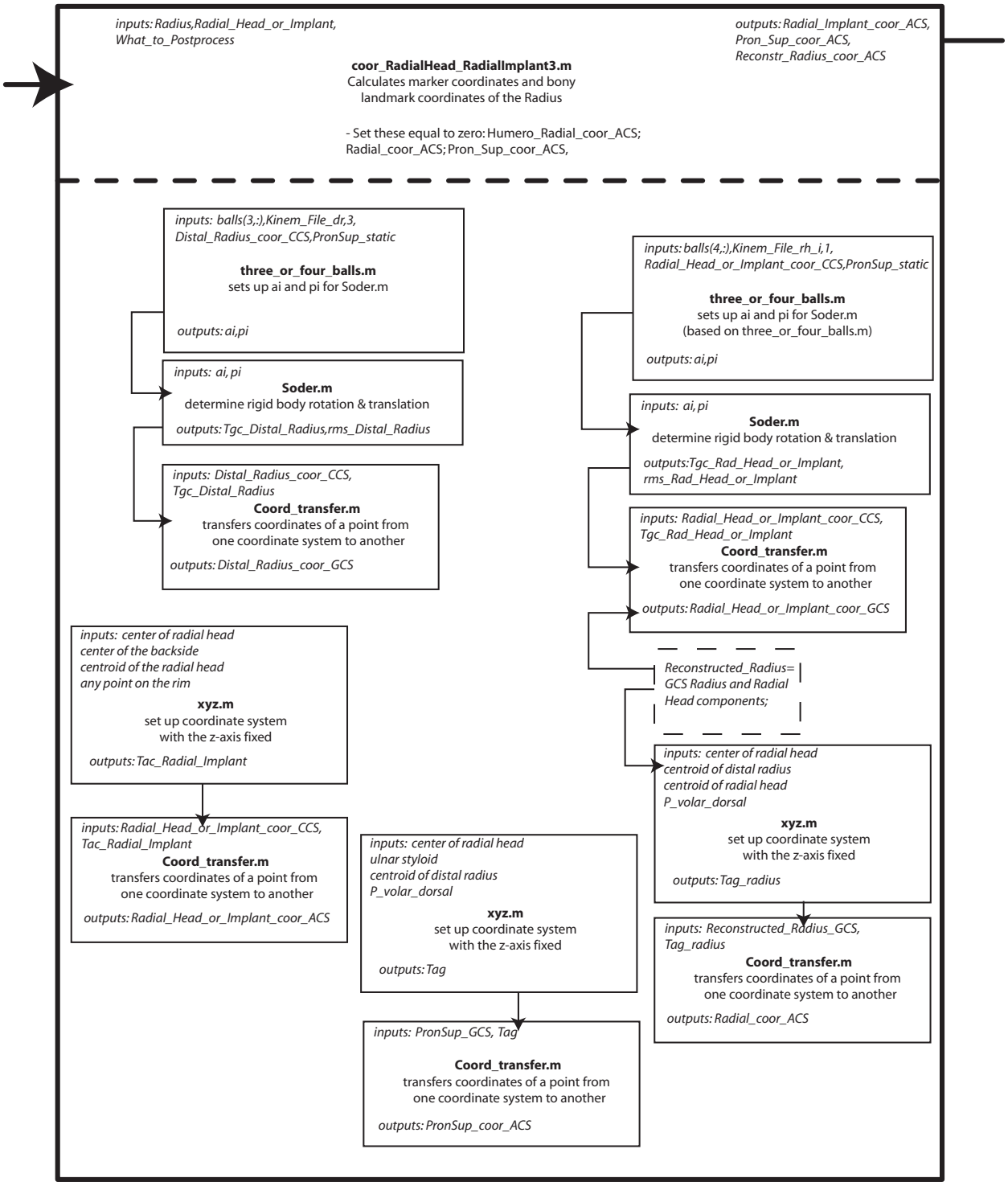


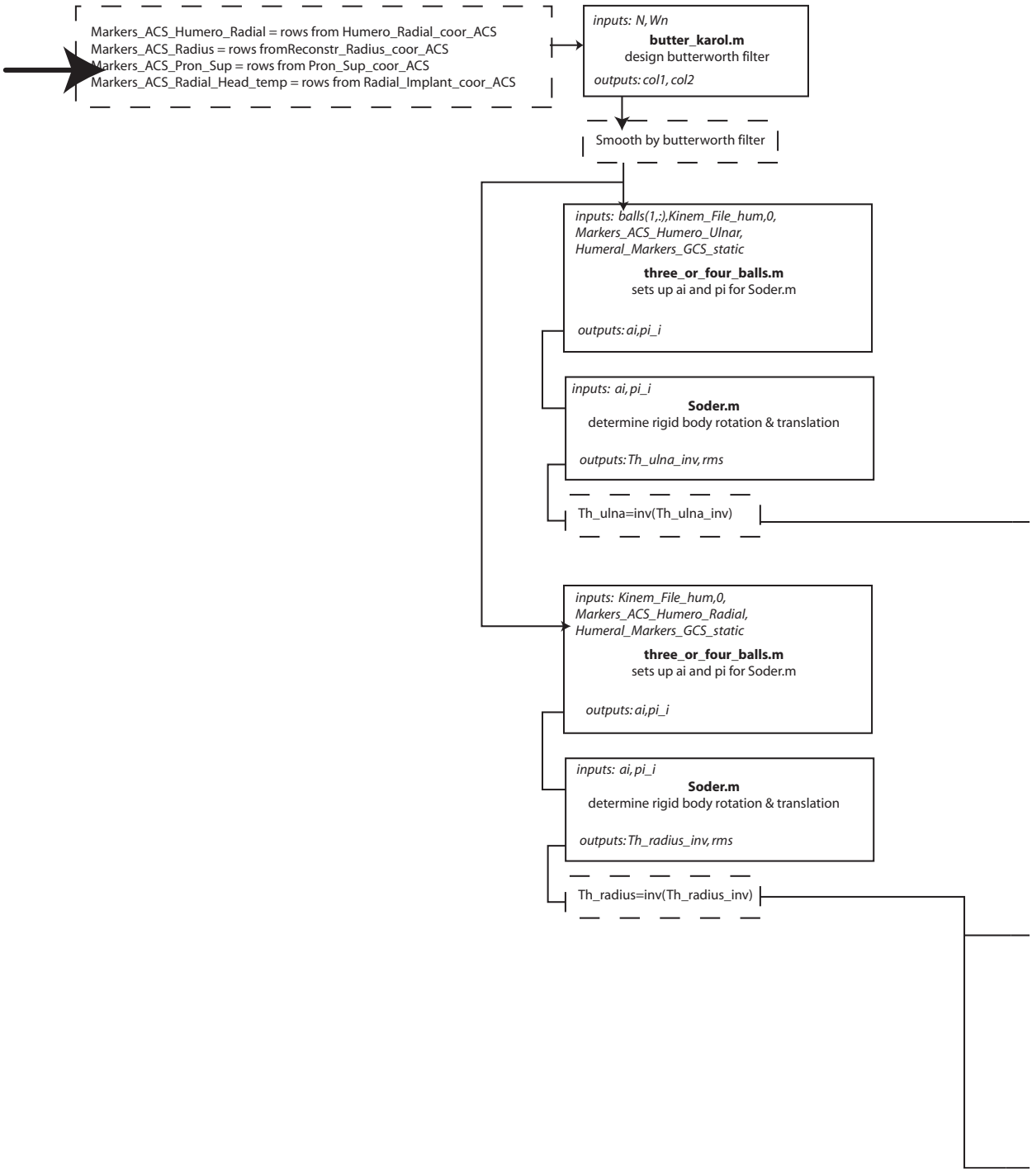
subfunction called (as part of main function)

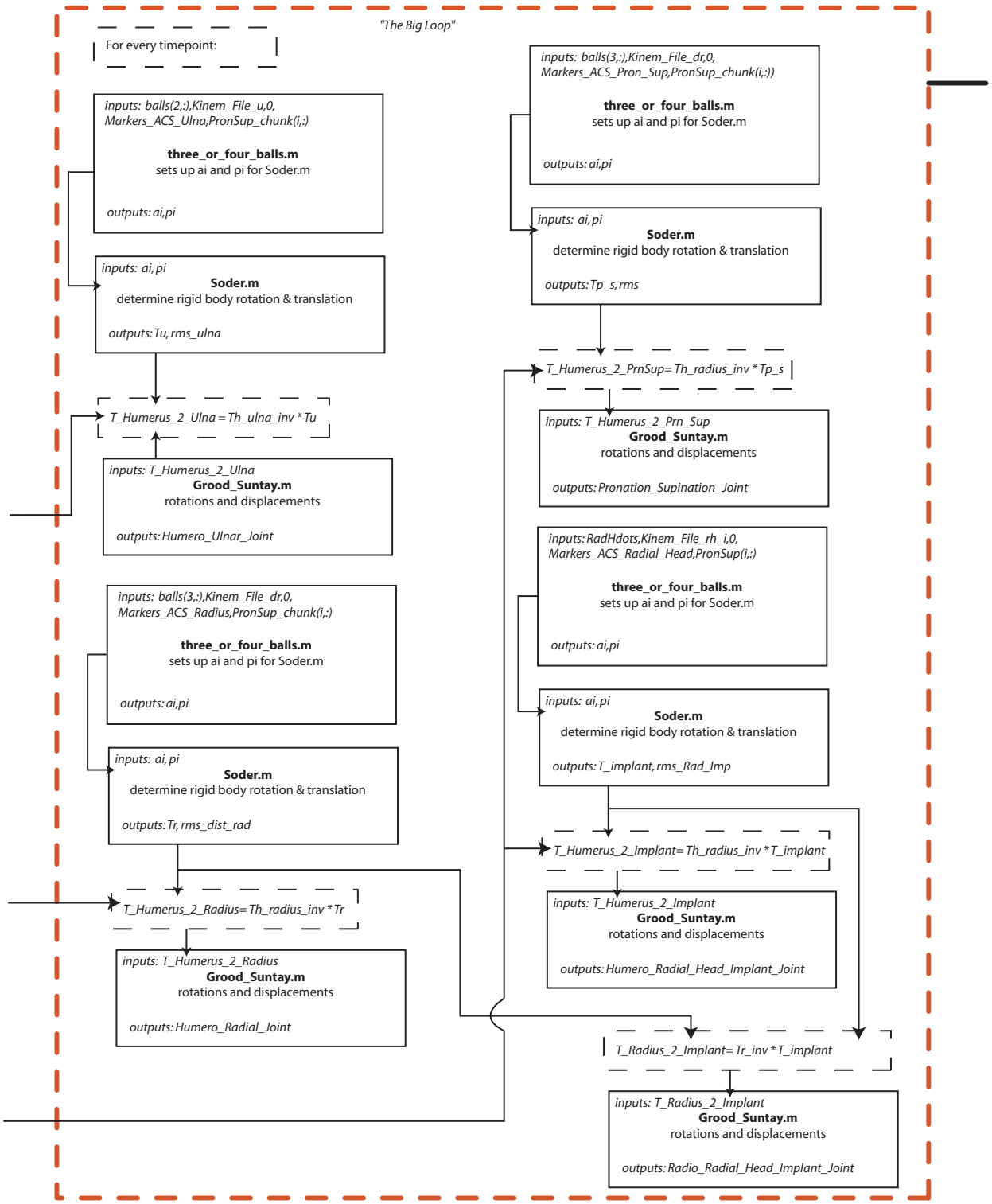
calculations made

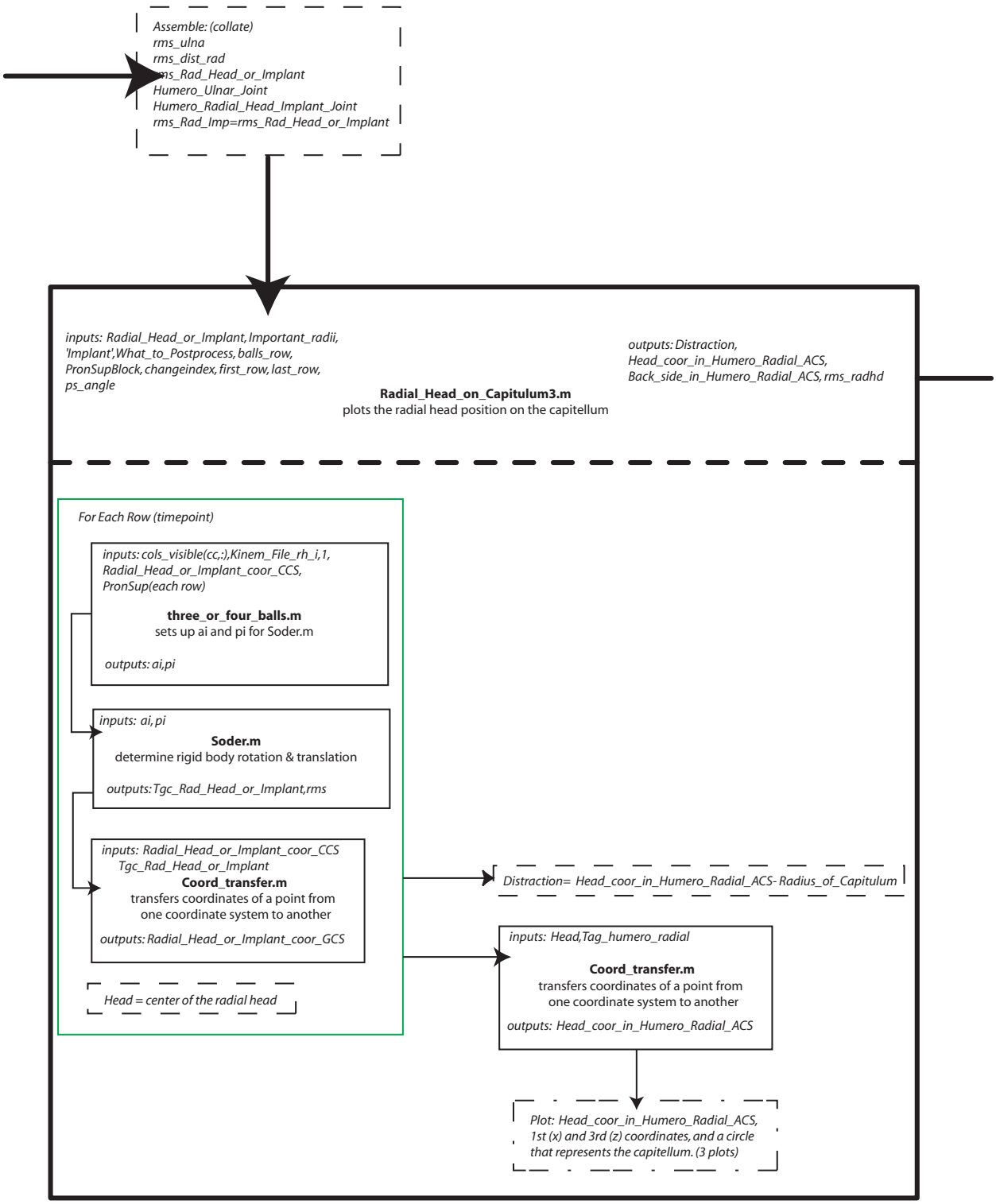
outcome


inputs: Humerus, Ulna, Radius,What_to_Postprocess
outputs: Humero_Ulnar_coor_ACS, Humero_Radial_coor_ACS, Radial_coor_ACS, Pron_Sup_coor_ACS
coor_CMM3.m
Calculates marker coordinates and bony landmark coordinates in ACS system for humerus, ulna and radius
- return zeros for Pron_Sup_coor_ACS if the radius is not involved (yes)
- return zeros for Radial_coor_ACS if Radius is not involved (YES)
- return zeroes for Humero_Radial_coor_ACS if Radius is not involved (YES)
inputs: centriod of trochlear groove; centroid of capitellum; centroid of the trochlear groove; centroid of humeral shaft
xyz_CMM.m
set up coordinate system from CMM data
outputs: Tca
inputs: Humeral_coor_CCS,Tac
Coord_transfer.m
transfers coordinates of a point from one coordinate system to another
outputs: Humero_Ulnar_coor_ACS
inputs: centriod of capitellum centroid of trochlear groove; centroid of the capitellum ulnar styloid
xyz.m
set up coordinate system from CMM data
outputs: Tag_humero_ulnar
inputs: Ulnar_coor_GCS,Tag
Coord_transfer.m
transfers coordinates of a point from one coordinate system to another
outputs: Ulnar_coor_ACS
inputs: balls(2,:),Kinem_File_u,2, Ulnar_coor_CCS,PronSup_static
three_or_four_balls.m
sets up ai and pi for Soder.m
outputs: ai,pi
inputs: ai, pi
Soder.m
determine rigid body rotation & translation
outputs: Tgc_ulna,rms_ulna
inputs: Ulnar_coor_CCS,Tgc_ulna
Coord_transfer.m
transfers coordinates of a point from one coordinate system to another
outputs: Ulnar_coor_GCS
inputs: balls(1,:),Kinem_File_hum,4, Humeral_coor_CCS,Humeral_Markers_GCS_static
three_or_four_balls.m
sets up ai and pi for Soder.m
outputs: ai,pi
inputs: ai, pi
Soder.m
determine rigid body rotation & translation
outputs: Tgc_humerus,rms_humeral
inputs: Humeral_coor_CCS,Tgc_humerus
Coord_transfer.m
transfers coordinates of a point from one coordinate system to another
outputs: Humeral_coor_GCS
inputs: Humeral_coor_GCS, Humero_Radial_coor_ACS
(which is zeroes since the radius is not involved)
Soder.m
determine rigid body rotation & translation; this is for helical axis calculations (i.e. not for radial head tracking)
outputs: Tag_humero_radial,rms
inputs: centriod of capitellum centroid of trochlear groove; centroid of the capitellum centroid of humeral shaft
xyz_CMM.m
set up coordinate system from CMM data
outputs: Tca_hum
inputs: Humeral_coor_CCS,Tac_hum
Coord_transfer.m
transfers coordinates of a point from one coordinate system to another
outputs: Humero_Radial_coor_ACS


inputs: Radius,Radial_Head_or_Implant,
What_to_Postprocess
outputs: Radial_Implant_coor_ACS,
Pron_Sup_coor_ACS,
Reconstr_Radius_coor_ACS
coor_RadialHead_RadialImplant3.m
Calculates marker coordinates and bony
landmark coordinates of the Radius
- Set these equal to zero: Humero_Radial_coor_ACS;
Radial_coor_ACS; Pron_Sup_coor_ACS,
inputs: balls(3,:),Kinem_File_dr,3,
Distal_Radius_coor_CCS,PronSup_static
three_or_four_balls.m
sets up ai and pi for Soder.m
outputs: ai,pi
inputs: ai, pi
Soder.m
determine rigid body rotation & translation
outputs: Tgc_Distal_Radius,rms_Distal_Radius
inputs: Distal_Radius_coor_CCS,
Tgc_Distal_Radius
Coord_transfer.m
transfers coordinates of a point from
one coordinate system to another
outputs: Distal_Radius_coor_GCS
inputs: balls(4,:),Kinem_File_rh_i,1,
Radial_Head_or_Implant_coor_CCS,PronSup_static
three_or_four_balls.m
sets up ai and pi for Soder.m
(based on three_or_four_balls.m)
outputs: ai,pi
inputs: ai, pi
Soder.m
determine rigid body rotation & translation
outputs:Tgc_Rad_Head_or_Implant,
rms_Rad_Head_or_Implant
inputs: Radial_Head_or_Implant_coor_CCS,
Tgc_Rad_Head_or_Implant
Coord_transfer.m
transfers coordinates of a point from
one coordinate system to another
outputs: Radial_Head_or_Implant_coor_GCS
Reconstructed_Radius=
GCS Radius and Radial
Head components;
inputs: center of radial head
center of the backside
centroid of the radial head
any point on the rim
xyz.m
set up coordinate system
with the z-axis fixed
outputs: Tac_Radial_Implant
inputs: Radial_Head_or_Implant_coor_CCS,
Tac_Radial_Implant
Coord_transfer.m
transfers coordinates of a point from
one coordinate system to another
outputs: Radial_Head_or_Implant_coor_ACS
inputs: center of radial head
ulnar styloid
centroid of distal radius
P_volar_dorsal
xyz.m
set up coordinate system
with the z-axis fixed
outputs: Tag
inputs: PronSup_GCS, Tag
Coord_transfer.m
transfers coordinates of a point from
one coordinate system to another
outputs: PronSup_coor_ACS
inputs: center of radial head
centroid of distal radius
centroid of radial head
P_volar_dorsal
xyz.m
set up coordinate system
with the z-axis fixed
outputs: Tag_radius
inputs: Reconstructed_Radius_GCS,
Tag_radius
Coord_transfer.m
transfers coordinates of a point from
one coordinate system to another
outputs: Radial_coor_ACS


Markers_ACS_Humero_Radial = rows from Humero_Radial_coor_ACS
Markers_ACS_Radius = rows fromReconstr_Radius_coor_ACS
Markers_ACS_Pron_Sup = rows from Pron_Sup_coor_ACS
Markers_ACS_Radial_Head_temp = rows from Radial_Implant_coor_ACS
inputs: N, Wn
butter_karol.m
design butterworth filter
outputs: col1, col2
Smooth by butterworth filter
inputs: balls(1,:),Kinem_File_hum,0,
Markers_ACS_Humero_Ulnar,
Humeral_Markers_GCS_static
three_or_four_balls.m
sets up ai and pi for Soder.m
outputs: ai,pi_i
inputs: ai, pi_i
Soder.m
determine rigid body rotation & translation
outputs: Th_ulna_inv, rms
Th_ulna=inv(Th_ulna_inv)
inputs: Kinem_File_hum,0,
Markers_ACS_Humero_Radial,
Humeral_Markers_GCS_static
three_or_four_balls.m
sets up ai and pi for Soder.m
outputs: ai,pi_i
inputs: ai, pi_i
Soder.m
determine rigid body rotation & translation
outputs: Th_radius_inv, rms
Th_radius=inv(Th_radius_inv)


"The Big Loop"
For every timepoint:
inputs: balls(2,:),Kinem_File_u,0,
Markers_ACS_Ulna,PronSup_chunk(i,:)
three_or_four_balls.m
sets up ai and pi for Soder.m
outputs: ai,pi
inputs: ai, pi
Soder.m
determine rigid body rotation & translation
outputs: Tu, rms_ulna
T_Humerus_2_Ulna = Th_ulna_inv * Tu
inputs: T_Humerus_2_Ulna
Grood_Suntay.m
rotations and displacements
outputs: Humero_Ulnar_Joint
inputs: balls(3,:),Kinem_File_dr,0,
Markers_ACS_Radius,PronSup_chunk(i,:)
three_or_four_balls.m
sets up ai and pi for Soder.m
outputs: ai,pi
inputs: ai, pi
Soder.m
determine rigid body rotation & translation
outputs: Tr, rms_dist_rad
T_Humerus_2_Radius= Th_radius_inv * Tr
inputs: T_Humerus_2_Radius
Grood_Suntay.m
rotations and displacements
outputs: Humero_Radial_Joint
inputs: balls(3,:),Kinem_File_dr,0,
Markers_ACS_Pron_Sup,PronSup_chunk(i,:))
three_or_four_balls.m
sets up ai and pi for Soder.m
outputs: ai,pi
inputs: ai, pi
Soder.m
determine rigid body rotation & translation
outputs: Tp_s, rms
T_Humerus_2_PrnSup= Th_radius_inv * Tp_s
inputs: T_Humerus_2_Prn_Sup
Grood_Suntay.m
rotations and displacements
outputs: Pronation_Supination_Joint
inputs: RadHdots,Kinem_File_rh_i,0,
Markers_ACS_Radial_Head,PronSup(i,:)
three_or_four_balls.m
sets up ai and pi for Soder.m
outputs: ai,pi
inputs: ai, pi
Soder.m
determine rigid body rotation & translation
outputs: T_implant, rms_Rad_Imp
T_Humerus_2_Implant= Th_radius_inv * T_implant
inputs: T_Humerus_2_Implant
Grood_Suntay.m
rotations and displacements
outputs: Humero_Radial_Head_Implant_Joint
T_Radius_2_Implant= Tr_inv * T_implant
inputs: T_Radius_2_Implant
Grood_Suntay.m
rotations and displacements
outputs: Radio_Radial_Head_Implant_Joint


Assemble: (collate)
rms_ulna
rms_dist_rad
rms_Rad_Head_or_Implant
Humero_Ulnar_Joint
Humero_Radial_Head_Implant_Joint
rms_Rad_Imp=rms_Rad_Head_or_Implant
inputs: Radial_Head_or_Implant, Important_radii, 'Implant',What_to_Postprocess, balls_row, PronSupBlock, changeindex, first_row, last_row, ps_angle
Radial_Head_on_Capitulum3.m
plots the radial head position on the capitellum
outputs: Distraction, Head_coor_in_Humero_Radial_ACS, Back_side_in_Humero_Radial_ACS, rms_radhd
For Each Row (timepoint)
inputs: cols_visible(cc,:),Kinem_File_rh_i,1, Radial_Head_or_Implant_coor_CCS, PronSup(each row)
three_or_four_balls.m
sets up ai and pi for Soder.m
outputs: ai,pi
inputs: ai, pi
Soder.m
determine rigid body rotation & translation
outputs:Tgc_Rad_Head_or_Implant,rms
inputs: Radial_Head_or_Implant_coor_CCS Tgc_Rad_Head_or_Implant
Coord_transfer.m
transfers coordinates of a point from one coordinate system to another
outputs: Radial_Head_or_Implant_coor_GCS
Head = center of the radial head
Distraction= Head_coor_in_Humero_Radial_ACS- Radius_of_Capitulum
inputs: Head,Tag_humero_radial
Coord_transfer.m
transfers coordinates of a point from one coordinate system to another
outputs: Head_coor_in_Humero_Radial_ACS
Plot: Head_coor_in_Humero_Radial_ACS, 1st (x) and 3rd (z) coordinates, and a circle that represents the capitellum. (3 plots)

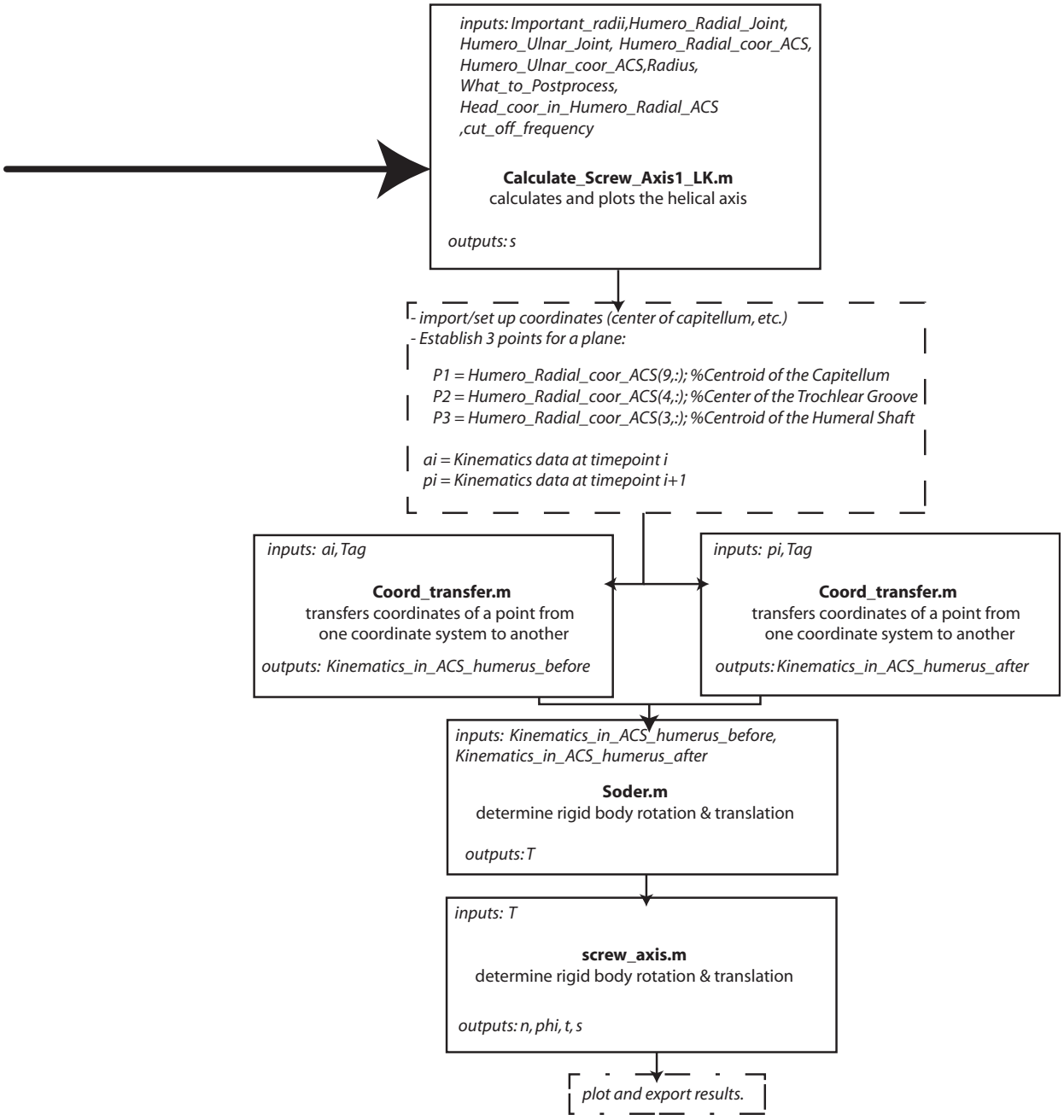


Figure 75: FHA MATLAB code schematic.

APPENDIX P

FINITE HELICAL AXIS TRACKING CODE

This Appendix presents the MATLAB code used for computing the finite helical axis, as described in Chapter 8.0 and shown schematically in the previous appendix. The functions are arranged in alphabetical order by title.

P.1 CALCULATE_SCREW_AXIS1_LK.M

```
1 function dummy_s =
    Calculate_Screw_Axis1_LK(Important_radii,Humero_Radial_Joint,Humero_Ulnar_
    Joint,...
2 Humero_Radial_coor_ACS,Humero_Ulnar_coor_ACS,Radial_coor_CCS,...
3 What_to_Postprocess,Head_coor_in_Humero_Radial_ACS,cut_off_frequency)
4 % (put it back to Karol's way to try Peak_CMM3_LK3_FHAtry2.mm)
5 % function dummy_s =
    Calculate_Screw_Axis1_LK(Important_radii,Humero_Radial_Joint,Humero_Ulnar_
    Joint,...
6 % Humero_Radial_coor_ACS,Humero_Ulnar_coor_ACS,Radial_coor_CCS,...
7 % What_to_Postprocess,cut_off_frequency)
8 % Laurel removed the input "Head_coor_in_Humero_Radial_ACS" since it is
9 % not needed. 7/16/07
10
11 % -This function will take Kinematic input file (PronSup) and calculate
12 % finite helical axis (FHA). Creates plots of Helical axis in 3D and
13 % projections of the mean helical axis at the styloid and radial head
14 % perpendicular to the mean helical axis.
15 % -You can pick whether the FHA should be calculated with marker array
16 % on the distal radius or the radial head. Only distal radius marker array
17 % should be used for implants.
18 %-Writes two files, one for each trial with "File_name"_FHA.FHA and
```

```
19 %one file where all data from one directory are written to file
20 %"Elbow_FHAs.FHA". This later file makes it easier to import all stuff
    into
21 %EXCEL at once.
22
23 % Inputs:- Tag_humero_radial- Transformation matrix from global to
    anatomical
24 % Humero_radial CS. Used for transfering all coordinates to humero-radial
    CS
25 % - Kinematics_file- Peaks camera input file
26 % - Humero_Radial_Joint- because of the stability problems at small angle
27 % increments (see e.g k.N. An, JK14) the (FHA) was calculated in 6
28 % (or other preset) degree of pronation supination increments.
29 % - Humero_Radial_coor_ACS- To draw the plane connected be the top
30 % of the humeral shaft, capitulum and trochlear groove
31 % -Head_coor_in_Humero_Radial_ACS to plot the travel of the RH along with
    the FHA
32 %
33 % Needed functions: "Screw_axis.m"
34 % "Soder.m
35 % "Plot_Intersection_Perpendicular_to_FHA.m"
36
37 global PronSup %Utilized in coor_CMM.m to average first 10 frames to be
38 %used for calculation of bony landmarks
39 global Tag_humero_radial %Comes from coor_CMM.m, used for Pronation-
    Supination
40 global Tag_humero_ulnar % used for calculating FHA in Flexion-Extension
41 global Ulnar_coor_GCS %To plot Ulnar styloid
42 global Reconstructed_Radius_GCS
43 global Ulnar_coor_GCS %To plot ulnar styloid on 3D FHA
44 global File_Name %Output File
45 global Left_or_Right
46 global sheet % Laurel added this one.
47
48 global Q %for "Plot_Intersection_Perpendicular_to_FHA.m"
49 global Important_radii %for "Plot_Intersection_Perpendicular_to_FHA.m"
50 % global Head_coor_in_Humero_Radial_ACS % LK commented out, 7/16/07
51
52 fid = fopen(strcat(File_Name, '.out'),'a'); %Open the file and append data
    % NOTE FROM LAUREL: Appending data could be dangerous.
53
54 Kinematics_File = PronSup;
55 radius_of_capitulum = Important_radii(1);
56
57 %-----------------------------------------------------------------
58 % Decision time
59 Angle_inc = 5; %Preset increment angle at which the FHA will be calculated
    % changed to 5 (from 10), LK, 6/26/06
60 plot_landmarks = 'Y'; %Plot the lines connecting the humeral and radial
    landmarks
61 plot_only_mean_FHA = 'N'; %Plot or not only the mean FHA (not all the
    FHA's)
62 DR_or_RH = 'DR'; %Calculations done using the RH marker array or the RH
    (implant) marker array
63 %'DR' or 'RH' - % NOTE FROM LAUREL - we always use DR
64 %since RH will be done separately in a new way from now
65 %on (with the itty-bitty "dot" markers.)
```

```
66
67 %-------------------------------------------------------------------------
   -
68 % Find 10 (or other preset) degree increment, remember indices of the
69 % Kinematics File at which the increment will occur
70
71 if What_to_Postprocess > 1
72 Joint = Humero_Radial_Joint;
73 col = 3; %3-Pronation Supination
74 title1 = 'Points for calculation of the FHA from Pronation-Supination';
75 Tag = Tag_humero_radial;
76 % You can pick whether the FHA should be calculated with marker array
77 % on the distal radius or the radial head
78 if DR_or_RH == 'DR';
79 % Kinem_File_ind = [17 18 19; 21 22 23; 25 26 27; 29 30 31]; %Distal
   Radius
80 Kinem_File_ind = [3 4 5; 6 7 8; 9 10 11; 12 13 14]; % for laurel
81 else
82 % Kinem_File_ind= [49 50 51; 53 54 55; 57 58 59; 61 62 63]; %Radial Head
   or Implant
83 Kinem_File_ind= [39 40 41; 42 43 44; 45 46 47; 48 49 50]; % for laurel;
84 end
85 %Below are 3 points to establish a XZ plane of intersection
86 P1 = Humero_Radial_coor_ACS(9,:); %Centroid of the Capitellum
87 P2 = Humero_Radial_coor_ACS(4,:); %Center of the Trochlear Groove
88 P3 = Humero_Radial_coor_ACS(3,:); %Centroid of the Humeral Shaft
89 % To check: Humero__Radial_coor_ACS(9,:)=[0,0,0] Origin of the Humeral
   ACS; Capitellum
90 % Humero__Radial_coor_ACS(4,:)=[d,0,0] where d is the distance between
91 %Trochlear groove and Capitulum
92 % Humero__Radial_coor_ACS(3,:)=[?,0,L] where L is the distance from
93
94 else %This may not work anymore since it hasn't been debugged for a while
   - this is the "else" for the "If What_to_postprocess >1", so it is only
   for o
ption 1, which we don't use.
95 Joint = Humero_Ulnar_Joint;
96 col = 1; %1-Flexion-Extension
97 title1 = 'Points for calculation of the FHA from Flexion-Extension';
98 Tag = Tag_humero_ulnar;
99 % Kinem_File_ind = [33 34 35; 37 38 39; 41 42 43; 45 46 47]; %Ulna
100 Kinem_File_ind = [15 16 17; 18 19 20; 21 22 23; 24 25 26]; %for laurel
101 %Below are 3 points to establish a YZ plane of intersection
102 % They are offset 200 mm to the left of the Trochlear groove
103 P1 = Humero_Ulnar_coor_ACS(4,:);
104 P2 = Humero_Ulnar_coor_ACS(3,:);
105 P1 = P1 + [200 0 0]; %Projection plane 200mm away from the trochlea
106 P2 = P2 + [200 0 0];
107 P3 = [200 10 0]; %Some point on the y axis
108 % To check: Humero_Ulnar_coor_ACS(4,:)=[0,0,0] Origin of the Humeral ACS;
   Troch. groove
109 % Humero_Ulnar_coor_ACS(9,:)=[-d,0,0] Capitulum: d is the distance
   between
110 %Trochlear groove and Capitellum
111 % Humero_Ulnar_coor_ACS(3,:)=[?,0,L] where L is the distance from
   Trochlear groove
112 %to the Center of the Humeral
```

```
113 %Shaft
114 end % end What_to_Postprocess > 1 if.
115
116 % Store indeces of the row numbers that will correspond to a preset
117 % increment angle of the pronation-supination (flexion-extension for
118 % humero-ulnar joint)
119
120 first = Joint(1,col);
121 remember_index(1) = 1;
122 [m,n] = size(Kinematics_File);
123 j = 1;
124 Joint % for debugging
125 for i = 1:m %number of rows
126 diff = abs(abs(Joint(i,col))-abs(first));
127 if diff >= Angle_inc
128 j = j+1;
129 remember_index(j) = i;
130 first = Joint(i,col);
131 end
132 end
133
134 p = size(remember_index,2); %Number of columns
135 %Plot which points of pronation supination curve were stored in
    "remember_index"
136 xx=1:1:m;
137 figure(18)
138 title(title1)
139 hold on
140 %for i = 1:(p-1)
141 % cislo = num2str(i);
142 % text(remember_index,Joint(remember_index,col),cislo)
143 %end
144 plot(xx,Joint(:,col),'LineWidth',2)
145 plot(remember_index,Joint(remember_index,col),'*r','MarkerSize',10)
146
147 hold off
148 clear first
149
150 %-------------------------------------------------------------------------
    --
151 % Use the "remember_index" from the previous step to extract info from
    the
152 % Kinematics_File. Transfer the Kinematic data to Humero-Radial_ACS
153 p
154 for i = 1:(p-1)
155 xplot(i)=i;
156 ai(1,:) = Kinematics_File(remember_index(i),Kinem_File_ind(1,1:3));
157 ai(2,:) = Kinematics_File(remember_index(i),Kinem_File_ind(2,1:3));
158 ai(3,:) = Kinematics_File(remember_index(i),Kinem_File_ind(3,1:3));
159 ai(4,:) = Kinematics_File(remember_index(i),Kinem_File_ind(4,1:3));
160 Kinematics_in_ACS_humerus_before = Coord_transfer(ai,Tag);
161 pi(1,:) = Kinematics_File(remember_index(i+1),Kinem_File_ind(1,1:3));
162 pi(2,:) = Kinematics_File(remember_index(i+1),Kinem_File_ind(2,1:3));
163 pi(3,:) = Kinematics_File(remember_index(i+1),Kinem_File_ind(3,1:3));
164 pi(4,:) = Kinematics_File(remember_index(i+1),Kinem_File_ind(4,1:3));
165 Kinematics_in_ACS_humerus_after = Coord_transfer(pi,Tag);
```

```
166 T =
    Soder(Kinematics_in_ACS_humerus_before,Kinematics_in_ACS_humerus_after);
167 [n,phi,t,s] = screw_axis(T);
168
169 % The FHA is given by a vector and one point (s) on the FHA. The point s
    was
170 % renamed to P5 to correspond with my derivations.
171 P4 = s + 10*n; %Second point on the FHA from parametric equation of line
    with t=1
172 %(equation [7] in my derivations). I multiplied it by 10 because
173 %n is an unit vector and the two points would be close to each other
174 phi_col(i)=phi;
175 FHA_matrix(i,:)=n'; %vector of FHA
176 t_col(i) = t;
177 P5_matrix(i,:) = s'; %First point on the FHA
178 P4_matrix(i,:) = P4'; %Second point on the FHA
179 end
180
181 % LAUREL COMMENTED THIS LINE OUT.
182 % FHA_matrix; %For debugging
183 % t_col;
184 % P5_matrix;
185 % P4_matrix;
186
187 % ------------------------------------------------------------------------
188 % Add the angles from FHA. For Pronation-Supination and at 90 degree
189 % of flexion, the FHA follows the y-axis. % THIS MEANS IT WILL HAVE TO
190 % CHANGE FOR OTHER ANGLES.
191 % For Flexion-Extension it should be x-axis. Positive x would mean
    flexion.
192 phi_add(1) = Joint(1,col); %Start where the Grood Suntay does
193 signn = sign(FHA_matrix(1,2)); % Laurel thinks that this is the line that
    might need to change. Oh, this initializes things. ok.
194 for i = 1:(p-1) % less the last point, the last point has to be done
    separately
195 if sign(FHA_matrix(i,2)) == signn %y-axis
196 phi_add(i+1) = phi_add(i) - phi_col(i);
197 else
198 phi_add(i+1) = phi_add(i) + phi_col(i);
199 end
200 end
201
202 figure(19)
203 % This figure should look similar to figure 18. The difference is that
    this
204 % figure plots the actual angle (blue line) as calculated from the
    helical
205 % axis. Sometimes, this blue curve doesn't follow the red curve but it
206 % doesn't cause a problem in the following calculations (cosmetic error).
    I
207 % figured that it had to do something the way the curve starts, it may
208 % locally go down (the red one) and then shoots up while the blue one
    will
209 % go only down.
210 hold on
211 title('Angle Phi from FHA vs. from Grood Suntay')
212 plot(xx,Joint(:,col),'r')
```

```
plot(remember_index,phi_add,'b',remember_index,phi_add,'*b')
legend('Phi from Grood Suntay','Phi from FHA')
hold off

remember_index_1=remember_index(2:end); %substract the first index which is just 1

% figure
% title('Displacement of the FHA')
% plot(remember_index_1,t_col)
% ylabel('Displacement (mm)')
% xlabel('N. of frames')

% Change the sign of the FHA vector so that they all are pointing the same way
% This is due to the fact that pronation and opposite supination produce
% opposite normals. The sense of rotation and the direction of FHA corresponds to the
% right-hand rule. Not doing this screws up averaging.
prvy = sign(FHA_matrix(1,2)); %y axis for Pronation_supination
for i = 1:(p-1)
FHA_One_Direction(i,:) = FHA_matrix(i,:);
if sign(FHA_matrix(i,2)) ~= prvy
FHA_One_Direction(i,:) = -FHA_matrix(i,:);
end
end
%FHA_matrix = FHA_One_Direction


%*********************************************************************
% STATISTICS
% Woltring, 1990 ,JK8, Biomechanics of Human Movement, pg 203-237
% Veeger, PRUJ 6.1 Orientation of axis in the elbow and forearm for
% biomechanical modeling (on the web)
% Stokdijk, 1999, PRUJ 7, Clinical Biomechanics, pg, 177-184
% DeLange, JK22, 1990, J. of Biomechanical Engr. pg 107-113

% Below is the equation 38 from Woltring
% According to Woltring if the Origin of Ey is projected onto FHA then
% the Q matrix is replaced by eye(3,3). In our case the P5_matrix is the point
% "s" which is the closest point on FHA from the Origin Ey
% This is essentially a simple averaging
Q_P5 = zeros(3,1);
for i = 1:(p-1)
Q_P5 = Q_P5 + eye(3,3)*P5_matrix(i,:)';
end

Q_P5_aver = Q_P5/(p-1);
P5_opt = eye(3,3)*Q_P5_aver;
%The P5_opt is the same as simple averaging below
%p=sum(P5_matrix)./(p-1)

%Similarly, calculate the optimal FHA vector
FHA = zeros(3,1);
for i = 1:(p-1)
FHA = FHA + eye(3,3)*FHA_One_Direction(i,:)';
```

```
end
FHA_ave = FHA/(p-1);
FHA_opt = eye(3,3)*FHA_ave;
FHA_opt=FHA_opt./norm(FHA_opt); %Normalize the vector
%The P5_opt is the same as simple averaging below
%FHA=sum(FHA_One_Direction)./(p-1)

%The P5_opt is the same as simple averaging below
%p=sum(P5_matrix)./(p-1)

% Calculate the intersection of the mean FHA
%p4_opt1=sum(P4_matrix)./(p-1) the same as below
P4_opt = P5_opt + 10*FHA_opt;

% from here its an unsuccsesfull attempt to do it by least square
Qsum = zeros(3,3);
Q_Ssum = zeros(3,1);
for i = 1:(p-1)
Q(:,:,i) = (eye(3,3) - FHA_One_Direction(i,:)'*FHA_One_Direction(i,:));
   %Eq 38b
Qsum = Qsum + Q(:,:,i);
Q_Ssum = Q_Ssum + Q(:,:,i)*P5_matrix(i,:)';
end
% Qaver = Qsum./(p-1);
% Q_Ssum_aver = Q_Ssum./(p-1);
% % the "s" optimal
% P5_optimal = inv(Qaver)*Q_Ssum_aver %Eq 38a
% % The end of unsuccsesfull attempt

% Position dispersion of the FHA, which is rms distance between FHA_opt
   and FHA's
% Woltring calls it deff, deLange (1990) calls it Ds. Since P5_opt is the
% closest point on the FHA_opt to the origin of the coordinate system,
   this
% is essentially a spread of the FHA at the Capitulum
% dsum = 0; %
% for i = 1:(p-1)
% dsum = dsum + (P5_opt-P5_matrix(i,:)')'*Q(:,:,i)*(P5_opt-
   P5_matrix(i,:)');
% end
% d_eff = sqrt(dsum/(p-1)); %Units in mm, Equation 38c from Woltring

% Direction dispersion of the FHA,
Lambda_sum = 0;
for i = 1:(p-1)
Lambda_sum = Lambda_sum +
   sin(acos(dot(FHA_opt,FHA_One_Direction(i,:))))^2; %Eq 40
end
Lambda = Lambda_sum/(p-1);
pi=3.14159265358979;
Chi_eff = asin(sqrt(Lambda))*180/pi; %Units in degrees, Eq 41

fprintf(fid,'\n');
fprintf(fid,'STATISTICS of FHA:\n');
fprintf(fid,' Angle increment= %5.3f \n',Angle_inc);
%fprintf(fid,' Position dispersion of FHA, d_eff= %5.3f [mm]\n',d_eff);
```

```
317 fprintf(fid,' Direction dispersion of FHA, Chi_eff= %5.3f
    [Deg]\n',Chi_eff);
318 fprintf(fid,'\n');
319 fclose(fid); %Close the output file
320
321 %
    ************************************************************************
322 % Calculate intersection of FHA's with a plane perpendicular to the mean
323 % FHA at either the Center of the Radial Head or at the Ulnar Styloid
324 % Note: Earlier I used intersection of FHA by plane determined by humerus
325 % but this is OK only for elbow flexion of 90 degrees. To see how it was
326 % done see "Calculate_Screw_Axis1_1.m"
327 % 7/16/07 - LK. So Laurel interprets the above statement to mean that
    this
328 % should work for angles besides 90degrees flexion.
329 %-----------------------------------------------------------------------
330 % Use the ulnar styloid to calculate axis limit.
331 Ulnar_coor_in_Humero_Radial_ACS = Coord_transfer(Ulnar_coor_GCS,Tag);
332
333 %1. at the Radial Head
334 figure(20)
335 title2='Intersection of FHAs at the Capitellum (plane perp. to mean
    FHA)';
336 FHA_import =
    Plot_Intersection_Perpendicular_to_FHA(p,P4_opt,P5_opt,FHA_opt,P1,P4_matri
    x,P5_matrix,...
337 FHA_matrix,What_to_Postprocess,title2);
338
339 FHA_data(1,1)=Chi_eff;
340 FHA_data(1,2:7) = FHA_import;
341
342 %2. At the Ulnar Styloid
343 figure(21)
344 title2='Intersection of FHA at the Ulnar Styloid (plane perp. to mean
    FHA)';
345 FHA_import =
    Plot_Intersection_Perpendicular_to_FHA(p,P4_opt,P5_opt,FHA_opt,...
346 Ulnar_coor_in_Humero_Radial_ACS(1,:),P4_matrix,P5_matrix,...
347 FHA_matrix,What_to_Postprocess,title2,Ulnar_coor_in_Humero_Radial_ACS);
348 FHA_data(1,8:13) = FHA_import;
349
350 name_of_program = strcat(File_Name, '.ATD');
351 s=which(name_of_program);
352 S = strrep(s,name_of_program,''); %This is the directory where this
    program resides
353 % %works properly only if the name of the file
354 % %has the same small or capital letters as
355 % %in the directory
356 output_file = strcat(File_Name, '_FHA.FHA'); %name of the output file
357 % %will be the name of the master file.out
358
359 %-----------------------------------------------------------------------
360 % Write FHA results to two files, one for each experiment and one where
    all
361 % results from one experiment are collected (makes it easire to
362 % postprocess)
363 %
```

```
% Output seguence:
% 1. Name of the trial 2. Rh or Dr (used for FHA calculations)
% 2. Angle Increment 3. Cut off frequency
% 4. Chi_eff
% 5:10 Capitulum: med_lat,proxi_distal,total_d,Ds,med_sup,med_pron
% 11:16 US: med_lat,proxi_distal,total_d,Ds,med_sup,med_pron
fid = fopen(strcat(S,output_file),'w');
fprintf(fid,'%s,', name_of_program')
fprintf(fid,'%s,', DR_or_RH')
fprintf(fid,'%5.0f,', Angle_inc')
fprintf(fid,'%5.0f,', cut_off_frequency')
fprintf(fid,'%5.2f, %5.2f, %5.2f, %5.2f, %5.2f, %5.2f, %5.2f,
   %5.2f, %5.2f, %5.2f, %5.2f, %5.2f, %5.2f\n',FHA_data);
fclose(fid) %Close the output file

%The following can be used to write the results to one file
% that is, all trials in one folder written to 1 file (easy to export to
% EXCEL)
output_file = 'Elbow_FHAs.FHA'
fid = fopen(strcat(S,output_file),'a');
fprintf(fid,'\n')
fprintf(fid,'%s,', name_of_program')
fprintf(fid,'%s,', DR_or_RH')
fprintf(fid,'%5.0f,', Angle_inc')
fprintf(fid,'%5.0f,', cut_off_frequency')
fprintf(fid,'%5.2f, %5.2f, %5.2f, %5.2f, %5.2f, %5.2f, %5.2f, %5.2f,
   %5.2f, %5.2f, %5.2f, %5.2f, %5.2f, %5.2f\n',FHA_data);
fclose(fid) %Close the output file


%******************************************************
% Plot the FHA in 3D
figure(22)

Radial_coor_in_Humero_Radial_ACS =
   Coord_transfer(Reconstructed_Radius_GCS,Tag);
hold on
t = linspace(-400,0); %Increases or decreases the lengths of lines
%on 3D plot of the FHA's

if plot_only_mean_FHA == 'N' %Skip if only the mean FHA is to be plotted
if Left_or_Right == 'L' %Right-hand rule, for left arm, positive y is
   pronation. %Positive numbers create negative y axis values
for i = 1:(p-1)
if sign(FHA_matrix(i,2)) == 1
x = P4_matrix(i,1) + FHA_matrix(i,1)*t;
y = P4_matrix(i,2) + FHA_matrix(i,2)*t;
z = P4_matrix(i,3) + FHA_matrix(i,3)*t;
plot3(x,y,z,'b','LineStyle',':')
plot3(x,y,z,'k','Linewidth',1)
cislo = num2str(i);
y_y = abs(y); %To find index close to y=0
[value,t_row] = min(y_y);
%text(x(t_row),y(t_row),z(t_row),cislo) %Plot numbers in the intersection
   plane
else
x = P4_matrix(i,1) - FHA_matrix(i,1)*t;
```

```
y = P4_matrix(i,2) - FHA_matrix(i,2)*t;
z = P4_matrix(i,3) - FHA_matrix(i,3)*t;
plot3(x,y,z,'r','LineStyle',':')
plot3(x,y,z,'k','Linewidth',1)
cislo = num2str(i);
y_y = abs(y); %To find index close to y=0
[value,t_row] = min(y_y);
%text(x(t_row),y(t_row),z(t_row),cislo) %Plot numbers in the intersection plane
end
end
end

if Left_or_Right == 'R' %Right hand, positive y is supination.
for i = 1:(p-1)
if sign(FHA_matrix(i,2)) == 1
x = P4_matrix(i,1) - FHA_matrix(i,1)*t;
y = P4_matrix(i,2) - FHA_matrix(i,2)*t;
z = P4_matrix(i,3) - FHA_matrix(i,3)*t;
plot3(x,y,z,'b','LineStyle',':')
plot3(x,y,z,'k','Linewidth',1)
cislo = num2str(i);
y_y = abs(y); %To find index close to y=0
[value,t_row] = min(y_y);
%text(x(t_row),y(t_row),z(t_row),cislo) %Plot numbers in the intersection plane
else
x = P4_matrix(i,1) + FHA_matrix(i,1)*t;
y = P4_matrix(i,2) + FHA_matrix(i,2)*t;
z = P4_matrix(i,3) + FHA_matrix(i,3)*t;
plot3(x,y,z,'r','LineStyle',':')
plot3(x,y,z,'k','Linewidth',1)
cislo = num2str(i);
y_y = abs(y); %To find index close to y=0
[value,t_row] = min(y_y);
%text(x(t_row),y(t_row),z(t_row),cislo) %Plot numbers in the intersection plane
end
end
end
end %Skip if only the mean FHA is to be plotted

%Plot the mean FHA

if sign(FHA_opt(2)) == 1
x = P4_opt(1) - FHA_opt(1)*t*10;
y = P4_opt(2) - FHA_opt(2)*t*10;
z = P4_opt(3) - FHA_opt(3)*t*10;
else
x = P4_opt(1) + FHA_opt(1)*t*10;
y = P4_opt(2) + FHA_opt(2)*t*10;
z = P4_opt(3) + FHA_opt(3)*t*10;
end
plot3(x,y,z,'m','Linewidth',4)

xlabel('X-axis')
ylabel('Y-axis')
```

```
zlabel('Z-axis')
title('FHA-s Blue is Pronation')

y_min = 0; %Center of the Capitulum
y_max = Ulnar_coor_in_Humero_Radial_ACS(1,2)+50; %Ulnar Styloid
axis equal;
axis([-100 100 y_min-20 y_max+10 -150 150])
grid off; box on
view([150,30])

%Draw a circle representing the Capitulum
%radius_of_capitulum = .48*25.4;
if plot_landmarks == 'Y'

% For now connect Capitulum, Trochlea and top of the shaft to a triangle
h_coor(1,:) = Humero_Radial_coor_ACS(9,:); %Centroid of the Capitulum
h_coor(2,:) = Humero_Radial_coor_ACS(4,:); %Center of the Trochlear
Groove
h_coor(3,:) = Humero_Radial_coor_ACS(3,:); %Centroid of the Humeral Shaft
h_coor(4,:) = Humero_Radial_coor_ACS(9,:); %Centroid of the Capitulum

plot3(h_coor(:,1),h_coor(:,2),h_coor(:,3),'Linewidth',4,'Marker','p','Colo
r','g')

% For now connect three points on the distal radius to a triangle
r_coor(1,:) = Radial_coor_in_Humero_Radial_ACS(1,:);
r_coor(2,:) = Radial_coor_in_Humero_Radial_ACS(2,:);
r_coor(3,:) = Radial_coor_in_Humero_Radial_ACS(3,:);
r_coor(4,:) = Radial_coor_in_Humero_Radial_ACS(1,:);
plot3(r_coor(:,1),r_coor(:,2),r_coor(:,3),'Linewidth',4,'Color','g')

% Plot Ulnar styloid as a point
u_coor(1,1) = Ulnar_coor_in_Humero_Radial_ACS(1,1); % u_coor(1,2) =
Ulnar_coor_in_Humero_Radial_ACS(1,1);
u_coor(1,2) = Ulnar_coor_in_Humero_Radial_ACS(1,2); %u_coor(2,2) =
Ulnar_coor_in_Humero_Radial_ACS(1,2);
u_coor(1,3) = Ulnar_coor_in_Humero_Radial_ACS(1,3); %u_coor(2,3) =
Ulnar_coor_in_Humero_Radial_ACS(1,3);

plot3(u_coor(:,1),u_coor(:,2),u_coor(:,3),'Linewidth',4,'Marker','o','Colo
r','b')

center=[0,0]; %Origin of the Capitulum
NOP = 50; %Number of points on the circle
pi = 3.14159265358979;
THETA=linspace(0,2*pi,NOP);
RHO=ones(1,NOP)*radius_of_capitulum;
[X,Z] = pol2cart(THETA,RHO);
X=X+center(1);
Z=Z+center(2);
Y = zeros(1,NOP);
H=plot3(X,Y,Z,'b-');

end %plot_landmarks = 'Y'


Ulnar_coor_in_Humero_Radial_ACS;
```

```
519 hold off
520 dummy_s = 0;
```

P.2 COOR_CMM3.M

```
1 function [Humero_Ulnar_coor_ACS,Humero_Radial_coor_ACS, Ulnar_coor_ACS,
    Radial_coor_ACS,Pron_Sup_coor_ACS]...
2 =
    coor_CMM3(Humeral_coor_CCS,Ulnar_coor_CCS,Radial_coor_CCS,What_to_Postproc
    ess)
3
4 % -Calculates marker coordinates and bony landmark coordinates
5 % in ACS system for humerus, ulna and radius
6 % MCS- Marker coordinate system
7 % GCS- Global coordinate system
8 % ACS- Anatomical coordinate system
9 % CCS- CMM coordinate system
10 % -Input: Matrix of landmarks for each of the bones.
11 % See file "Determination of Bony Landmarks.doc" for more details
12 % -Output: Marker and Landmark Coordinates of all three bones in their
    respective ACS's
13 % Since we established two different coordinate systems, one humero-ulnar
    with
14 % the origin in the trochlear groove and second humero-radial with the
    origin
15 % in the capitulum there are 2 different humeral ACS coordinates
16 % (Humero_Ulnar_coor_ACS,Humero_Radial_coor_ACS)
17 % -Pronation-Supination ACS based on the ulnar styloid and center of the
    radial head
18
19 global PronSup %Utilized in coor_CMM.m to average first 10 frames to be
20 %used for calculation of bony landmarks
21 global Humeral_Markers_GCS_static %File containing 4 markers of the
    humerus
22 global Ulnar_coor_GCS %To transfer it to coor_RadialHead_RadialImplant.m
    for
23 %Pron-Sup axis
24 global Tag_humero_radial %Comes from coor_CMM.m, used for calculating FHA
25 global Tag_humero_ulnar % used for calculating FHA in Flexion-Extension
26 global File_Name
27 global rms_humeral %For plotting in the main program
28 global balls %array (file) that tells which balls from marker array are
    used
29 global Kinem_File_hum Kinem_File_u Kinem_File_dr
30 global Left_or_Right
31 global Tac_hum %For transfering cloud of points of the 3D humerus
32 global Tgc_ulna %For transfering cloud of points of the 3D ulna
33
34 %-------------------------------------------------------------------------
    -------
35 % Use first 10 frames of the experiment to establish ACS's % hmm, this
36 % could be bad for Laurel. Maybe we'd want to use somethign else?
```

```
37 PronSup_static = PronSup;
38 [N_of_frames,n] = size(PronSup_static);
39 if N_of_frames ~= 1 %Because the function "sum" sums rows instead of
    columns
40 %if only 1 row is present
41 PronSup_static = sum(PronSup_static(1:10,:))./10; %Average first 10 frames
42 end
43 % % Laurel test - seems to not matter if you use the last 10 frames
44 % instead.
45 % if N_of_frames ~= 1 %Because the function "sum" sums rows instead of
    columns
46 % %if only 1 row is present
47 % PronSup_static = sum(PronSup_static(end-10:end,:))./10; %Average last 10
    frames
48 % end
49
50
51
52 %@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@@@@@@@@@@@@
53 % HUMERO-ULNAR JOINT COORDINATE SYSTEM
54
55 %*********************************************************
56 % HUMERUS (for humero-ulanr joint)
57 % The origin of the humeral coordinate system is center of the Trochlear
    Groove
58 % The origin of the ulnar coordinate system is in the center of the
    Trochlear Groove
59 %+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
60
61 %---------------------------------------------------------------------
62 %Step1: Find Humeral ACS in CCS using three landmarks
63 % We get {Xca}
64 % LK- 7/23/07 - this is all based on anatomy, not balls, so it should stay
65 % the same.
66
67 ww(1,:) = Humeral_coor_CCS(4,:); %Center of the Trochlear groove
68 ww(2,:) = Humeral_coor_CCS(9,:); %Centroid of the Capitulum
69 ww(3,:) = ww(1,:); %If the Center of the Trochlear groov is the origin
70 ww(4,:) = Humeral_coor_CCS(3,:); %Center of the humeral Shaft
71 Tca = xyz_CMM(1,ww); %X-axis is fixed, Transformation matrix [TCA]
    ACShumerus-->CCS
72 %1- change sign on x-axis
73 if Left_or_Right == 'R'
74 Tca = xyz_CMM(0,ww); %X-axis is fixed, Transformation matrix [TCA]
    ACShumerus-->CCS
75 end
76
77%
78 Tac = inv(Tca); % [TAC] CCS-->ACShumerus
79
80 %---------------------------------------------------------------------
    ----------------
81 % Step 2, Transfer marker and landmark coordinates into ACShumerus,
    {Xa}=[TAC]{Xc}
82 Humero_Ulnar_coor_ACS = Coord_transfer(Humeral_coor_CCS,Tac); % transfers
    the whole humeral pointing file. This is still ok, LK, 7/23/07
```

```
83
84 % To check: Humero_Ulnar_coor_ACS(4,:)=[0,0,0] Origin of the Humeral ACS;
   Trochlear groove
85 % Humero_Ulnar_coor_ACS(9,:)=[-d,0,0] Capitulum: d is the distance between
86 %Trochlear groove and Capitulum
87 % Humero_Ulnar_coor_ACS(3,:)=[?,0,L] where L is the distance from
   Trochlear groove
88 %to the Center of the Humeral Shaft
89
90 %-------------------------------------------------------------------------
91 % Step 3, To transfer Flexion-Extension axis from humerus to ulna
92 % we have to use marker coordinates in the CCS and in GCS
93 % Find [TGC] (from the CCS to GCS)
94 % Use Soder.m instead of Spoors algorithm
95 % ai- {Xcm} Marker coordinates in CCS
96 % pi- {Xgm} Marker coordinates in GCS
97
98 % From the Spatial Model "Elbow12" :the size of matrix
   is=PronSup_static(nframes,48)
99 % After we added radial head implant it increased the size to 64
100
101 [ai,pi] =
   three_or_four_balls(balls(1,:),Kinem_File_hum,4,Humeral_coor_CCS,Humeral_M
   arkers_GCS_static);
102 [Tgc_humerus,rms] = Soder(ai,pi);
103 rms_humeral = rms;
104
105 %-----------------------------------------------------------------
106 % Step 4, Bony landmarks of the Humerus will now be transfered to GCS
107 %{Xg}=[Tgc]{Xc}
108 Humeral_coor_GCS = Coord_transfer(Humeral_coor_CCS,Tgc_humerus);
109
110 name_of_program = strcat(File_Name,'.ATD');
111 s=which(name_of_program);
112 S = strrep(s,name_of_program,'');
113 output_file = strcat(File_Name,'.out'); %name of the output file
114 %will be the name of the master file.out
115 fid = fopen(strcat(S,output_file),'a');
116
117
118 fprintf(fid,'Humeral Landmarks in GCS:\n');
119 fprintf(fid,' Trochlear Groove:
   %5.2f,%5.2f,%5.2f\n',Humeral_coor_GCS(4,:));
120 fprintf(fid,' Capitulum : %5.2f,%5.2f,%5.2f\n',Humeral_coor_GCS(9,:));
121 fprintf(fid,' Top of the Shaft:
   %5.2f,%5.2f,%5.2f\n',Humeral_coor_GCS(3,:));
122 fprintf(fid,'***rms_humeral, CCS to GCS: %5.3f\n',rms_humeral);
123 fprintf(fid,'\n');
124 fprintf(fid,'Humeral Landmarks in ACS (Humero-Ulnar joint):\n');
125 fprintf(fid,' Trochlear Groove:
   %5.2f,%5.2f,%5.2f\n',Humero_Ulnar_coor_ACS(4,:));
126 fprintf(fid,' Capitulum :
   %5.2f,%5.2f,%5.2f\n',Humero_Ulnar_coor_ACS(9,:));
127 fprintf(fid,' Top of the Shaft:
   %5.2f,%5.2f,%5.2f\n',Humero_Ulnar_coor_ACS(3,:));
128 fprintf(fid,'\n');
129
```

```
%*****************************************************************
% ULNA (for now, left arm)
%+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

% we will not have ulna in a meaningful way.
%-----------------------------------------------------------------------
---
% Step 5, Find the transformation matrix from ulnar CCS to GCS
[ai,pi] =
three_or_four_balls(balls(2,:),Kinem_File_u,2,Ulnar_coor_CCS,PronSup_stati
c);
[Tgc_ulna,rms] = Soder(ai,pi); % From ulnar CCS to GCS
rms_ulnar = rms;

%-----------------------------------------------------------------------
---
% Step 6, The Ulnar Styloid coordinates have to be transfered to GCS
% so that it can be used with the F-E axis (from step 5) to establish
% ACS(ulna) in GCS
%{Xg}=[Tgc]{Xc}
Ulnar_coor_GCS = Coord_transfer(Ulnar_coor_CCS,Tgc_ulna); %Contains
Styloid and Markers

%To check if we got meaningfull results, look at Humeral_coor_GCS(4,:)
%which is trochlear groove in humerus and check it with
Ulnar_coor_GCS(2,:)
%which is trochlear groove in the ulna. Should be very close
% Humeral_coor_GCS(4,:);
% Ulnar_coor_GCS(2,:);
% figure(50); plot(Humeral_coor_GCS(4,:), 'b-'); hold on;
plot(Ulnar_coor_GCS(2,:), 'r-' ) % Laurel added this line to check.
% Um....how close is close?
%-----------------------------------------------------------------------
%Step 7: Find Ulnar ACS in GCS using three landmarks

ww(3,:) = Humeral_coor_GCS(9,:); %Centroid of the Capitulum: from Humerus
ww(1,:) = Humeral_coor_GCS(4,:); %Center of the Trochlear groove: from
Humerus
ww(4,:) = ww(1,:); %If the Centroid of the Capitulum is the origin
ww(2,:) = Ulnar_coor_GCS(1,:); %Ulnar Styloid
Tga = xyz(ww); % Z-axis is fixed,Transformation matrix [TGA] ACSulna--
>GCS

Tag = inv(Tga); % [TAG] GCS-->ACSulna
Tag_humero_ulnar = Tag;

%-----------------------------------------------------------------------
% Step 8, Transfer marker and landmark coordinates into ACSulna,
{Xa}=[TAG]{Xg}
% Add the bony landmarks from Humerus into ULna
Ulnar_coor_GCS(7,:) = Humeral_coor_GCS(9,:); %Centroid of the Capitulum:
from Humerus
Ulnar_coor_GCS(8,:) = Humeral_coor_GCS(4,:); %Center of Trochlear
groove:from Humerus
Ulnar_coor_ACS = Coord_transfer(Ulnar_coor_GCS,Tag);

fprintf(fid,'Ulnar Landmarks in GCS:\n');
```

```
175 fprintf(fid,' Ulnar Styloid: %5.2f,%5.2f,%5.2f\n',Ulnar_coor_GCS(1,:));
176 fprintf(fid,' Trochlear ridge: %5.2f,%5.2f,%5.2f\n',Ulnar_coor_GCS(2,:));
177 fprintf(fid,'***rms_ulnar, CCS to GCS: %5.3f\n',rms_ulnar);
178 fprintf(fid,'\n');
179 fprintf(fid,'Ulnar Landmarks in ACS:\n');
180 fprintf(fid,' Ulnar Styloid: %5.2f,%5.2f,%5.2f\n',Ulnar_coor_ACS(1,:));
181 fprintf(fid,' Trochlear ridge: %5.2f,%5.2f,%5.2f\n',Ulnar_coor_ACS(2,:));
182 fprintf(fid,'\n');
183
184
185 % To check: Ulnar_coor_ACS(8,:)=[0,0,0] Origin of the ulnar ACS
186 % Ulnar_coor_ACS(7,:)=[-d,0,0] where d is the distance between
187 % Trochlear groove and Capitulum
188 % Ulnar_coor_ACS(1,:)=[?,0,L] where L is the distance from
189 % %Trochlear groove to the Ulnar styloid
190
191
192 %@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@@@@@@@@@@@@
193 % HUMERO-RADIAL JOINT COORDINATE SYSTEM
194 % The origin of the humeral coordinate system is in the Capitulum
195 % The origin of the radial coordinate system is in the center of the
    Radial Head
196
197 Humero_Radial_coor_ACS = zeros(10,3); %Zero out to be faster
198 Radial_coor_ACS = zeros(10,3); %Return something if Radius is not
    involved
199 Pron_Sup_coor_ACS = zeros(8,3); %Return something if Radius is not
    involved
200
201
202
    %*************************************************************************
203 % Humerus (humero-radial CS)
204
    %+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
205 % For left elbow, the x-axis is positive to the medial side (from
    capitulum
206 % to the trochlear groove).
207 % For right elbow, the x-axis stays in the same direction but this time
208 % pointing to the lataral side (from trochlear groove to the capitulum)
209 % This means that for postprocessing, negative x should always be to the
210 % right (set(gca,'Xdir','reverse')) but the signs for "Medial" and
211 % "Lateral" should be switched depending on the elbow- true for the
    picture with
212 % capitulum as a circle (usaully Figure#2).
213 %-------------------------------------------------------------------
214 %Step 9: Find Humeral ACS in CCS using three landmarks
215 % We get {Xca}
216 ww(1,:) = Humeral_coor_CCS(9,:); %Centroid of the Capitulum
217 ww(2,:) = Humeral_coor_CCS(4,:); %Center of the Trochlear groove
218 ww(3,:) = ww(1,:); %If the Centroid of the Capitulum is the origin
219 ww(4,:) = Humeral_coor_CCS(3,:); %Center of the humeral Shaft
220 Tca = xyz_CMM(0,ww); %X-axis is fixed, Transformation matrix [TCA]
    ACShumerus-->CCS
221 %0-do not change sign on X-axis
222 if Left_or_Right == 'R'
```

```
223 Tca = xyz_CMM(1,ww); %X-axis is fixed, Transformation matrix [TCA]
    ACShumerus-->CCS
224 end
225
226 Tac = inv(Tca); % [TAC] CCS-->ACShumerus
227 Tac_hum = Tac; %For transfering cloud of points of the 3D humerus
228
229 %-------------------------------------------------------------------------
230 % Step 10, Transfer marker and landmark coordinates into ACShumerus,
    {Xa}=[TAC]{Xc}
231
232 Humero_Radial_coor_ACS = Coord_transfer(Humeral_coor_CCS,Tac);
233
234
235 % The Humero_Radial_coor_ACS is needed for What_to_Postprocess=2,3,4.
236 % For What_to_Postprocess=1 it is not needed but won't harm to have it.
237
238 % To check: Humero_Radial_coor_ACS(9,:)=[0,0,0] Origin of the Humeral
    ACS; Capitulum
239 % Humero_Radial_coor_ACS(4,:)=[d,0,0] where d is the distance between
240 %Trochlear groove and Capitulum
241 % Humero_Radial_coor_ACS(3,:)=[?,0,L] where L is the distance from
242 %Capitulum to the Center of the Humeral Shaft
243 %-------------------------------------------------------------------------
244 % Step 10.1, Find Tag_humerus based on Humero-radial coordinate system
245 % This will be used for finite helical axis calculations
246 [Tag_humero_radial,rms] = soder(Humeral_coor_GCS,Humero_Radial_coor_ACS);
247 rms_AG_humero_radial = rms; % [TAG] GCS-->ACShumero_radial
248
249 fprintf(fid,'Humeral Landmarks in ACS (Humero-Radial joint):\n');
250 fprintf(fid,' Trochlear Groove:
    %5.2f,%5.2f,%5.2f\n',Humero_Radial_coor_ACS(4,:));
251 fprintf(fid,' Capitulum :
    %5.2f,%5.2f,%5.2f\n',Humero_Radial_coor_ACS(9,:));
252 fprintf(fid,' Top of the Shaft:
    %5.2f,%5.2f,%5.2f\n',Humero_Radial_coor_ACS(3,:));
253 fprintf(fid,'***rms_AG_humero_radial, ACS to GCS:
    %2.5f\n',rms_AG_humero_radial);
254
255 fprintf(fid,'\n');
256
257
258 if What_to_Postprocess == 2 %Do the following if the Radius is in
    calculations
259 % but the radial head was not cut off for the Radial Head implants.
260 %Otherways, skip to the end of the file
261 %If skipped, the Radius will be taken care of in
    coor_RadialHead_RadialImplant.m
262
263
    %*************************************************************************
264 % Radius (for now, left arm)
265
    %+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
266
267 clear Radial_coor_ACS Pron_Sup_coor_ACS
268
```

```
% ------------------------------------------------------------------------
-----
% Step 11, ACSradius is found (in CCS)
%

%Center of the distal radius
Center_Distal_End = ((Radial_coor_CCS(2,:) + Radial_coor_CCS(3,:))./2 +
Radial_coor_CCS(1,:))./2;
P_Volar_Dorsal = (Radial_coor_CCS(2,:) + Radial_coor_CCS(3,:))./2; %Point
in the middle

ww(1,:) = Radial_coor_CCS(4,:); %Center of the radial head, Origin of the
ACSradius
ww(2,:) = Center_Distal_End; %Center of the distal radius
ww(3,:) = ww(2,:); %If the Center of the Radial head is the origin
ww(4,:) = P_Volar_Dorsal; %

Tca = xyz(ww); %Z-axis is fixed, Transformation matrix [TCA] ACSradius--
>CCS
Tac = inv(Tca);

%------------------------------------------------------------------------
--------------------
% Step 12, Transfer marker and landmark coordinates into ACSradius,
{Xa}=[TAC]{Xc}
% Add the calculated bony landmarks (Center_Distal_End and the
P_Volar_Dorsal)
Radial_coor_CCS(9,:) = P_Volar_Dorsal; %Point in the middle between
Dorsal and Volar
%radioulnar joint points
Radial_coor_CCS(10,:) = Center_Distal_End; %Center of the distal radius
Radial_coor_ACS = Coord_transfer(Radial_coor_CCS,Tac)

fprintf(fid,'Radial Landmarks in ACS:\n');
fprintf(fid,' Center of the distal radius:
%5.2f,%5.2f,%5.2f\n',Radial_coor_ACS(10,:));
fprintf(fid,' Point in the middle between Dorsal and Volar:
%5.2f,%5.2f,%5.2f\n',Radial_coor_ACS(9,:));
fprintf(fid,' Center of the radial head:
%5.2f,%5.2f,%5.2f\n',Radial_coor_ACS(4,:));
fprintf(fid,'\n');


% To check: Radial_coor_ACS(4,:)=[0,0,0] Origin of the radial ACS- Radial
head
% Radial_coor_ACS(9,:)=[?,0,L] where L is the distance between the Radial
% Head and the P_Dorsal_Volar (length of the radius)
% Radial_coor_ACS(10,:)=[0,0,L] Center of the distal radius

%@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@@@@@@@@@@@@@
% PRONATION-SUPINATION AXIS
% Transfomation to Pron-Sup Axis which runs from the center of the radial
head
% to the ulnar styloid
%------------------------------------------------------------------------
--------------
```

```
310
311 % Step 13, Find the transformation matrix from radial CCS to GCS
312 % The landmarks from ulna (in GCS) will be combined with landmarks in
    radius (in GCS)
313 [ai,pi] = three_or_four_balls(balls(3,:),Kinem_File_dr,3,
    Radial_coor_CCS,PronSup_static);
314 [Tgc_radial,rms] = Soder(ai,pi); % From radial CCS to GCS
315 rms_radial = rms;
316
317 %-------------------------------------------------------------------------
    -------------
318 % Step 14, The Radial landmark coordinates are transferred to GCS
319 % so that the Pronation-Supination Joint coordinate system can be created
320 % %{Xg}=[Tgc]{Xc}
321 Radial_coor_GCS = Coord_transfer(Radial_coor_CCS,Tgc_radial); %Landmarks
    and Markers
322
323 fprintf(fid,'Radial Landmarks in GCS:\n');
324 fprintf(fid,' Center of the distal radius:
    %5.2f,%5.2f,%5.2f\n',Radial_coor_GCS(10,:));
325 fprintf(fid,' Point in the middle between Dorsal and Volar:
    %5.2f,%5.2f,%5.2f\n',Radial_coor_ACS(9,:)); % Laurel changed the
    Radial_coor_AGS
to Radial_coor_ACS.
326 fprintf(fid,' Center of the radial head:
    %5.2f,%5.2f,%5.2f\n',Radial_coor_GCS(4,:));
327 fprintf(fid,'***rms radial, CCS to GCS: %2.5f\n',rms_radial);
328 fprintf(fid,'\n');
329
330
331 disp('Humeral trochlear groove in GCS')
332 Humeral_coor_GCS(4,:)
333 disp('Center of the radial head in GCS')
334 Radial_coor_GCS(4,:)
335
336
337 %----------
338 %Step 15
339 ww(1,:) = Radial_coor_GCS(4,:); %center of the Radial Head
340 ww(2,:) = Ulnar_coor_GCS(1,:); %Ulnar Styloid
341 ww(3,:) = Radial_coor_GCS(10,:) %%Center of the distal radius
342 ww(4,:) = Radial_coor_GCS(9,:); %P_Volar_Dorsal;
343
344 Tga = xyz(ww); %Z-axis is fixed, Transformation matrix [TCA] ACSradius--
    >GCS
345 Tag = inv(Tga);
346
347 %-------------------------------------------------------------------------
    ---------------
348 % Step 16, Transfer marker and landmark coordinates into ACSPron_Sup,
    {Xa}=[TAG]{Xg}
349 % Since the supination-pronation coordinate system does not have a marker
    array
350 % radial marker array will be used
351 Pron_Sup_GCS(1,:) = ww(1,:); %center of the radial head
352 Pron_Sup_GCS(2,:) = ww(2,:); %Ulnar Styloid
353 Pron_Sup_GCS(3,:) = ww(4,:); %%P_Volar_Dorsal
```

```
354 Pron_Sup_GCS(4:7,:) = Radial_coor_GCS(5:8,:) %marker arrays
355 Pron_Sup_coor_ACS = Coord_transfer(Pron_Sup_GCS,Tag);
356
357 end %end of " What_to_Postprocess == 2" (nondestrupted Radius)
358
359 fclose(fid) %Close the outup file
```

P.3 COOR_RADIALHEAD_RADIALIMPLANT3.M

```
1 function
    [Radial_Head_or_Implant_coor_ACS,Pron_Sup_coor_ACS,Radial_coor_ACS]...
2 =
    coor_RadialHead_RadialImplant3(Distal_Radius_coor_CCS,Radial_Head_or_Impla
    nt_coor_CCS,What_to_Postprocess)
3
4 % -Calculates marker coordinates and bony landmark coordinates of the
    Radius with either
5 % natural Radial Head or Radial Head Implants.
6 % Since the radial head will be cut off, the bony landmarks and ACS of
    Radius have to be
7 % created after coordinates of the bony landmarks were transferred from two
    separate pieces.
8 % One piece (common for Radial Head and All Radial Head implants) is from
    the file
9 % "Distal_Radius_coor_CCS".
10 %The other piece depends on whether it is the Radial Head or Implant
    "Radial_Head_or_Implant"
11 % These two files are in two different CCS systems. That means that the
    reconstruction
12 % will be done in GCS system by using average of the first 10 frames from
    Peak Cameras
13 % in CCS system for humerus, ulna and radius
14 % MCS- Marker coordinate system
15 % GCS- Global coordinate system
16 % ACS- Anatomical coordinate system
17 % CCS- CMM coordinate system
18 % -Input: Landmarks and markers of Distal Radius and either Radial Head or
    Implant.
19 % -Output: Marker and Landmark Coordinates in their respective ACS's for:
20 % a, radial implant
21 % b, Pronation-Supination axis
22 % c, reconstructed radius
23
24 global PronSup % Kinematics File, Average first 10 frames for calculation
    of bony landmarks
25 global Ulnar_coor_GCS %To transfer it from coor_CMM.m for Pron-Sup axis
26 global File_Name %For printing to the output file
27 global Reconstructed_Radius_GCS %For calculate_screw_axis.m
28 global balls %array (file) that tells which balls from marker array are
    used
29 global Kinem_File_dr Kinem_File_rh_i
30 global Tgc_Distal_Radius %To create 3D Distal Radius in D3_animation.m
```

```
31 global Tgc_Rad_Head_or_Implant %To create 3D Radial Head in D3_animation.m
32
33 PronSup_static = PronSup;
34 %-------------------------------------------------------------------------
   ----------
35 % Use first 10 frames of the experiment to establish ACS's
36 [N_of_frames,n] = size(PronSup_static);
37 if N_of_frames ~= 1 %The function "sum" sums rows instead of columns if #
   of rows=1
38 PronSup_static = sum(PronSup_static(1:10,:))./10; %Average first 10 frames
39 end
40
41 % Laurel tried using the last 10 frames below - as with the other one, it
42 % seems to not affect results.
43 % if N_of_frames ~= 1 %The function "sum" sums rows instead of columns if
   # of rows=1
44 % PronSup_static = sum(PronSup_static(end-10:end,:))./10; %Average first
   10 frames
45 % end
46
47 %@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@@@@@@@@@@@@@
48 % HUMERO-RADIAL JOINT COORDINATE SYSTEM
49 % The origin of the humeral coordinate system is in the Capitulum
50 % The origin of the radial coordinate system is in the center of the
   Radial Head
51
52 Humero_Radial_coor_ACS = zeros(10,3); %Zero out field, should be faster
53 Radial_coor_ACS = zeros(10,3); %Zero out field, should be faster
54 Pron_Sup_coor_ACS = zeros(8,3); %Zero out field, should be faster
55
56 %**************************************************************************
57 % RADIUS RECONSTRUCTION
58 % -Since the radial head is cut off, the radius has to be reconstructed
59 % from distal radius and from radial head (implant).
60 %- If What_to_Postprocess == 3 the Reconstructed_Radius_GCS(5:8,:) will
61 % contain Radial Head marker arrays
62 % - If What_to_Postprocess == 4 the Reconstructed_Radius_GCS(5:8,:) will
63 % contain Distal Radius marker arrays
64
65 % Step 1 Distal Radius (for now, left arm)
66 %++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
67
68 % Find the transformation matrix from radial CCS to GCS
69 [ai,pi] = three_or_four_balls(balls(3,:),Kinem_File_dr,3,
   Distal_Radius_coor_CCS,PronSup_static);
70 [Tgc_Distal_Radius,rms] = Soder(ai,pi); % From ulnar CCS to GCS
71 rms_Distal_Radius = rms;
72
73 %**************************************************************************
74 % Step 2 Radial Head or Radial Head Implant (for now, left arm)
75 %++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
76 % Find the transformation matrix from radial CCS to GCS
77 [ai,pi] = three_or_four_balls(balls(4,:),Kinem_File_rh_i,1,
   Radial_Head_or_Implant_coor_CCS,PronSup_static);
78 [Tgc_Rad_Head_or_Implant,rms] = Soder(ai,pi); % From ulnar CCS to GCS
79 rms_Rad_Head_or_Implant = rms;
```

```
80
81 %-------------------------------------------------------------------------
   ------------
82 % Step 3, The Radial landmark coordinates from two parts have to be
   transfered to GCS
83 % %{Xg}=[Tgc]{Xc}
84
85 Distal_Radius_coor_GCS =
   Coord_transfer(Distal_Radius_coor_CCS,Tgc_Distal_Radius);
86 Radial_Head_or_Implant_coor_GCS =
   Coord_transfer(Radial_Head_or_Implant_coor_CCS,Tgc_Rad_Head_or_Implant);
87
88 % Combine landmarks from two different files
89 Reconstructed_Radius_GCS(1,:) = Distal_Radius_coor_GCS(1,:); %Radial
   Styloid
90 Reconstructed_Radius_GCS(2,:) = Distal_Radius_coor_GCS(2,:); %Volar
   Radioulanr joint
91 Reconstructed_Radius_GCS(3,:) = Distal_Radius_coor_GCS(3,:); %Dorsal
   Radioulanr joint
92 Reconstructed_Radius_GCS(4,:) = Radial_Head_or_Implant_coor_GCS(1,:);
   %Radial Head center2
93 %see "Determination of Bony Landmarks.doc" for more details
94
95 if (What_to_Postprocess==3)|(What_to_Postprocess==5)
96 Reconstructed_Radius_GCS(5:8,:) = Radial_Head_or_Implant_coor_GCS(2:5,:);
   %Marker array
97 % on Radial Head
98 else
99 Reconstructed_Radius_GCS(5:8,:) = Distal_Radius_coor_GCS(4:7,:); %Marker
   array
100 % on Distal radius
101 end
102
103 %-------------------------------------------------------------------------
   --------------------
104 % Step 4, ACSradius is found (in GCS)
105 %Center of the Distal Radius
106 Center_Distal_End = (( Reconstructed_Radius_GCS(2,:) +
   Reconstructed_Radius_GCS(3,:))./2 + Reconstructed_Radius_GCS(1,:))./2;
107 P_Volar_Dorsal = (Reconstructed_Radius_GCS(2,:) +
   Reconstructed_Radius_GCS(3,:))./2; %Point in the middle
108
109 ww(1,:) = Reconstructed_Radius_GCS(4,:); %Center of the Radial Head
110 ww(2,:) = Center_Distal_End; %Center of the Distal Radius
111 ww(3,:) = ww(2,:); %If the Center of the Radial Head is the origin
112 ww(4,:) = P_Volar_Dorsal; %
113
114 Tga = xyz(ww); %Z-axis is fixed, Transformation matrix [TCA] ACSradius--
   >GCS
115 Tag = inv(Tga);
116 Tag_radius = Tag; %Not used for anything right now
117
118 %-------------------------------------------------------------------------
   -------------
119 % Step 5, Transfer marker and landmark coordinates into ACSradius,
   {Xa}=[TAG]{Xg}
```

```
%Add the calculated bony landmarks (Center_Distal_End and the P_Volar_Dorsal)
Reconstructed_Radius_GCS(9,:) = P_Volar_Dorsal; %Point in the middle between Dorsal
%and Volar radioulnar joint points
Reconstructed_Radius_GCS(10,:) = Center_Distal_End; %Center of the Distal Radius

Radial_coor_ACS = Coord_transfer(Reconstructed_Radius_GCS,Tag);
% To check: Radial_coor_ACS(4,:)=[0,0,0] Origin of the radial ACS- Radial Head
% Radial_coor_ACS(9,:)=[d,0,s] d- distance between Trochlear groove and Capitulum
% there will be some small s because x-axis was not fixed
% Radial_coor_ACS(10,:)=[0,0,L] Center of the Distal Radius, L-length of the Radius

name_of_program = strcat(File_Name,'.ATD');
s=which(name_of_program);
S = strrep(s,name_of_program,'');
output_file = strcat(File_Name,'.out'); %name of the output file
%will be the name of the master file.out
fid = fopen(strcat(S,output_file),'a');

fprintf(fid,'Radial Landmarks from reconstructed radius, in GCS:\n');
fprintf(fid,' Center of the distal radius : %5.2f,%5.2f,%5.2f\n',Reconstructed_Radius_GCS(10,:));
fprintf(fid,' Point in the middle between Dorsal and Volar: %5.2f,%5.2f,%5.2f\n',Reconstructed_Radius_GCS(9,:));
fprintf(fid,' Center of the radial head : %5.2f,%5.2f,%5.2f\n', Reconstructed_Radius_GCS(4,:));
fprintf(fid,'***rms Distal Radius, CCS to GCS : %2.5f\n',rms_Distal_Radius);
fprintf(fid,'***rms Radial Head or Implant, CCS to GCS: %2.5f\n',rms_Rad_Head_or_Implant);
fprintf(fid,'\n');

fprintf(fid,'Radial Landmarks from reconstructed radius, in ACS:\n');
fprintf(fid,' Center of the distal radius : %5.2f,%5.2f,%5.2f\n',Radial_coor_ACS(10,:));
fprintf(fid,' Point in the middle between Dorsal and Volar: %5.2f,%5.2f,%5.2f\n',Radial_coor_ACS(9,:));
fprintf(fid,' Center of the radial head : %5.2f,%5.2f,%5.2f\n', Radial_coor_ACS(4,:));
fprintf(fid,'\n');

%@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
% PRONATION-SUPINATION AXIS
% Transfomation to Pron-Sup Axis which runs from the Center of the Radial Head
% to the Ulnar Styloid
%--------------------------------------------------------------------------------------
%Step 6
ww(1,:) = Reconstructed_Radius_GCS(4,:); %Center of the Radial Head
ww(2,:) = Ulnar_coor_GCS(1,:); %Ulnar Styloid
```

```
160 ww(3,:) = Reconstructed_Radius_GCS(10,:);
161 %Center of the Distal Radius
162 ww(4,:) = Reconstructed_Radius_GCS(9,:); %P_Volar_Dorsal;
163
164 Tga = xyz(ww); %Z-axis is fixed, Transformation matrix [TCA] ACSradius--
    >GCS
165 Tag = inv(Tga);
166
167 %-------------------------------------------------------------------------
    -----------------
168 % Step 7, Transfer marker and landmark coordinates into ACSPron_Sup,
    {Xa}=[TAG]{Xg}
169 % Since the supination-pronation coordinate system does not have a marker
    array
170 % radial marker array will be used
171 Pron_Sup_GCS(1,:) = ww(1,:); %Center of the Radial Head
172 Pron_Sup_GCS(2,:) = ww(2,:); %Ulnar Styloid
173 Pron_Sup_GCS(3,:) = ww(4,:); %P_Volar_Dorsal
174 Pron_Sup_GCS(4:7,:) = Reconstructed_Radius_GCS(5:8,:); %marker arrays
175 Pron_Sup_coor_ACS = Coord_transfer(Pron_Sup_GCS,Tag);
176
177 Radial_Head_or_Implant_coor_ACS = zeros(7,3);
178
179 if (What_to_Postprocess==4)|(What_to_Postprocess==6)
180
181 %@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@@@@@@@@@@@@
182 % Radial Implant or Radial Head coordinate system
183 % Establishes ACS_Radial_Head_or_Implant
184 % This will be used to calculate relative motion between the implant and
    the Radius
185 % and between the implant and the Capitulum.
186 %-------------------------------------------------------------------------
    ---------------
187 %Step 8
188
189 ww(1,:) = Radial_Head_or_Implant_coor_CCS(1,:); %Center of the Radial
    Head or
190 %for implant it is the center of the contacting surface
191 ww(2,:) = Radial_Head_or_Implant_coor_CCS(7,:); %Center of the Radial
    Head but
192 %on the other side of the implant
193 %Z offset by the thickness of the implant if the implant was pointed
194 %by laying flat on the surface. It is important for Z axis to go
195 %z=ww(2,:)-ww(1,:) so that Z axis is pointing towards the radius
196 ww(3,:) = ww(1,:); %If the Center of the Trochlear groove is the origin
197 ww(4,:) = Radial_Head_or_Implant_coor_CCS(6,:); %Any point that is the
    radius of the
198 %implant away from the Center of the Radial head implant contacting point
199 %ww(1,:). This can be easily done by adding or substracting r from x or y
200 %coordinates of the ww(1,:) point (done in the
    "Radial_Head_or_Implant_Landmarks").
201 %This can be only done if the radial implant was lying flat on CMM
    machine.
202 %Note, this point is not on the rim of the implant.
203 Tca = xyz(ww); %Z-axis is fixed, Transformation matrix [TCA]
    ACS_Radial_Implant-->CCS
```

```
204 Tac = inv(Tca); % [TAC] CCS-->ACS_Radial_Implant
205
206 %-------------------------------------------------------------------------
    -----------------
207 % Step 9, Transfer marker and landmark coordinates into
    ACS_Radial_Implant, {Xa}=[TAC]{Xc}
208 Radial_Head_or_Implant_coor_ACS =
    Coord_transfer(Radial_Head_or_Implant_coor_CCS,Tac);
209
210 % To check: Radial_Head_or_Implant_coor_ACS(1,:)=[0,0,0] Origin, Center
    of the Radial head
211 % Radial_Head_or_Implant_coor_ACS(6,:)=[+-r,0,0]
212 % or Radial_Head_or_Implant_coor_ACS(6,:)=[0,+-r,0] depending on how the
    ww(4,:) was made
213 % Radial_Head_or_Implant_coor_ACS(7,:)=[0,0,w] where w is the width of
    the implant
214
215 end % if (What_to_Postprocess==4)|(What_to_Postprocess==6)
216
217 fclose(fid) %Close the output file
```

P.4 GROOD_SUNTAY.PDF

```
1 function Angles_Displacements = Grood_Suntay(T)
2 % Calculates Rotation angles and Displacements in a Joint coordinate system
3 % acc. to Grood, E. S. and Suntay, W. J., 1983: A Joint Coordinate System
4 % for Clinical Description of Three-Dimensional Motions: Application to the
    Knee.
5 % Journal of Biomechanical Engineering. 105, 136-144
6
7% Input parameters: Transformation matrix
8 % Output : One row: Three angles and three dispalacements arranged in a row
9 %
10 Angles_Displacements = zeros(1,6);
11
12 pi= 3.14159265358979;
13 alpha = atan(T(3,4)/T(4,4)); %Flexion, Extension
14
15 if alpha < 0 %Flexion extension has to be positive (my argument)
16 alpha = pi - abs(alpha);
17 end
18 %if T(4,4) < 0
19 % alpha = pi - abs(alpha); %from Kanes
20 %end
21
22 beta = acos(T(2,4)); %Valgus, Varus
23 gama = atan2(T(2,3),T(2,2)); %Internal, External Rotation of ulna or
    radius
24 %gama has to be used because Matlab has a function gamma
25
26 shift = T(2,1); %Medial-Lateral shift of the ulna or radius
```

```
27 glide = T(3,1)*cos(alpha)-T(4,1)*sin(alpha); %Anterior-Posterior glide of
    the ulna or radius
28 distraction = -T(2,4)*T(2,1)-T(3,4)*T(3,1)-T(4,4)*T(4,1);
29
30 Angles_Displacements(1,1) = alpha*180/pi;
31 Angles_Displacements(1,2) = beta*180/pi;
32 Angles_Displacements(1,3) = gama*180/pi;
33 Angles_Displacements(1,4) = shift;
34 Angles_Displacements(1,5) = glide;
35 Angles_Displacements(1,6) = distraction;
```

P.5 LIST_OF_EXPERIMENTS_LK

An abbreviated version of this file is shown here: omitted lines represent other instantiations of its use to process different data files.

```
1 function Input_Files = List_of_experiments;
2
3 % %-------------------------------------------------------------------------
4 % % List of all experiments, the last one not commented out is the one to
5 % % be run
6 % % Sequence of input parameters:
7 % % Input_Files =
    {What_to_Postprocess,cut_off_frequency,Kinematics_file,Humeral_Landmarks,
8 % % Ulnar_Landmarks(distal),Radial_Landmarks(distal),
9 % % Radial_Head_or_Implant_Landmarks,Proximal_Ulna_Landmarks, balls, sheet
    };
10 %
…
874 % 7/16/07
875 % Trying to debug the helical axis stuff with some "perfect" data - this
876 % sheet is the one with the perfect data, not the other.
877
    Input_Files={3,'N','R','RotatedOutputs.xls','RotatedOutputs.xls','MOCKHUM.
    dat','MOCKULNA.dat','MOCKDRAD.dat','MOCKRADH.dat','balls_all.d
at','fromViconPaper'};
```

P.6 PEAK_CMM3_LK3_FHA_FINAL.M

```
1 % This program is based on Peack_CMM3_LK3, which is itself based on
2 % Peak_CMM3 from Karol. Laurel has modified it so that one can calculate
3 % the FHA in a meaningful way without having explicity radial head markers.
4 %
5 % That is, the only meaningful results from this piece of code are the FHA
```

```
6 % results. The actual FHA part of the code is way down at the very bottom
7 % of this file.
8 %
9 % TO USE, one must add a new, active (uncommented) line to
10 % "List_of_experiments_LK", and comment out all other lines. (This is the
11 % way that Karol set it up.) Then just type "Peak_CMM3_LK3_FHA_try2" at
12 % the Matlab command prompt.
13 %
14 % During data collection, one must have arrays of 4 balls each on the
    ulna,
15 % the radius, and the humerus. The data file must be structured so that
16 % the columns of x,y,z data are arranged as described below. ** columns
17 % not used must still have numbers in them so that Matlab reads in the
18 % whole file correctly.
19 %
20 % TO VALIDATE The line in "List_of_experiments_LK" that needs to be active
    is this one,
21 % which is currently ~ line 877 of that file:
22 %
    Input_Files={3,'N','R','RotatedOutputs.xls','RotatedOutputs.xls','MOCKHUM.
    dat','MOCKULNA.dat','MOCKDRAD.dat','MOCKRADH.dat','balls_all.d
at','fromViconPaper'};
23 %
24 % (many of the comments below are from Karol, some modified by Laurel)
25 %-------------------------------------------------------------------------
    -
26 % Program that postprocesses results from Elbow experiments
27 % Needs functions:
28 % -xyz.m (creates CS based on 3 points, Z-axis fixed, Radius, RH and
    Implant)
29 % -xyz_CMM.m (creates CS based on 3 points, X-axis fixed, Humerus, Ulna)
30 % (-Transform.m (Spoor, Veldpaus algorithm)-not used-did not work)
31 % -Soder.m (used instead of Spoor's method)
32 % -Coord_transfer.m (Transfers coordinates of a point from one coordinate
33 % system to another, {Xgm}=[Tgm]*{Xmg} for example
34 % -coor_CMM3.m (Calculates marker coordinates and bony landmark
35 % coordinates in ACS system for certain type of bone)
36 % -coor_RadialHead_RadialImplant3.m (-Calculates marker coordinates and
37 % bony landmark coordinates of the radius with either natural
38 % Radial Head or Radial Head Implants)
39 % -Grood_Suntay.m (Establishes Joint Coordinate systems, returns
40 % rotations and displacements)
41 % -Plot_Rotations_Displacements.m- plots the above (NOT USED ANYMORE
42 % FOR FHA ONLY, LK, 11/29/07)
43 % -Calculate_Screw_Axis1_LK.m (Calculates screw axis parameters)
44 % -Butter_karol.m -my modification of the Butterworth filter. The only
45 % modification I did was to collect all functions and put them
46 % into one file (Butter_Karol.m)
47 % -Filtfilt.m -Matlab function for back and forth filtering
48 % -Radial_Head_on_Capitulum3.m -draws pictures of how the Radial head
49 % tracked on Capitulum
50 % -Plot_against_Styloids3.m- original idea was to plot the results
51 % against angle given by ulnar and radial styloids
52 % (real pronation-supination). The current version actually plots
53 % the results against external-internal rotation from pro-supination
54 % joint which is a close approximation of the real
55 % pro-supination. THIS FUNCTION IS NOT USED TO ONLY CALCULATE
```

```
56 % THE FHA. (LK, 11/29/07)
57 % -three_or_four_balls.m- makes it possible to have only 3 balls from a
58 % marker array to go to transformations instead of all 4. This
59 % is needed when one of the balls got bumped.
60 % other files needed:
61 % - balls.dat (see Karol's description below)
62 % - screw_axis.m (needed by Calculate_Screw_Axis1_LK)
63 % - Plot_Intersection_Perpindicular_to_FHA (from Karol)
64 %
65 %
66 % MCS- Marker coordinate system (Not really used)
67 % GCS- Global coordinate system
68 % ACS- Anatomical coordinate system
69 % CCS- CMM coordinate system
70 %
71 % Marker array names: 1. Humeral
72 % 2. Ulnar (Distal Ulna)
73 % 3. Radial (Distal Radius)
74 % 4. Radial head or implant
75 % 5. Ulnar proximal
76 % THings that might need to be changed in the future: In the butterworth
77 % filter section, adjust the calculation of Wn (currently line 396) to be
78 % 1/2 whatever the motion sampling frequency is.
79
80
81 clear all
82 close all % Close all figures
83
84 global PronSup %Utilized in coor_CMM.m to average first 10 frames to be
85 %used for calculation of bony landmarks
86 global Humeral_Markers_GCS_static %File containing 4 markers of the
    humerus
87 global Tag_humero_radial % used for calculating FHA in Pronation-
    Supination
88 global Tag_humero_ulnar % used for calculating FHA in Flexion-Extension
89 global File_Name
90 global rms_humeral %From cmm.m for plotting in "Check rms" section
91 global balls %array (file) that tells which balls from marker array are
    used
92 global Kinem_File_hum Kinem_File_u Kinem_File_dr Kinem_File_rh_i
    Kinem_File_pu
93 global Left_or_Right
94 global sheet % Laurel added this one.
95
96 %-------------------------------------------------------------------------
    -
97 %The sequence of columns from Peak system (Note: the Ulna and the Distal
98 %radius are switched in the program to acommodate for earlier sequence
    that
99 %the program was written for
100 % Kinem_File_hum = [1 2 3; 5 6 7; 9 10 11; 13 14 15]; %Humerus
101 % Kinem_File_u = [33 34 35; 37 38 39; 41 42 43; 45 46 47]; %Ulna
102 % Kinem_File_dr = [17 18 19; 21 22 23; 25 26 27; 29 30 31]; %Distal
    Radius
103 % Kinem_File_rh_i = [49 50 51; 53 54 55; 57 58 59; 61 62 63]; %Radial
    head or Implant
```

```
% Kinem_File_pu = [65 66 67; 69 70 71; 73 74 75; 77 78 79]; %Proximal Ulna
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%THIS IS THE PART TO CHANGE IN THE FUTURE IF NEEDED.
%%%%%These next few lines establish which columns of the data go with which
%%%%%markers.
%%%%% IMPORTANT: For this piece of code ONLY, we trick it into thinking
%%%%% that we have radial head markers when in reality we do not. That is,
%%%%% Kinem_file_rh_i should be the same as Kinem_File_dr . (The same is
%%%%% true for Kinem_file_pu and Kinem_File_u.) This is because it was
%%%%% easier to do it this way than to pull apart existing code.
% CURRENT:
Kinem_File_hum = [27 28 29; 30 31 32; 33 34 35; 36 37 38]; %Humerus
Kinem_File_u = [15 16 17; 18 19 20; 21 22 23; 24 25 26]; %Ulna
Kinem_File_dr = [3 4 5; 6 7 8; 9 10 11; 12 13 14]; %Distal Radius
Kinem_File_pu = [15 16 17; 18 19 20; 21 22 23; 24 25 26]; %Proximal Ulna - WE DO NOT HAVE THIS (so tell it to use the regular ulna array)
Kinem_File_rh_i = [3 4 5; 6 7 8; 9 10 11; 12 13 14]; %Radial head or Implant (we won't have this, so just use the distal radius.)
%------------------------------------------------------------
% New part added on 4/21/2004 (by Karol)
% It will be possible to use only 3 balls from a marker array instead of 4
% in case one of them moved and made rms too high. To figure out which ball
% moved use the program "Check_rms_manually.m". Once it is found which ball
% has to be excluded, the information can be stored in "balls.dat" file. If
% all balls are fine, use the file "balls_all.dat" file which looks like the
% columns on the left side. If, for example, IC2 is excluded (ball2 on
% radial head) then the file should look like columns on the right.
% 1 2 3 4 [ Humerus] 1 2 3 4
% 1 2 3 4 [Distal Ulna] 1 2 3 4
% 1 2 3 4 [Distal Radius] 1 2 3 4
% 1 2 3 4 [Radial Head or Implant] 1 2 4 0
% 1 2 3 4 [Proximal Ulna] 1 2 3 4
%balls=[1 2 3 4; 1 2 3 4; 1 2 3 0; 1 2 3 0; 1 2 3 4];
% Read the experiment to be postprocessed
Input_Files = List_of_experiments_LK;
%------------------------------------------------------------------
%
% Input Landmark files, kinematic file and what to postprocess
% The sequence is as following:
%Input_Files =
```

```
150
    %{What_to_Postprocess,cut_off_frequency,Kinematics_file,Humeral_Landmarks,
151 % Ulnar_Landmarks(distal),Radial_Landmarks(distal),
152 % Radial_Head_or_Implant_Landmarks,Proximal_Ulna_Landmarks };
153
154 What_to_Postprocess = Input_Files{1,1};
155 cut_off_frequency = Input_Files{1,2}
156 Left_or_Right = Input_Files{1,3};
157 Kinematics_file = Input_Files{1,4}; %peak kinematic file
158 Humeral_File = Input_Files{1,5}; %Peak static file
159 Humeral_Landmarks = Input_Files{1,6}; %Pointing file
160 Ulnar_Landmarks = Input_Files{1,7}; %Pointing file
161 balls = Input_Files{1,10};
162 sheet = Input_Files{1,11};
163
164 switch What_to_Postprocess
165 case 1
166
167 case 2
168 %barely used
169 % Laurel put this in
170 Radial_Landmarks = Input_Files{1,8};
171 Radial_Head_or_Implant_Landmarks = Input_Files{1,9}; %Radial head
    Pointing %%LAUREL MOVED THIS UP HERE.
172 case 3
173 Radial_Landmarks = Input_Files{1,8}; %Distal Radius Pointing file
174 Radial_Head_or_Implant_Landmarks = Input_Files{1,9}; %Radial head
    Pointing
175 case 4
176 Radial_Landmarks = Input_Files{1,8}; %Distal Radius Pointing
177 Radial_Head_or_Implant_Landmarks = Input_Files{1,9}; %Implant Pointing
178 end
179
180 %What_to_Postprocess = % 1- Only Humero-Ulnar joint
181 % 2- (Humero-Ulnar joint + Radius that is not cut off for Radial
182 % head implant) Should be retired- not used anymore
183 % 3- Humero-Ulnar joint + Radius with cut off Radial head
184 % 4- Humero-Ulnar joint + Radial head implant
185 % 5- As 3 + contact with Proximal Ulna -Not in this
186 % program anymore, see Peak_Radio_Ulnar_Contact.m
187 % 6- As 4 + contact with Proximal Ulna -Not in this
188 % program anymore, see Peak_Radio_Ulnar_Contact.m
189
190 %If radial head implants are used this file does not contain the radial
    head.
191 %In this case it is not used in coo_CMM.m but in
    coor_RadialHead_RadialImplant.m
192 %and contains distal landmarks with the radial marker
193 %%%%%%%%%%%%%NOTE FROM LAUREL: from now on, use "4" for
194 %%%%%%%%%%%%%everything.
195
196 balls = load(balls);
197
198 %%%%%LAUREL EDIT%%%%%%%%%
199 PronSup = xlsread(Kinematics_file, sheet); %It is called PronSup because
    I started with Pron-Supination
200 %File actually contains the output from Peaks
```

```
%system to be postprocessed (Karol liked to
%use "load". I prefer something I can look
%at in Excel. LK.)

% Humeral_Markers_GCS = load(Humeral_File);
%%%%%%%ANOTHER LAUREL EDIT HERE.
Humeral_Markers_GCS = xlsread(Humeral_File, sheet);

Humeral_Markers_GCS_static = Humeral_Markers_GCS;
[N_of_frames,n] = size(Humeral_Markers_GCS_static);
if N_of_frames ~= 1 %Because the function "sum" sums rows instead of
columns
%if only 1 row is present
Humeral_Markers_GCS_static = sum(Humeral_Markers_GCS_static(1:10,:))./10;
end

%Program was written for the following sequence of Peaks data
postprocessing:
% 1. Humerus
% 2. Radius
% 3. Ulna
% 4. Radial head or Radial head implant
% Starting from "Second Elbow Experiment" we changed the sequence to
% 1. Humerus
% 2. Ulna (Distal- to cover styloid)
% 3. Radius (Distal- to cover three distal radial landmarks)
% 4. Radial head or Radial head implant
% Starting from "Fourth Elbow Experiment (second pointing)", fifth
% marker array was added.
% 5. Proximal Ulna (to cover contact points on the ulna)
%

%-------------------------------------------------------------------------
-------------
%
% Write results into an output file
% The output file has to be closed before entering any functions.
% In functions, new info is appended into the output file
[m,n] = size(Kinematics_file); %Length of the string
File_Name1 = Kinematics_file(1,1:(n-4)); %cut the .atd suffix
File_Name=strcat(File_Name1, sheet);
name_of_program = strcat(File_Name,'.ATD');
s=which(name_of_program);
S = strrep(s,name_of_program,''); %This is the directory where this files
resides
%works properly only if the name of the file has the same small
%or capital letters as in the directory
output_file = strcat(File_Name, '.out'); %name of the output file
%will be the name of the master file.out
fid = fopen(strcat(S,output_file),'w');

fprintf(fid,'The following are results obtained for the following
experiment:\n');
switch What_to_Postprocess
case 1
fprintf(fid,'Only Humero-Ulnar joint- (What_to_Postprocess=1)\n');
case 2
```

```
253 fprintf(fid,'Humero-Ulnar joint + Radius that is not cut off for Radial
    head implant\n');
254 case 3
255 fprintf(fid,'Humero-Ulnar joint + Radius with cut off Radial head\n');
256 case 4
257 fprintf(fid,'Humero-Ulnar joint + Radial head implant\n');
258 end
259
260 d = date; %Current Date
261 fprintf(fid,'The experiment was postprocessed on %s\n',d');
262 fprintf(fid,'\n');
263 fprintf(fid,'The input files were as following:\n');
264 fprintf(fid,'Kinematic File- %s\n',Kinematics_file');
265 fprintf(fid,'Humeral Landmarks- %s\n',Humeral_Landmarks');
266 fprintf(fid,'Ulnar Landmarks- %s\n',Ulnar_Landmarks');
267
268 if What_to_Postprocess > 1
269 fprintf(fid,'Radial Landmarks- %s\n',Radial_Landmarks');
270 end
271 if What_to_Postprocess > 2
272 fprintf(fid,'Radial head or Radial Implant-
    %s\n',Radial_Head_or_Implant_Landmarks');
273 end
274 fprintf(fid,'\n');
275
276 %-------------------------------------------------------------------------
    ----------------------
277 %
278 % Establish local coordinate systems in the bones
279 % -The following matrices (Humero_Ulnar_coor_ACS...) contain both
    landmark and marker coordinates
280 % -Humero_Ulnar_coor_ACS- coordinates of humeral bony landmarks and
    markers used for
281 % Humero-Ulnar JCS. This means that the origin is in the Trochlear Groove
282 % -Humero_Radial_coor_ACS- coordinates of humeral bony landmarks and
    markers used for
283 % Humero-Radial JCS. This means that the origin is in the Capitellum
284 Radius_of_Capitulum = 0; %
285 Radius_of_Trochlear_Groove = 0; %From humerus, needed
286 Radius_of_Trochlear_Ridge =0; %From Proximal Ulna, informational purpose
    only
287 Radius_of_Radial_Head = 0;
288 Radius_of_Implant = 0;
289
290 Humerus = load(Humeral_Landmarks); Humerus(:,1) = []; %Get rid off the
    first column
291 Ulna = load(Ulnar_Landmarks); Ulna(:,1) = []; %Get rid off the first
    column
292 Humerus = Humerus.*25.4; % CMM does it in inches
293 Ulna = Ulna.*25.4; % CMM does it in inches
294 Radius_of_Capitulum = Humerus(9,4); % LK, 7/18/07 - why are these 3 radii
    not in inches? It must be taken care of somewhere else.
295 Radius_of_Trochlear_Groove = Humerus(4,4);
296 Radius_of_Trochlear_Ridge = Ulna(2,4);
297
298 % Radius refers to Distal Radius
```

```
299 Radius = zeros(8,4); %Bogus matrix if the Radius is not involved in
    calculations
300 if What_to_Postprocess ~= 1 %If the radial head is cut off then this file
    does not contain the radial head.
301 Radius = load(Radial_Landmarks); Radius(:,1) = []; %Get rid off the first
    column
302 Radius = Radius.*25.4; % CMM does it in inches
303 end
304 if What_to_Postprocess >= 2 %Radial Head or implant %%%Laurel added the
    "=" part of the >=2.
305 Radial_Head_or_Implant = load(Radial_Head_or_Implant_Landmarks);
306 Radial_Head_or_Implant(:,1) = []; %Get rid off the first column
307 Radial_Head_or_Implant = Radial_Head_or_Implant.*25.4; % CMM does it in
    inches
308 end
309 if What_to_Postprocess == 3 %Radial Head
310 Radius_of_Radial_Head = Radial_Head_or_Implant(1,4); % already in inches
    (from above)
311 end
312 if What_to_Postprocess == 4 %Implant
313 Radius_of_Implant = Radial_Head_or_Implant(1,4); % already in inches
    (from above)
314 end
315
316 Humerus(:,4) = []; %Delete the column with the radii
317 Ulna(:,4) = []; %Delete the column with the radii
318 Radius(:,4)=[];
319 Radial_Head_or_Implant(:,4) = []; %Delete the column with the radii
320
321 %Write it to the output file
322 fprintf(fid,'Radius of Capitulum= %5.2f mm\n',Radius_of_Capitulum');
323 fprintf(fid,'Radius of Trochlear_Groove= %5.2f
    mm\n',Radius_of_Trochlear_Groove');
324 fprintf(fid,'Radius of Trochlear_Ridge= %5.2f
    mm\n',Radius_of_Trochlear_Ridge');
325 fprintf(fid,'Radius of Radial_Head= %5.2f mm\n',Radius_of_Radial_Head');
326 fprintf(fid,'Radius of Implant= %5.2f mm\n',Radius_of_Implant');
327 fprintf(fid,'\n');
328
329 fclose(fid); %Close the output file
330
331 % For easier transport to functions put the radii into an array
332 Important_radii(1) = Radius_of_Capitulum;
333 Important_radii(2) = Radius_of_Trochlear_Groove;
334 Important_radii(3) = Radius_of_Trochlear_Ridge;
335 Important_radii(4) = Radius_of_Radial_Head;
336 Important_radii(5) = Radius_of_Implant;
337
338 %-------------------------------------------------------------------------
    --
339 %Function "coor_CMM.m"- establishes anatomical coordinates for humerus,
340 %ulna and radius not cut
341 [Humero_Ulnar_coor_ACS, Humero_Radial_coor_ACS, Ulnar_coor_ACS,
    Radial_coor_ACS, Pron_Sup_coor_ACS]...
342 = coor_CMM3(Humerus,Ulna,Radius,What_to_Postprocess);
343
344 %Retrieve Marker coordinates in ulnar ACS for Humerus
```

```
345 Markers_ACS_Humero_Ulnar = Humero_Ulnar_coor_ACS(5:8,:);
346
347 %Retrieve Marker coordinates in ACS for Ulna
348 Markers_ACS_Ulna = Ulnar_coor_ACS(3:6,:);
349
350 %Retrieve Marker coordinates in ACS for humerus and radius and Pronation-
    Supination axis
351 if What_to_Postprocess == 2 %If the Radius is involved, but not the
    Implants
352 %Also, if the radial head is cut off do not use the 3 lines
353 Markers_ACS_Humero_Radial = Humero_Radial_coor_ACS(5:8,:);
354 Markers_ACS_Radius = Radial_coor_ACS(5:8,:);
355 Markers_ACS_Pron_Sup = Pron_Sup_coor_ACS(4:7,:);
356 %Not fixed like the ones above because What_to_Postprocess == 2 is not
357 %likely to be used
358 end
359
360 %-----------------------------------------------------------------------
    -------------
361 % Function "coor_RadialHead_RadialImplant.m"- establishes anatomical
    coordinates for reconstructed radius
362 % and Radial head or Implant
363 %Postprocess results from radius whose radial head was cut off
364 %or results from radial head implant
365 if What_to_Postprocess > 2
366 [Radial_Implant_coor_ACS,Pron_Sup_coor_ACS,Reconstr_Radius_coor_ACS]=...
367
    coor_RadialHead_RadialImplant3(Radius,Radial_Head_or_Implant,What_to_Postp
    rocess);
368
369 %Retrieve Marker coordinates in radial ACS for humerus
370 Markers_ACS_Humero_Radial = Humero_Radial_coor_ACS(5:8,:); %Humeral
    markers
371
372 %Retrieve Marker coordinates in ACS for reconstructed Radius
373 Markers_ACS_Radius = Reconstr_Radius_coor_ACS(5:8,:); %Radial markers
374
375 %Retrieve Marker coordinates in ACS for Pronation Supination
376 %Radial marker array will be from Distal Radius for What_to_Postprocess=4
377 % and from Radial Head for What_to_Postprocess=3
378 Markers_ACS_Pron_Sup = Pron_Sup_coor_ACS(4:7,:);
379
380 %Retrieve Marker coordinates in ACS for Radial Head or Implant
381 Markers_ACS_Radial_Head = Radial_Implant_coor_ACS(2:5,:);
382
383 end % of What_to_Postprocess > 2
384
385 %-----------------------------------------------------------------------
    -----
386 % Smoothing by Butterworth Filter
387 [m,n] = size(PronSup);
388
389 %Design the filter butter_karol(N,Wn)
390 %B Nth order lowpass digital. The cut-off frequency Wn must be 0.0 < Wn <
    1.0,
391 %with 1.0 corresponding to half the sample rate.
392 % For example, we tape at 60Hz so 0.5*60=30Hz is half the sample rate
```

```
393 % King and Veeger use 1.5 Hz cut-off frequency => Wn=1/(30/1.5)=0.05
394 %Wn = 1/(30/1.5); ,Cut-off frequency Wn must be 0.0 < Wn < 1.0
395 %-The fundamental frequency was calculated from looking at the pron-sup
396 %graph and counting the number of frames it took for one part of the main
397 %curve- uphill motion. f_fundamental=1(number of frames/60seconds).
398 %Accoording to Woltring, JB 1994, p.1428, the optimal smoothing was
    around
399 %for Butterworth filter was for cut-off frequency of 5 times the
    fundamental
400 %frequency of the motion.
401
402 % Wn = 1/(30/cut_off_frequency); should this be 1/60/cutoff?
403 Wn = 1/(60/cut_off_frequency); % this seems to make more sense and makes
    things look much better.
404 N = 4; %Nth order lowpass digital filter
405 % N=8; %trying this.
406 [col1,col2] = butter_karol(N,Wn);
407
408 fid = fopen(strcat(File_Name,'.out'),'a'); %Open the file from main
    program and append data
409 fprintf(fid,'BUTTERWORTH FILTER parameters: \n');
410 fprintf(fid,'Cut off frequency for Butterworth Filter= %5.3f
    \n',cut_off_frequency);
411 fprintf(fid,'Order of lowpass digital filter= %5.3f \n',N);
412 fprintf(fid,'\n');
413 fclose(fid); %Close the output file
414
415 if cut_off_frequency ~= 'N' %if we are *not* doing No filtering
416 for i=1:n %Number of columns of the kinematics file
417 Dirty_Stuff = PronSup(:,i);
418 Filtered_Stuff = filtfilt(col1,col2,Dirty_Stuff);
419 PronSup(:,i) = Filtered_Stuff;
420 end
421 clear Dirty_Stuff Filtered_Stuff
422 end
423
424 %-----------------------------------------------------------------
425 % Go through all frames and analyze kinematics
426 %-----------------------------------------------------------------
427 %
428 % Improvement introduced on 2/25/04: The humeral marker array will be
429 % postprocesed in a separate file. Since the humerus is not moving it can
430 % be set constant throghout the motion but it should not be set static
431 % during postprocessing of Peaks tapes due to lower precision.
432 %
433 % Since the marker array balls in bone ACS's were arrainged to go from
    1:4
434 % the offset_num going to "three_or_four_balls.m" function will be=0 for
435 % all bones (see the part of the program after functions "coor_cmm.m" and
436 % "coor_RadialHead_RadialImplant.m"
437
438 %Humerus in Humero-Ulnar coordinate system
439 [ai,pi_i] =
    three_or_four_balls(balls(1,:),Kinem_File_hum,0,Markers_ACS_Humero_Ulnar,H
    umeral_Markers_GCS_static);
440 [Th_ulna,rms] = Soder(ai,pi_i); %[Tga]hum_ulnar
441 Th_ulna_inv = inv(Th_ulna); %[Tag]hum_ulnar
```

```
442
443 %Humerus in Humero-Radial coordinate system
444 if What_to_Postprocess > 1
445 [ai,pi_i] =
    three_or_four_balls(balls(1,:),Kinem_File_hum,0,Markers_ACS_Humero_Radial,
    Humeral_Markers_GCS_static);
446 [Th_radius,rms] = Soder(ai,pi_i); %[Tga]hum_radial
447 Th_radius_inv = inv(Th_radius); %[Tag]hum_radial
448 end
449 %-----------------------------------------------
450 % The big Loop
451
452 for i = 1:size(PronSup,1)
453 xplot(i)=i;
454
455 %---------
456 %Ulna
457 [ai,pi] =
    three_or_four_balls(balls(2,:),Kinem_File_u,0,Markers_ACS_Ulna,PronSup(i,:
    ));
458 [Tu,rms] = Soder(ai,pi); %[Tga]ulna
459 rms_ulna(i) = rms;
460
461 % Humero-Ulnar JCS
462 T_Humerus_2_Ulna = Th_ulna_inv * Tu; %[Tag]hum_ulnar*[Tga]ulna
463 Humero_Ulnar_Joint(i,:) = Grood_Suntay(T_Humerus_2_Ulna);
464
465 %-----------
466 %Radius
467 if (What_to_Postprocess==2)|(What_to_Postprocess==4)
468 % Radius is represented by the distal radius marker array (not
469 % radial head)
470 [ai,pi] =
    three_or_four_balls(balls(3,:),Kinem_File_dr,0,Markers_ACS_Radius,PronSup(
    i,:));
471 [Tr,rms] = Soder(ai,pi); %[Tga]Dr
472 rms_dist_rad(i) = rms;
473
474 % Humero-Radial JCS
475 T_Humerus_2_Radius = Th_radius_inv * Tr; %[Tag]hum_radial*[Tga]Dr
476 Humero_Radial_Joint(i,:) = Grood_Suntay(T_Humerus_2_Radius);
477
478 % Pronation-Supination JCS
479 [ai,pi] =
    three_or_four_balls(balls(3,:),Kinem_File_dr,0,Markers_ACS_Pron_Sup,PronSu
    p(i,:));
480 [Tp_s,rms] = Soder(ai,pi); %[Tga]PS
481 T_Humerus_2_Prn_Sup = Th_radius_inv * Tp_s; %[Tag]hum_radial*[Tga]PS
482 Pronation_Supination_Joint(i,:) = Grood_Suntay(T_Humerus_2_Prn_Sup);
483
484 end % (What_to_Postprocess==2)|(What_to_Postprocess==4)
485
486 if (What_to_Postprocess==3)
487 % (What_to_Postprocess=3 uses Radial Head marker array
488 [ai,pi] =
    three_or_four_balls(balls(4,:),Kinem_File_rh_i,0,Markers_ACS_Radius,PronSu
    p(i,:));
```

```
489 [Tr,rms] = Soder(ai,pi); %[Tga]RH
490 rms_Rad_Head(i) = rms;
491
492 % Humero-Radial JCS
493 Th_radius_inv = inv(Th_radius);
494 T_Humerus_2_Radius = Th_radius_inv * Tr; %[Tag]hum_radial*[Tga]RH
495 Humero_Radial_Joint(i,:) = Grood_Suntay(T_Humerus_2_Radius);
496
497 % Pronation-Supination JCS
498 [ai,pi] =
    three_or_four_balls(balls(4,:),Kinem_File_rh_i,0,Markers_ACS_Pron_Sup,Pron
    Sup(i,:));
499 [Tp_s,rms] = Soder(ai,pi);
500 T_Humerus_2_Prn_Sup = Th_radius_inv * Tp_s;
501 Pronation_Supination_Joint(i,:) = Grood_Suntay(T_Humerus_2_Prn_Sup);
502
503 end %(What_to_Postprocess==3)
504
505 %-----------------------------------------------
506 %Radial Head implant
507 if What_to_Postprocess==4 % Radial Head implant
508 [ai,pi] =
    three_or_four_balls(balls(4,:),Kinem_File_rh_i,0,Markers_ACS_Radial_Head,P
    ronSup(i,:));
509 [T_implant,rms] = Soder(ai,pi); %[Tga]Implant
510 rms_Rad_Imp(i) = rms;
511
512 %Humero- Radial Head implant JCS
513 T_Humerus_2_Implant = Th_radius_inv * T_implant;
    %[Tag]hum_radial*[Tga]Implant
514 Humero_Radial_Head_Implant_Joint(i,:) =
    Grood_Suntay(T_Humerus_2_Implant);
515
516 %Radio- Radial Head implant JCS
517 Tr_inv = inv(Tr);
518 T_Radius_2_Implant = Tr_inv * T_implant; %[Tag]DR*[Tga]Implant
519 Radio_Radial_Head_Implant_Joint(i,:) = Grood_Suntay(T_Radius_2_Implant);
520
521 end % What_to_Postprocess==4
522
523 end %End of the Big loop
524
525 %-----------------------------------------------------------
526 % Check rms
527 % It was noticed that sometimes, Peaks could start tracking different
528 % ball if two balls come close to each other. Making the "Search radius"
529 % small helps to avoid this problem. This procedure will
530 % calculate rms for the whole kinematics file.
531 %
532 % It could have been done in the "Big loop" but the distal radius is not
533 % calculated
534
535 %Humerus
536 for i = 1:size(PronSup,1)
537 rms_hum(i)= rms_humeral;
538 end
539
```

```
%Radial Head marker array
if (What_to_Postprocess == 3)
for i = 1:size(PronSup,1)
[ai,pi] =
    three_or_four_balls(balls(3,:),Kinem_File_dr,3,Radius,PronSup(i,:));
i;
ai;
pi;
[T,rms] = Soder(ai,pi);
rms_dist_rad(i) = rms;

end
end

% Laurel adds this part:
if (What_to_Postprocess == 4)
rms_hd=rms_Rad_Imp;
else
rms_hd=rms_Rad_Head;
end % end if

% End of "Check rms"
%-------------------------------------------------------------
% Postprocessing - do not need a lot of this
% plotting stuff for FHA only.
%Postprocess motion of the Radial Head on Capitellum
if (What_to_Postprocess==3)

    [Distraction,Head_coor_in_Humero_Radial_ACS,Back_side_in_Humero_Radial_ACS
    ] = Radial_Head_on_Capitulum3...
(Radial_Head_or_Implant,Important_radii,'',What_to_Postprocess); % This
    is the correct one to use - I just didn't realize it before. LK, 9/25/06
end

%Postprocess motion of the Radial Head on Capitellum
if (What_to_Postprocess==4) %Radial Head implant

    [Distraction,Head_coor_in_Humero_Radial_ACS,Back_side_in_Humero_Radial_ACS
    ] = Radial_Head_on_Capitulum3...
(Radial_Head_or_Implant,Important_radii,'Implant',What_to_Postprocess); %
    This is the correct one to use - I just didn't realize it before. LK, 9/
25/06
end
close all;

%% REMOVE THIS FOR NOW, FIX LATER. LAUREL, 9/16/05
% % FINITE HELICAL AXIS
if What_to_Postprocess > 1 %Not for humero-ulnar joint
s =
    Calculate_Screw_Axis1_LK(Important_radii,Humero_Radial_Joint,Humero_Ulnar_
    Joint,...

    Humero_Radial_coor_ACS,Humero_Ulnar_coor_ACS,Radius,What_to_Postprocess,He
    ad_coor_in_Humero_Radial_ACS,cut_off_frequency);
end
```

P.7 PLOT_AGAINST_STYLOIDS.M

```
1 function dummy_s = Plot_against_Styloids(Ulnar_coor_CCS,
   Distal_Radius_coor_CCS,...
2 Pronation_Supination_Joint,Distraction,Head_coor_in_Humero_Radial_ACS,...
3 What_to_Postprocess,Back_side_in_Humero_Radial_ACS)
4
5 % -Option #1 Plot results against Ulnar (US) and Radial (RS) styloids
   positions. When the
6 % RS is on the highest Z position in the humero-radial ACS then
7 % it should be on a vertical line with the US marking neatral
8 % pro-supination.
9 % -Option #2 (currently applied) Plot results against external-internal
   rotation
10 % obtained from pronation-supination joint. This ext-int. rotation should
   be close to the
11 % pronation-supination angle
12
13
14 global PronSup
15 global Ulnar_coor_GCS % Ulnar styloid (US)
16 global Reconstructed_Radius_GCS % Radial styloid (RS)
17 global Tag_humero_radial % To transfer the US and RS into Humeral ACS
18 global File_Name
19
20 Kinematics_File = PronSup;
21 Tag = Tag_humero_radial;
22 % Transfer US and RS into humeral ACS with origin in the Capitulum.
23 %Then we know that the highest positive Z coordinate of RS is in neutral
24 %pro-supination.
25 Radial_coor_in_Humero_Radial_ACS =
   Coord_transfer(Reconstructed_Radius_GCS,Tag);
26 Ulnar_coor_in_Humero_Radial_ACS = Coord_transfer(Ulnar_coor_GCS,Tag);
27
28 [m,n] = size(Kinematics_File);
29
30 Kinem_File_ind_Dr = [17 18 19; 21 22 23; 25 26 27; 29 30 31]; %Distal
   radius
31 Kinem_File_ind_U = [33 34 35; 37 38 39; 41 42 43; 45 46 47]; %Ulna
32
33 %Ulna- The procedure below is similar to coor_CMM2.m procedure
34 for i = 1:m
35 ai(1:4,:) = Ulnar_coor_CCS(3:6,:);
36 pi(1,:) = Kinematics_File(i,Kinem_File_ind_U(1,1:3));
37 pi(2,:) = Kinematics_File(i,Kinem_File_ind_U(2,1:3));
38 pi(3,:) = Kinematics_File(i,Kinem_File_ind_U(3,1:3));
39 pi(4,:) = Kinematics_File(i,Kinem_File_ind_U(4,1:3));
40 [Tgc_ulna,rms] = Soder(ai,pi); % From ulnar CCS to GCS
41 Ulnar_coor_GCS = Coord_transfer(Ulnar_coor_CCS,Tgc_ulna); %Contains
   Styloid and Markers
42 Ulnar_Coor_Humero_Radial_ACS = Coord_transfer(Ulnar_coor_GCS,Tag);
43 US_in_Humero_Radial_ACS(i,:) = Ulnar_Coor_Humero_Radial_ACS(1,:); %US
44 end
45
46 %Radius-
```

```
for i = 1:m
ai(1:4,:) = Distal_Radius_coor_CCS(4:7,:); %Top to CW3
pi(1,:) = Kinematics_File(i,Kinem_File_ind_Dr(1,1:3));
pi(2,:) = Kinematics_File(i,Kinem_File_ind_Dr(2,1:3));
pi(3,:) = Kinematics_File(i,Kinem_File_ind_Dr(3,1:3));
pi(4,:) = Kinematics_File(i,Kinem_File_ind_Dr(4,1:3));
[Tgc_radial,rms] = Soder(ai,pi); % From radial CCS to GCS
Distal_Radius_coor_GCS =
    Coord_transfer(Distal_Radius_coor_CCS,Tgc_radial); %Landmarks and Markers
Distal_Radius_Coor_Humero_Radial_ACS =
    Coord_transfer(Distal_Radius_coor_GCS,Tag);
RS_in_Humero_Radial_ACS(i,:) = Distal_Radius_Coor_Humero_Radial_ACS(1,:);
    %RS
Center_Distal_End = ((Distal_Radius_Coor_Humero_Radial_ACS(2,:) +
    Distal_Radius_Coor_Humero_Radial_ACS(3,:))./2 ...
+ Distal_Radius_Coor_Humero_Radial_ACS(1,:))./2; %Used later in this
    function
Center_Distal_End_in_Humero_Radial_ACS(i,:) = Center_Distal_End;
end

% figure
% for i =1:m
%
    plot3(US_in_Humero_Radial_ACS(i,1),US_in_Humero_Radial_ACS(i,2),US_in_Hume
    ro_Radial_ACS(i,3))
% end
%
    text(US_in_Humero_Radial_ACS(1,1),US_in_Humero_Radial_ACS(1,2),US_in_Humer
    o_Radial_ACS(1,3),'US')
% hold
% for i =1:m
%
    plot3(RS_in_Humero_Radial_ACS(i,1),RS_in_Humero_Radial_ACS(i,2),RS_in_Hume
    ro_Radial_ACS(i,3),'r','LineStyle',':')
% end

%Establish reference position. The reference position is taken from the
%first frame. This assumes that the experiment was started with the US
%and RS on the top of each other (neutral pro-supination).

init_vector = RS_in_Humero_Radial_ACS(1,:) - US_in_Humero_Radial_ACS(1,:);
    %Initial pos.

pi = 3.14159265358979;
for i =1:m
next_vector = RS_in_Humero_Radial_ACS(i,:) - US_in_Humero_Radial_ACS(1,:);
Beta =acos(dot(init_vector,next_vector)...
/((norm(init_vector)*norm(next_vector))));
Beta_ang = Beta*180/pi;
Beta_ang_vector(i) = Beta_ang;
end
x_starting = RS_in_Humero_Radial_ACS(1,1); %Based on the x-coordinate of
    the RS we can find out
%which way the arm pronated. For left arm, higher x is pronation
for i =1:m
if RS_in_Humero_Radial_ACS(i,1)>x_starting %Pronation, left arm
Beta_ang_vector(i) = -Beta_ang_vector(i); %Pronation negative
```

```
91 end
92 end
93
94 %-------------------------------------------------------------------------
95 %Other way of plotting- using the Internal-External rotation from the
96 %Supination-Pronation Joint CS
97 for i =1:m
98 Beta_PS(i) = Pronation_Supination_Joint(i,3) -
    Pronation_Supination_Joint(1,3); % Laurel did not start at neutral.
99 % Beta_PS(i) = Pronation_Supination_Joint(i,3) - 45;
100 end
101 %Beta_PS and Beta_ang_vector should be similar but the later proved to
    give
102 %smaller angles. Most probably because the line connecting US and RS is
103 %not perpendicular to the ulnar axis.
104
105 %----------------------------------------
106 %Plot results against this angle
107 %
108 % Distraction
109 figure
110 plot(Beta_PS,Distraction)
111 title('Distraction')
112 xlabel('Pronation Supination')
113 ylabel('(mm)')
114
115 %
116 figure
117 plot(Beta_PS,Head_coor_in_Humero_Radial_ACS(:,1))
118 title('Medio-Lateral shift')
119 xlabel('Pronation Supination')
120 ylabel('Medial Lateral')
121
122 figure
123 plot(Beta_PS,Head_coor_in_Humero_Radial_ACS(:,3))
124 title('Proximo-Distal shift')
125 xlabel('Pronation Supination')
126 ylabel('Distal Proximal')
127
128 %-----------------------------------------------
129 % Do the previous for only certain increments of angle
130 Angle_inc=10;
131 first = Beta_PS(1);
132 remember_index(1) = 1;
133 j = 1;
134 for i = 1:m %number of rows
135 diff = abs(abs(Beta_PS(i))-abs(first));
136 if diff >= Angle_inc
137 j = j+1;
138 remember_index(j) = i;
139 first = Beta_PS(i);
140 end
141 end
142
143 p = size(remember_index,2); %Number of columns
144 for i = 1 :p
145 Head_kinematics(i,1) = Beta_PS(remember_index(i));
```

```
Head_kinematics(i,2) = Distraction(remember_index(i));
Head_kinematics(i,3) =
Head_coor_in_Humero_Radial_ACS(remember_index(i),1);
Head_kinematics(i,4) =
Head_coor_in_Humero_Radial_ACS(remember_index(i),3);
end

[Head_kinematics(:,1) Head_kinematics(:,2) Head_kinematics(:,3)
Head_kinematics(:,4)]

fid = fopen(strcat(File_Name,'.prn'),'w');
fprintf(fid,'%5.2f, %5.2f,%5.2f,%5.2f\n', Head_kinematics');
fclose(fid);



%For Radial Heads only
%Plot the difference between the vector v1 connecting the Centroid of the
%Capitulum with the center of the head and the vector v2 connecting
%the connecting the center of the head with the back center
%of the distl radius
if (What_to_Postprocess==3)|(What_to_Postprocess==5) % Radial Head
for i =1:m
v1 = [0 0 0] - Head_coor_in_Humero_Radial_ACS(i,:);
v2 = Head_coor_in_Humero_Radial_ACS(i,:) -
Center_Distal_End_in_Humero_Radial_ACS(i,:);
Alpha =acos(dot(v1,v2)/(norm(v1)*norm(v2)));
Alpha_ang = Alpha*180/pi;
Alpha_ang_vector(i) = Alpha_ang;
end
figure
plot(Beta_PS,Alpha_ang_vector)
title('Radius wobling')
xlabel('Pronation Supination')
ylabel('Angle missalignment')
end


%For implants only
%Plot the difference between the vector v1 connecting the Centroid of the
%Capitulum with the front center of the head (implant) and the vector v2
connecting
%the connecting the front center of the head (implant) with the back
center
%of the implant.
if (What_to_Postprocess==4)|(What_to_Postprocess==6) % Radial Head
implant
for i =1:m
v1 = [0 0 0] - Head_coor_in_Humero_Radial_ACS(i,:);
v2 = Head_coor_in_Humero_Radial_ACS(i,:) -
Back_side_in_Humero_Radial_ACS(i,:);
Alpha =acos(dot(v1,v2)/(norm(v1)*norm(v2)));
Alpha_ang = Alpha*180/pi;
Alpha_ang_vector(i) = Alpha_ang;
end
figure
plot(Beta_PS,Alpha_ang_vector)
```

```
195 title('Implant wobling')
196 xlabel('Pronation Supination')
197 ylabel('Angle missalignment')
198 end
199
200
201
202 dummy_s = 0;
```

P.8 PLOT_INTERSECTION_PERPENDICULAR_TO_FHA.M

```
1 function FHA_export =
    Plot_Intersection_Perpendicular_to_FHA(p,P4_opt,P5_opt,FHA_opt,P1,P4_matri
    x,P5_matrix,...
2 FHA_matrix,What_to_Postprocess,title2,Ulnar_coor_in_Humero_Radial_ACS)
3 %Plot the intersections of FHA with a plane perpendicular to the mean FHA
4 %at either the Radial Head or the Ulnar Styloid (P1 point). The procedure
    is similar
5 %to the procedure derived earlier for projection on the humerus, but the
6 %three points on the humerus are not needed. Instead the P1 point becomes
7 %either the Radial head or the Ulnar Styloid
8 % -Also prints the statistics
9 % -Recognizes (if nargin == 11) whether there are 10 inputs (RH) or 11 (US)
10
11
12 %Imbedded function: "Intersect_function"
13
14 n = FHA_opt; %FHA is the normal of the plane
15
16 global File_Name %Output File
17 global Left_or_Right
18
19 global Q
20 global Important_radii
21 global Head_coor_in_Humero_Radial_ACS %to plot the travel of the RH along
    with the FHA
22
23 %-----------------------------------------------------------------
24 % Time to make decission
25 Connect_points_with_lines = 'N';
26 Plot_average_pron_sup = 'N';
27 Should_FHAs_be_numbered = 'Y';
28 Plot_the_travel_of_RH = 'N';
29 Plot_FHAs_small ='N'; % Plot FHA's small and all one size and color
30 % It was done for a picture for our first paper
31
32 hold on
33 %-----------------------------------------------------------------
34 % Travel of the radial head imported from
    "Radial_Head_on_Capitulum_Elbow3.m"
35 if nargin ~= 11 %Don't plot it at the US
36 if Plot_the_travel_of_RH == 'Y'
```

```
37
    plot(Head_coor_in_Humero_Radial_ACS(:,1),Head_coor_in_Humero_Radial_ACS(:,
    3),'LineWidth',2)
38 end
39 end
40
41 %Parameter t from parametric equation of line x=x4+(x5-x4)*t
42 radius_of_capitulum = Important_radii(1);
43 [P_intersect_matrix,P_inter]
    =Intersect_function(P5_matrix,P4_matrix,P4_opt,P5_opt,P1,n,p);
44
45 %-----------------------------------------------------
46 %Plot the points of intersection in 2D
47 %figure
48 title(title2)
49 if What_to_Postprocess > 1
50 xlabel(' Medial [mm] Lateral ')
51 if Left_or_Right == 'R'
52 xlabel(' Lateral [mm] Medial ')
53 end
54 else
55 xlabel('Y-axis')
56 end
57 ylabel('Posterior [mm] Anterior')
58
59 % The helical axis changes direction with supination and pronation. Since
60 % the T matrix is supplied in humeral ACS, at 90 degrees of flexion, the
61 % main component of helical axis is on y-axis.
62 % Left arm: Possitive y (FHA_matrix(i,2)) means pronation
63 % Right elbow: Negative y (FHA_matrix(i,2)) means pronation
64 % It also separates the P4 and P5 points into pronation and supination to
65 % see if there was any difference
66 j=1; %for pronation
67 k=1; %for supination
68 axis equal % Square will be square, circle will be circle
69
70 if nargin ~= 11 %If it is not the Ulnar styloid
71 center=[0,0]; %Origin of the Capitellum
72 NOP = 50; %Number of points on the circle
73 pi = 3.14159265358979;
74 THETA=linspace(0,2*pi,NOP);
75 RHO=ones(1,NOP)*radius_of_capitulum;
76 [X,Y] = pol2cart(THETA,RHO);
77 X=X+center(1);
78 H=plot(X,Y,'k-','LineWidth',3); %Plot the Capitellum as a circle
79 end
80
81 P5_sup = zeros(1,3); P4_sup = zeros(1,3); %Dummies, just in case they
    don't exist
82 P5_pron = zeros(1,3); P4_pron = zeros(1,3);
83
84 if Left_or_Right == 'L' %Right-hand rule, for left arm, positive y is
    pronation.
85 for i = 1:(p-1)
86 if sign(FHA_matrix(i,2)) == 1
87 if Plot_FHAs_small == 'Y'
```

```
88
    plot(P_intersect_matrix(i,1),P_intersect_matrix(i,3),'Marker','*','Color',
    'k','MarkerSize',4)
89 else
90 plot(P_intersect_matrix(i,1),P_intersect_matrix(i,3),'bo')
91 end
92 cislo = num2str(i);
93 if Should_FHAs_be_numbered == 'Y'
94 text(P_intersect_matrix(i,1),P_intersect_matrix(i,3),cislo)
95 end
96 P5_pron(j,:) = P5_matrix(i,:); %Separate them into pronation and
    supination
97 P4_pron(j,:) = P4_matrix(i,:);
98 j=j+1;
99 else
100 if Plot_FHAs_small == 'Y'
101
    plot(P_intersect_matrix(i,1),P_intersect_matrix(i,3),'Marker','*','Color',
    'k','MarkerSize',4)
102 else
103 plot(P_intersect_matrix(i,1),P_intersect_matrix(i,3),'r*')
104 end
105 cislo = num2str(i);
106 if Should_FHAs_be_numbered == 'Y'
107 text(P_intersect_matrix(i,1),P_intersect_matrix(i,3),cislo)
108 end
109 P5_sup(k,:) = P5_matrix(i,:); %Separate them into pronation and
    supination
110 P4_sup(k,:) = P4_matrix(i,:);
111 k=k+1;
112 end
113 end
114 if Connect_points_with_lines == 'Y'
115 plot(P_intersect_matrix(:,1),P_intersect_matrix(:,3)) %Connect points
116 end
117 end
118 if Left_or_Right == 'R' %Right hand, positive y is supination.
119 for i = 1:(p-1)
120 if sign(FHA_matrix(i,2)) == 1
121 if Plot_FHAs_small == 'Y'
122
    plot(P_intersect_matrix(i,1),P_intersect_matrix(i,3),'Marker','*','Color',
    'k','MarkerSize',4)
123 else
124 plot(P_intersect_matrix(i,1),P_intersect_matrix(i,3),'r*')
125 end
126 cislo = num2str(i);
127 if Should_FHAs_be_numbered == 'Y'
128 text(P_intersect_matrix(i,1),P_intersect_matrix(i,3),cislo)
129 end
130 P5_sup(k,:) = P5_matrix(i,:); %Separate them into pronation and
    supination
131 P4_sup(k,:) = P4_matrix(i,:);
132 k=k+1;
133 else
134 if Plot_FHAs_small == 'Y'
```

```
135
    plot(P_intersect_matrix(i,1),P_intersect_matrix(i,3),'Marker','*','Color',
    'k','MarkerSize',4)
136 else
137 plot(P_intersect_matrix(i,1),P_intersect_matrix(i,3),'bo')
138 end
139 cislo = num2str(i);
140 if Should_FHAs_be_numbered == 'Y'
141 text(P_intersect_matrix(i,1),P_intersect_matrix(i,3),cislo)
142 end
143 P5_pron(j,:) = P5_matrix(i,:); %Separate them into pronation and
    supination
144 P4_pron(j,:) = P4_matrix(i,:);
145 j=j+1;
146 end
147 end
148 if Connect_points_with_lines == 'Y'
149 plot(P_intersect_matrix(:,1),P_intersect_matrix(:,3)) %Connect points
150 end
151 end
152
153 %-----------------------------------------------------------------
154 % Find average helical axis projections for supination and pronation
155 %Averaging of P5 and P4 points to see if there is any difference
156 n_rows_pron = size(P5_pron,1);
157 n_rows_sup = size(P5_sup,1);
158 P5_ave_sup = sum(P5_sup)./n_rows_sup;
159 P4_ave_sup = sum(P4_sup)./n_rows_sup;
160 P5_ave_pron = sum(P5_pron)./n_rows_pron;
161 P4_ave_pron = sum(P4_pron)./n_rows_pron;
162 %P_inter_sup and P_inter_pron should be the same as selected rows of
    P_intersect_matrix
163
164 if n_rows_sup ~= 1 %No supination, just the dummy, skip
165 p =n_rows_sup+1;
166 [P_inter_mat,P_inter_sup]
    =Intersect_function(P5_sup,P4_sup,P4_ave_sup',P5_ave_sup',P1,n,p);
167 if Plot_average_pron_sup == 'Y'
168
    plot(P_inter_sup(1,1),P_inter_sup(3,1),'Marker','*','Color','r','MarkerSiz
    e',15) %
169 end
170 end
171 if n_rows_pron ~= 1 %No pronation, just the dummy, skip
172 p =n_rows_pron+1;
173 [P_inter_mat,P_inter_pron]
    =Intersect_function(P5_pron,P4_pron,P4_ave_pron',P5_ave_pron',P1,n,p);
174 if Plot_average_pron_sup == 'Y'
175
    plot(P_inter_pron(1,1),P_inter_pron(3,1),'Marker','o','Color','b','MarkerS
    ize',15) %
176 end
177 end
178 end
179
180 % Plot the mean FHA as magenda star and Ulnar Styloid as green circle
```

```
181
    plot(P_inter(1,1),P_inter(3,1),'Marker','p','Color','m','MarkerSize',20,'M
    arkerFaceColor','b') %Mean FHA
182
183 if nargin == 11 %If it is at the Ulnar styloid
184
    plot(Ulnar_coor_in_Humero_Radial_ACS(1,1),Ulnar_coor_in_Humero_Radial_ACS(
    1,3),...
185 'Marker','o','Color','g','MarkerSize',8,'MarkerFaceColor','c')
186 end
187 set(gca,'Xdir','reverse') %negative x to the right, doesn't matter
    whether
188 %the elbow is left or right
189 v=axis;
190
191 %--------------------------------------------------------------------
192 %Custom Legend (Writes legend out of the way for plotted stuff)
193 yrange = abs(v(4)-v(3)); xrange = abs(v(2)-v(1));
194 xadd = xrange/3; % add it so that the custom legend won't interfere with
    points
195 axis([v(1) v(2)+xadd v(3) v(4)])
196 text(v(2)+2*xrange/12,v(4)-yrange/20,'Mean FHA') %Upper right corner
197 plot(v(2)+6*xrange/24,v(4)-
    yrange/20,'Marker','p','Color','m','MarkerSize',10)
198 x1 = num2str(P_inter(1,1)); y1 = num2str(P_inter(3,1));
199 x_mean=['at x=' x1]; y_mean=[' z=' y1];
200 text(v(2)+2*xrange/12,v(4)-yrange/10,x_mean); text(v(2)+2*xrange/12,v(4)-
    yrange/7,y_mean)
201 if Plot_FHAs_small == 'N';
202 plot(v(2)+6*xrange/24,v(4)-yrange/3,'bo')
203 text(v(2)+2*xrange/12,v(4)-yrange/3,'Pronation')
204 plot(v(2)+6*xrange/24,v(4)-yrange/2.5,'r*')
205 text(v(2)+2*xrange/12,v(4)-yrange/2.5,'Supination')
206 end
207 if nargin == 11
208 text(v(2)+2*xrange/12,v(4)-yrange/5,'US') %Upper right corner
209 plot(v(2)+6*xrange/24,v(4)-
    yrange/5,'Marker','o','Color','g','MarkerSize',8)
210 x1 = num2str(Ulnar_coor_in_Humero_Radial_ACS(1,1)); y1 =
    num2str(Ulnar_coor_in_Humero_Radial_ACS(1,3));
211 x_mean=['at x=' x1]; y_mean=[' z=' y1];
212 text(v(2)+2*xrange/12,v(4)-yrange/4,x_mean); text(v(2)+2*xrange/12,v(4)-
    yrange/3.5,y_mean)
213 end
214 % End of Custom Legend
215 hold off
216
217 % --------------------------------------------------------------------
218 % Statistics in the plane of the cut (print to a file)
219 %
220 fid = fopen(strcat(File_Name,'.out'),'a'); %Open the file and append data
221
222 %Spread index, Duck, 2001 PRUJ5, 47th ORS, pg 0761, online
223 % No longer used. It didn't seem to work properly, when switched to RH
    for
224 % FHA calculating it was still showing higher rms at the Capitulum.
225 % SDx = 0; SDy = 0; SDz = 0;
```

```
% for i = 1:(p-1)
% SDx = SDx + abs(P_inter(1,1)-P_intersect_matrix(i,1));
% SDy = SDy + abs(P_inter(2,1)-P_intersect_matrix(i,2));
% SDz = SDz + abs(P_inter(3,1)-P_intersect_matrix(i,3));
% end
% SDx = SDx/(p-1); SDy = SDy/(p-1); SDz = SDz/(p-1);
% SI=sqrt(SDx^2+SDy^2+SDz^2);

dsum = 0; %
for i = 1:(p-1)
dsum = dsum + (P_inter-P_intersect_matrix(i,:)')'*Q(:,:,i)*(P_inter-
P_intersect_matrix(i,:)');
end
d_eff_new = sqrt(dsum/(p-1)); %Units in mm, Equation 38c from Woltring

%Distance between the mean FHA and Capitellum
if nargin ~= 11
x1=num2str(abs(P_inter(1,1)));
x2=num2str(abs(P_inter(3,1)));
%disp('Location of the mean FHA at the Capitellum')
str1 = 'The mean FHA ';
if sign(P_inter(1,1)) == 1
text1=['The mean FHA is ' x1 ' mm medially'];
str2=[' is ' x1 ' mm medially'];
else
text1=['The mean FHA is ' x1 ' mm lateraly'];
str2=[' is ' x1 ' mm laterally'];
end
if sign(P_inter(3,1)) == 1
text1=['The mean FHA is ' x2 ' mm anteriorly'];
str3=[' and ' x2 ' mm anteriorly'];
else
text1=['The mean FHA is ' x2 ' mm posteriorly'];
str3=[' and ' x2 ' mm posteriorly'];
end
total_d=sqrt(P_inter(1,1)^2+P_inter(3,1)^2);
td=num2str(total_d); %SI_s=num2str(SI);
str_f=strcat(str1,str2,str3);
d_eff_s = num2str(d_eff_new);
str4= ['at total distance of ' td ' mm and deff of ' d_eff_s ' mm'];
fprintf(fid,'-The mean FHA and Capitellum:\n');
fprintf(fid,' %s\n',str_f');
fprintf(fid,' %s\n',str4');
fprintf(fid,' from the Capitellum.\n');
end

%Distance between the mean FHA and US
if nargin == 11
x1=num2str(abs(P_inter(1,1)-Ulnar_coor_in_Humero_Radial_ACS(1,1)));
x2=num2str(abs(P_inter(3,1)-Ulnar_coor_in_Humero_Radial_ACS(1,3)));
%disp('mean FHA location at the US')
str1 = 'The mean FHA ';
if Ulnar_coor_in_Humero_Radial_ACS(1,1) < P_inter(1,1)
text1=['The mean FHA is ' x1 ' mm medially'];
str2=[' is ' x1 ' mm medially'];
else
text1=['The mean FHA is ' x1 ' mm lateraly'];
```

```
str2=[' is ' x1 ' mm laterally'];
end
if Ulnar_coor_in_Humero_Radial_ACS(1,3) < P_inter(3,1)
text1=['The mean FHA is ' x2 ' mm anteriorly'];
str3=[' and ' x2 ' mm anteriorly'];
else
text1=['The mean FHA is ' x2 ' mm posteriorly'];
str3=[' and ' x2 ' mm posteriorly'];
end
total_d=sqrt((P_inter(1,1)-Ulnar_coor_in_Humero_Radial_ACS(1,1))^2+...
+(P_inter(3,1)-Ulnar_coor_in_Humero_Radial_ACS(1,3))^2);
td=num2str(total_d); %SI_s=num2str(SI);
str_f=strcat(str1,str2,str3);
d_eff_s = num2str(d_eff_new);
str4= ['at total distance of ' td ' mm and deff of ' d_eff_s ' mm'];
fprintf(fid,'-The mean FHA and US:\n');
fprintf(fid,' %s\n',str_f');
fprintf(fid,' %s\n',str4');
fprintf(fid,' from the US.\n');
end

t=0; %dummy

fclose(fid); %Close the output file

%-----------------------------------------------------------------
% Statistics done new way, easy to export to EXCEL

%At the Capitellum
% Medial and Proximal will have positive signs even thought the the x-
  axis
% of the humeral ACS is always positive to the left.
if nargin ~= 11
proxi_distal = P_inter(3,1);
med_lat = P_inter(1,1);
if n_rows_sup ~= 1 %No supination, just the dummy, skip
med_sup = P_inter_sup(1,1)-P_inter(1,1);%Substract mean FHA from
  P_iner_sup
else
med_sup =0;
end
if n_rows_pron ~= 1 %No pronation, just the dummy, skip
med_pron = P_inter_pron(1,1)-P_inter(1,1);
%med_sup = P_inter_sup(1,1) %for easier statistics, see note below
%med_pron = P_inter_pron(1,1) %for easier statistics, see note below
else
med_pron = 0;
end
if Left_or_Right == 'R'
med_lat = -P_inter(1,1);
med_sup = -med_sup;
med_pron = -med_pron;
end
total_d=sqrt(P_inter(1,1)^2+P_inter(3,1)^2);
end

if nargin == 11 %Ulnar Styloid
```

```
med_lat = (P_inter(1,1)-Ulnar_coor_in_Humero_Radial_ACS(1,1));
proxi_distal = (P_inter(3,1)-Ulnar_coor_in_Humero_Radial_ACS(1,3));
if n_rows_sup ~= 1 %No supination, just the dummy, skip
med_sup = P_inter_sup(1,1)-P_inter(1,1);
else
med_sup =0;
end
if n_rows_pron ~= 1 %No pronation, just the dummy, skip
med_pron = P_inter_pron(1,1)-P_inter(1,1);
%med_sup = P_inter_sup(1,1)-Ulnar_coor_in_Humero_Radial_ACS(1,1) for easier statistics, see note below
%med_pron = P_inter_sup(1,1)-Ulnar_coor_in_Humero_Radial_ACS(1,1) for easier statistics, see note below
else
med_pron = 0;
end
if Left_or_Right == 'R'
med_lat = -med_lat;
med_sup = -med_sup;
med_pron = -med_pron;
end
total_d=sqrt(proxi_distal^2+med_lat^2);
end

% Export this info to the "calculate_Screw_axis1.m"

% The med_lat and proxi_distal are for average FHA.
% The med_sup and med_pron are basically med_lat for an average pronation
% or supination FHA. Note, I didn't calculate this for proximo-distal
% direction. The med_sup and med_pron are calculated with respect to the
% average FHA at Capitellum or Ulnar styloid. For statistics I had to recalculate
% this (in Excel, "Helical axis, exact results") so that it was calculated
% with respect to the Capitellum or Ulnar Styloid, otherways I wouldn't be
% able to do the statistics
FHA_export =[med_lat,proxi_distal,total_d,d_eff_new,med_sup,med_pron];

function [P_intersect_matrix,P_inter] =Intersect_function(P5_matrix,P4_matrix,P4_opt,P5_opt,P1,n,p)
% Calculates intersection of helical axis with predefined points in the
% space (Capitellum, Ulnar styloid) and is perpendicular to the mean FHA
for i = 1:(p-1)
P5 = P5_matrix(i,:);
P4 = P4_matrix(i,:);
t = dot(n,(P1-P4))/dot(n,(P5-P4)); %Equation [6] on my derivation sheet
%t>0 and t<1 : The intersection occurs between the two points
%t=0 :The intersection falls on the first point
%t=1 :The intersection falls on the second point
%t>1 :The intersection occurs beyond second point
%t<0 :The intersection occurs before first point
P_intersect = P4+(P5-P4).*t; %Equation [1] on my derivation sheet
P_intersect_matrix(i,:) = P_intersect;
end

%Mean Helical axis
```

```
388 t = dot(n,(P1-P4_opt'))/dot(n,(P5_opt'-P4_opt')); %Equation [6] on my
    derivation sheet
389 P_inter = P4_opt +(P5_opt -P4_opt ).*t; %Equation [1] on my derivation
    sheet
```

P.9 PLOT_ROTATIONS_DISPLACEMENTS.M

```
1 function blaa = Plot_Rotations_Displacements(xplot,Joint,Title_text);
2 % Plots Rotations and Displacements
3
4 figure
5 SUBPLOT(3,1,1), plot(xplot,Joint(:,1))
6 title_T=strcat('Rotations in',Title_text);
7 title(title_T)
8 ylabel('Degrees'), legend('Flexion-Extension')
9 SUBPLOT(3,1,2), plot(xplot,Joint(:,2))
10 ylabel('Degrees'), legend('Valgus-Varus')
11 SUBPLOT(3,1,3), plot(xplot,Joint(:,3))
12 xlabel('Number of Frames'), ylabel('Degrees'), legend('Ext-Int Rotation')
13
14 figure
15 SUBPLOT(3,1,1), plot(xplot,Joint(:,4))
16
17 title_T=strcat('Displacements in',Title_text);
18 title(title_T)
19 ylabel('mm'), legend('Shift')
20 SUBPLOT(3,1,2), plot(xplot,Joint(:,5))
21 ylabel('mm'), legend('Glide')
22 SUBPLOT(3,1,3), plot(xplot,Joint(:,6))
23 xlabel('Number of Frames'), ylabel('mm'), legend('Distraction')
24
25 blaa=1;
```

P.10 PS_POT_SYNC.M

```
1 % Laurel Kuxhaus, lck4@pitt.edu, laurel.k@gmail.com
2 % 27 November 2007
3 % Orthopaedic Biomechanics Laboratory, Allegheny General Hospital and
4 % University of Pittsburgh
5
6 % This code is based on MA_sync.m
7 % It will read in the potentiometer data, sync it with the motion file
8 %
9 % this code ASSUMES that the p/s sampling frequency will be both higher
10 % than the radial head sampling frequency, *and* an even multiple thereof.
11 % This code also ASSUMES that the p/s angle is already in degrees.
12 % p/s file is CSV.
13
```

```
14 % inputs needed:
15 % ps_file: This will be a .csv file that has the sync signal in the first
16 % column and the p/s angle from the potentiometer in the 2nd. (If these
17 % change, will need to modify code indicated below.)
18 % ps_samp: The sampling frequency of the p/s data. ** It is assumed that
19 % this is faster than the sampling frequency of the motion data!**
20 % radhd_coords: The radial head coordinates, as calculated in
21 % LaurelsProgram.
22 % trajectories: The big motion data file that has the motion sync signal
    in it.
23 %
24 % radhd_sync_column: The column of "trajectories" that contains the sync
    signal.
25 %
26 % radhd_samp: The sampling frequency of the motion data.
27 %
28 % degree_increment: the interval (in degrees) that you want for the output
29 % file.
30 % filename: The name of the output file. This will preferrably end in
    ".xls".
31 %
32 % outputs will be:
33 % radhd_coords_final = the final (truncated, trimmed, to-be-exported)
34 % radial head coordinates.
35 % radhd_sync_final - the final (trimmed) sync signal from the radial head.
36 % ps_final - the final (to-be-exported) p/s positions.
37
38
39
40 function [radhd_coords_final, radhd_sync_final,
    ps_final]=ps_pot_sync(ps_file, ps_samp, radhd_coords, trajectories,
    radhd_sync_column, radhd_sa
mp, degree_increment, filename)
41
42
43 threshold=10.5; % May need to change this later.
44 ps_angle_column=2; % if the p/s angle is not in the 2nd column, will need
    to adjust this.
45 ps_sync_column=1; % if the p/s sync signal is not in the 1st column, will
    need to change this.
46
47 % Step 1: find Sync Signal in p/s data.
48 ps_data=csvread(ps_file);
49
50 for ii=1:length(ps_data)
51 if ps_data(ii,1)<=threshold
52 ps_hiorlow(ii)=1;
53 else
54 ps_hiorlow(ii)=0;
55 end % end if
56 end % end for
57
58 % find difference btwn hiorlow(ii) and hiorlow(ii-1) - it will be 1 when
59 % the signal goes up, and 0 when it goes down.
60
61 for jj=2:length(ps_hiorlow)
62 ps_diff(jj)=ps_hiorlow(jj)-ps_hiorlow(jj-1);
```

```
63 end % end for
64
65 kk=0; ll=1;
66 while kk==0
67 if ps_diff(ll)==0
68 kk==0;
69 ll=ll+1;
70 else kk=1;
71 end % end if
72 end % end while
73
74
75 % step 1b: throw out everything before the "high" of the signal.
76 ll;
77 ps=ps_data(ll:end, ps_angle_column);
78
79 % Step 2: find Sync Signal in radial head data.
80 radhd_sync_data=trajectories(:, radhd_sync_column);
81
82 for mm=1:length(radhd_sync_data)
83 if radhd_sync_data(mm)<=threshold
84 radhd_sync_hiorlow(mm)=1;
85 else
86 radhd_sync_hiorlow(mm)=0;
87 end % end if
88 end % end for
89
90 % find difference btwn hiorlow(ii) and hiorlow(ii-1) - it will be 1 when
91 % the signal goes up, and 0 when it goes down.
92
93 for nn=2:length(radhd_sync_hiorlow)
94 radhd_sync_diff(nn)=radhd_sync_hiorlow(nn)-radhd_sync_hiorlow(nn-1);
95 end % end for
96
97 oo=0; pp=1;
98 while oo==0
99 if radhd_sync_diff(pp)==0 %& pp<length(radhd_sync_data)
100 % if pp<length(radhd_sync_data)
101 oo==0;
102 pp=pp+1;
103 % end % end pp if
104 else
105 oo=1;
106 end % end if
107 end % end while
108
109 % step 2b: throw out everything before.
110 pp;
111 radhd_sync=radhd_sync_data(pp:end,:); % pp was oo.
112 % repeat for the radial head travel coordinates
113 radhd_coords=radhd_coords(pp:end,:);
114
115 % step 3: downsample everything to lowest frequency, which will be the
    motion data, radhd_samp.
116 size(ps);
117 ps_down=downsample_LK(ps_samp/radhd_samp, ps);
118
```

```
% Step 4: trim off any extra at the ends.

datalength=min([length(radhd_sync), length(ps_down)]);
radhd_coords_final=radhd_coords(1:datalength,:);
radhd_sync_final=radhd_sync(1:datalength,:);
ps_final=ps_down(1:datalength,:);

% step 5: make plots
figure;
subplot(3,1,1)
title('travel vs. p/s angle')
hold on;
xlabel('p/s angle (degrees)');
plot(ps_final, radhd_coords_final(:,1));
ylabel('m/l travel (mm)');

subplot(3,1,2)
hold on;
xlabel('p/s angle (degrees)');
plot(ps_final, radhd_coords_final(:,2));
ylabel('distraction (mm)');

subplot(3,1,3)
hold on;
xlabel('p/s angle (degrees)');
plot(ps_final, radhd_coords_final(:,3));
ylabel('a/p travel (mm)');

% step 6: make output file.

%degree_increment, filename

%% stuff below is based on the end of "Radial_Head_on_Capitulum3", a
   program
%% by Karol Galik and modified by Laurel Kuxhaus

first = round(ps_final(1)/degree_increment)*degree_increment;
rounded_angle(1)=first; % see if that helps.
remember_index(1) = 1;
j = 1;
for i = 1:length(ps_final) %number of rows
diff = abs(ps_final(i)-first);
if diff >= degree_increment
j = j+1;
remember_index(j) = i;
if j>1
first = round(ps_final(i)/degree_increment)*degree_increment;
end
rounded_angle(j) = first;
end
end
size(ps_final)
ps_final
size(rounded_angle)
p = size(rounded_angle,2);
for i = 1 :p
Head_kinematics(i,1) = rounded_angle(i);
```

```
175 Head_kinematics(i,2) = radhd_coords(remember_index(i),1);
176 Head_kinematics(i,3) = radhd_coords(remember_index(i),2);
177 Head_kinematics(i,4) = radhd_coords(remember_index(i),3);
178
179 end
180
181 % Laurel likes to use dlmwrite.
182 dlmwrite(filename, Head_kinematics,'\t')
```

P.11 RADIAL_HEAD_ON_CAPITULUM3.M

```
1 function
    [Distraction,Head_coor_in_Humero_Radial_ACS,Back_side_in_Humero_Radial_ACS
    ] = ...
2
    Radial_Head_on_Capitulum_Elbow3(Radial_Head_or_Implant_coor_CCS,Important_
    radii,figure_title,...
3 What_to_Postprocess)
4
5 % This function will plot the motion of the Radial Head on Capitulum
6 % Output variables "Distraction" , "Head_coor_in_Humero_Radial_ACS"
7 % ,"Back_side_in_Humero_Radial_ACS" and the
    "Center_Distal_End_in_Humero_Radial_ACS"
8 %are used in "Plot_against_Styloids.m"
9
10 global Tag_humero_radial
11 global PronSup
12 global Kinem_File_rh_i
13 global balls
14 global Left_or_Right
15
16 Radius_of_Capitulum = Important_radii(1);
17
18 % Find the transformation matrix from radial CCS to GCS
19
20 m = size(PronSup,1);
21 n_points = m-1 %30; %Number of points to postprocess
22 interval = floor(m/n_points);
23 pocet = 1;
24
25 for i = 1:(n_points+1)
26 [ai,pi] = three_or_four_balls(balls(4,:),Kinem_File_rh_i,1,
    Radial_Head_or_Implant_coor_CCS,PronSup(pocet,:));
27 [Tgc_Rad_Head_or_Implant,rms] = Soder(ai,pi); % From CCS to GCS
28 Radial_Head_or_Implant_coor_GCS =
    Coord_transfer(Radial_Head_or_Implant_coor_CCS,...
29 Tgc_Rad_Head_or_Implant);
30 Head(i,:) = Radial_Head_or_Implant_coor_GCS(1,:); %Only radial head or
    implant
31 if (What_to_Postprocess==4)|(What_to_Postprocess==6) % Radial Head implant
32 Back_side(i,:) = Radial_Head_or_Implant_coor_GCS(7,:);
33 end
```

```
pocet = 1+i*interval;
end

%needed for "Plot_against_Styloids.m"
Back_side_in_Humero_Radial_ACS=0; %Just return something
Head_coor_in_Humero_Radial_ACS = Coord_transfer(Head,Tag_humero_radial);
if (What_to_Postprocess==4)|(What_to_Postprocess==6) % Radial Head implant
Back_side_in_Humero_Radial_ACS =
 Coord_transfer(Back_side,Tag_humero_radial);
end

Distraction = Head_coor_in_Humero_Radial_ACS(:,2)- Radius_of_Capitulum;

figure
% Figure #4
%Draw a circle representing the Capitellum
center=[0,0]; %Origin of the capitellum
NOP = 50; %Number of points on the circle
pi = 3.14159265358979;
THETA=linspace(0,2*pi,NOP);
RHO=ones(1,NOP)*Radius_of_Capitulum;
[X,Y] = pol2cart(THETA,RHO);
X=X+center(1);
Y=Y+center(2);
H=plot(X,Y,'k-','LineWidth',3);
axis square;
hold on
% When looking form the front- anterior view:
xlabel('Medial [mm] Lateral ')
if Left_or_Right == 'R'
xlabel(' Lateral [mm] Medial ')
% Head_coor_in_Humero_Radial_ACS(:,1) = -
 Head_coor_in_Humero_Radial_ACS(:,1);
end
title('')
% For the medio-lateral motion the Head_coor_in_Humero_Radial_ACS(:,1)
% doesn't have to be inverted if the arm is right because the x axis
 always
% points left no matter whether the arm is left or right.
%This has to be done in "Plot_against_Styloids.m" to have it correctly
%graphed.
ylabel('Posterior [mm] Anterior')
title(figure_title)

 plot(Head_coor_in_Humero_Radial_ACS(:,1),Head_coor_in_Humero_Radial_ACS(:,
 3),'LineWidth',2)
set(gca,'Xdir','reverse')
hold off

X_laurel=Head_coor_in_Humero_Radial_ACS(:,1);
Y_laurel=Head_coor_in_Humero_Radial_ACS(:,3);

save LaurelFilter.mat X_laurel Y_laurel X Y

% Figure #5
% figure(3)
% xplot=1:1:(n_points+1);
```

```
86 % %plot(xplot,Head_coor_in_Humero_Radial_ACS(:,2))
87 % plot(xplot,Distraction)
88 % title('Distraction')
89 % xlabel('Frames')
90
91 % Figure #2 and Figure #3 could be combined to a 3D picture
92 %3D Capitulum
93 %figure
94 % [X,Y,Z] = sphere(30);
95 % X = X.* Radius_of_Capitulum;
96 % Y = Y.* Radius_of_Capitulum;
97 % Z = Z.* Radius_of_Capitulum;
98 %
99 % %surf(X,Y,Z)
100 % view(-173,20); axis tight; grid on;
101 % shading interp
102 % colormap hot
103 % camlight headlight; lighting gouraud;
104 % rc = Radius_of_Capitulum;
105 % axis([-(rc+.25*rc) rc+.25*rc 0 rc+.25*rc -(rc+.25*rc) rc+.25*rc])
106 % hold on
107 %
    %plot3(Head_coor_in_Humero_Radial_ACS(:,1),Head_coor_in_Humero_Radial_ACS(
    :,2),Head_coor_in_Humero_Radial_ACS(:,3))
108 % %arrow(Head_coor_in_Humero_Radial_ACS)
109 % xlabel('Medial [x] Lateral')
110 % ylabel('Radius [y]')
111 % zlabel('Distal Humerus [z] Proximal')
```

P.12 SCREW_AXIS.M

```
1 function [n,phi,t,s] = screw_axis(T,intersect);
2 % Calculation of the screw axis
3 % function [n,point,phi,t]=screw(T)
4 % Input: T matrix containing the rotation matrix and transl. vector
5 % [1 0 0 0;[t1,t2,t3]'; R]
6 % intersect location of the screw axis where it intersects either the x=0
    (intersect=1),
7 % the y=0 (intersect=2), or the z=0 (intersect=3) plane.
8 % default: intersect=3
9 % Output: n unit vector with direction of helical axis
10 % point point on helical axis
11 % phi rotation angle (in deg)
12 % t amount of translation along screw axis
13 %
14 % Comments: Note that phi is between 0 and 180 deg. Right handed screw
15 % axis system. The "sign" of phi can be checked with direction
16 % of the unit vector (n).
17 % References: (1) Spoor and Veldpaus (1980) Rigid body motion calculated
18 % from spatial co-ordinates of markers.
19 % J Biomech 13: 391-393
20 % (2) Berme, Cappozzo, and Meglan. Rigid body mechanics
```

```
% as applied to human movement studies. In Berme and
% Cappozzo: Biomechanics of human movement.
%
% Author: Christoph Reinschmidt, HPL, UofCalgary
% Date: Oct. 03, 1994
% Last Changes: August 2003 by Karol Galik

if nargin == 1, intersect = [3]; end

R = T(2:4,2:4); %Rotation matrix extracted from the whole transformation
 matrix

% tmp is matrix in equ. 31 (Spoor and Veldpaus, 1980)
tmp = [R(3,2)-R(2,3);R(1,3)-R(3,1);R(2,1)-R(1,2)];

%Calculating n using equ. 31 and 32 (Spoor and Veldpaus, 1980)
n = tmp/norm(tmp);

% Calculating phi either with equ. 32 or 34 (Spoor and Veldpaus, 1980)
% depending on wheter sin(phi) is smaller or bigger than 0.5*SQRT(2)
if norm(tmp) <= sqrt(2)
phi = (asin(0.5*norm(tmp)))*180/pi;
else phi = (acos(0.5*(R(1,1)+R(2,2)+R(3,3)-1)))*180/pi;
end

%If phi approaches 180 deg it is better to use the following:


%(see Spoor and Veldpaus Eq. 35,36)
% Very large rotations-most probably won't happen
if phi > 135;
b = [0.5*(R+R')-cos(phi/180*pi) * eye(3)];
b1 = [b(:,1)]; b2=[b(:,2)]; b3=[b(:,3)];
btmp = [b1'*b1;b2'*b2;b3'*b3];
[bmax,i] = max(btmp);
n = b(:,i)/sqrt(bmax);
if sign(R(3,2)-R(2,3)) ~= sign(n(1,1)); n=n.*(-1); end;
end

t = n'*T(2:4,1); %Spoor and Veldpaus, Equation 37
% Radius vector of a point on the helical axis. (eq. 38)
% n ans s are orthogonal

phi_rad = phi*pi/180;
ss = cross(n,T(2:4,1));
s = -0.5*cross(n,ss) + sin(phi_rad)/(2*(1-cos(phi_rad)))*ss;

if phi == 0 %No Rotation, only translation (Spoor and Veldpaus, Equation
 39)
display('Only translation, no rotation of the screw axis')
t = sqrt(T(2:4,1)'*T(2:4,1));
s = 0;
%n = T(2:4,1)./t;
end

% Calculate where the screw axis intersects the plane as defined in
 'intersect'
```

```
75 %Q=R-eye(3);
76 %Q(:,intersect)=-n;
77 %point=Q\[T(2:4,1).*[-1]];
78 %point(intersect,1)=[0];
```

P.13 THREE_OR_FOUR_BALLS.M

```
1 function [ai,pi] =
    three_or_four_balls(balls_ind,Kinem_File_u,off_set,balls_coor,Kin_file);
2
3 % Input: balls_ind - Indicis of balls for a particular marker array
4 % Kinem_File_u- Row numbers from Peak cameras
5 % off_set - Offset number from pointing. Each marker array has different
6 %sequence of balls from pointing
7 % balls_coor - coordinates of balls from pointing on CMM
8 % Kin_File - coordinates of balls from Peaks
9
10 Kinem_File_ind(1,:) = Kinem_File_u(balls_ind(1),:);
11 Kinem_File_ind(2,:) = Kinem_File_u(balls_ind(2),:);
12 Kinem_File_ind(3,:) = Kinem_File_u(balls_ind(3),:);
13 ai(1,:) = balls_coor(balls_ind(1)+off_set,:);
14 ai(2,:) = balls_coor(balls_ind(2)+off_set,:);
15 ai(3,:) = balls_coor(balls_ind(3)+off_set,:);
16 if balls_ind(1,4) ~= 0
17 Kinem_File_ind(4,:) = Kinem_File_u(balls_ind(4),:);
18 ai(4,:) = balls_coor(balls_ind(4)+off_set,:);
19 end
20
21 pi(1,:) = Kin_file(:,Kinem_File_ind(1,1:3));
22 pi(2,:) = Kin_file(:,Kinem_File_ind(2,1:3));
23 pi(3,:) = Kin_file(:,Kinem_File_ind(3,1:3));
24 if balls_ind(1,4) ~= 0
25 pi(4,:) = Kin_file(:,Kinem_File_ind(4,1:3));
26 end
```

APPENDIX Q

ADDITIONAL TEST AND RESULTS FROM THE SPICATEK SYSTEM

Q.1 METHODS

Following the presentation of the results shown in Chapter 9.0 , additional tests were conducted to evaluate the suitability of the Spicatek system. The same test apparatus was used with some modifications. To address the issue of glare in the larger markers and to permit automatic tracking of the larger markers, the large 9 mm markers were replaced with smaller 3.97 mm markers. The divots and smaller spherical markers remained unchanged. The updated apparatus is shown in Figure 76. The small balls used to simulate the native radial head, and the divots in the implant, remained exactly as before. Additionally, a white felt sheet was hung in the background of the test apparatus to enhance the contrast between the markers and the background. Care was taken to focus the cameras well prior to camera calibration at the start of the tests.

As in Chapter 8.0 , static, translational, and rotational trials were collected for both the Implant and Ring (native head) cases. The cluster of markers to simulate the native radial head was not used. For the translational trials, the direction of motion was in the plane of the capitellum for the Ring case, and perpendicular to this plane for the Implant case. These

directions are indicated below in Figure 76. All trials to simulate the Implant case were collected first, followed by repositioning of the apparatus for the Ring trials. In between the Implant and Ring trials, the focus on one of the cameras was adjusted to ensure a clean image of the repositioned apparatus. Care was taken to do this without moving the camera's position. It is also important to note that the calibration frame was used without the presence of one marker, and other markers had previously been reattached without measurement verification.

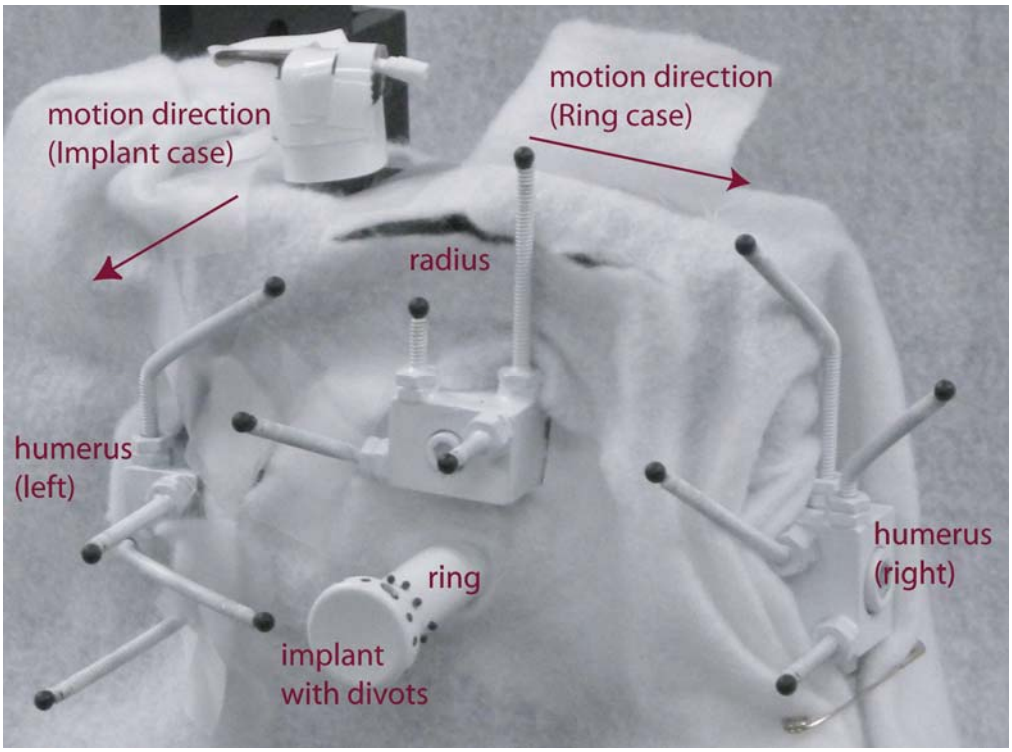


Figure 76: Revised test apparatus for Spicatek tests.

Three trials of each type were captured. The results were processed as described in Chapter 8.0 , and are presented below. The cutoff frequency for the filtered results was again 1

Hz. The speed of the movements was similar to that described in Chapter 8.0 – translations of about 1 mm occurred over about 2 seconds, and monotonic rotations of about 9° occurred over 3 seconds or more.

Q.2 RESULTS AND DISCUSSION

Q.2.1 General observations

In general, the results collected in this latest evaluation of the Spicatek system indicate improvement over those described in Chapter 9.0 . This is attributed to the increased use of the software's ability to automatically track the larger markers, now 3.7 mm in diameter. However, it is important to note that the tracking was not flawless. In all trials, one or more of the larger markers needed to be manually tracked for one or more frames. The amount of manual tracking required per marker ranged from only a few frames on one camera's images to all frames from one camera and some on the second camera. In general, the Ring trials seemed to require more manual tracking than the Implant trials. This could be due to the position of the apparatus in the calibrated space, or the refocusing of the camera between the Implant and Ring data collection. Furthermore, it was noted during the tracking process that some of the spherical markers created shadows which also appeared black and may have been included in the automatic tracking algorithms' estimation of marker centroid positions.

Q.2.2 Implant results

Representative results for the Implant case are shown in Figure 77, Figure 78, and Figure 79. Figure 77 is from a static trial, using the circle-fitting algorithm to track the position of the implant. The expected result is a point at exactly (0,0). As with the other figures, the resulting radial head path is far from the center of the capitellum, which could be due to the calibration factors mentioned above, or due to the sensitivity of the results to the exact plane of projection, as discussed below in Section Q.2.5. Regardless of the projected position on to the plane of the capitellum, the filtering markedly reduced the amount of noise, indicating that the system may operate at a sufficiently low noise level to track the radial head movements to sub-millimeter accuracy. Figure 78 shows the results from a representative translation trial for the implant case. Note by looking at the components of the travel shown in the lower portion of the figure that the main direction of travel was in the distraction direction, or out of the plane of the capitellum. Thus the majority of the translation is not observable by projection on to this plane. Like the static case, these results suggest that sub-millimeter accuracy is achievable with filtering. In Figure 79, the results from a representative rotation case are shown. If the implant was exactly centered on the capitellum, the expected result is a point centered at (0,0) in capitellar coordinates. As with the static case, the center of the implant is a few millimeters from the center of the capitellum. However, the total motion measured, after filtering, was at the sub-millimeter level. It is possible that the center of the implant and the center of the capitellum were not perfectly aligned which may have contributed to this off-center result, or that the calibration was not sufficiently accurate in this region to permit truly accurate marker tracking. Additionally, the manual tracking of some markers may have influenced these results.

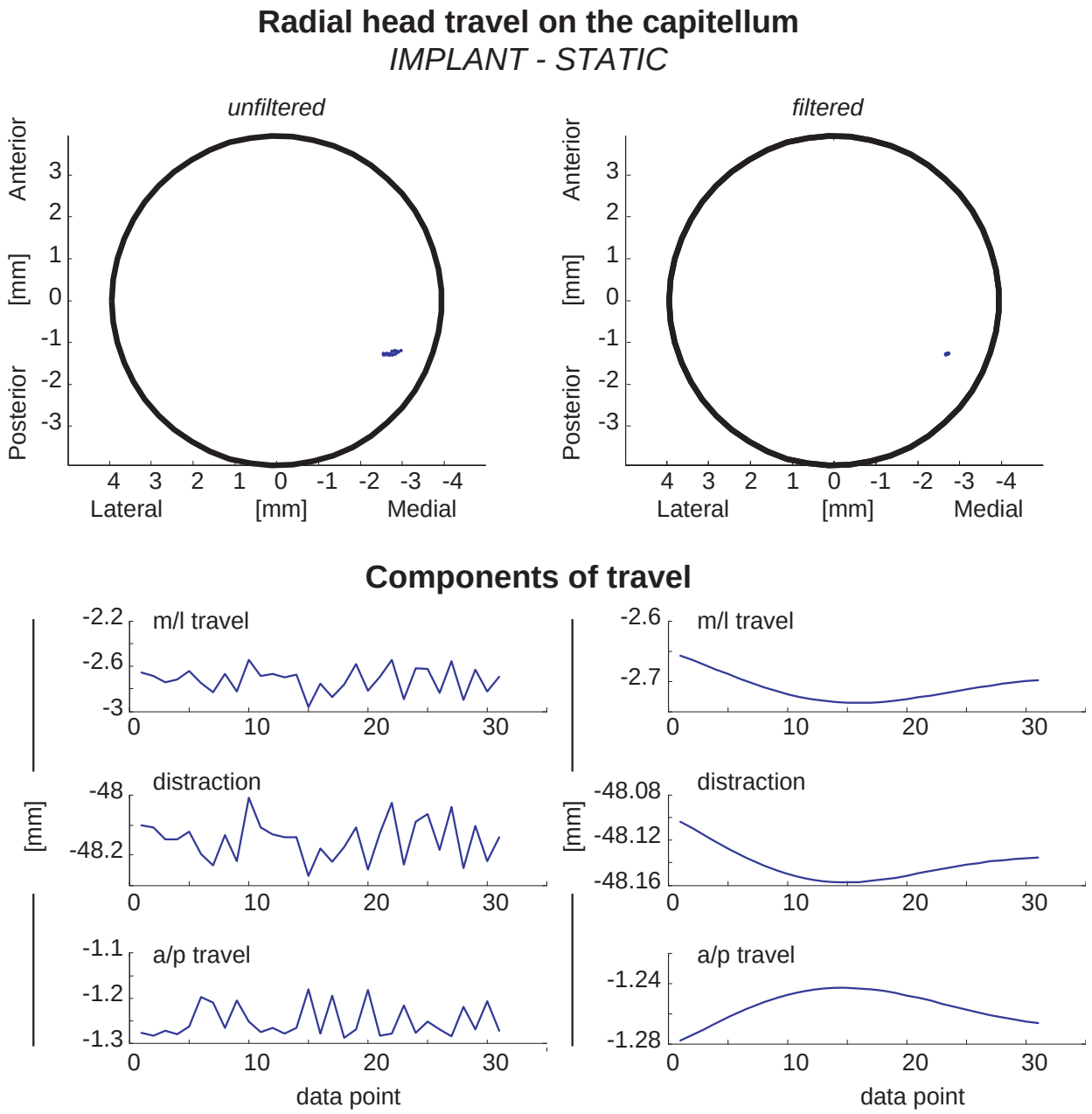


Figure 77: Implant results as collected with the Spicatek system for the static case.

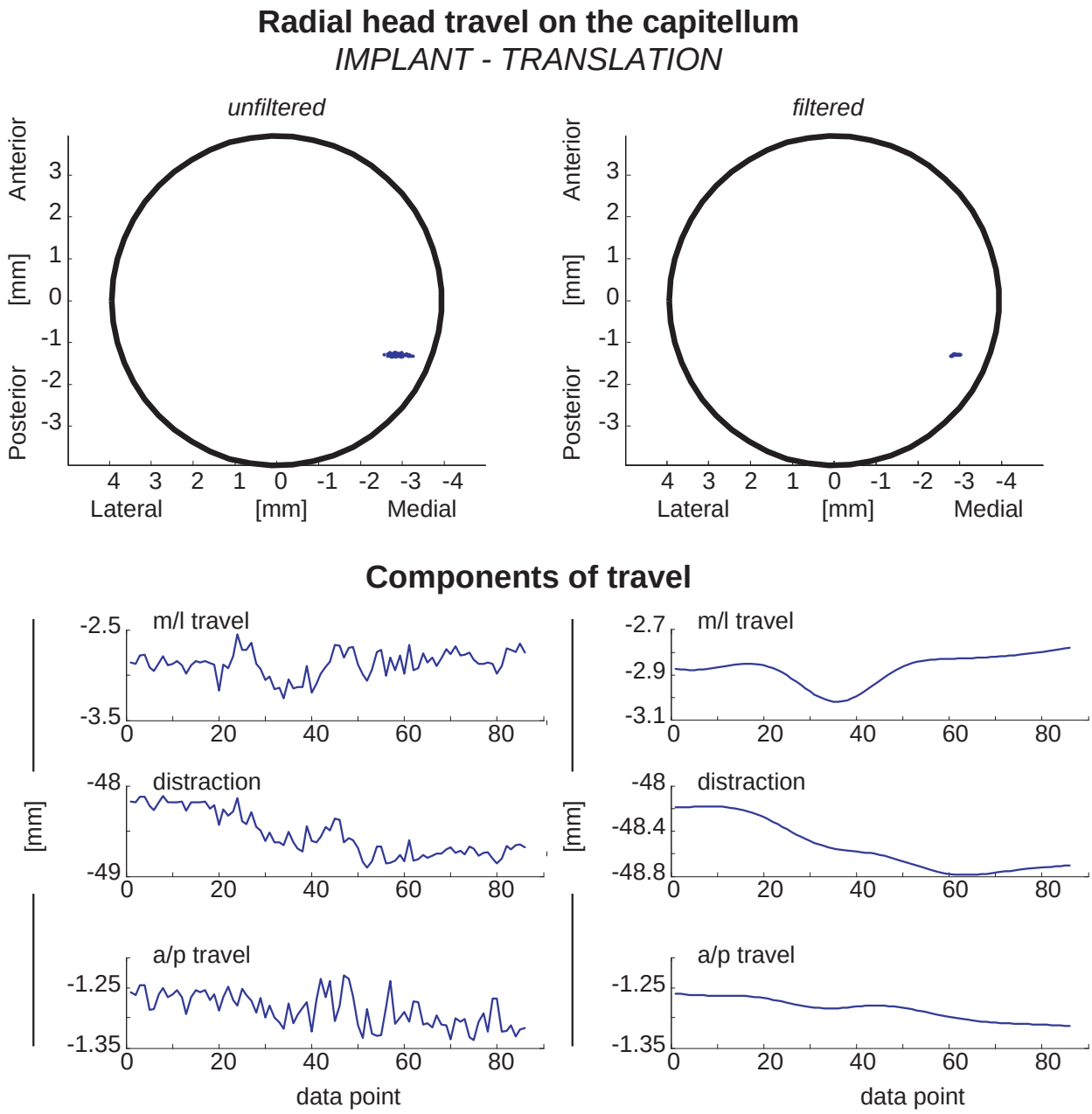


Figure 78: Implant results as collected with the Spicatek system for the translation case.

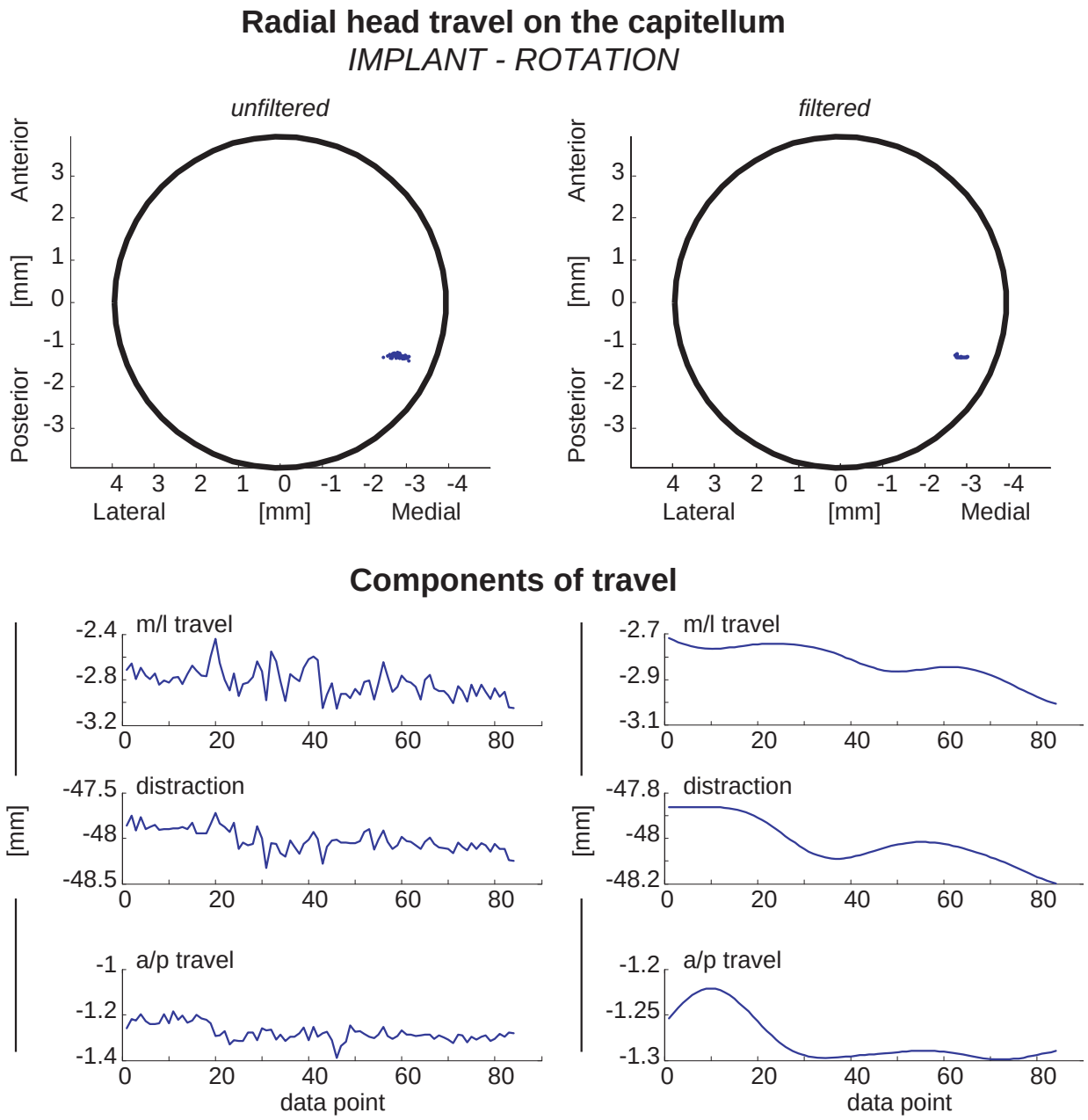


Figure 79: Implant results as collected with the Spicatek system for the rotation case.

The table below summarizes the numeric results for the Implant trials. It lists the total motion observed, both raw and filtered, as well as the expected motion. Note that, for the Static and Rotation trials in which no motion was expected, the maximum observed motion was 1.0366 mm. This seems spurious in comparison to the other results, which had a maximum motion, after filtering, of 0.4455 mm. In general, the amount of perceived motion during the Rotation trials was greater than that during the Static trials. This could be attributed to any slight misalignment between the implant and the center of rotation of the micrometer table. The amount of measured motion during the translation trials is concerning: the trial named "Implant_Translate_2" came the closest to measuring the true translation, with an error of 0.093 mm before filtering, and 0.01 mm after. The computed translations from the remaining Translation trials is concerning and grossly underestimates the amount of translation that occurred. One possible explanation may have been human error in the reading of the micrometer dial. It is also worth noting that the trial that performed best produced motion in a direction opposite to the other two trials. Future measurements are needed to assess the sources of error between the Spicatek system and the micrometer table.

Table 31: Motion measured and expected, Implant case.

	motion, in mm		
Trial name:	*Raw*	*Filtered*	*Expected*
Implant_Static_1	0.36	0.10	0.00
Implant_Static_2	0.27	0.07	0.00
Implant_Static_3	0.30	0.14	0.00
Implant_Translate_1	0.76	0.60	1.18
Implant_Translate_2	0.76	0.68	0.67
Implant_Translate_3	0.88	0.65	1.18
Implant_Rotate_1	0.83	0.33	0.00
Implant_Rotate_2	10.46	1.04	0.00
Implant_Rotate_3	0.54	0.45	0.00

Q.2.3 Ring Results

The next three figures show the results from the data collected to track the motion of a Ring of spherical markers which represent the native radial head tracking case. Similar to the results presented for the Implant case, the figures show representative examples for the static, translational, and rotational trials. In general, the results for the Ring trials are closer to the expected position (on the center of the capitellum) than those from the Implant trials. This could be because of the refocusing of one camera in between the collection of the Ring and Implant trials given that the Ring trials were collected first.

Figure 80 shows a representative result from a static trial which simulated native head tracking with the Ring of spherical markers. The expected outcome was that the center of the radial head would be at exactly the center of the capitellum and would not move with time. As seen in the right-hand side of the figure, with filtering, the motion during the trial is small, 0.04 mm before filtering and 0.02 after, but not the expected 0.0 mm. There is some concern about the fact that it is not centered on the capitellum. To be more specific, the average coordinates of the radial head are (-0.87, -0.51) for the raw case, and (-0.87, -0.52) after filtering. However, as discussed above for the Implant case, it is possible that the calibration suffered due to the absence of one marker on the calibration frame, and it is also possible that any required manual tracking or shadows from the markers contributed to these results. These factors may have contributed to the error in position calculation.

In Figure 81, a representative example from a translation trial, it is observed that the translation occurred primarily in the m/l direction. There is some motion in the other two directional components that is not eliminated with filtering. This may be due to the manual actuation of the dial to create the motion. The a/p position is expected to be at zero, and in

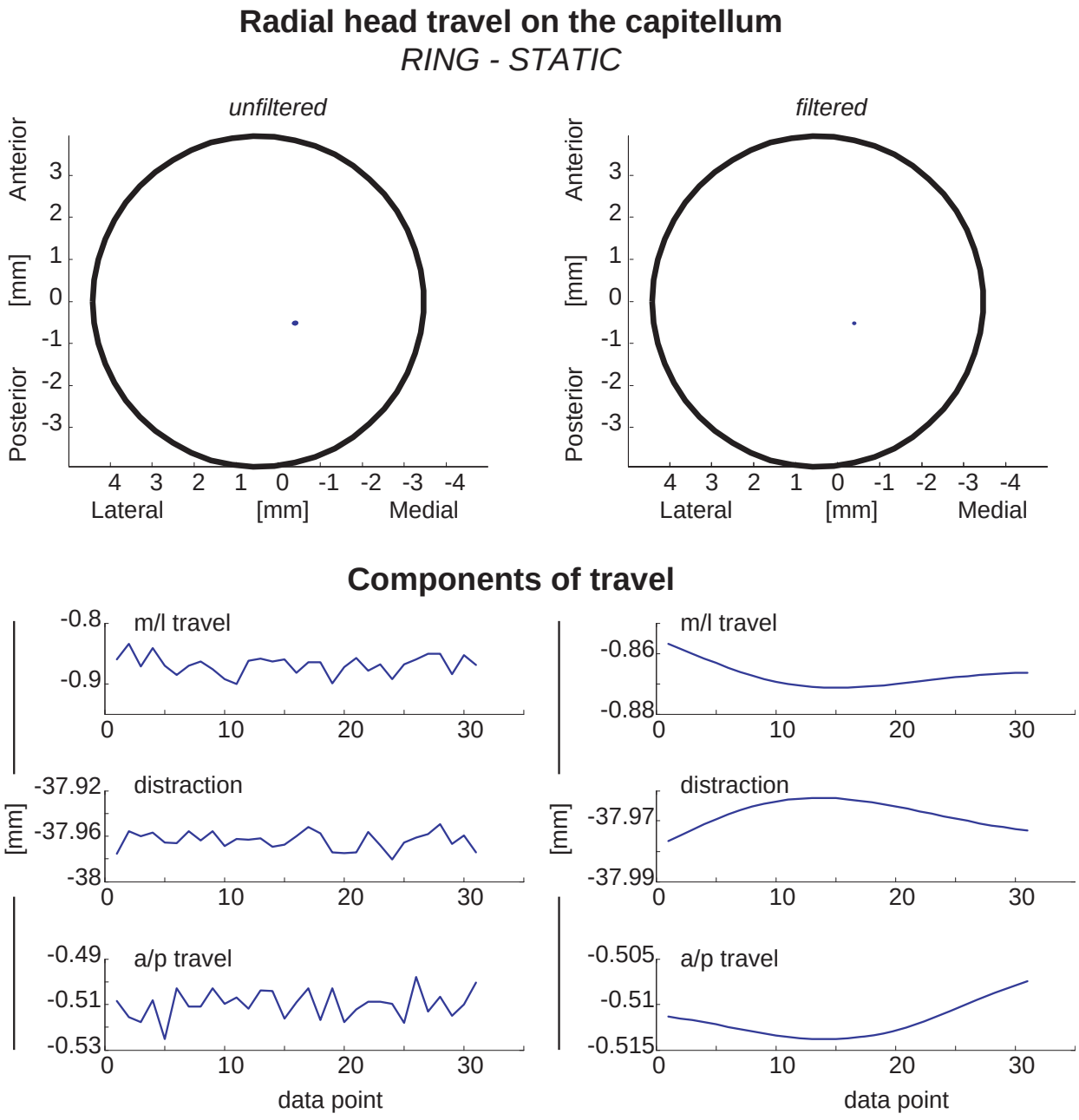


Figure 80: Native head (Ring) results as collected with the Spicatek system for the static case.

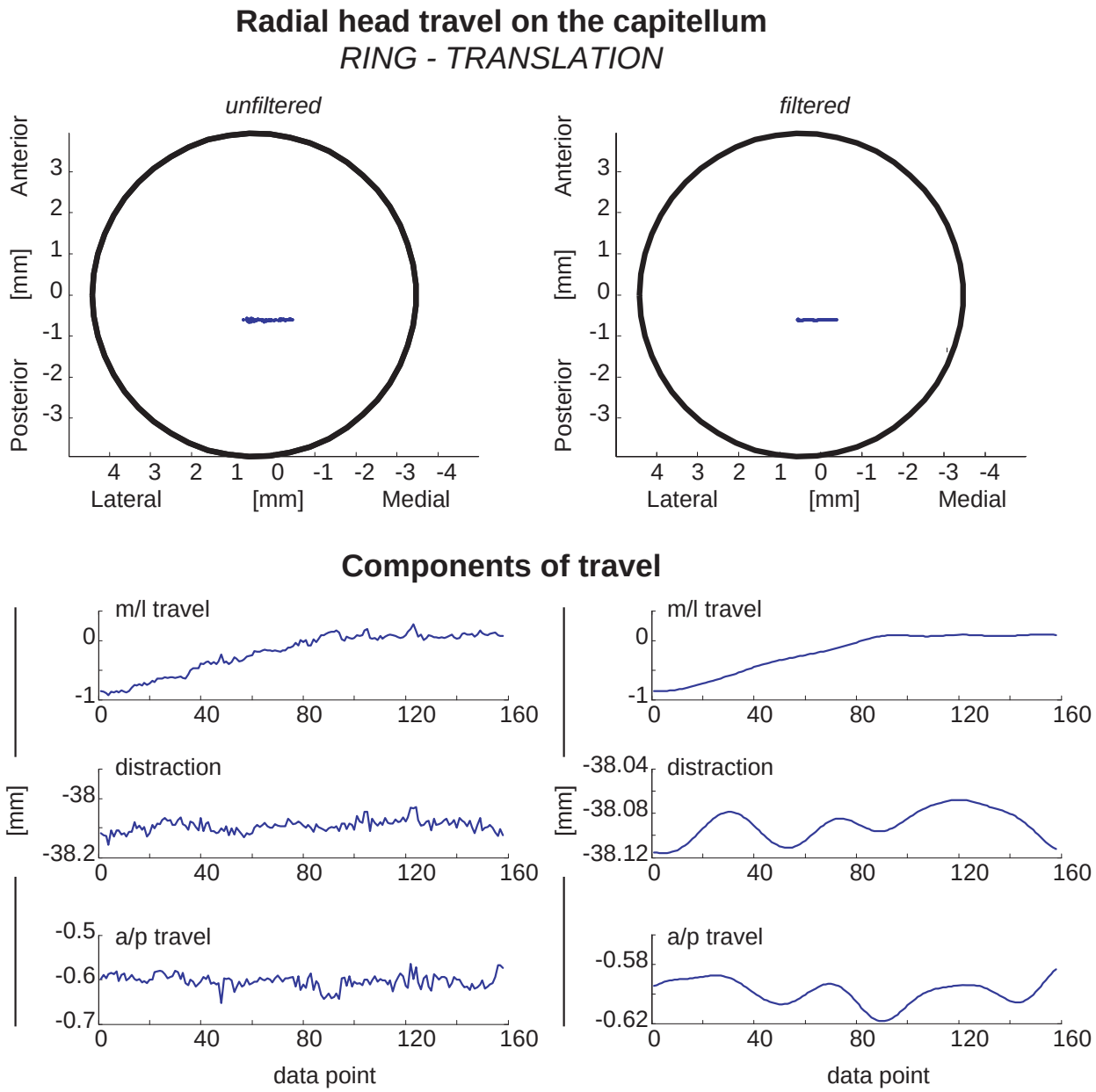


Figure 81: Native head (Ring) results as collected with the Spicatek system for the translation case.

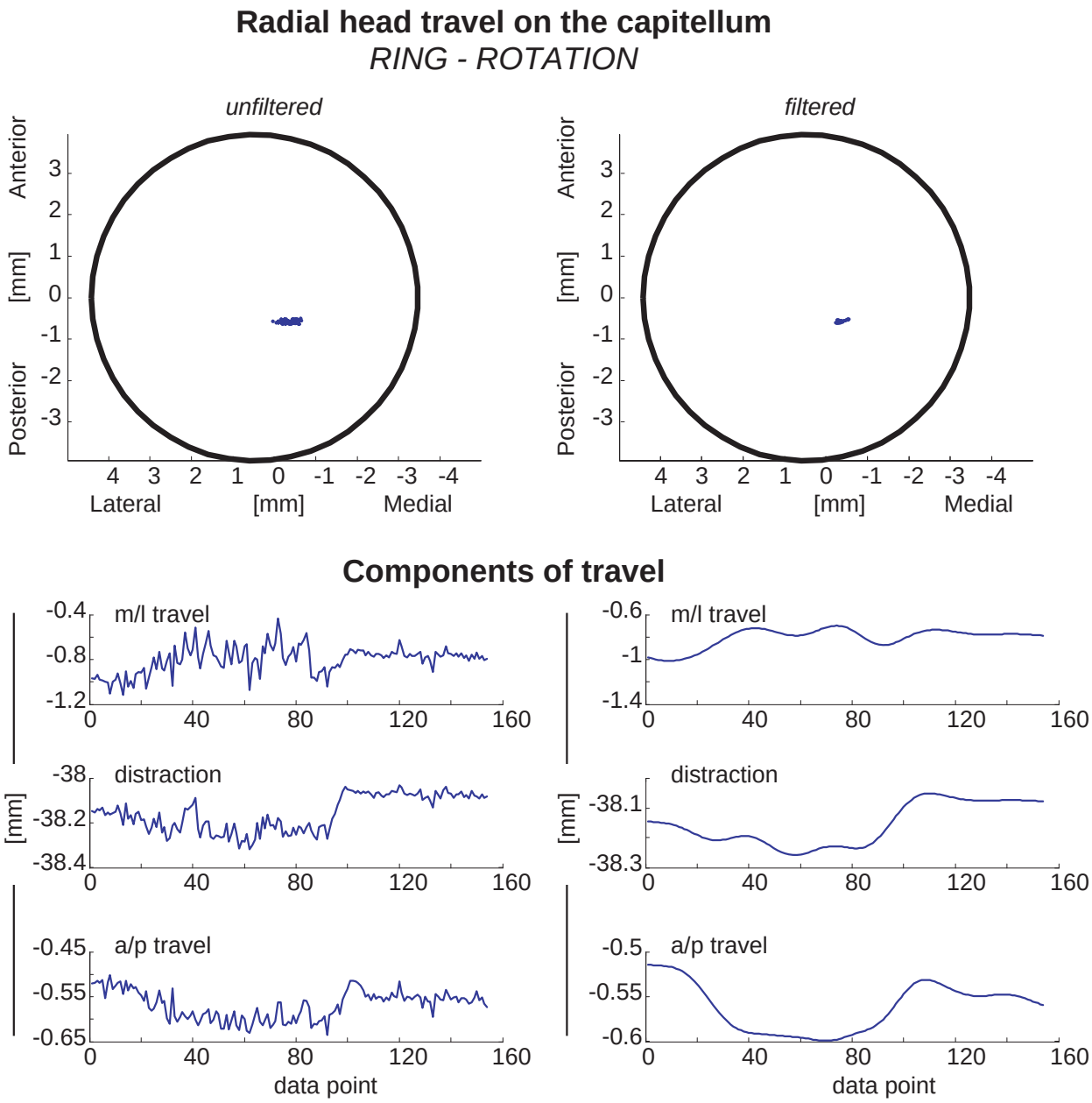


Figure 82: Native head (Ring) results as collected with the Spicatek system for the rotation case.

reality is incorrect by 0.5 mm. Possible explanations for this include the aforementioned issues with calibration and marker shadows.

Figure 82 shows an example from a rotation trial for the native head (Ring) case. The expected result is that the position of the radial head will always be at the center of the capitellum. In reality, there is some motion at the sub-millimeter level. This could be in part due to the manual actuation of the rotation table, or due to the changes in lighting (shadows) caused by the manual actuation of the rotation table. If the rod simulating the radial head was slightly off-center from the center of the rotation table, this may have also contributed. Additionally, any inaccuracies due to the incomplete calibration frame may have influenced the results.

The total motion measured for the Ring case is shown below in Table 32. Similar to Table 31, the measured distances are given for both the raw and filtered cases, along with the expected distance moved. Like the Implant case, the calculated motion during the Static trials was considerably less than that during the Rotation trials. During the Translation trials, unlike the Implant case, the computed motions are close to the expected motions, with a maximum error of 0.22 mm using the raw data and 0.06 mm with filtering. It is worth noting that for the Ring_Translate_1 and Ring_Translate_3 trials, the motion direction was the same, and the Spicatek system overestimated the amount of travel by at least 0.04 mm. The change in direction for the Ring_Translate_2 trial could have encountered backlash, which could explain the overestimate by a smaller amount of 0.02 mm.

Table 32: Motion measured and expected, Native Head (Ring) case.

	motion, in mm		
Trial name:	Raw	Filtered	Expected
Ring_Static_1	0.04	0.02	0.00
Ring_Static_2	0.27	0.17	0.00
Ring_Static_3	0.26	0.02	0.00
Ring_Translate_1	1.13	0.98	0.94
Ring_Translate_2	0.97	0.93	0.91
Ring_Translate_3	1.13	0.96	0.91
Ring_Rotate_1	0.85	0.40	0.00
Ring_Rotate_2	0.54	0.31	0.00
Ring_Rotate_3	1.50	0.38	0.00

Q.2.4 Marker noise during static trials

Similar to the calculations presented in Chapter 9, the noise of each marker was computed for the static trials, where noise was again defined as the standard deviation in marker position over the entire static trial. Since the larger 3.7 mm markers were mostly automatically tracked, the noise for these markers was computed in addition to that for the smaller spheres and divot markers. The results of the computations are shown below in Table 33 and Table 34. For the smaller markers, the maximum noise component in any direction was 0.11 mm, which indicates that sub-millimeter measurements are feasible with the Spicatek system. Looking at Table 34, which

shows the noise components for the larger (3.7 mm) markers, the maximum component noise is 0.57 mm. Markers 5-12 were not automatically tracked sufficiently in the implant Static 1 and 3 trials to permit computation of the noise of these markers. The higher noise level could be due to the use of manual tracking when tracking some of the larger markers, or due to the influence of shadows, which created additional dark regions around some of the markers. However, since there was no human movement during the static data collection, it is expected that the effects of shadows creating excessive dark regions would have been constant and would not have affected the noise measured.

Table 33: Standard deviation of small marker positions during static trials. Units are mm.

		marker				
		1	2	3	4	5
Implant Static 1	x	0.11	0.03	0.03	0.02	
	y	0.02	0.01	0.01	0.01	
	z	0.00	0.01	0.00	0.01	
Implant Static 2	x	0.07	0.01	0.01	0.01	
	y	0.01	0.00	0.00	0.00	
	z	0.00	0.00	0.00	0.00	
Implant Static 3	x	0.04	0.03	0.00	0.03	
	y	0.01	0.01	0.00	0.01	
	z	0.00	0.00	0.00	0.01	
Ring Static 1	x	0.03	0.01	0.03	0.02	0.02
	y	0.01	0.01	0.00	0.00	0.01
	z	0.03	0.00	0.01	0.02	0.00
Ring Static 2	x	0.05	0.06	0.07	0.05	0.02
	y	0.03	0.03	0.03	0.02	0.01
	z	0.02	0.00	0.01	0.02	0.00
Ring Static 3	x	0.02	0.06	0.03	0.03	0.03
	y	0.00	0.02	0.02	0.01	0.02
	z	0.01	0.01	0.02	0.01	0.01

Table 34: Standard deviation of larger (3.7 mm) marker positions during static trials. Units are mm.

		marker											
		1	2	3	4	5	6	7	8	9	10	11	12
Implant Static 1	x	0.01	0.01	0.01	0.01								
	y	0.00	0.00	0.01	0.01								
	z	0.00	0.01	0.00	0.01								
Implant Static 2	x	0.01	0.02	0.01	0.01	0.01	0.01	0.02	0.01	0.02	0.02	0.02	0.01
	y	0.00	0.01	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.01	0.00
	z	0.00	0.00	0.01	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.01	0.00
Implant Static 3	x	0.02	0.02	0.01	0.01								
	y	0.01	0.01	0.01	0.00								
	z	0.00	0.00	0.00	0.01								
Ring Static 1	x	0.05	0.04	0.02	0.09	**0.57**	0.02	0.02	0.08	0.02	0.02	0.04	0.02
	y	0.03	0.02	0.01	0.02	0.26	0.01	0.01	0.04	0.01	0.01	0.02	0.01
	z	0.01	0.01	0.01	0.02	0.05	0.00	0.01	0.01	0.01	0.01	0.01	0.00
Ring Static 2	x	0.21	0.13	0.16	0.40	0.04	0.03	0.04	0.03	0.02	0.03	0.36	0.04
	y	0.14	0.08	0.09	0.24	0.02	0.01	0.02	0.02	0.01	0.01	0.17	0.02
	z	0.02	0.01	0.02	0.02	0.01	0.00	0.01	0.01	0.01	0.01	0.05	0.01
Ring Static 3	x	0.08	0.06	0.04	0.07	0.03	0.02	0.02	0.02	0.06	0.02	0.03	0.08
	y	0.05	0.03	0.02	0.03	0.01	0.01	0.01	0.01	0.03	0.01	0.01	0.05
	z	0.01	0.01	0.00	0.03	0.01	0.00	0.01	0.00	0.01	0.01	0.01	0.00

Q.2.5 Sensitivity to projection plane of the capitellum

Upon viewing the results presented above, a primary concern was that the radial head position was not centered on the capitellum and seemed to show a consistent bias towards the medial side. Table 35 below summarizes the average position offset for the Static and Rotation trials, which had a predicted position of (0,0). Note that filtering does not necessarily bring the radial head position closer to the center of the capitellum. This offset could possibly be attributed to the angle from which the cameras viewed the test apparatus. Alternatively, it could be due to inaccuracies in the computation of the center of the capitellum and the plane onto which the radial head position is projected. To this end, the coordinates representing the humerus as measured on the CMM were examined. The construction of the projection plane relies on accurate measurements of points representing the center of the capitellum, the center of the trochlear groove, and the center of the humeral shaft. During the measurement on the CMM, the rotational table was oriented to be parallel to the CMM measurement table. The three points used to construct the projection plane were represented by points on the rotation table, thus their vertical coordinates were expected to be identical. The vertical coordinate of one of these points

differed by 1.75 mm from the others. Since these points were expected to be in exactly the same plane, the vertical coordinate of all three points was chosen to be identical. One static trial from the Implant case was then processed, with filtering, with the newly edited humerus measurements. The results are shown below in Figure 83. The left-hand side shows the radial head position in capitellar coordinates as originally processed, and the right-hand side shows the same as processed with the modified humeral landmark locations.

Table 35: Average distance from the center of the capitellum to the radial head position for the Static and Rotation trials.

	distance from center (mm)	
Trial Name:	Raw	Filtered
Implant_Static_1	3.00	2.99
Implant_Static_2	3.06	3.05
Implant_Static_3	3.23	3.24
Implant_Rotate_1	2.71	2.71
Implant_Rotate_2	3.28	3.19
Implant_Rotate_3	1.99	3.10
Ring_Static_1	1.01	1.01
Ring_Static_2	1.02	1.02
Ring_Static_3	1.01	1.01
Ring_Rotate_1	1.05	1.04
Ring_Rotate_2	0.98	0.98
Ring_Rotate_3	0.78	0.76

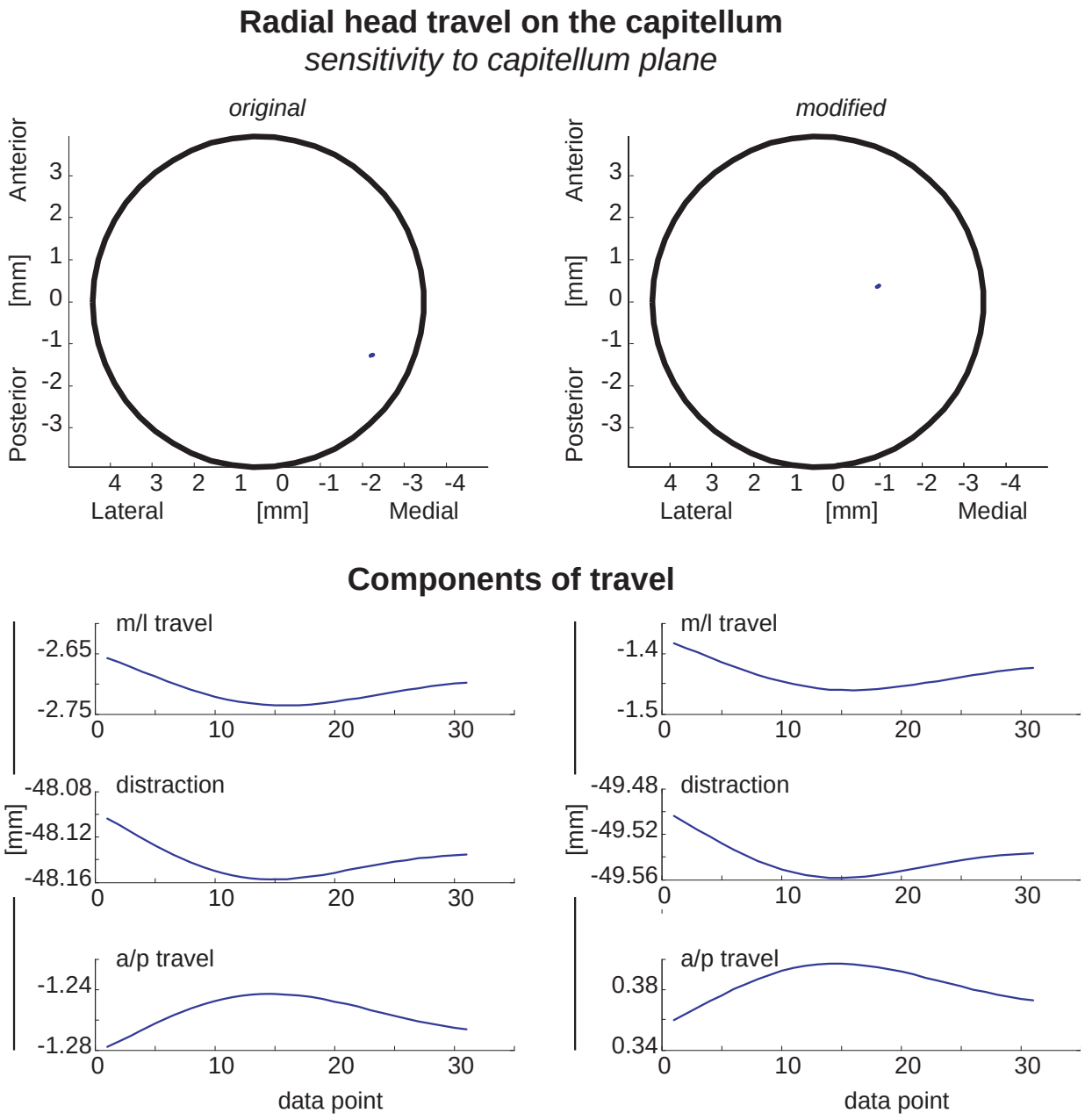


Figure 83: Sensitivity of radial head position to the precise measurement of capitellar coordinates.

As seen in Figure 83, the changes in the position of the projection plane, as calculated from the measured humerus landmarks, resulted in a change in position of the radial head center of 2.08 mm. Table 36 enumerates these results. This illustrates the sensitivity of the computed radial head position to the measurement of the humeral landmarks, an additional element that may have contributed to the radial head positioning showing a consistent bias towards the medial side.

Table 36: Distance and position coordinates of the average radial head position for the originally-measured and truly-planar capitellum.

	mm		
	distance from center	m/l coordinate	a/p coordinate
original	2.99	-2.71	-1.256
true plane	1.49	-1.44	0.38

To overcome this problem in the future, additional techniques may be employed to improve the accuracy of the computation of the projection plane. One approach would be to improve the estimation of the humeral landmark positions by estimating them optically using redundant markers. In essence, this would mean surrounding the capitellum with markers used to compute its center. Another option would be to project radial head travel on to a plane centered on the capitellum but whose orientation was computed with respect to the radial head instead of with respect to other humeral landmarks. If these methods do not improve the

accuracy with which the projection plane can be estimated, the measurements of radial head travel will still be valid, though the absolute positioning of the radial head may not be interpreted as unequivocally being in a specific region of the capitellum.

Q.3 SUMMARY

The Spicatek system is promising for use in the tracking of the native radial head and implants, but concerns remain. This second round of tests yielded improved results over the first set of tests, with less noise in the static trials and results closer to those expected for the dynamic ones. The apparent bias of the computed radial head position to the medial side could be due to imperfect calibration of the camera system, the influence of reflections and shadows in and around the markers as captured on the videos, and possibly due to inaccuracies in the computation of the capitellum-centered projection plane. Future work can explore alternative lighting configurations, such as indirect lighting, which may reduce the glare in the larger markers. Additionally, a more robust calibration frame may offer an improved calibration which may in turn reduce the amount of manual marker tracking necessary. For small gaps of only a few frames, interpolation may give better results than manual tracking. In conclusion, even further investigation to fine-tune the data acquisition and processing methods is needed.

BIBLIOGRAPHY

1. Abu-Faraj, Z.O., et al. A clinical system for the analysis of three-dimensional pediatric foot and ankle motion. in Engineering in Medicine and Biology society, 1997. Proceedings of the 19th Annual International Conference of the IEEE. 1997.

2. Admiraal, M.A., W.P. Medendorp, and C.C. Gielen, Three-dimensional head and upper arm orientations during kinematically redundant movements and at rest. Exp Brain Res, 2002. 142(2): p. 181-92.

3. Aggarwal, J.K. and Q. Cai, Human motion analysis: A review. Computer Vision and Image Understanding, 1999. 73(33): p. 428-40.

4. Allin, J. and G.F. Inbar, FNS control schemes for the upper limb. IEEE Trans Biomed Eng, 1986. 33(9): p. 818-28.

5. Allin, J. and G.F. Inbar, FNS parameter selection and upper limb characterization. IEEE Trans Biomed Eng, 1986. 33(9): p. 809-17.

6. Amis, A.A., Muscle strengths and musculo-skeletal geometry of the upper limb. Engineering in Medicine, 1979. 8: p. 41-47.

7. Amis, A.A., D. Dowson, and V. Wright, Analysis of elbow forces due to high-speed forearm movements. J Biomech, 1980. 13(10): p. 825-31.

8. An, K.N., et al., Muscles across the elbow joint: a biomechanical analysis. J Biomech, 1981. 14(10): p. 659-69.

9. An, K.N., K.R. Kaufman, and E.Y. Chao, Physiological considerations of muscle force through the elbow joint. J Biomech, 1989. 22(11-12): p. 1249-56.

10. An, K.N., et al., Determination of muscle and joint forces: a new technique to solve the indeterminate problem. J Biomech Eng, 1984. 106(4): p. 364-7.

11. An, K.N. and B.F. Morrey, Biomechanics of the elbow, in The elbow and its disorders. 2003, W.B. Saunders: Philadelphia. p. 43-60.

12. An, K.N., et al., Tendon excursion and moment arm of index finger muscles. J Biomech, 1983. 16(6): p. 419-25.

13. Anglin, C. and U.P. Wyss, Review of arm motion analyses. Proc Inst Mech Eng [H], 2000. 214(5): p. 541-55.

14. Armstrong, A.D., et al., Rehabilitation of the medial collateral ligament-deficient elbow: an in vitro biomechanical study. J Hand Surg [Am], 2000. 25(6): p. 1051-7.

15. Armstrong, A.D., et al., Single-strand ligament reconstruction of the medial collateral ligament restores valgus elbow stability. J Shoulder Elbow Surg, 2002. 11(1): p. 65-71.

16. Arner, O., K. Ekengren, and T. Von Schreeb, Fractures of the head and neck of the radius; a clinical and roentgenographic study of 310 cases. Acta Chir Scand, 1957. 112(2): p. 115-34.

17. Bach, J.M. and M.L. Hull, A new load application system for in vitro study of ligamentous injuries to the human knee joint. J Biomech Eng, 1995. 117(4): p. 373-82.

18. Bach, J.M. and M.L. Hull, Strain inhomogeneity in the anterior cruciate ligament under application of external and muscular loads. J Biomech Eng, 1998. 120(4): p. 497-503.

19. Bain, G.I., A review of complex trauma to the elbow. Aust N Z J Surg, 1999. 69(8): p. 578-81.

20. Bazzucchi, I., et al., Coactivation of the elbow antagonist muscles is not affected by the speed of movement in isokinetic exercise. Muscle Nerve, 2006. 33(2): p. 191-9.

21. Beckett, K.S., et al., Variations in the normal anatomy of the collateral ligaments of the human elbow joint. J Anat, 2000. 197 Pt 3: p. 507-11.

22. Beingessner, D.M., et al., The effect of radial head fracture size on radiocapitellar joint stability. Clin Biomech (Bristol, Avon), 2003. 18(7): p. 677-81.

23. Beingessner, D.M., et al., The effect of radial head fracture size on elbow kinematics and stability. J Orthop Res, 2005. 23(1): p. 210-7.

24. Beredjiklian, P.K., et al., Prosthetic radial head components and proximal radial morphology: a mismatch. J Shoulder Elbow Surg, 1999. 8(5): p. 471-5.

25. Bernotas, L.A., P.E. Crago, and H.J. Chizeck, Adaptive control of electrically stimulated muscle. IEEE Trans Biomed Eng, 1987. 34(2): p. 140-7.

26. Bizzi, E., et al., Posture control and trajectory formation during arm movement. J Neurosci, 1984. 4(11): p. 2738-44.

27. Boninger, M.L., et al., Shoulder and elbow motion during two speeds of wheelchair propulsion: a description using a local coordinate system. Spinal Cord, 1998. 36(6): p. 418-26.

28. Borghese, N. and P. PRigiroli. Tracking densely moving markers. in IEEE Proceedings - 3D Processing Visualization and Transmission. 2002.

29. Borghese, N.A., P. Cerveri, and P. Rigiroli, A fast method for calibrating video-based motion analysers using only a rigid bar. Med Biol Eng Comput, 2001. 39(1): p. 76-81.

30. Bottlang, M., et al., Radiographic determinants of the elbow rotation axis: experimental identification and quantitative validation. J Orthop Res, 2000. 18(5): p. 821-8.

31. Brand, P.W., R.B. Beach, and D.E. Thompson, Relative tension and potential excursion of muscles in the forearm and hand. J Hand Surg [Am], 1981. 6(3): p. 209-19.

32. Brand, P.W., K.C. Carville, and J.C. Ellis, Tendons and pulleys at the metacarpophalangeal joint of a finger. J Bone Joint Surg Am, 1975. 57-A: p. 779-84.

33. Bremer, A.K., et al., Moment arms of forearm rotators. Clin Biomech (Bristol, Avon), 2006. 21(7): p. 683-91.

34. Buchanan, T.S., G.P. Rovai, and W.Z. Rymer, Strategies for muscle activation during isometric torque generation at the human elbow. J Neurophysiol, 1989. 62(6): p. 1201-12.

35. Butler, A.L., et al. Translational kinematics of the radial head before and after radial head replacement. in Annual Meeting of the American Society for Surgery of the Hand. 2005. San Antonio.

36. Cai, Q. and J.K. Aggarwal. Automatic tracking of human motion in indoor scenes across multiple synchronized video streams. in Computer Vision, 1998. Sixth International Conference on. 1998.

37. Cai, Q. and J.K. Aggarwal, Tracking human motion in structured environments using a distributed-camera system. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 1999. 21(11): p. 1241-1247.

38. Caldwell, D., G. Medrano-Cerda, and M. Goodwin, Control of pneumatic muscle actuators. IEEE Control Systems, 1995: p. 40-48.

39. Caldwell, G.E. and M. Van Leemputte, Elbow torques and EMG patterns of flexor muscles during different isometric tasks. Electromyogr Clin Neurophysiol, 1991. 31(7): p. 433-45.

40. Campbell, K.R. Expert vision, data acquisition and analysis. in ASME International Computers in Engineering Conference and Exhibition. 1987. New Yor, NY.

41. Cappozzo, A., et al., Surface-marker cluster design criteria for 3-D bone movement reconstruction. Biomedical Engineering, IEEE Transactions on, 1997. 44(12): p. 1165-1174.

42. Cappozzo, A., et al., Position and orientation in space of bones during movement: experimental artefacts. Clin Biomech (Bristol, Avon), 1996. 11(2): p. 90-100.

43. Captier, G., et al., Biometry of the radial head: biomechanical implications in pronation and supination. Surg Radiol Anat, 2002. 24(5): p. 295-301.

44. Carr, C.R. and J.W. Howard, Metallic cap replacement of radial head following fracture. West J Surg Obstet Gynecol, 1951. 59(10): p. 539-46.

45. Castberg, T. and E. Thing, Treatment of fractures of the upper end of the radius. Acta Chir Scand, 1953. 105(1-4): p. 62-9.

46. Chadwick, E.K. and A.C. Nicol, Elbow and wrist joint contact forces during occupational pick and place activities. J Biomech, 2000. 33(5): p. 591-600.

47. Challis, J.H., A procedure for determining rigid body transformation parameters. J Biomech, 1995. 28(6): p. 733-7.

48. Chao, E.Y. and B.F. Morrey, Three-dimensional rotation of the elbow. J Biomech, 1978. 11(1-2): p. 57-73.

49. Charlton, I.W. and G.R. Johnson, Application of spherical and cylindrical wrapping algorithms in a musculoskeletal model of the upper limb. J Biomech, 2001. 34(9): p. 1209-16.

50. Chaudhari, A.M., et al. A video-based, markerless motion tracking system for biomechanical analysis in an arbitrary environment. in ASME Bioengineering. 2001.

51. Chen, L., C.W. Armstrong, and D.D. Raftopoulos, An investigation on the accuracy of three-dimensional space reconstruction using the direct linear transformation technique. J Biomech, 1994. 27(4): p. 493-500.

52. Chen, Q., et al., Technical note: validation of a motion analysis system for measuring the relative motion of the intermediate component of a tripolar total hip arthroplasty prosthesis. Med Eng Phys, 2005. 27(6): p. 505-12.

53. Cherry, J.C., Use of acrylic prosthesis in the treatment of fracture of the head of the radius. J Bone Joint Surg Br, 1953. 35-B(1): p. 70-1.

54. Chiari, L., et al., Human movement analysis using stereophotogrammetry. Part 2: instrumental errors. Gait Posture, 2005. 21(2): p. 197-211.

55. Chong, S.Y., et al. Determining the optinal axis of the radius about the ulna in-vivo. in 5th World Congress of Biomechanics. 2006. Munich, Germany: J Biomech.

56. Colbaugh, R. and K. Glass, Hierarchical Control of Human Joint Motion Simulators. Comp Elec Eng, 1993. 19(3): p. 213-230.

57. Colbaugh, R., K. Glass, and J. Ryu, Adaptive control of a human wrist motion simulator. International Journal of Modelling and Simulation, 1991. 11(3): p. 93-102.

58. Coleman, D.A., W.F. Blair, and D. Shurr, Resection of the radial head for fracture of the radial head. Long-term follow-up of seventeen cases. J Bone Joint Surg Am, 1987. 69(3): p. 385-92.

59. Conn, J. and P.A. Wade, Injuries of the Elbow: A Ten Year Review. Journal of Trauma, 1961. 1: p. 248.

60. Coope, I.D., Circle fitting by linear and nonlinear least squares. Journal of Optimization Theory and Applications, 1993. 76(2): p. 381-8.

61. Corazza, S. and T.P. Andriacchi. Posturographic Analysis Is Possible Without Ground Reaction Forces Measurement Through Markerless Motion Capture. in ASB. 2007. Stanford, CA.

62. Corcos, D.M., et al., Fatigue induced changes in phasic muscle activation patterns for fast elbow flexion movements. Exp Brain Res, 2002. 142(1): p. 1-12.

63. Debicki, D.B., et al., Kinematics of wrist joint flexion in overarm throws made by skilled subjects. Exp Brain Res, 2004. 154(3): p. 382-94.

64. Debski, R.E., et al., A new dynamic testing apparatus to study glenohumeral joint motion. J Biomech, 1995. 28(7): p. 869-74.

65. Delp, S.L., D.A. Ringwelski, and N.C. Carroll, Transfer of the rectus femoris: effects of transfer site on moment arms about the knee and hip. J Biomech, 1994. 27(10): p. 1201-11.

66. Desjardins, P., et al., Handling missing marker coordinates in 3D analysis. Med Eng Phys, 2002. 24(6): p. 437-40.

67. Dingwell, J.B., C.D. Mah, and F.A. Mussa-Ivaldi, Manipulating objects with internal degrees of freedom: evidence for model-based control. J Neurophysiol, 2002. 88(1): p. 222-35.

68. Dingwell, J.B., C.D. Mah, and F.A. Mussa-Ivaldi, Experimentally confirmed mathematical model for human control of a non-rigid object. J Neurophysiol, 2004. 91(3): p. 1158-70.

69. Dockstader, S.L. and A.M. Tekalp, Multiple camera tracking of interacting and occluded human motion. Proceedings of the IEEE, 2001. 89(10): p. 1441-1455.

70. Duck, T.R., et al., Application of screw displacement axes to quantify elbow instability. Clin Biomech (Bristol, Avon), 2003. 18(4): p. 303-10.

71. Duck, T.R., et al., Variability and repeatability of the flexion axis at the ulnohumeral joint. J Orthop Res, 2003. 21(3): p. 399-404.

72. Duck, T.R., G.J.W. King, and J.A. Johnson. Screw Displacement Axes of Forearm Pro-Supination. in 47th Annual Meeting, ORS. 2001. San Francisco, California.

73. Dunning, C.E., et al., Simulated active control produces repeatable motion pathways of the elbow in an in vitro testing system. J Biomech, 2001. 34(8): p. 1039-48.

74. Dunning, C.E., et al., Quantifying translations in the radiohumeral joint: application of a floating axis analysis. J Biomech, 2003. 36(8): p. 1219-23.

75. Dunning, C.E., et al., Development of a motion-controlled in vitro elbow testing system. J Orthop Res, 2003. 21(3): p. 405-11.

76. Edinger, D.H., Accuracy of a robotic system for the reproduction of condylar movements: a preliminary report. Quintessence Int, 2004. 35(7): p. 519-23.

77. Ehara, Y., et al., Comparison of the performance of 3D camera systems II. Gait & Posture, 1997. 5(3): p. 251-255.

78. Ehara, Y., et al., Comparison of the performance of 3D camera systems. Gait & Posture, 1995. 3(3): p. 166-169.

79. Enoka, R.M., Multi-joint systems, in Neuromechanics of human movement. 2002, Human Kinetics: Champaign, IL. p. 313-358.

80. Ervilha, U.F., et al., Effect of load level and muscle pain intensity on the motor control of elbow-flexion movements. Eur J Appl Physiol, 2004. 92(1-2): p. 168-75.

81. Ettema, G.J.C., G. Styles, and V. Kippers, The moment arms of 23 muscle segments of the upper limb with varying elbow and forearm positions: Implications for motor control. Human Movement Science, 1998. 17: p. 201-220.

82. Ewald, F.C., Total elbow replacement. Orthop Clin North Am, 1975. 6(3): p. 685-96.

83. Fagergren, A., O. Ekeberg, and H. Forssberg, Precision grip force dynamics: a system identification approach. IEEE Trans Biomed Eng, 2000. 47(10): p. 1366-75.

84. Feldman, A.G. and M.L. Latash, Testing hypotheses and the advancement of science: recent attempts to falsify the equilibrium point hypothesis. Exp Brain Res, 2005. 161(1): p. 91-103.

85. Figueroa, P.J., N.J. Leite, and R.M. Barros, A flexible software for tracking of markers used in human motion analysis. Comput Methods Programs Biomed, 2003. 72(2): p. 155-65.

86. Fisher, B., Least-square fitting of circles to 3D point data. 2000.

87. Flash, T., The control of hand equilibrium trajectories in multi-joint arm movements. Biol Cybern, 1987. 57(4-5): p. 257-74.

88. Florou, G. and R. Mohr. What accuracy for 3D measurements with cameras? in Pattern Recognition, 1996., Proceedings of the 13th International Conference on. 1996.

89. Frey, M., et al., Optimised robot-based system for the exploration of elastic joint properties. Med Biol Eng Comput, 2004. 42(5): p. 674-8.

90. Funk, D.A., et al., Electromyographic analysis of muscles across the elbow joint. J Orthop Res, 1987. 5(4): p. 529-38.

91. Galik, K., et al., The effect of the annular ligament on kinematics of the radial head. J Hand Surg [Am], 2007. 32(8): p. 1218-1224.

92. Gardinier, J.D. and R.V. Gonzalez. Pronation-supination moment arms in the human forearm. in 2003 Summer Bioengineering Conference. 2003. Key Biscayne, FL.

93. Georgopoulos, A.P., et al., Neural coding of finger and wrist movements. J Comput Neurosci, 1999. 6(3): p. 279-88.

94. Gerbeaux, M., E. Turpin, and G. Lensel-Corbeil, Musculo-articular modelling of the triceps brachii. J Biomech, 1996. 29(2): p. 171-80.

95. Ghafouri, M. and A.G. Feldman, The timing of control signals underlying fast point-to-point arm movements. Exp Brain Res, 2001. 137(3-4): p. 411-23.

96. Gialas, N. and Y. Matsuoka. Muscle actuator design for the ACT hand. in IEEE International Conference on Robotics and Automation. 2004. New Orleans, LA.

97. Gleicher, M., Comparing Constraint-Based Motion Editing Methods. Graphical Models, 2001. 63(2): p. 107-134.

98. Gleicher, M. and N. Ferrier. Evaluating video-based motion capture. in Computer Animation, 2002. Proceedings of. 2002.

99. Gomi, H. and M. Kawato, Human arm stiffness and equilibrium-point trajectory during multi-joint movement. Biol Cybern, 1997. 76(3): p. 163-71.

100. Gonzalez, R.V., et al., Development and evaluation of a musculoskeletal model of the elbow joint complex. J Biomech Eng, 1996. 118(1): p. 32-40.

101. Gordon, M. and P.G. Bullough, Synovial and osseous inflammation in failed silicone-rubber prostheses. J Bone Joint Surg Am, 1982. 64(4): p. 574-80.

102. Gottlieb, G.L., On the voluntary movement of compliant (inertial-viscoelastic) loads by parcellated control mechanisms. J Neurophysiol, 1996. 76(5): p. 3207-29.

103. Gottlieb, G.L., et al., Coordinating two degrees of freedom during human arm movement: load and speed invariance of relative joint torques. J Neurophysiol, 1996. 76(5): p. 3196-206.

104. Gowitzke, B.A. and M. Milner, Scientific Bases of Human Movement. 3 ed. 1988, Baltimore: Williams & Wilkins.

105. Greenwood, D.T., Principles of dynamics. 2nd ed. 1988, Englewood Cliffs, NJ: Prentice Hall.

106. Gribble, P.L., R. Laboissiere, and D.J. Ostry, Control of human arm and jaw motion: issues related to musculo-skeletal geometry, in Self-Organization, Computational Maps and Motor Control, P. Morasso and V. Sanguineti, Editors. 1997, Elsevier North-Holland: Amsterdam. p. 483-506.

107. Grood, E.S. and W.J. Suntay, A joint coordinate system for the clinical description of three-dimensional motions: application to the knee. J Biomech Eng, 1983. 105(2): p. 136-44.

108. Grood, E.S., et al., Biomechanics of the knee-extension exercise. Effect of cutting the anterior cruciate ligament. J Bone Joint Surg Am, 1984. 66(5): p. 725-34.

109. Guigon, E., P. Baraduc, and M. Desmurget, Computational motor control: redundancy and invariance. J Neurophysiol, 2007. 97(1): p. 331-47.

110. Gupta, G.G., G. Lucas, and D.L. Hahn, Biomechanical and computer analysis of radial head prostheses. J Shoulder Elbow Surg, 1997. 6(1): p. 37-48.

111. Haggard, P. and A.M. Wing, Assessing and reporting the accuracy of position measurements made with optical tracking systems. J Mot Behav, 1990. 22(2): p. 315-21.

112. Hatwell, M.S., et al., The development of a model reference adaptive controller to control the knee joint of paraplegics. Automatic Control, IEEE Transactions on, 1991. 36(6): p. 683-691.

113. Hogan, N., An organizing principle for a class of voluntary movements. J Neurosci, 1984. 4(11): p. 2745-54.

114. Hollister, A.M., H. Gellman, and R.L. Waters, The relationship of the interosseous membrane to the axis of rotation of the forearm. Clin Orthop Relat Res, 1994(298): p. 272-6.

115. Hore, J., D.B. Debicki, and S. Watts, Braking of elbow extension in fast overarm throws made by skilled and unskilled subjects. Exp Brain Res, 2005. 164(3): p. 365-75.

116. Hotchkiss, R.N., Displaced Fractures of the Radial Head: Internal Fixation or Excision? J Am Acad Orthop Surg, 1997. 5(1): p. 1-10.

117. Hotchkiss, R.N. and A.J. Weiland, Valgus stability of the elbow. J Orthop Res, 1987. 5(3): p. 372-7.

118. Hunter, S.K., et al., Activation among the elbow flexor muscles differs when maintaining arm position during a fatiguing contraction. J Appl Physiol, 2003. 94(6): p. 2439-47.

119. Hutchins, E.L., The musculoskeletal geometry of the human elbow and wrist: an analysis using torque-angle relationships, in Engineering. 1993, University of Texas at Austin: Austin, TX. p. 88.

120. Ito, M., Long-term depression as a memory process in the cerebellum. Neurosci Res, 1986. 3(6): p. 531-9.

121. Ito, M., Long-term depression. Annu Rev Neurosci, 1989. 12: p. 85-102.

122. Ives, J.C., L. Abraham, and W. Kroll, Neuromuscular control mechanisms and strategy in arm movements of attempted supranormal speed. Res Q Exerc Sport, 1999. 70(4): p. 335-48.

123. Ives, J.C., W.P. Kroll, and L.L. Bultman, Rapid movement kinematic and electromyographic control characteristics in males and females. Res Q Exerc Sport, 1993. 64(3): p. 274-83.

124. Jagacinski, R.J. and J.M. Flach, Control theory for humans. 2003, Mahway, NJ: Lawrence Erlbaum Associates. 379.

125. Jan Nijhof, E. and D.A. Gabriel, Maximum isometric arm forces in the horizontal plane. J Biomech, 2006. 39(4): p. 708-16.

126. Jensen, S.L., et al., Laxity of the elbow after experimental excision of the radial head and division of the medial collateral ligament. Efficacy of ligament repair and radial head prosthetic replacement: a cadaver study. J Bone Joint Surg Br, 2003. 85(7): p. 1006-10.

127. Jo, S. and S.G. Massaquoi, A model of cerebellum stabilized and scheduled hybrid long-loop control of upright balance. Biol Cybern, 2004. 91(3): p. 188-202.

128. Jobbagy, A. and E.H. Furnee, Marker centre estimation algorithms in CCD camera-based motion analysis. Med Biol Eng Comput, 1994. 32(1): p. 85-91.

129. Johnson, G.W., A Follow-Up of One Hundred Cases of Fracture of the Head of the Radius with a Review of the Literature. Ulster Medicine Journal, 1952. 31: p. 51-56.

130. Johnson, J.A., et al., Simulation of elbow and forearm motion in vitro using a load controlled testing apparatus. J Biomech, 2000. 33(5): p. 635-9.

131. Jorgensen, K. and S. Bankov, Maximum strength of elbow flexors with pronated and supinated forearm, in Medicine and Sport, vol. 6: Biomechanics II. 1971, Karger: Basel. p. 174-180.

132. Kadaba, M.P., et al., Repeatability of phasic muscle activity: performance of surface and intramuscular wire electrodes in gait analysis. J Orthop Res, 1985. 3(3): p. 350-9.

133. Karniel, A. and G.F. Inbar, Human motor control: learning to control a time-varying, nonlinear, many-to-one system. Systems, Man and Cybernetics, Part C, IEEE Transactions on, 2000. 30(1): p. 1-11.

134. Kassubek, J., et al., Changes in cortical activation during mirror reading before and after training: an fMRI study of procedural learning. Brain Res Cogn Brain Res, 2001. 10(3): p. 207-17.

135. Kaucic, R., et al. A unified framework for tracking through occlusions and across sensor gaps. in Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on. 2005.

136. Kawakami, Y., et al., Specific tension of elbow flexor and extensor muscles based on magnetic resonance imaging. Eur J Appl Physiol Occup Physiol, 1994. 68(2): p. 139-47.

137. Kawato, M., K. Furukawa, and R. Suzuki, A hierarchical neural-network model for control and learning of voluntary movement. Biol Cybern, 1987. 57(3): p. 169-85.

138. Kepple, T.M., S.J. Stanhope, and A.H. Rich. The presentation and evaluation of a video based, six degree-of-freedom approach for analyzing human motion. in Engineering in Medicine and Biology Society, 1988. Proceedings of the Annual International Conference of the IEEE. 1988.

139. King, G.J., et al., Single-strand reconstruction of the lateral ulnar collateral ligament restores varus and posterolateral rotatory stability of the elbow. J Shoulder Elbow Surg, 2002. 11(1): p. 60-4.

140. King, G.J., D.C. Evans, and J.F. Kellam, Open reduction and internal fixation of radial head fractures. J Orthop Trauma, 1991. 5(1): p. 21-8.

141. King, G.J., et al., Motion and laxity of the capitellocondylar total elbow prosthesis. J Bone Joint Surg Am, 1994. 76(7): p. 1000-8.

142. King, G.J., et al., A standardized method for assessment of elbow function. Research Committee, American Shoulder and Elbow Surgeons. J Shoulder Elbow Surg, 1999. 8(4): p. 351-4.

143. King, G.J., et al., Metallic radial head arthroplasty improves valgus stability of the elbow. Clin Orthop, 1999(368): p. 114-25.

144. Kistemaker, D.A., A.J. Van Soest, and M.F. Bobbert, Is equilibrium point control feasible for fast goal-directed single-joint movements? J Neurophysiol, 2006. 95(5): p. 2898-912.

145. Kitamura, T., K. Matsuda, and H. Akashi, Adaptive control technique for artificial hearts. IEEE Trans Biomed Eng, 1986. 33(9): p. 839-44.

146. Klein, P., S. Mattys, and M. Rooze, Moment arm length variations of selected muscles acting on talocrural and subtalar joints during movement: an in vitro study. J Biomech, 1996. 29(1): p. 21-30.

147. Knight, D.J., et al., Primary replacement of the fractured radial head with a metal prosthesis. J Bone Joint Surg Br, 1993. 75(4): p. 572-6.

148. Kodek, T. and M. Munih, An identification technique for evaluating body segment parameters in the upper extremity from manipulator-hand contact forces and arm kinematics. Clin Biomech (Bristol, Avon), 2006. 21(7): p. 710-6.

149. Kopparapu, S. and P. Corke, The Effect of Noise on Camera Calibration Parameters. Graphical Models, 2001. 63(5): p. 277-303.

150. Kurosawa, K., et al., Joint angle control by FES using a feedback error learning controller. IEEE Trans Neural Syst Rehabil Eng, 2005. 13(3): p. 359-71.

151. Kuxhaus, L., et al. Reproducing physiologic moment arms with an elbow simulator. in Amer Soc Biomech. 2005. Cleveland, OH.

152. Lam, T., M. Anderschitz, and V. Dietz, Contribution of feedback and feedforward strategies to locomotor adaptations. J Neurophysiol, 2006. 95(2): p. 766-73.

153. Lemay, M.A. and P.E. Crago, A dynamic model for simulating movements of the elbow, forearm, an wrist. J Biomech, 1996. 29(10): p. 1319-30.

154. Liew, V.S., et al., The effect of metallic radial head arthroplasty on radiocapitellar joint contact area. Clin Biomech (Bristol, Avon), 2003. 18(2): p. 115-8.

155. Lin, G.T., et al., Functional anatomy of the human digital flexor pulley system. J Hand Surg [Am], 1989. 14(6): p. 949-56.

156. Liu, H., C. Holt, and S. Evans, Accuracy and repeatability of an optical motion analysis system for measuring small deformations of biological tissues. J Biomech, 2007. 40(1): p. 210-4.

157. Loftice, J., et al., Biomechanics of the elbow in sports. Clin Sports Med, 2004. 23(4): p. 519-30, vii-viii.

158. London, J.T., Kinematics of the elbow. J Bone Joint Surg Am, 1981. 63(4): p. 529-35.

159. Loren, G.J., et al., Human wrist motors: biomechanical design and application to tendon transfers. J Biomech, 1996. 29(3): p. 331-42.

160. Lujan, J.L. and P.E. Crago, Computer-based test-bed for clinical assessment of hand/wrist feed-forward neuroprosthetic controllers using artificial neural networks. Med Biol Eng Comput, 2004. 42(6): p. 754-61.

161. Lujan, T.J., et al., Simultaneous measurement of three-dimensional joint kinematics and ligament strains with optical methods. J Biomech Eng, 2005. 127(1): p. 193-7.

162. Madey, S.M., et al., Hinged external fixation of the elbow: optimal axis alignment to minimize motion resistance. J Orthop Trauma, 2000. 14(1): p. 41-7.

163. Magnusen, J.P., Design and fabrication of an elbow motion simulator, in Mechanical Engineering. 2004, University of Pittsburgh: Pittsburgh, PA. p. 70.

164. Maletsky, L.P. and B.M. Hillberry, Simulating dynamic activities using a five-axis knee simulator. J Biomech Eng, 2005. 127(1): p. 123-33.

165. Maletsky, L.P., J. Sun, and N.A. Morton, Accuracy of an optical active-marker system to track the relative motion of rigid bodies. J Biomech, 2007. 40(3): p. 682-5.

166. Malis, E. and A. Bartoli. Euclidean reconstruction independent on camera intrinsic parameters. in Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on. 2004.

167. Markolf, K.L., et al., Load-sharing at the wrist following radial head replacement with a metal implant. A cadaveric study. J Bone Joint Surg Am, 2004. 86-A(5): p. 1023-30.

168. Massaquoi, S.G., LinkModeling the function of the cerebellum in scheduled linear servo control of simple horizontal planar arm movements, in Electrical Engineering. 1999, MIT: Boston. p. 240.

169. Matheson Rittenhouse, D., et al., A neural network model for reconstructing EMG signals from eight shoulder muscles: Consequences for rehabilitation robotics and biofeedback. J Biomech, 2005.

170. McGill, S.M., N. Patt, and R.W. Norman, Measurement of the trunk musculature of active males using CT scan radiography: implications for force and moment generating capacity about the L4/L5 joint. J Biomech, 1988. 21(4): p. 329-41.

171. McLean, C.A. and A.M. Ahmed, Design and development of an unconstrained dynamic knee simulator. J Biomech Eng, 1993. 115(2): p. 144-8.

172. Melhorn, J.M., Cumulative trauma disorders and repetitive strain injuries. The future. Clin Orthop Relat Res, 1998(351): p. 107-26.

173. Michel, D. Learning from human's strategies of motor control: a challenge for robotic systems design. in Robot and Human Communication, 1996., 5th IEEE International Workshop on. 1996.

174. Mikic, Z.D. and S.M. Vukadinovic, Late results in fractures of the radial head treated by excision. Clin Orthop Relat Res, 1983(181): p. 220-8.

175. Miller, C., A. Mulavara, and J. Bloomberg, A quasi-static method for determining the characteristics of a motion capture camera system in a "split-volume" configuration. Gait Posture, 2002. 16(3): p. 283-7.

176. Miller, M.C., et al. The effect of annular ligament resection of radial head kinematics. in ASME Bioengineering. 2005. Vail, Colorado.

177. Miller, M.C., et al. Transection of the annular ligament affects radial head travel but not the axis location in pronation-supination of the forearm. in ORS. 2006. Chicago, IL.

178. Miller, M.C., et al. Translation and travel of monoblock and bipolar radial head replacements during supination-pronation. in ORS. 2006. Chicago, IL.

179. Miller, M.C., et al., The effect of component placement on knee kinetics after arthroplasty with an unconstrained prosthesis. J Orthop Res, 2001. 19(4): p. 614-20.

180. Moro, J.K., et al., Arthroplasty with a metal radial head for unreconstructible fractures of the radial head. J Bone Joint Surg Am, 2001. 83-A(8): p. 1201-11.

181. Morrey, B.F., Complex instability of the elbow. Instr Course Lect, 1998. 47: p. 157-64.

182. Morrey, B.F., Radial head fracture, in The elbow and its disorders. 2000, W.B. Saunders: Philadelphia. p. 341-364.

183. Morrey, B.F. and K.N. An, Articular and ligamentous contributions to the stability of the elbow joint. Am J Sports Med, 1983. 11(5): p. 315-9.

184. Morrey, B.F. and K.N. An, Functional anatomy of the ligaments of the elbow. Clin Orthop Relat Res, 1985(201): p. 84-90.

185. Morrey, B.F., L. Askew, and E.Y. Chao, Silastic prosthetic replacement for the radial head. J Bone Joint Surg Am, 1981. 63(3): p. 454-8.

186. Morrey, B.F., L.J. Askew, and E.Y. Chao, A biomechanical study of normal functional elbow motion. J Bone Joint Surg Am, 1981. 63(6): p. 872-7.

187. Morrey, B.F. and E.Y. Chao, Passive motion of the elbow joint. J Bone Joint Surg Am, 1976. 58(4): p. 501-8.

188. Morrey, B.F., E.Y. Chao, and F.C. Hui, Biomechanical study of the elbow following excision of the radial head. J Bone Joint Surg Am, 1979. 61(1): p. 63-8.

189. Morrey, B.F., S. Tanaka, and K.N. An, Valgus stability of the elbow. A definition of primary and secondary constraints. Clin Orthop, 1991(265): p. 187-95.

190. Mühlich, M., The role of total least squares in motion analysis, in Lecture notes in computer science: Computer Vision - ECCV 1998. 1998, Springer Berlin: Berlin. p. 305.

191. Murray, R., Fractures of the head and neck of the radius. Br. J. Surg, 1940. 28: p. 106.

192. Murray, W.M., The functional capacity of the elbow muscles: anatomical measurements, computer modeling, and anthropometric scaling, in Biomedical Engineering. 1997, Northwestern University: Evanston, IL. p. 149.

193. Murray, W.M., T.S. Buchanan, and S.L. Delp, Scaling of peak moment arms of elbow muscles with upper extremity bone dimensions. J Biomech, 2002. 35(1): p. 19-26.

194. Murray, W.M., S.L. Delp, and T.S. Buchanan, Variation of muscle moment arms with elbow and forearm position. J Biomech, 1995. 28(5): p. 513-25.

195. Myers, J.B., et al., Three dimensional scapular movements in throwing athletes. Am J Sports Med, 2005. 33(2): p. 263-71.

196. Nemeth, G. and H. Ohlsen, In vivo moment arm lengths for hip extensor muscles at different angles of hip flexion. J Biomech, 1985. 18(2): p. 129-40.

197. Nguyen, Q.T. and D. Kleinfeld, Positive feedback in a brainstem tactile sensorimotor loop. Neuron, 2005. 45(3): p. 447-57.

198. Ogata, K., Modern Control Engineering. 3rd ed. 1997, Upper Saddle River, NJ: Prentice-Hall, Inc. 997.

199. Olsen, B.S., et al., Posterolateral elbow joint instability: the basic kinematics. J Shoulder Elbow Surg, 1998. 7(1): p. 19-29.

200. Olsen, B.S., et al., Lateral collateral ligament of the elbow joint: anatomy and kinematics. J Shoulder Elbow Surg, 1996. 5(2 Pt 1): p. 103-12.

201. Otis, J.C., et al., Changes in the moment arms of the rotator cuff and deltoid muscles with abduction and rotation. J Bone Joint Surg Am, 1994. 76(5): p. 667-76.

202. Park, M.C. and C.S. Ahmad, Dynamic contributions of the flexor-pronator mass to elbow valgus stability. J Bone Joint Surg Am, 2004. 86-A(10): p. 2268-74.

203. Park, S. and J.K. Hodgins, Capturing and animating skin deformation in human motion. ACM Transactions on Graphics (SIGGRAPH 2006), 2006. 25(3).

204. Pigeon, P., L. Yahia, and A.G. Feldman, Moment arms and lengths of human upper limb muscles as functions of joint angles. J Biomech, 1996. 29(10): p. 1365-70.

205. Pomianowski, S., et al., Contribution of monoblock and bipolar radial head prostheses to valgus stability of the elbow. J Bone Joint Surg Am, 2001. 83-A(12): p. 1829-34.

206. Pribanic, T., P. Sturm, and M. Cifrek. Camera parameter initialization for 3D kinematic systems. in Image and Signal Processing and Analysis, 2005. ISPA 2005. Proceedings of the 4th International Symposium on. 2005.

207. Pribyl, C.R., et al., The effect of the radial head and prosthetic radial head replacement on resisting valgus stress at the elbow. Orthopedics, 1986. 9(5): p. 723-6.

208. Prochazka, A., D. Gillard, and D.J. Bennett, Implications of positive feedback in the control of movement. J Neurophysiol, 1997. 77(6): p. 3237-51.

209. Prochazka, A., D. Gillard, and D.J. Bennett, Positive force feedback control of muscles. J Neurophysiol, 1997. 77(6): p. 3226-36.

210. Rainis, E.J., Characterizing the mechanical properties of the glenohumeral capsule: implications for finite element modeling., in Bioengineering. 2007, University of Pittsburgh: Pittsburgh. p. 147.

211. Reich, J. and W.J. Daunicht, A rigid body model of the forearm. J Biomech, 2000. 33(9): p. 1159-68.

212. Richards, J.G., The measurement of human motion: A comparison of commercially available systems. Human Movement Science, 1999. 18(5): p. 589-602.

213. Riddle, C.N. and S.N. Baker, Manipulation of peripheral neural feedback loops alters human corticomuscular coherence. J Physiol, 2005. 566(Pt 2): p. 625-39.

214. Ring, D., J.B. Jupiter, and J. Zilberfarb, Posterior dislocation of the elbow with fractures of the radial head and coronoid. J Bone Joint Surg Am, 2002. 84-A(4): p. 547-51.

215. Ring, D., J. Quintero, and J.B. Jupiter, Open reduction and internal fixation of fractures of the radial head. J Bone Joint Surg Am, 2002. 84-A(10): p. 1811-5.

216. Sampath, G., et al. Design and development of an active marker based system for analysis of 3-D pediatric foot and ankle motion. in Engineering in Medicine and Biology Society, 1998. Proceedings of the 20th Annual International Conference of the IEEE. 1998.

217. Sauerbier, M., et al., The dynamic radioulnar convergence of the Darrach procedure and the ulnar head hemiresection interposition arthroplasty: a biomechanical study. J Hand Surg [Br], 2002. 27(4): p. 307-16.

218. Schimoler, P., et al. Control System for an Elbow Joint Motion Simulator. in IMECE. 2007. Seattle, WA.

219. Scott, S.H., Optimal feedback control and the neural basis of volitional motor control. Nat Rev Neurosci, 2004. 5(7): p. 532-46.

220. Seireg, A. and R. Arkivar, Biomechanical analysis of the musculoskeletal structure for medicine and sports. 1989, New York: Hemisphere Pub. Corp. 818.

221. Shadmehr, R., Control of Equilibrium Position and Stiffness Through Postural Modules. J Mot Behav, 1993. 25(3): p. 228-241.

222. Sharkey, N.A. and A.J. Hamel, A dynamic cadaver model of the stance phase of gait: performance characteristics and kinetic validation. Clin Biomech (Bristol, Avon), 1998. 13(6): p. 420-433.

223. Sheridan, T.B., Man-Machine Systems: Information, Control, and Decision Models of Human Performance. 1974, Cambridge, MA: MIT Press.

224. Singh, K. and S.H. Scott, A motor learning strategy reflects neural circuitry for limb control. Nat Neurosci, 2003. 6(4): p. 399-403.

225. Skalski, K., et al., Radial head prosthesis with a mobile head. J Shoulder Elbow Surg, 2004. 13(1): p. 78-85.

226. Soderkvist, I. and P.A. Wedin, Determining the movements of the skeleton using well-configured markers. J Biomech, 1993. 26(12): p. 1473-7.

227. Sojbjerg, J.O., J. Ovesen, and C.E. Gundorf, The stability of the elbow following excision of the radial head and transection of the annular ligament. An experimental study. Arch Orthop Trauma Surg, 1987. 106(4): p. 248-50.

228. Song, R. and K.Y. Tong, Using recurrent artificial neural network model to estimate voluntary elbow torque in dynamic situations. Med Biol Eng Comput, 2005. 43(4): p. 473-80.

229. Soutas-Little, R.W. and D.J. Inman, Engineering Mechanics: Statics. 1998, Upper Saddle River, NJ: Prentice-Hall. 564.

230. Soutas-Little, R.W. and D.J. Inman, Engineering Mechanics: Dynamics. 1999, Upper Saddle River, NJ: Prentice-Hall. 528.

231. Speed, K., Ferrule caps for the head of the radius. Surg Gynecol Obstet, 1941. 73: p. 845–850.

232. Spoor, C.W., et al., Estimation of instantaneous moment arms of lower-leg muscles. J Biomech, 1990. 23(12): p. 1247-59.

233. Spoor, C.W. and F.E. Veldpaus, Rigid body motion calculated from spatial co-ordinates of markers. J Biomech, 1980. 13(4): p. 391-3.

234. Stagni, R., et al., Quantification of soft tissue artefact in motion analysis by combining 3D fluoroscopy and stereophotogrammetry: a study on two subjects. Clin Biomech (Bristol, Avon), 2005. 20(3): p. 320-9.

235. Stein, G.P. Accurate internal camera calibration using rotation, with analysis of sources of error. in Computer Vision, 1995. Proceedings., Fifth International Conference on. 1995.

236. Stokes, I.A. and M.G. Gardner-Morse, Strategies used to stabilize the elbow joint challenged by inverted pendulum loading. J Biomech, 2000. 33(6): p. 737-43.

237. St-Onge, N., S.V. Adamovich, and A.G. Feldman, Control processes underlying elbow flexion movements may be independent of kinematic and electromyographic patterns: experimental study and modelling. Neuroscience, 1997. 79(1): p. 295-316.

238. Stroeve, S., A learning feedback and feedforward neuromuscular control model for two degrees of freedom human arm movements. Human Movement Science, 1997. 16: p. 621-651.

239. Sutherland, D.H., The evolution of clinical gait analysis. Part II kinematics. Gait Posture, 2002. 16(2): p. 159-79.

240. Suzuki, M., et al. Reformulation of dynamics equation of planar two-joint arm movements. in Systems, Man, and Cybernetics, 1999. IEEE SMC '99 Conference Proceedings. 1999 IEEE International Conference on. 1999.

241. Swanson, A.B., S.H. Jaeger, and D. La Rochelle, Comminuted fractures of the radial head. The role of silicone-implant replacement arthroplasty. J Bone Joint Surg Am, 1981. 63(7): p. 1039-49.

242. Taylor, D.M., S.I. Tillery, and A.B. Schwartz, Direct cortical control of 3D neuroprosthetic devices. Science, 2002. 296(5574): p. 1829-32.

243. Taylor, K.D., et al., An automated motion measurement system for clinical gait analysis. J Biomech, 1982. 15(7): p. 505-16.

244. Thomas, T.T., Fractures of the head of the radius. Univ. Pa. Med. Bull., 1905. 18: p. 184-221.

245. Todorov, E. and M.I. Jordan, Optimal feedback control as a theory of motor coordination. Nat Neurosci, 2002. 5(11): p. 1226-35.

246. Tomaino, M.M., et al., Reconstruction of the interosseous ligament of the forearm reduces load on the radial head in cadavers. J Hand Surg [Br], 2003. 28(3): p. 267-70.

247. Trousdale, R.T., et al., Radio-ulnar dissociation. A review of twenty cases. J Bone Joint Surg Am, 1992. 74(10): p. 1486-97.

248. Unal, G. and A. Yezzi. A variational approach to problems in calibration of multiple cameras. in Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on. 2004.

249. van Riet, R.P., et al., The effect of the orientation of the noncircular radial head on elbow kinematics. Clin Biomech (Bristol, Avon), 2004. 19(6): p. 595-9.

250. van Riet, R.P., et al., The effect of the orientation of the radial head on the kinematics of the ulnohumeral joint and force transmission through the radiocapitellar joint. Clin Biomech (Bristol, Avon), 2006. 21(6): p. 554-9.

251. van Riet, R.P., et al., The noncircular shape of the radial head. J Hand Surg [Am], 2003. 28(6): p. 972-8.

252. van Zuylen, E.J., A. van Velzen, and J.J. Denier van der Gon, A biomechanical model for flexion torques of human arm muscles as a function of elbow angle. J Biomech, 1988. 21(3): p. 183-90.

253. Vander Linden, D.W., S.J. Carison, and R.L. Hubbard, Reproducibility and accuracy of angle measurements obtained under static conditions with the Motion Analysis video system. Physical Therapy, 1992. 72(4): p. 300-305.

254. Vanderwilde, R.S., et al., Inflammatory arthritis after failure of silicone rubber replacement of the radial head. J Bone Joint Surg Br, 1994. 76(1): p. 78-81.

255. Vangsness, C.T., Jr., et al., The origin of the long head of the biceps from the scapula and glenoid labrum. An anatomical study of 100 shoulders. J Bone Joint Surg Br, 1994. 76(6): p. 951-4.

256. Veeger, H.E.J., B. Yu, and K.-N. An. Orientation of Axes in the Elbow and Forearm for Biomechanical Modeling. in First Conference of the International Shoulder Group. 1997: Maastricht, Delft.

257. Visser, J.J., et al., Length and moment arm of human leg muscles as a function of knee and hip-joint angles. Eur J Appl Physiol Occup Physiol, 1990. 61(5-6): p. 453-60.

258. Viviani, P. and T. Flash, Minimum-jerk, two-thirds power law, and isochrony: converging approaches to movement planning. J Exp Psychol Hum Percept Perform, 1995. 21(1): p. 32-53.

259. Weinberg, A.M., et al., A new kinematic model of pro- and supination of the human forearm. J Biomech, 2000. 33(4): p. 487-91.

260. Weiss, A.-P., C. and H. Hastings II, The Anatomy of the Proximal Radioulnar Joint. Journal of Shoulder and Elbow Surgery, 1992. 1(4): p. 193-199.

261. Welk, M., et al., Median and related local filters for tensor-valued images. Tensor Signal Processing, 2007. 87(2): p. 291-308.

262. Werner, F.W., et al., Wrist joint motion simulator. J Orthop Res, 1996. 14(4): p. 639-46.

263. Williams, S., et al., An upper body model for the kinematical analysis of the joint chain of the human arm. J Biomech, 2005.

264. Winters, J., L. Stark, and A.H. Seif-Naraghi, An analysis of the sources of musculoskeletal system impedance. J Biomech, 1988. 21(12): p. 1011-25.

265. Winters, J.M. and D.G. Kleweno, Effect of initial upper-limb alignment on muscle contributions to isometric strength curves. J Biomech, 1993. 26(2): p. 143-53.

266. Winters, J.M. and L. Stark, Estimated mechanical properties of synergistic muscles involved in movements of a variety of human joints. J Biomech, 1988. 21(12): p. 1027-41.

267. Woltring, H., Data Processing and Error Analysis, in Biomechanics of Human Movement, N. Berme and A. Capozzo, Editors. 1990, bertec Corporation: Worthington. p. 203-237.

268. Yamane, K., T. Kuroda, and Y. Nakamura. High-precision and high-speed motion capture combining heterogeneous cameras. in Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on. 2004.

269. Youm, Y., et al., Biomechanical analyses of forearm pronation-supination and elbow flexion-extension. J Biomech, 1979. 12(4): p. 245-55.