3D Musculoskeletal Modeling of the Human Lower Extremity

by

Jasmine C. Feddema

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science

Faculty of Kinesiology, Sport, and Recreation

University of Alberta

Abstract

Modeling of human musculature is of interest to anatomists and biomechanists for its ability to predict human motion. In current models, muscles are portrayed as connecting lines starting at the origin and ending at the insertion, however this modeling often does not account for the wrapping that occurs at muscles like the vasti. In particular, the vastus medialis oblique, has a unique oblique orientation, which may indicate a specialized action on the femur. Additionally, the specific geometry of the bones is not typically incorporated in current models. The omission of proper musculoskeletal geometry from models limits their ability to accurately predict muscular action and human motion. The purpose of this study was to develop a comprehensive three-dimensional model that included specific bone geometry and proper muscle paths, that could determine the contribution of the vasti muscles to a moment about the longitudinal axis of the femur.

It was hypothesized that the vastus medialis oblique would exert the highest moment about the longitudinal axis of the femur because of its oblique fiber orientation. Specific bone geometry was collected using three sets of lower extremity cadaver bones with surface geometry captured using three-dimensional photogrammetry. Ten participants were recruited to perform maxima vertical jumps to obtain knee joint kinematics and kinetics. They also performed maximal isometric knee extension tests to determine the upper boundary for the knee extensor moment. The digitized bony landmarks of each cadaver model were transformed to follow the limb kinematics of each participant's vertical jumps, and forces and moments experienced at the patella and femur were calculated. It was found that most of the moments exerted by the vasti muscles about the longitudinal axis of the femur were meaningful (>0.12 Nm/Kg), although the vastus medialis oblique alone did not exert a meaningful moment (<0.06 Nm/Kg). A net internal rotation moment was found, driven by the vastus intermedius and the vastus lateralis (≥0.10 Nm/Kg). When comparing bone models, it was found that there were meaningful differences between models that highlighted the importance of bone geometry on the force and moment production. The most unexpected finding was that the muscles attached to the patella were all pulling medially at the highest incidence of knee extensor moment, alluding to another counter force other than the patellar ligament, quadriceps tendon, and vastus medialis oblique tendon that is active within the knee joint and requiring further investigation. The findings in this study are beneficial for progression of musculoskeletal modeling techniques and uncover further insight about the functionality of the knee extensor mechanism.

Preface

This thesis is an original work by Jasmine Feddema. The research project, of which this thesis is a part, received research ethics approval from the University of Alberta Research Ethics Board, titled "3D MUSCULOSKELETAL MODELING OF THE HUMAN LOWER EXTREMITY" study identification Pro00135286, on October 6th, 2023.

Acknowledgements

I would like to express my gratitude to everyone who has contributed to the completion of this piece of work. My supervisor, Dr. Loren Chiu, who was responsible for the generation of the premise of this project, and co-supervisor Dr. Michael Kennedy for his support and interest in the project. I am thankful for Hugh Barret and Jason Papirny who were greatly supportive of our anatomical research and moved mountains to help with whatever crazy new idea we had. Thank you to my undergraduate assistants Claire Hopkins, Jane Hopkins, Emma Klein, Shelby Shmidt, and Lauryn Tremblay, for all your help, and your company when things got difficult. My external examiner Dr. Lauren Welte, and my Chairman, Dr. Kelvin Jones.

Thank you to my research participants for volunteering their time and giving their best effort in whatever task was given to them.

I am also beyond grateful for my family and teammates. You know who you are, and my special words of gratitude will be reserved for them alone.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

RF= Rectus Femoris

VML= Vastus Medialis Longus

VMO= Vastus Medialis Oblique

VI= Vastus Intermedius

PCSA= Physiological Cross- Sectional Area

ACSA= Anatomical Cross-Sectional Area

PL= Patellar Ligament

QT= Quadriceps Tendon

SSM= Statistical Shape Model

CT= Computed Tomography

MRI= Magnetic Resonance Imagery

Glossary of Terms

Global Coordinate System: A fixed three-dimensional reference frame defined by orthogonal X, Y, and Z axes, used to describe the position and orientation of rigid bodies.

Joint Angle: The angle between the proximal and distal segment related to the joint of interest.

Local Coordinate System: The coordinate system related to a segment within a global coordinate system that defines the position and orientation of a segment in the global coordinate system.

Lower Extremity: Consists of the femur, leg, patella, and foot, upon the pelvis.

Moment: The force acting perpendicular to a measured distance from an axis of rotation.

Moment Arm: The distance between the axis of rotation and a force acting perpendicular to this vector.

Physiological Cross-Sectional Area: The cross-sectional area of a muscle that is perpendicular to its fibers.

Segment Angle: Angular Orientation of a rigid body in 3D space relative to a global coordinate system.

Unit Vector: A vector with a magnitude of one whose components describe the directional contribution of a related vector.

Vector: A quantity that expresses magnitude and direction.

Chapter 1: Introduction

1.1. Motivation

In the field of biomechanics, the pursuit of developing a fully functioning and accurate computerized musculoskeletal model is coveted [1], [2]. The completion of such a model could result in substantial improvements in numerous fields such as sports performance, injury reduction, education, and more. This goes beyond evaluation of current movement and opens the door to predictive analysis by allowing a computer to determine effectiveness of different techniques and methods of action.

Although the topic has been well explored, and current models of the human musculoskeletal system are advanced, there are still shortcomings in their development that can be improved upon. Early models were often created as "stick figures" with string-like structures acting upon the extremities in the form of muscles. The modelling of bones as dimensionless lines ignores the unique and complex geometry of human bones, which are integral to the specialized directionality of muscles. Additionally, modelling muscles as strings indicate a unidirectional contraction of the muscle and discard the presence of pennation, which causes multi-directional vectors. It also assumes a straight line pull from origin to insertion, which ignores wrapping of muscles around bones that changes its line of action, and force application [3].

Researchers have attempted to address these issues by creating three-dimensional (3D) bone structures using geometrical shapes from digitization or medical images to mimic their influence on muscle lines of action [4]. They may also include additional supplementary shapes such as semicircles, ellipsoids, or cylinders for muscles to wrap about [5]. Magnetic resonance imagery (MRI) and computed tomography (CT) have been used to obtain specific bony geometry through non-invasive measures [6]. The downfall of these methods is that finer defining characteristics of the bones are overlooked, while gross anatomical features such as the greater and lesser trochanters, and the linea aspera are retained. Loss of important fine characteristics arises from the scan thickness used in CT and MRI [7]. For a given thickness of MRI or CT slice, a smaller thickness of data is actually collected, and geometric properties are extrapolated. A thin slice will give a sharper image, but less information will be included, whereas a thicker slice will have a less sharp image but include more information for a larger part of the body. However, there is a space kept between slices to prevent crosstalk between images that may cause distortion. This results in the loss of information within these spaces, which is where extrapolation is most pertinent, and to a loss of detail that is crucial for anatomical research. Details found on the bony surface are important to investigating muscular action. Deviations in shape from a general bone shape can be evidence of muscular action since muscular pull can cause stress on the bone and cause adaptations to support the demands of the muscle [8]. Additionally, whole body scans using MRI and CT are expensive, and in the case of CT, exposes participants to unreasonable amounts of radiation due to the time needed to take a full body scan. For this reason, CT and MRI information is usually collected for specific regions of the body (hip, knee, etc.).

Inclusion of precise bone geometry could enhance the accuracy of musculoskeletal models. We propose that the creation of computerized 3D bone models by use of 3D photogrammetry and cadaver bones could improve upon previous model properties. Use of 3D photogrammetry to model bones is reliable and accurate [9]Because the bones are extracted from a cadaver, the opportunity to observe the separate muscles more intimately than a scan allows for proper investigation into the functions of the muscle based on their attachment points and natural separations. Furthermore, observation of the specific muscle attachment points aids in a biomechanical analysis of the function of the muscles that is not limited to the explanations in anatomy textbooks. This method could more accurately estimate the direction of forces exerted by muscles on bones because the musculature would follow its proper path around an anatomically correct shape as opposed to a generically generated one. The importance of this is highlighted in muscles such as the vasti which wrap around the femur and may generate force along a curved path that cannot be well described without inclusion of bony geometry and path constraints [10].

Clinically, the vasti muscle that receives higher levels of attention is the vastus medialis, due to its proposed division into the vastus medialis longus (VML) and vastus medialis oblique (VMO). Observation of cadaveric vastus medialis muscles shows a difference in fiber orientation between the two muscles, with the VML fibers running more parallel to the femoral shaft, and the VMO fibers running obliquely and horizontally relative to the femoral shaft. This difference suggests that distinct lines of action may be present for each part of the vastus medialis which would influence the moments exerted on the femur, leading to an interest in modelling the muscle.

Although its function has been long debated, previous research has mainly focused on the sagittal and frontal plane actions of the vastus medialis on the patella, yet the transverse plane action upon the femur has been ignored. The primary goal output of this model is to elucidate the function of the vastus medialis oblique through investigation into the 3D forces experienced in the transverse plane. The model is integral to this finding because of the emphasis on the proper path of the muscle.

The purpose of this project is to create a 3D model of the human lower extremity by combining bone geometry models and muscle measurements, and functional movement motion capture to determine the function of the vasti. There is special interest in the investigation of the anatomical and functional individuality of the vastus medialis oblique and vastus medialis longus. Understanding the functions of these muscles will broaden our knowledge of the musculoskeletal system and aid in the discovery of new therapeutic, performance, and training techniques that optimize the usage of this muscle. The ultimate hope is for the methods used to be accessible by all for individual and academic use.

1.2. Objectives

The objective of this research is to create a functional 3D model of the human lower extremity using inputs from the primary photogrammetry research, and from 3D motion analysis data. Collection of 3D motion analysis is the final factor in developing an accurate model in conjunction with the bone models. The specific objectives of the project are listed below:

1. Capture 3D motion analysis data of a common task using corresponding bony landmarks to the photogrammetry outputs to obtain kinematic information of human movement.
2. Use photogrammetry outputs, 3D motion analysis, and maximal isometric contraction to analyze forces and moments at the knee joint.
3. Determine the moments about the femur's longitudinal axis exerted by the vasti.

Chapter 2: Background

This chapter highlights previous research about the femur, the quadriceps, and musculoskeletal modelling. It concludes with a purpose statement outlining the goals of using musculoskeletal modelling to investigate the vastus medialis oblique.

2.1. Anatomy of the Thigh

2.1.1. Femoral Geometry

An understanding of femoral geometry is important for grasping its relevance in the function of the vasti muscles. The shape of the femur is directly related to its function, which guides the hypothesis that the vastus medialis oblique acts transversely on the femur. A diagram of the femur can be found in Figure 2-1.

The femur is a long bone that contains a shaft with a femoral head and trochanters on the proximal end, and condyles on the distal end [11]. The femoral head is on the medial side and protrudes from the femoral neck like a mushroom that could be described to be two thirds of a sphere. This portion of the femur articulates with the pelvis in the acetabular fossa, and a small groove can be found on the femoral head where the ligament to anchor this connection attaches. The smoothness and roundness of the femoral head contributes to low friction experienced in the hip joint. On the opposing lateral side of the proximal femur, the greater and lesser trochanter protrude and serve mainly as gluteal region and adductor muscle attachment sites respectively.

The shaft of the femur contains significant bony landmarks predominately on the posterior side. The linea aspera is a raised prominence running along the femoral shaft and is considered to be the proximal attachment location for the vastus medialis and vastus lateralis muscles, as well as adductor group and posterior thigh muscles. At the distal end of the shaft, there are medial and lateral condyles which are rounded at the end to articulate with the tibia and patella. On the anterior surface between the condyles, there is a small groove intended for the patella to sit in.

The femur does not project directly downwards from the pelvis, but instead creates a triangle with the femoral neck and shaft. This shape is created by the femoral neck protruding laterally from the pelvic hip joint, with the most lateral aspect of the femur, the greater trochanter, being physically lateral to the knee joint when considered in anatomical position. The angle between the femoral neck and the femoral shaft is on average 125 degrees [11]. The femoral shaft is angled medially to meet with the knee joint and tibia, creating an angled mechanical axis in relation to the femoral shaft axis. Previously, most research has ignored the mechanical axis and treated the femoral shaft as the longitudinal axis of the femur. As the vastus lateralis and vastus medialis wrap around the femur their orientation would create a transverse plane action. However, these actions would be different depending on whether the mechanical axis versus the femoral shaft axis is defined as the transverse plane axis.

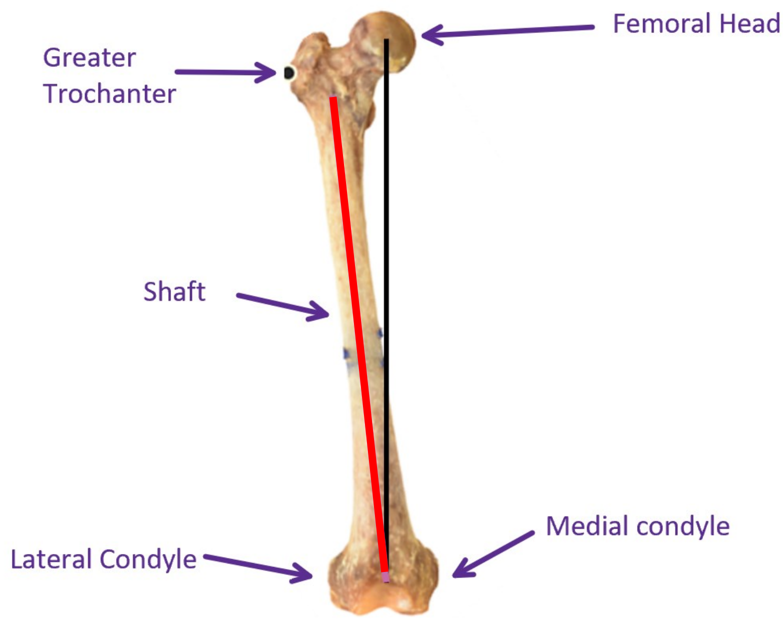


Figure 2-1:Femoral geometry and the interpretation of the mechanical axis of the femur corresponding with the longitudinal axis labelled in black, as opposed to the femoral axis labelled in red.

2.1.2. Quadriceps Anatomy

The quadriceps are made up of four muscles: the rectus femoris, vastus lateralis, vastus medialis, and the vastus intermedius [10]. The rectus femoris is the most superficial of the four muscles and is a fusiform bipennate muscle. It is the only quadricep muscle that crosses two joints and does not directly attach to the femur. The vastus lateralis is typically the largest of the four muscles and is on the lateral side of the anterior thigh. Its proximal attachment is broad, covering the lateral lip of the linea aspera, the anterior intertrochanteric line, the greater trochanter, and may arise from portions of the gluteal tendons. This broad proximal attachment creates a much fleshier mass when compared to the rectus femoris which culminates into a thickened tendon attaching to the lateral border of the patella and contributing to the patellar tendon that attaches to the tibial tuberosity. The vastus medialis lies on the medial side of the anterior thigh and is similar to the vastus lateralis with a broad proximal attachment location upon the intertrochanteric line, medial lip of the linea aspera, and in some cases the adductor magnus tendons. The fibers are directionally anterior and downward, attaching to the medial border of the patella and into the patellar tendon. The vastus intermedius arises broadly from the surface of the anterior and outer shaft on the proximal half, which eventually distally inserts into the base of the patella and the tibial tuberosity via the patellar tendon. It lies deep to the rectus femoris and is closely bound to both the vastus lateralis and vastus medialis, often blending fibers with the vastus medialis.

Although each muscle has a different orientation from one another, their action is known to be upon the bones that they attach to. The rectus femoris does not act upon the femur because it does not directly attach to it, but rather attaches proximally to the pelvis on the anterior inferior iliac spine. Therefore, the rectus femoris is not involved in femoral action, but rather flexion of the pelvis at the hip, and extension of the leg at the knee. The vasti are mainly recognized for leg extension when the thigh is fixed, as seen in a leg extension machine. However, it is important to also recognize that the vasti will extend the thigh when the leg is fixed. It is often overlooked that the vasti muscles can act upon the femur in a proximal to distal action, where the leg is fixed, and the thigh is moving. An example of this may be a sit to stand, where the foot and leg is planted, and the thigh is moving relative to the stationary leg.

Additionally, because of the wrapping of the vasti around the femur, there could be action in the transverse plane applied to the femur when the leg is fixed. This is a concept that garners little attention, and with proper analysis can uncover the full extent of the action of the vasti muscles.

2.2. Vastus Medialis Oblique Function

The vastus medialis has been debated to have two subsections, the vastus medialis longus (VML), and the vastus medialis oblique (VMO). Furthermore, these subsections can be argued to have separate functions upon the knee joint and patella. Although the vastus medialis is well established as an extensor of the leg at the knee, observation of the fiber orientation has led anatomists to study the muscle in more depth.

Upon observation of cadaveric vasti muscles, the more medially located vastus medialis fibers can be seen to attach to the patella upon the medial border, while the more laterally located fibers attach at the base of the patella [12]. The orientations of the fibers in the distinct portions of the muscles are unique from one another, where the medially located fibers are oriented obliquely into the patella, whereas the laterally located fibers are oriented vertically into the patella. The difference in orientation of the medial and lateral fibers of the vastus medialis, of oblique and longitudinal orientations respectively, introduced the concept of two separate portions of the muscles. This separation being the vastus medialis oblique (medially located), and the vastus medialis longus (laterally located). Each portion of the muscle has distinct origins and insertions, which explains the difference in fiber orientation.

The fiber orientation of the more medially located vastus medialis fibers have been hypothesized to act upon the patella by pulling posteriorly and anchoring the patella onto the knee joint, while contributing to knee extension [13]. An illustration of the expected muscular pull upon the patella can be seen below in Figure 2-2. However, the concept of the subsections have been considered to be a functional separation, rather than a physical one because of the inability to establish a physical fascial plane [14]. Despite this, there has been interest in identifying a fascial plane between the vastus medialis longus and vastus medialis oblique to anatomically distinguish the different muscles. But there has been minimal evidence of a clear division, leading earlier research to believe there was no separation at all [15]. However, the presence of obliquely oriented fibers has prevailed as the determining characteristic and evidence of a vastus medialis oblique and vastus medialis longus existence [16].

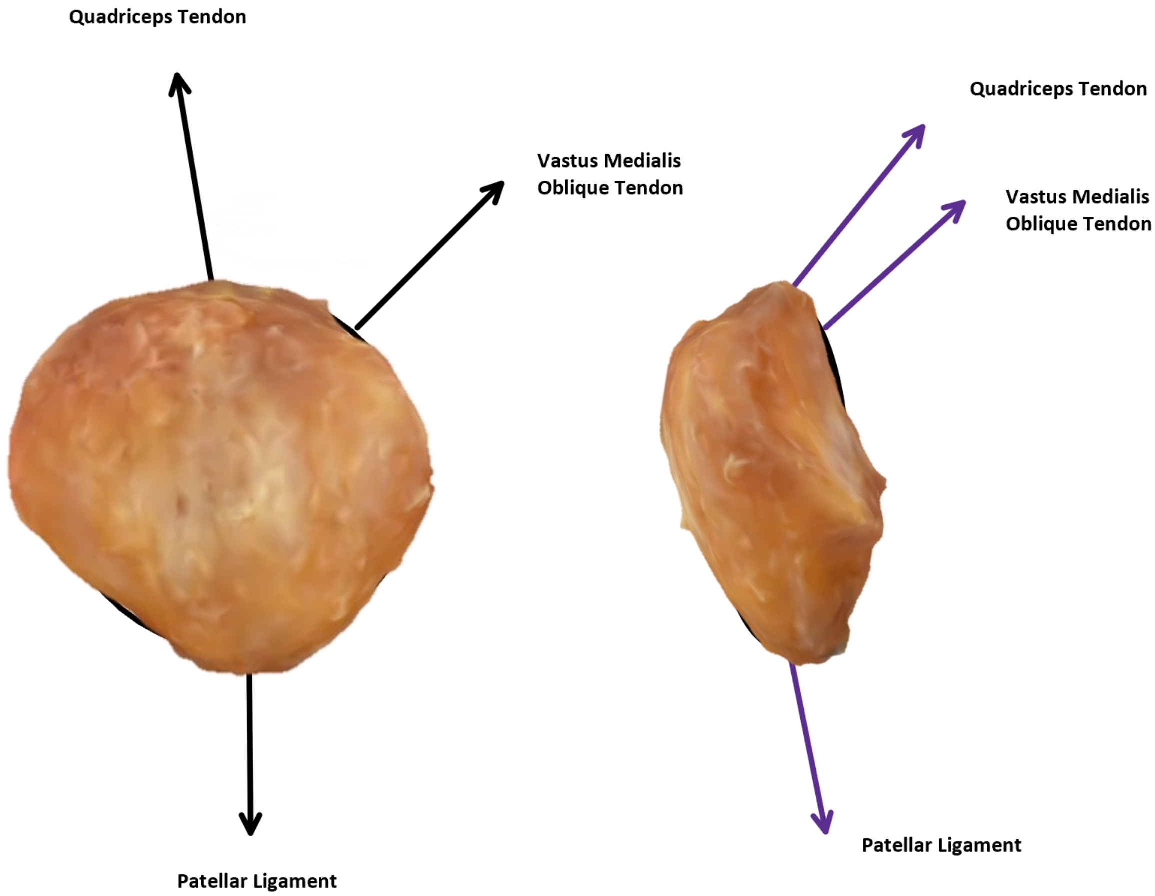


Figure 2-2: Muscular vectors upon the patella in frontal plane (left) and sagittal plane (right).

These observations have led to findings supporting the function of the vastus medialis oblique upon the patella and its role in patellar tracking [17]. A unique role of medial support against lateral strain, and subsequent prevention of patellofemoral syndrome has been elucidated from this information and has led to development of treatment plans centered around the vastus medialis oblique. Although this information is important, there is a lack of knowledge concerning the function of the vastus medialis oblique when movement requires a proximal-to-distal action. A proximal-to-distal action is usually associated with weight bearing tasks such as a sit-to-stand, the stance phase of walking, or jumping. Because of this, there is a gap in our understanding of the action of the muscle upon its proximal attachment point, the femur. This must be acknowledged to fully understand the purpose of this muscle.

2.3. Musculoskeletal Modeling

Computerized modeling of the human musculoskeletal system is a non-invasive technique for analyzing human motion and predicting outcomes. A common downfall of current systems is their modeling of muscles as following a straight line or curved path acting upon a normative skeletal system. There are two problems with this approach, the first being the inaccurate usage of point-to-point muscle lines, and secondly the exclusion of proper bony anatomy in a generic model. This results in a poor understanding of human movement and limits progression of knowledge in a broad range of interests such as ergonomics, injury prevention, and kinesiology.

2.3.1. Quadricep Architecture and the Muscle Force

The physiological cross-sectional area (PCSA) of a muscle is considered to be responsible for its force production abilities [18]. Larger PCSA is associated with greater levels of force production and is therefore a relevant marker for muscular strength. Theoretically, the larger arrangement of fibers in a parallel fashion within a cross-section demonstrates a greater contraction ability along the associated longitudinal axis [19]. It is important to recognize that PCSA depends on the cross section that is perpendicular to the direction of the muscle fibers, which contrasts with anatomical cross-sectional area (ACSA) where the cross section is taken to include the greatest amount of contractile material perpendicular to the long shaft. A common method of evaluating the PCSA of a muscle is to use MRI to determine the volume of the muscles and dividing that value by the length of the muscle fascicle [19]. Although volume is also correlated with the strength of the muscle, PCSA measurements are now considered to be similar within small margins, meaning that there are multiple effective ways of estimating muscle force production abilities.

The quadriceps are the only muscle group to cross the knee on the anterior side of the thigh, making them responsible for knee extension. This is achieved through the patellar tendon which attaches to the anterior tibia and is the common insertion for all the quadricep muscles. A measured knee extensor moment would be the combination of force from each muscle, of which individual muscle contributions cannot be immediately determined. Rather, the total sum of force is the only measurable outcome. The method of optimization could be used to mathematically estimate each muscle force contribution according to an agreement upon common human movement tendencies related to central nervous system efficiency [20] However, there are different theories pertaining to muscular control to simplify task performance or to regulate joint stressors, influencing the individual muscle contribution [21]. The forces could also be estimated by partitioning according to PCSA contribution, since PCSA content is directly related to force output. Ultimately, the selection of method to partition forces will influence the outcome and produce different individual muscle forces [22]

When the PCSA of individual muscles can be known, in conjunction with the total force output, the relative force contribution of the muscles can be equated to the relative PCSA contribution to the whole of the muscle. The equation below demonstrates how the relative contribution of PCSA is related to the relative contribution of force. The quotient calculated when dividing the individual muscle force (F_i) by the total muscle force (F_{Total}) is equivalent to the quotient calculate when dividing the individual muscle PCSA content ($PCSA_i$) by the ($PCSA_{Total}$), and results in the percent contribution of the muscle to the whole ($\%PCSA_i$).

$$\frac{F_i}{F_{Total}} = \frac{PCSA_i}{PCSA_{Total}} = \%PCSA_i; \textit{ i refers to an individual quadriceps muscles}$$

It is important to consider exactly what muscles would be involved when considering a single joint movement such as knee extension or flexion. In the case of knee extension, the only muscles involved would be the quadriceps muscles, limiting the force output to those muscles.

2.3.2. Optimization

Musculoskeletal modelling may use optimization techniques to estimate muscle activation and subsequent forces. Optimization is a criterion-based selection of an ideal element from a set of different options[23]. There are two categories of optimization: static and dynamic. Static optimization models will calculate the force values of the involved muscles at each time point through a movement using inverse dynamics[24]. Static optimization is a mathematical method of estimating the muscle force exerted by individual muscles by resolving net joint moments [25]. The calculation involves a cost minimization function, typically with the use of a sum of squares of the muscle activation, relating to the energy cost of the muscle. These equations are governed by constraints related to minimum and maximum boundaries of metabolic cost, energy expenditure, and muscle fatigue [20] . Static optimization relies heavily on accurate kinematic values from motion capture because of its use of inverse dynamics[24].

Dynamic optimization uses forward dynamics and require input of muscular physiological parameters as movement patterns are governed by established differential equations associated with human movement, and muscle activation properties [26]. Forward dynamics predict movements based on modelled forces and moments, meaning that recorded kinematics are not needed. Because of the inclusion of muscular physiological parameters, dynamic optimization is preferred for movements that involve substantial co-contraction and dynamic muscle activation [27].

2.3.3. Modelling Muscle Forces

Inverse dynamics allows the calculation of joint moments within a model of connected rigid bodies [28]. Equations are solved consecutively, beginning with the segment containing the most amount of information. Information about segment weights are typically based on normative anthropometric data and locations of each segment can be identified using motion capture techniques for human movement. Calculations are also informed by ground reaction forces obtained from a force plate. The combination of this information allows for the calculation of the moments experienced at each junction of the rigid body model, or the joints.

To find the forces exerted by the muscles upon the joint that result in the moments that are calculated during inverse dynamics, the line of action and moment arm of the muscle must be known. This information must be in the two dimensions that the moment is not acting in, as to determine the forces acting in the separate dimensions to cause the moment. For example, if a moment is found acting about the Z-axis, the muscle line of action and moment arm values must be known along the X and Y axis. The moment is a result of the cross-product of the moment arm and the force in the opposing dimensions as can be seen in the following equations.

$$M_z = r \times F = r_x * F_y + r_y * F_x$$

$$M_y = r \times F = r_z * F_x + r_x * F_z$$

$$M_x = r \times F = r_y * F_z + r_z * F_y$$

Knowledge of the moment arm and muscle line of action can be found within modelling software. Analysis can occur in an individually programmed software or within a purpose-specific software. Individually programmed models can be created multiple ways with different programs, with one example being a coding software such as Matlab. These models allow for individual interest to be explored, and specific parameters applied to the outcome of interest. Purpose-specific software is useful and convenient but has limitations in its modelling approach and ability to manipulate parameters. An example of the latter is OpenSim, which is an open-source software that specializes in human muscle modelling. Although there are many ways to analyze the moments and find the forces, each method is inherently different and may produce different results. It is important to consider that bone geometry can influence the moment arms and lines of action, and as such should be considered when selecting modelling techniques.

2.3.4. Bone Geometry

Specific bone geometry is often overlooked in modelling in favor of averaged bone models. Although general geometry of human bones is similar, scaling for sizing alone does not include the fine details of individual uniqueness of bones. When considering muscular attachment, the apparently small differences in geometry can significantly change the line of action or moment arm of the associated muscle [29]. As previously discussed, knowledge of the line of action and moment arm is critical in discerning muscular force output, highlighting the importance of accurate bone geometry. Beyond surface geometry, the material of bone and the stressors placed upon the bone in response to muscle pull has also been studied. The surface geometry can be used in finite element analysis (FEA) to estimate the internal stresses experienced in the bone which is informed by the calculated forces.

2.3.5. How to Obtain Bone Geometry

The creation of subject-specific models is highly sought after because of its high degree of applicability in sports, injury prevention, and the disabled community. There are two main methods of collecting information for subject-specific modelling in medical imaging and statistical shape modelling.

The 3D geometry of bones has been assessed through magnetic resonance imaging (MRI) and computed tomography (CT) techniques [6]. CT scanning is considered the gold standard for obtaining bony geometry with high levels of accuracy, while MRI is considered to also produce highly accurate bone images [30]. Use of scanning allows for the inclusion of subject-specific bone models, allowing for analysis of a living person's anatomy. Although these techniques are non-invasive, they are expensive and may be hard to access. In the case of CT, there is exposure to ionizing radiation.

Statistical shape modelling (SSM) uses motion capture and population-based metrics from imaging and normative data[31]. An individual's bone geometry shape can be estimated from a population-informed model [32]. Bony landmarks tracked during motion capture are used to construct the model in conjunction with the average shape to find a participant-specific bone model [31]. This is a non-invasive method of finding bony geometry and presents a safer alternative to extensive medical imaging. However, as this is an emerging approach, the effect of SSM on musculoskeletal modeling is not fully known [31]

The outputs of CT, MRI, and SSM can be used within multiple different methods such as Finite Element Analysis, OpenSim, and in mathematical models to evaluate their influence on forces and moments experienced.

As musculoskeletal modeling inputs are often obtained from cadavers, obtaining the cadaver bones' geometry may enhance model parameters. An accessible and affordable technique to capture an object's geometry is 3D photogrammetry. Photogrammetry computer programs use images taken of an object at all angles to construct a 3D computerized model. This can be used to create computerized models of human bones which can capture small details and present them in the model. Having multiple models of each segment of the lower extremity allows the user to construct specific bone models to build a musculoskeletal system upon. The reliability and accuracy of photogrammetry for creation of human bone models has been established [9].

2.4. Rationale

Given the information drawn from the literature, there is a clear purpose for a continued pursuit of improvement of 3D musculoskeletal models, and the subsequent finding of vastus medialis oblique functioning. The first highlight of the research is the inclusion of bony geometry and subsequent muscle lines of action into the model characteristics, which would improve upon previous outputs degree of accuracy The second highlight is that this research will study a weight-bearing task, where the thigh moves relative to the leg. Combined, this may ultimately allow us to demonstrate the function of the vasti, and specifically the vastus medialis oblique during weight-bearing by examining the wrapping of the vasti around the femur. The goal of this project is to improve musculoskeletal modeling by including bone geometry and using this model to assess the potentially unique vastus medialis oblique function.

Chapter 3: Methods

3.1. Overview

The goal of this research is to develop a model that can calculate the forces of the vasti muscles, and their magnitude and vectors, upon the femur. Additionally, this information can be used to calculate the moments experienced on all three planes. For this to occur, there are two components which must be satisfied to produce a functional model. An overview of the methods used for data collection can be seen below in Figure 3-1.

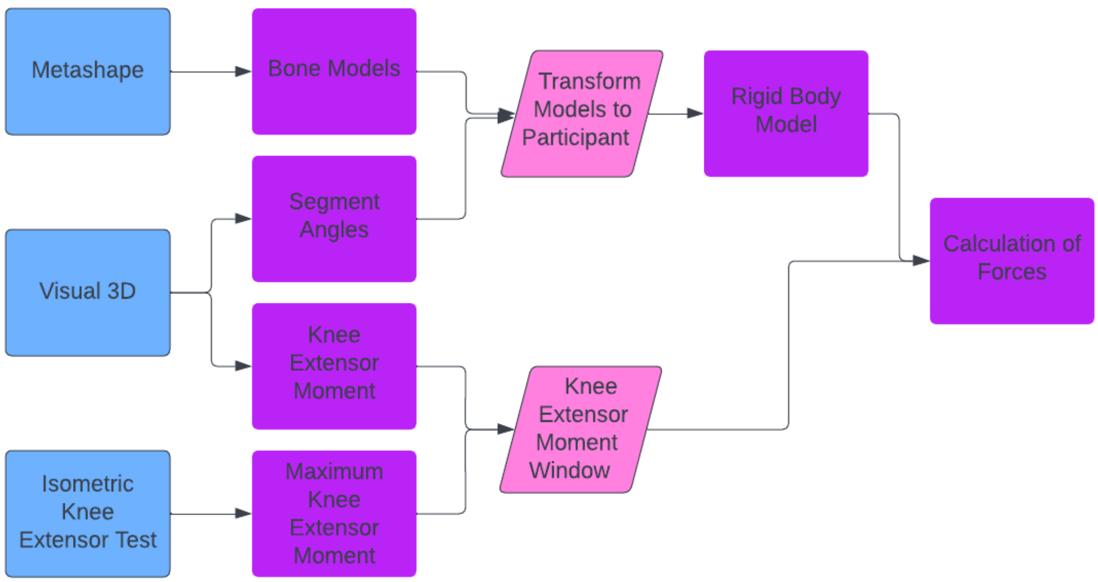


Figure 3-1: Overview of data collection methods.

The first component needed is bony geometry, which can provide the vector for the forces calculated from the kinetics of the vertical jump. Bony shape, such as shaft diameter and curvature may influence parameters such as moment arms and change the direction muscles exert force. Consequently, the moment caused by the muscle and experienced on the bone can be found, and the effect of bone geometry on the value of the moment can be assessed. The methods of obtaining the computerized models of the lower extremity bones can be found in Appendix B. Briefly, 3D photogrammetry was applied to digitally reconstruct bones from cadaver specimens.

The knee joint orientation is needed to properly calculate the forces and moments. Therefore, the second component is motion analysis, which will be used to animate and properly orient bones to calculate muscle force. Force plate and segment locational information can describe the movement of the bones of interest which can inform the translation of the base bone models, as well as provide net joint moment values experienced at the joints.

3.2. Hypothesis

It is expected that the vasti exert a transverse plane moment upon the femur (i.e. about the femur's longitudinal axis) because of the wrapping nature of the muscles. It is also expected that the differing bone geometries used for investigation will affect the moments experienced. The vastus medialis oblique is hypothesized to be responsible for the largest moment experienced in the transverse plane.

3.3. Obtaining Bony Geometry

3.3.1. Purpose

Digitally reconstructed bones were modeled in MATLAB software [33]. The associated code can be found in Appendix H. Once modeled, the objective was to virtually map the vasti musculature onto the bones. The muscles were mapped by anchoring attachment points at the origin and insertion points, which were informed by visual assessments of the cadaver specimens dissected. During dissection, the muscles of interest were manually removed, and the specific attachment location positions were identified.

3.3.2. Digitization of Bony Landmarks

Three sets of cadaver bones were photogrammetrically modelled, and were labelled as Model A, Model B, and Model C. These models differed in sex and height to offer variability within the bone sets. The object and image files from the construction of the bone models from photogrammetry were inputted into a Matlab script to obtain the texturized model with the capacity to digitize landmarks on the bone. These landmarks were similar to the landmarks used for motion capture. However, the expected locations of the center of the femoral head and vasti attachment points were also digitized, as they cannot be tracked using motion capture. The description of how landmarks were chosen during digitization can be found in Appendix C.

The femoral head is spherical and the pcfitsphere function was used to define the femoral head as a spherical object [33]. The center of this sphere was considered the origin of the femoral coordinate system. The greater trochanter, medial and lateral epicondyles, the proximal attachment points of the vasti, and the transition points of the vasti into the quadriceps tendon and vastus medialis oblique tendon were digitized on the femur.

The patellar apex, base, medial and lateral facet ridges, and the vastus medialis oblique tendon attachment point were digitized on the patella.

The medial and lateral tibial condyles, tibial tuberosity, and medial and lateral malleoli were digitized for the leg.

The posterior calcaneus, navicular tuberosity, and first and fifth metatarsal heads were digitized on the foot.

3.3.3. Orientation of Bones

Following the digitization of the models of the bones, they must be properly scaled and positioned. The initial reconstruction will not demonstrate the true size of the bones but will demonstrate the relative sizing within the singular reconstruction. Therefore, the previously measured actual size of the bone can be used to scale the bone to its actual size within MATLAB by coordinating the bony landmarks used for the original measurements and determining a scaling factor. The length of the femur and leg, the height of the patella, and the distance between the calcaneus and first metatarsal were used to determine the scaling factors for each model. These lengths were manually measured during dissection using calipers.

The bones also needed to be oriented properly to align with anatomical position. We used a standard right hand coordinate system to determine orientation of the bones. This uses bony landmarks specific to each bone to align properly within each coordinate axis. The code used for bone scaling and orientation can be found in Appendix I.

The origin of the femoral coordinate system was set to be the femoral head center. The line between the femoral head center and the center point between the lateral and medial epicondyles coincided with the superior-inferior axis (Z-axis). The Y-axis is the mutual perpendicular of the vector connecting the medial and lateral epicondyles and the Z-axis. The X-axis is the cross-product of the Y and Z-axes.

The origin of the patella was set as the center point of the patella, which was defined as the midpoint between the lateral facet and medial facet. The Z-axis was defined as line connecting the apex and the center of the patella. The Y-axis is the cross-product of the line between the medial and lateral facets and the X-axis is the cross-product of the Y and Z axes.

The midpoint between the lateral and medial tibial condyles established the origin of the leg. The line from the midpoint of the medial and lateral tibial condyles and the midpoint between the malleoli aligned with the Z-axis. The cross-product of vectors from the medial malleolus to the origin, and lateral malleolus to the origin defined the Y-axis.

For all bones, the X-axis was calculated from the cross-product of the Y- and Z-axes.

3.4. Participant Recruitment

Participants were recruited from the university population, with a focus on students engaged in sports or other high-intensity physical activities. An equal number of male (5) and female (5) participants were recruited. Participants were physically active, which is defined as having a sport or dance background where jumping regularly occurs, meaning that they would not be subject to more risk than they assumed daily. Participants were between the ages of 18-35 and did not have any knee or ankle injuries. Included participants were recruited by information letters shared with students in the faculty and within the local varsity athletics community.

Table 3-1: Participant Information.

Participant Number	Sex	Height (m)	Weight (kg)
1	Male	1.72	78.0
2	Male	1.88	71.4
3	Female	1.76	72.3
4	Female	1.73	69.8
5	Male	1.73	78.2
6	Male	1.69	62.0
7	Male	1.71	63.9
8	Female	1.82	95.0
9	Female	1.70	65.1
10	Female	1.68	51.9

3.5. Motion Analysis

3.5.1. Rationale for Activity

The task that was analyzed was a vertical jump. The rationale for selecting this task was its universality across physically active populations. For instance, vertical jumping serves as a commonly used test of physical fitness in many sports and occupations. Furthermore, vertical jumping requires a substantial knee extensor moment, and performance is influenced by quadriceps strength [34].

3.5.2. Kinematics and Kinetics

To obtain the lower extremity kinematics and kinetics from the vertical jump, eight Miqus M3 motion capture cameras (Qualisys, Gothenburg, Sweden) and a force plate (OR6-6; AMTI, Watertown, MA) were used. This motion capture system records movement of retro-reflective markers and has been validated [30].. This is conducted in Matlab, but the locations throughout the trials are extracted from Qualisys. Information from the force plate is used to determine the net joint moment values at the ankle, knee, and hip.

3.5.3. Marker Placement

Motion capture markers were placed on the lower extremities and pelvis to capture all lower body movements. A table detailing the marker placement locations can be found in Appendix E, and the rationale for marker decisions is explained in the next section. There are different marker sets for the static and dynamic trials, each of which are distinguished in Appendix E, and can be seen below in Figure 3-2. There are more markers in the static state to accurately determine the locations of the participants bony landmarks in relation to the cluster sets. For the dynamic trials, cluster markers and select pelvis markers remain since the motion capture software can calculate the location of the static markers in proximity of the dynamic marker set during the vertical jump. This is to prevent skin and clothes motion artifact when the participant is moving and retain the accuracy of bony landmark locations that is achieved in the static state. Cluster markers experience minimal motion artifact as they are attached to a segment rather than a bony landmark, making them more ideal for motion capture than individual markers.

The pelvis is a unique segment that cannot hold a cluster marker set like the thigh and leg can. Because of this, bony landmarks were used to both define the pelvis and track its motion.

Markers were placed on the left and right anterior superior iliac spine (ASIS), posterior superior iliac spine (PSIS), and the fifth lumbar and first sacral vertebrae joint landmarks. The definition of coordinate systems for the femur, patella, and shank, based on bony landmarks, were the same for motion capture data and the cadaver bones.

.

Figure 3-2: Motion Capture Marker Locations. Above: Static marker set. Below: Dynamic marker set.

3.5.4. Vertical Jump

. Participants were instructed to stand off the force plate before all trials for the force plate to zero out any external signals or forces that are not contributable to the participant. For the reference trial, participants were instructed to stand motionless on the force plate within the scope of the infrared cameras to establish a reference position. After the completion of the static trial, static markers were removed, which are listed in Appendix E. The static trial is important for the motion capture software to use for reference, as the locations of the static markers are recorded and can be mathematically located during the dynamic trials within Visual 3D. This decreases the amount of skin artifact that is commonly experienced by infrared markers during dynamic activities and improves the accurate tracking of the bony landmarks.

Following the static trial, the participants performed a maximum effort vertical jump five separate times, ensuring that they did not occlude any markers with their bodies. Participants were instructed to keep their hands at shoulder height to prevent occlusion of markers, and to land in a bent knee, loaded position to mitigate any instance of injury. Each participant completed five jump trials which were combined into their individual files. The best jump was used for each participant to best represent their maximum force production. The best jump was determined by the greatest achieved height of the pelvis through the L5/S1 marker. This information gives the participant a tertiary numerical marker that designates which trial to use for the segment and joint information.

3.6. Knee Moment during Maximal Contraction

When completing a vertical jump, co-contraction most likely occurs between the quadriceps and the hamstrings. Although a net knee extensor moment is observed, the contribution of the knee extensors is detracted from by the knee flexors. The quadriceps contribute as knee extensors during vertical jumping. However, the hamstrings muscles may co-contract and exert a knee flexor moment. This net effect of quadriceps and hamstrings is calculated as the net joint moment that indicates the minimum moment required from the quadriceps. To determine the possible maximum knee extensor moment experienced at the knee, participants completed three maximum voluntary isometric contractions. The apparatus used to test for the maximum knee moment can be seen below in Figure 3-3. The purpose of using a maximal isometric test was to determine an upper boundary maximum knee extensor moment to compare the vertical jump to. Completing a maximum contraction allows for the assumption to be made that all muscles within the knee extensor group were contributing to the full extent of their abilities and antagonistic muscles were not competing against the quadriceps. Isolating knee extension allows for a reliable estimate of the maximum knee extensor moment at the measured joint angle.

The voluntary maximal isometric contraction was done to establish a reference value for the maximal force of the quadriceps. The isometric contractions were performed sitting on a dynamometer that allowed the knee joint to be set as close to ninety degrees as possible, approximately the same maximum knee flexion angle expected during the vertical jumps. Since the participants were sitting, the hip joint was set at ninety degrees, which minimized any hip involvement. However, a goniometer was used to verify knee joint angle as it most likely was not exactly ninety degrees. Measured knee joint angles can be seen below in Table 3-2.

Table 3-2: Participant knee joint angles during isometric maximal knee extension.

Participant	External Knee Joint Angle (^{o})
1	65
2	64
3	68
4	67
5	65
6	65
7	58
8	60
9	65
10	72

It also allowed for minimal effects of body weight contributing to the need for force output that may occur if the leg were needed to be lifted to 45 degrees before commencing a contraction. Nevertheless, a baseline value was recorded for any force experienced at the resting position. Given the anatomical structure of the patellar ligament, and the assumption that there is negligible contribution in the medial-lateral direction, the extensor moment recorded during trials can be expected to be completely contributed to the action of the quadriceps via the patellar tendon.

A brief warm-up of perceived 30%, 60%, and 90% sub-maximal effort contractions was performed prior to the maximal efforts.. The maximal voluntary isometric contraction was only performed on the right side and was conducted three times. The maximum moment value was taken from the three trials as this corresponds to their absolute max value.

Figure 3-3: Maximum isometric leg extension apparatus.

3.7. Rigid Body Model

The motion analysis and force plate data collected were imported into Visual 3D software [35] for rigid-body modeling and inverse dynamics calculations. Only the pelvis and right leg lower extremity was modeled.

The pelvis was based on the location of the right and left anterior superior iliac spine (ASIS), and the right and left posterior superior iliac spine (PSIS). Tracking markers to help in the construction of the pelvis included the right and left iliac crest and the L5/S1 spinal level marker. The femur relied on the greater trochanter, and medial and lateral femoral epicondyles. The hip joint centre was estimated from the ASIS and PSIS locations [36]. .. Tracking markers for the femur included the five marker clusters. The leg location was created using the lateral and medial tibial condyles and lateral and medial malleoli. Tracking markers included the five marker clusters. The foot was determined using the lateral and medial malleolus and the first and fifth metatarsal heads, as well as a three-marker cluster on the lateral heel. The patella location was created using the apex and base, with the medial and lateral facet locations as tracking markers.

From the established bone locations and force information, the joint angles, and joint moments from the vertical jump of the hip, knee, and ankle were obtained from Visual 3D and can be tracked and graphed. Joint and segment angles were obtained using a X-Y-Z Cardan sequence. The moment values can be used to help determine individual quadricep muscle involvement when combined with the assumption of relative PCSA involvement.

For purposes of orientation of the digital bone models, the segment angles relative to the global coordinate system (the lab) were also collected. These identify the relative location of the rigid bodies in space and can be used to translate the bone models to the relative locations to each other. Segment angles were collected for the foot, leg, thigh, pelvis, and patella. Since the patella is not associated with a calculated joint angle or moment, the segment angle of the patellar segment is the only identifying characteristic that can be collected from visual 3D. The knee moment collected acts across the patella and into the patellar ligament, which leaves the moments experienced at the patella unmeasured. Knowledge of its location relative to the thigh and leg is crucial for the identification of the forces experienced at the patella.

Chapter 4: Muscle Force Computation

Following the collection of inputs from motion capture, leg extension, and bone models, they are amalgamated to produce the desired end result, which is the transverse moment experienced at the knee. The overview and order of calculation methods can be seen in Figure 4-1.

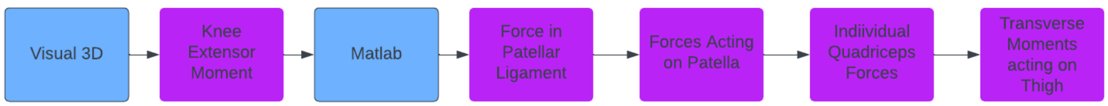


Figure 4-1: Overview of methods and outputs.

4.1. Matching of Bone Models to Motion Capture

Visual 3D software (version 2020.03.26; C-Motion, Germantown, MD) was used to analyze the motion capture data to generate a rigid-body model about the skeletal locations and relative placement of individual bones. This software also allows for calculation of segment and joint angles and joint moments. Rotation matrices for the shank, patella, and femur were extracted to inform the translation of the digital bone models to the motion capture data and the knee extensor net joint moment was determined to inform the calculation of forces acting on these bones.

4.2. Force Calculation

4.2.1. Calculation of Force in the Patella Ligament from Tibial Tuberosity

The patellar ligament attaches proximally to the apex of the patella, and distally at the tibial tuberosity. It is considered to possess the force of the entirety of the quadriceps since the quadriceps and vastus medialis oblique tendon coalesce into the patella and the subsequent patellar ligament. The muscular moment recorded at the knee during motion capture is experienced at the knee joint center, and with the known length of moment arm can be used to calculate the force experienced in the patellar tendon.

When considering the path of the patellar ligament, there is little medial-lateral (X-axis) change, making the change in the superior-inferior (Z-axis) and anterior-posterior (Y-axis) directions the majority of the vector contributions. Given this observation, the assumption can be made that there is no medial-lateral contribution to the force vector of the patellar ligament. Therefore, the directional vector can be described in the YZ plane.

The moments experienced at the knee are described as net joint moments, meaning that although the direction of the moment indicates extension, there is most likely influence from the co-contraction of hamstrings, gastrocnemius, and the influence of ligaments. Therefore, it must be assumed that the moment recorded about the X-axis during the vertical jump is the minimum contribution of the patellar ligament, as it must overcome the flexor moments created by antagonistic muscles.

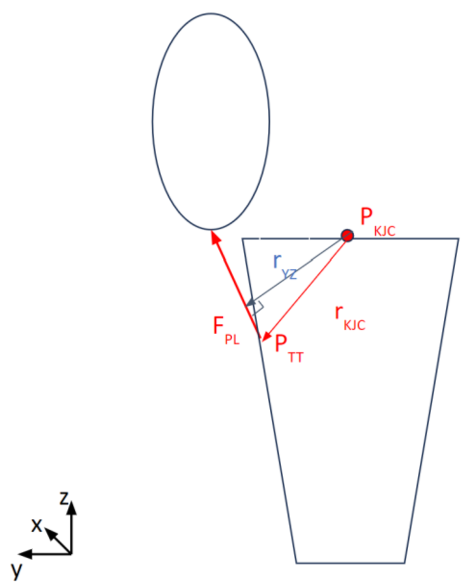


Figure 4-2: Conceptual Positioning of Patella and Tibia in YZ plane.

The direction of the force vector of the patellar ligament can be calculated since the proximal and distal attachment locations are known. This vector will be known as V_{PL}. The unit vectors associated with the directional contributions can be calculated below. These values indicate the contributions of each direction towards the total vector. Each unit vector shall be described as U_{PL}. It is assumed that there is no X component, so only the Y and Z components will be calculated with a normative V_{PL} being determined with only Y and Z components.

$$U_{PL_y} = \frac{V_{PL_y}}{|V_{PL}|} \quad U_{PL_z} = \frac{V_{PL_z}}{|V_{PL}|}$$

Since the direction of the force vector is known, a perpendicular moment arm can be calculated using the cross product of the known distance between the axis of rotation (AOR) of the leg at the proximal end position, and the attachment point at the tibial tuberosity (r_{KJC}), and the unit vectors of the patellar ligament force vector.

$$r_{YZ} = r_{KJC}\ X\ U_{PL}$$

The equation to calculate the moment about an axis usually contains four terms because the force needs to be evaluated perpendicularly to the moment arm, which occurs in two directions. However, a single perpendicular moment arm is calculated to account for both the Y and Z directions. Therefore, only one equation is needed in the YZ plane to calculate the moment in the X. The equation used to calculate the moment in X can be expressed as such.

$$M_x = r_{yz} \times F_{yz}$$

Since the moment in the X direction is known from the motion capture data, the equation can be rearranged to determine the force in the YZ plane. It should be noted that r_{yz} is perpendicular to the F_{yz} component.

$$F_{yz} = \frac{M_x}{r_{yz}}$$

4.2.2. Calculation of Forces acting on the Patella

It can be assumed that the force experienced at the tibial tuberosity will be equal and opposite to the force experienced at the apex of the patella from the patellar ligament.

A similar process can be used to determine the forces of the quadriceps tendon and the vastus medialis oblique tendon on the superior side of the patella. The center of mass of the patella was defined as the center of the patella by finding the midpoint between the apex, base, medial facet, and lateral facet. The patellar attachment locations of all three tendons were digitized on the patella, with the patellar ligament attachment location on the apex, quadriceps tendon attachment location on the base, and the vastus medialis oblique attachment location on the medial-superior corner of the patella. The femoral attachment locations for the VMO and Quadriceps tendon were located on the ridge superior to the medial femoral condyle and approximately five centimeters above the patellar groove respectively.

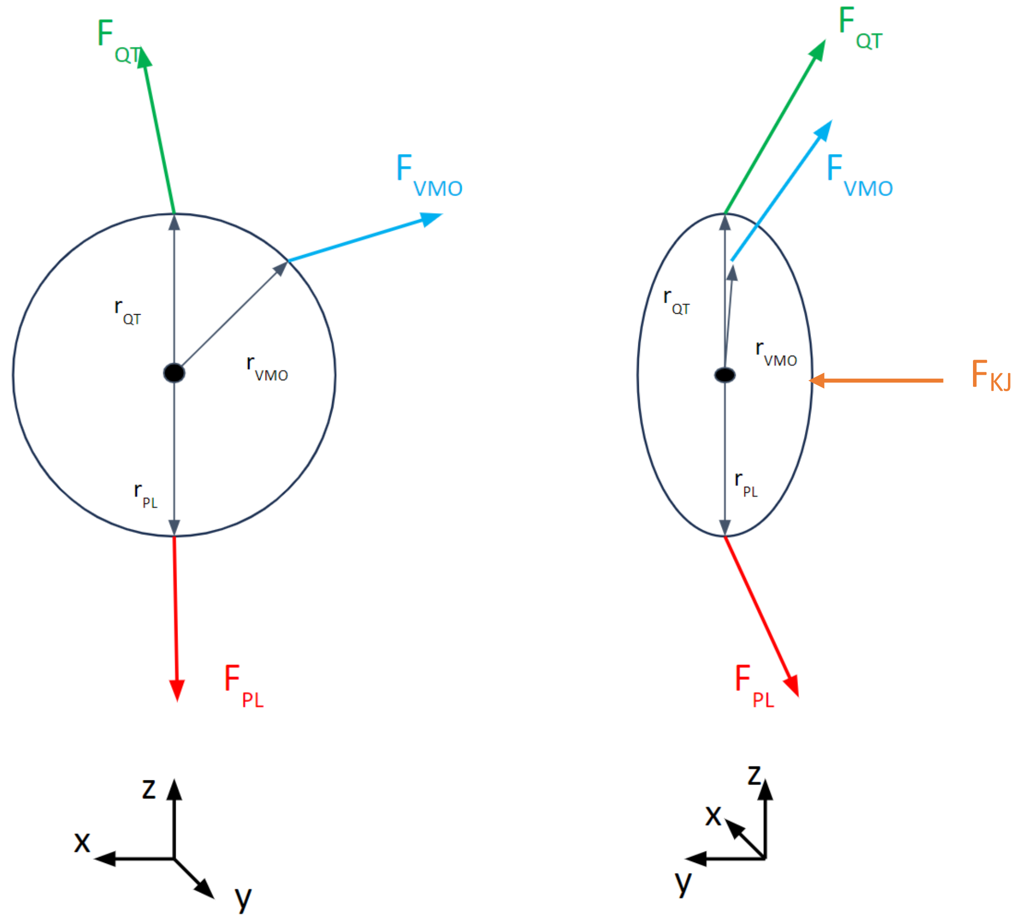


Figure 4-3: Conceptual Representation of Force Acting upon the Patella. Left-Frontal Plane. Right- Sagittal Plane.

Distances from the center of mass to the patellar ligament, quadriceps tendon, and vastus medialis oblique attachment points were calculated, as well as force vector directions from the patella to corresponding bony attachment sites. Although the force could not be known at this point, the unit vector specifications for directional proportions could be determined from the force vector information, allowing knowledge of how much force would be attributed to each direction. The vector of the force will be known as V_F.

$$V_F = V_{F_x} * \hat{\imath} + V_{F_y} * \hat{\jmath} + V_{F_z} * \hat{k}$$

To determine a perpendicular moment arm from the center of mass to each component of force, the cross product of the distance from the center of mass to the attachment point and the corresponding unit vector of the force vector was taken for the patellar ligament, quadriceps tendon, and vastus medialis oblique tendon. The moment arm found is described in the YZ plane, ZX plane, and XY plane. A sample equation is below.

$$U_x = \frac{V_x}{|V|}\; U_y = \frac{V_y}{|V|} \quad U_z = \frac{V_z}{|V|}$$

$$r = r_{AOR} \times U_V$$

Since the force at the patellar ligament is known as being equal and opposite from the force calculated from the leg, there are two unknown forces caused by the vastus medialis oblique and quadriceps tendon on the patella. It is assumed that the sum of moments in each direction active on the patella has a net value of zero at a given moment in time. The total force of the quadriceps tendon and the vastus medialis oblique tendon can be multiplied by their unit vector to determine their contribution in each plane. This allows the same unknown variable to be inputted into each moment equation which makes substitution possible. As such, each moment equation can be written as follows.

$$\sum M_x = 0 = r_{PL_{yz}} * F_{PL_{yz}} + r_{VMO_{yz}} * |F_{VMO}| * U_{VMO_{yz}} + r_{QT_{yz}} * |F_{QT}| * U_{QT_{yz}}$$

$$\sum M_y = 0 = r_{PL_{zx}} * F_{PL_{zx}} + r_{VMO_{zx}} * |F_{VMO}| * U_{VMO_{zx}} + r_{QT_{zx}} * |F_{QT}| * U_{QT_{zx}}$$

$$\sum M_z = 0 = r_{PL_{xy}} * F_{PL_{xy}} + r_{VMO_{xy}} * |F_{VMO}| * U_{VMO_{xy}} + r_{QT_{xy}} * |F_{QT}| * U_{QT_{xy}}$$

It should be noted that there is also a force exerted by the patellofemoral articulation on the posterior patella. However, it was assumed that the location of this force was close to or overlapping with the center of the patella, therefore, the moment arm for this force would be negligible [37]. By using substitution methods, the value of the force in one tendon can be calculated, followed by the calculation of the other unknown force using the newly calculated first force. One way of rearranging this equation to obtain the force of the vastus medialis oblique is shown below, followed by the equation to determine the force in the quadriceps tendon.

$$|F_{VMO}| = \frac{r_{QT_{yz}} * U_{QT_{yz}} * r_{PL_{zx}} * F_{PL_{zx}} + r_{PL_{yz}} * F_{PL_{yz}} * r_{QT_{zx}} * U_{QT_{zx}}}{r_{QT_{yz}} * U_{QT_{yz}} * r_{VMO_{zx}} * U_{VMO_{zx}} + r_{VMO_{yz}} * U_{VMO_{yz}} * r_{QT_{zx}} * U_{QT_{zx}}}$$

$$|F_{QT}| = \frac{-r_{VMO_{yz}} * U_{VMO_{yz}} * |F_{VMO}| * -r_{PL_{yz}} * F_{PL_{yz}}}{r_{QT_{yz}} * U_{QT_{yz}}}$$

Following calculation of each muscular force, the forces were inputted back into the moment equations to verify the validity of the equations. The forces calculated as acting upon the patella were considered to be equal to the force exerted on their opposite attachment point, which are the femur and leg for the VMO and QT, and PL respectively.

4.2.3. Calculation of Moments on the Femur

The moment experienced at the proximal attachment point of the quadriceps muscles can be calculated by using a cylindrical coordinate system. The VL, VML, and VMO wrap around from the posterior-proximal side of the femur to the anterior-distal side of the femur, therefore, resemble cables wrapped around a cylinder. A cylindrical coordinate system is used to address the issue of wrapping. In the transverse plane, the line from the femur's longitudinal axis to the proximal attachment can be considered the cylinder's radius and the muscle's moment arm at the proximal attachment. The force that is found to act tangentially to the cylinder is included in the calculation of the transverse moment. Therefore, the transverse plane moment is the force acting in the XY plane multiplied by the cylinder's radius. It should be discussed that an assumption is made regarding the path of the muscle, as it does not follow the path of a cylinder perfectly. Rather, the muscles follow the unique shape of the femur, and using a cylinder to model the direction of the muscle pull is a limitation.

The radius of the cylinder which represents the distance from the vertical axis of the femur to the proximal attachment points can be calculated as the resultant length of the X and Y coordinates of the digitized attachment locations. The height of the cylinder can be determined by finding the difference between the Z coordinates of the proximal and distal muscle attachment locations. This is done within the femoral coordinate system rather than the global coordinate system since the cylinder of interest will move with the femur throughout the participant action, and therefore the landmark coordinates should not change.

Within a cylindrical coordinate system, three cylindrical unit vectors, U_r, U_θ, and U_z, can be calculated from the cartesian unit vectors.

$$\hat{\imath} = \frac{x}{A} \quad \hat{\jmath} = \frac{y}{A} \quad \hat{k} = \frac{z}{A}$$

$$U_r = \hat{\imath} * \cos\theta + \hat{\jmath} * \sin\theta \quad U_\theta = -\hat{\imath} * \sin\theta + \hat{\jmath} * \cos\theta \quad U_z = \hat{k}$$

U_θ is perpendicular to the radius of the cylinder, and they are both constructed with X and Y components, which when combined can allow for the calculation of a moment about the Z axis, or a transverse plane moment. Similar to previous methods, the total force of the muscle can be multiplied by the cylindrical unit vector to determine how much of the force contributes to the given direction, and then multiplied by the radius which serves as a moment arm length. Using this method allows for the calculation of a moment applied by a vector that follows a three-dimensional curved path. This is done for each vasti muscle as the curved paths are different for each muscle. A diagram modelling the cylindrical coordinate system can be seen below in Figure 4-4: Cylindrical coordinate system.

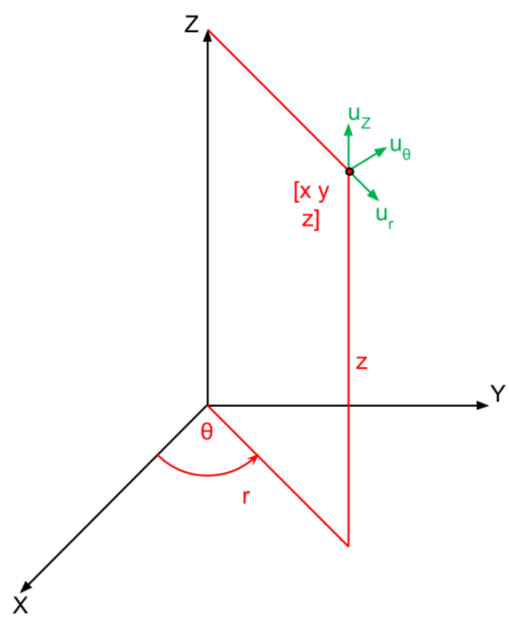


Figure 4-4: Cylindrical coordinate system. The Z-axis is related to the longitudinal axis of the femur, and the Z height corresponds to the height difference between the related proximal and distal locations.

Chapter 5: Results

5.1. Knee Net Joint Moment

During the vertical jump, the value of the net joint moment of the knee was recorded in all three axes. Only the sagittal plane moment was used for analysis, which was always a knee extensor moment. The voluntary maximum knee extensor moment was recorded with an isometric knee extension exercise. The mass normalized knee extensor moment for participant 1 can be seen below in Figure 5-1, for which the associated knee joint angle was sixty-eight degrees. The mass normalized results of the maximum knee extensor moments from the vertical jump and isometric knee extension can be found below in Table 5-1.

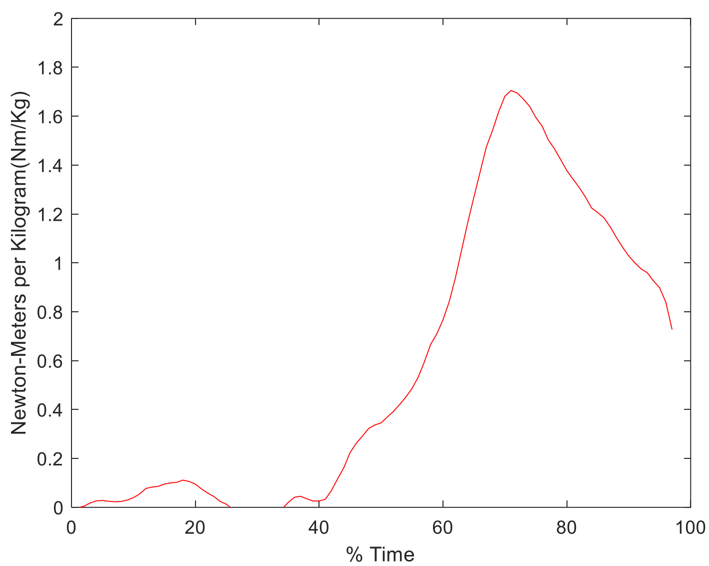


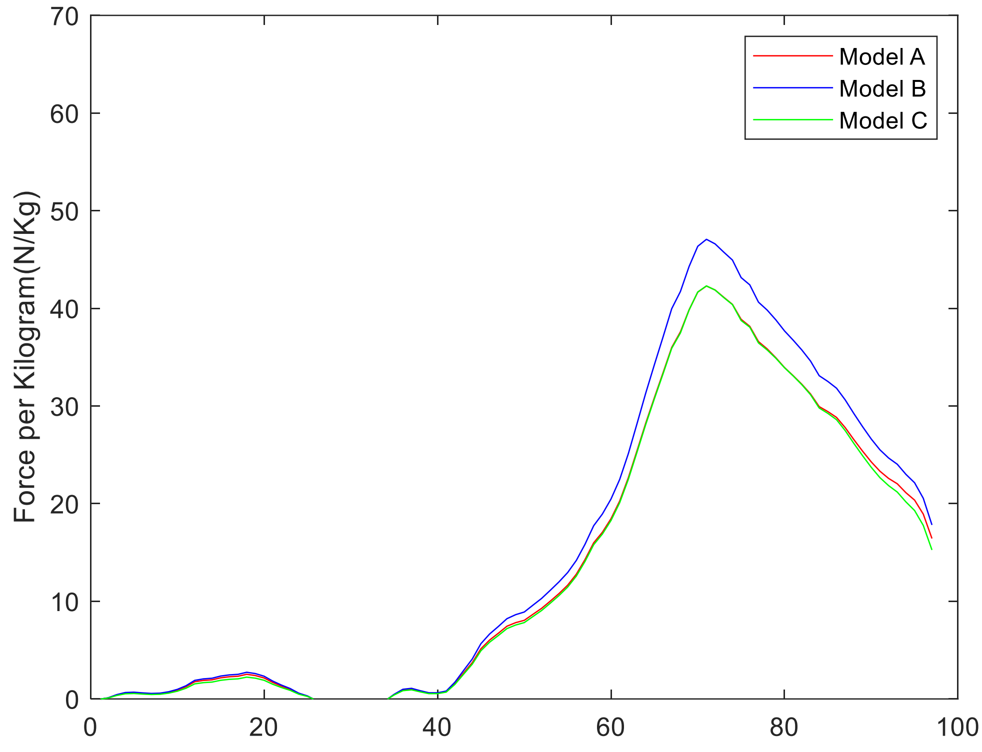
Figure 5-1: Mass normalized knee extensor moment during vertical jump for Participant 1.

Table 5-1: Maximum normalized knee extensor moments for each participant in the vertical jump and isometric knee extension exercise.

Participant	Mass (Kg)	Vertical Jump (Nm/Kg)	Isometric (Nm/Kg)
1	78.0	1.71	2.47
2	71.4	1.63	2.58
3	72.3	1.47	1.99
4	69.8	1.32	2.34
5	78.2	1.50	2.61
6	62.0	0.89	1.97
7	63.9	1.47	2.80
8	95.0	1.03	2.84
9	65.1	0.96	2.35
10	51.9	1.08	2.58
Average±SD		1.31±0.28	2.45±0.28

5.2. Forces Acting on the Leg.

The resultant forces of the patellar ligament upon the leg were calculated. An example of the forces for P1 can be seen below in Figure 5-2: Example mass normalized resultant forces upon the leg Participant 1. Across all participants, the model using cadaver B consistently recorded the highest forces comparative to the other models. The shapes of the leg force graphs were similar, and the maximum force aligned closely with the maximum knee extensor moment instance. The mass normalized forces upon the leg for all participants can be found in Appendix F.



% Time

Figure 5-2: Example mass normalized resultant forces upon the leg Participant 1

The forces upon the leg at the instance of the maximum knee extensor moment were recorded to show the similarities between participants and across models. They are found below in Table 5-2. Model B consistently outputs the highest force across the models.

Table 5-2: Resultant forces upon leg via the Patellar Ligament at instance of maximum knee extensor moment.

Participant	**Max Force (N/Kg)**		
	Model A	**Model B**	**Model C**
1	42.3	47.0	42.3
2	41.6	47.1	42.9
3	52.5	62.9	53.7
4	37.6	43.9	37.3
5	32.7	36.2	32.3
6	22.2	25.0	21.5
7	42.8	49.4	44.1
8	25.9	29.9	26.9
9	23.9	27.4	22.7
10	32.9	36.6	32.8
Average±SD	35.4±9.2	40.5±11.1	35.7±9.8

5.3. Forces Acting on the Patella.

The directional forces of the QT, VMO, and PL upon the patella were calculated. An example of the results from P2 can be seen below in Figure 5-3. The maximum forces experienced by each model in all three directions for the QT, VMO, and PL can be found in Table 5-3, Table 5-4, and Table 5-5 respectively. The graphical representations of the patellar forces for all participants can be found in Appendix F. It should be noted that values from Participant 3 were presented in the tables and in Appendix F, but the values recorded were highly inconsistent with all other participants and were therefore considered outliers and removed from averaging calculations and discounted during analysis. Further information is found in section 6.5.

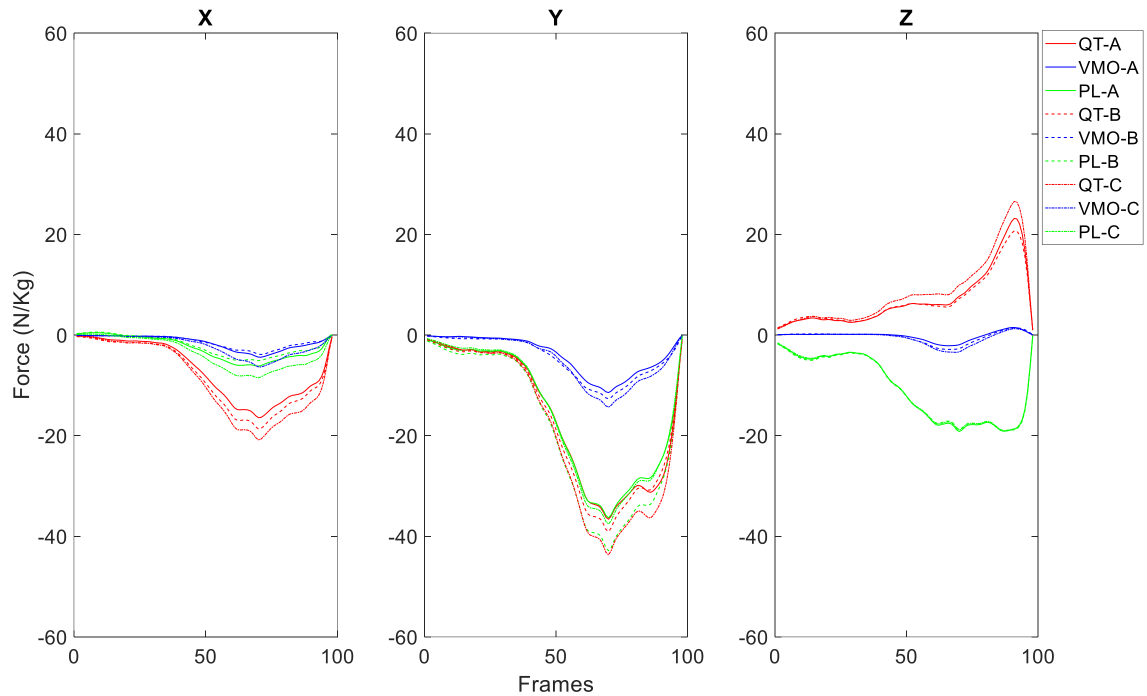


Figure 5-3: Participant 2 example of mass normalized forces upon the patella from the Patellar Ligament, Quadriceps Tendon, and Vastus Medialis Oblique

Table 5-3: Maximum force of Quadriceps Tendon upon patella across participants

	Max Force (N/Kg)								
Participant	**Model A**			**Model B**			**Model C**		
	X	**Y**	**Z**	**X**	**Y**	**Z**	**X**	**Y**	**Z**
1	1.3	-53.5	23.4	-3.2	-45.2	18.4	-3.9	-59.9	25.9
2	-15.8	-34.7	22.2	-18.0	-37.1	20.0	-20.1	-41.3	25.5
3	26.5	29.5	12.4	29.1	29.9	11.9	34.5	33.7	14.8
4	-9.7	-30.7	3.6	-13.4	-35.6	4.8	-13.7	-33.8	4.6
5	-4.4	-31.6	16.8	-6.3	-30.6	15.4	-8.0	-35.8	19.3
6	2.5	-25.8	15.0	-1.4	-23.1	13.9	-2.3	-29.1	17.8
7	-2.8	-48.8	22.8	-6.4	-45.3	19.9	-6.4	-55.8	27.6
8	-3.0	-17.0	11.4	-5.0	-20.2	12.5	-5.6	-20.3	13.2
9	-8.1	-17.7	12.4	-9.5	-18.7	12.6	-9.9	-18.0	13.2
10	-8.7	-20.4	6.5	-11.8	-25.7	7.2	-11.5	-23.7	7.8
Average ±SD	-5.4±5.4	-31.0±12.2	14.9±6.7	-8.3±5.0	-31.2±9.6	13.7±4.8	-9.0±5.2	-35.3±14.0	17.2±7.7

Note: Values from Participant 3 were removed from averaging calculations as values were highly inconsistent with all other participants.

Table 5-4: Maximum force of Vastus Medialis Oblique upon patella across participants

Participant	Max Force (N/Kg)								
	Model A			Model B			Model C		
	X	Y	Z	X	Y	Z	X	Y	Z
1	1.2	-29.2	3.2	2.8	-22.7	1.7	-2.8	-34.8	1.1
2	-4.1	-10.6	-2.1	-3.5	-11.8	-2.8	-5.9	-13.3	-3.4
3	22.3	11.6	-5.1	15.6	6.8	-3.9	25.0	13.7	-7.6
4	-2.2	-17.2	-4.2	0.5	-18.6	-5.1	-5.6	-19.4	-6.1
5	-1.3	-5.9	0.8	-0.7	-4.9	0.6	-2.6	-7.1	0.2
6	1.1	-10.6	2.7	1.3	-7.9	1.9	-1.7	-12.5	2.1
7	-0.3	-21.1	0.2	1.9	-17.2	-0.8	-3.5	-24.0	-2.2
8	-0.2	-1.1	0.1	0.4	-3.2	-0.2	-0.5	-2.2	-0.2
9	-3.1	-4.2	1.3	-2.8	-5.1	1.3	-3.5	-4.3	0.8
10	-1.6	-3.8	0.5	-1.6	-5.1	-0.8	-2.3	-4.7	-0.9
Average ±SD	-1.2±1.7	-11.5±8.8	0.3±2.1	-0.2±2.0	-10.7±6.8	-0.5±2.2	-3.2±1.6	-13.6±10.2	-1.2±2.3

Note: Values from Participant 3 were removed from averaging calculations as values were highly inconsistent with all other participants.

Table 5-5: Maximum force of Patellar Ligament upon patella across participants

Participant	Max Force (N/Kg)								
	Model A			Model B			Model C		
	X	Y	Z	X	Y	Z	X	Y	Z
1	-4.1	-36.1	-9.7	-3.0	-40.6	-9.0	-6.0	-35.8	-9.5
2	-5.9	-34.4	-18.5	-5.0	-40.4	-18.9	-8.1	-35.5	-18.4
3	-11.0	43.1	-22.8	-12.1	53.6	-22.9	-9.3	44.1	-23.6
4	-12.0	-28.4	-20.9	-12.8	-35.5	-22.2	-13.3	-27.7	-20.5
5	-5.0	-26.1	-19.9	-4.0	-30.6	-20.3	-5.9	-26.1	-19.8
6	-3.7	-17.4	-12.3	-3.5	-20.9	-12.7	-4.4	-17.3	-12.1
7	-9.7	-37.1	-16.7	-9.5	-44.2	-16.6	-11.8	-38.2	-16.5
8	-2.0	-16.2	-19.8	2.5	-20.3	-21.3	-2.7	-17.2	-20.3
9	-5.8	-16.2	-15.4	-6.6	-19.8	-16.5	-5.7	-14.9	-15.0
10	-3.5	-21.9	-25.4	-3.3	-27.7	-28.4	-4.3	-22.2	-25.3
Average ±SD	-5.7±3.0	-26.0±8.1	-17.6±4.5	-5.0±4.1	-31.1±9.0	-18.4±5.3	-6.9±3.3	-26.1±8.4	-17.5±4.5

Note: Values from Participant 3 were removed from averaging calculations as values were highly inconsistent with all other participants.

Based on Figure 5-3, the forces in the X direction were all negative, with the QT producing the highest negative force, or the medial direction. The QT produced the most force in the negative X direction across all participants. Some participants did see positive contribution from the VMO or QT, but any positive contribution was minimal, and the instance of positive contribution was very low.

The forces in the Y direction were all negative, which is consistent with expectations that the muscles will pull posterior to retain the patella upon the knee joint. The quadriceps also produced the highest magnitude of force, and the most force in the negative direction across all participants.

The forces in the Z direction were spilt, with the QT and VMO pulling in the positive direction (superior) and the PL pulling in the negative direction (inferior). This is expected as the QT and VMO attach on the superior aspect of the patella and the PL attaches inferiorly on the patella. The magnitude of the QT and PL were similar and opposite in sense. The VMO provided minimal positive pull, but with a much smaller magnitude than either the PL or QT. This is also expected given the oblique nature of the VMO tendon.

5.4. Individual Quadriceps Forces

The resultant forces of the quadriceps upon the patella were calculated. The breakdown of the RF, VL, VI, and VML was determined using the QT force upon the patella and dividing by PCSA content. An example of the forces for P1 can be seen below in Figure 5-4, followed by the force values at the instance of peak knee extensor moment. The values for RF, VML, VI, VL, and VMO can be found in Table 5-6, Table 5-7, Table 5-8, Table 5-9, and Table 5-10 respectively. The graphical representations of all participants individual quadriceps forces upon the patella can be found in Appendix F.

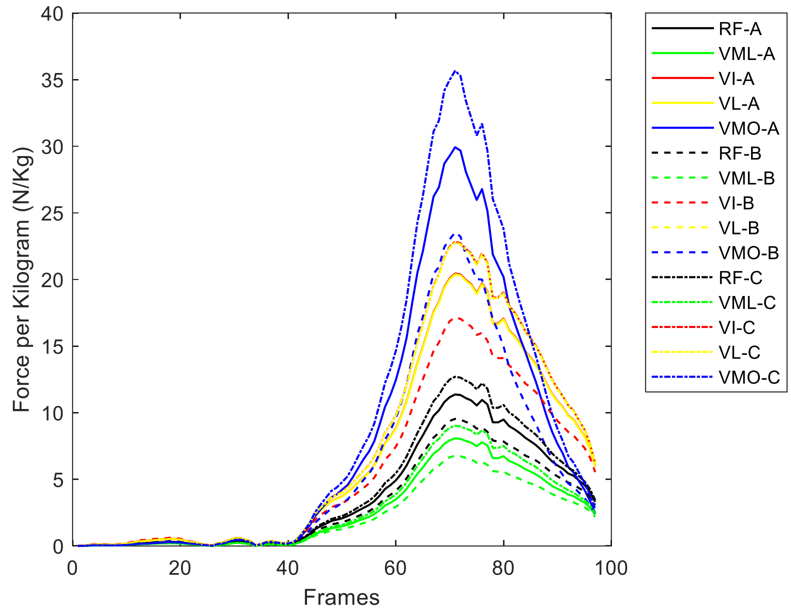


Figure 5-4: Example of mass normalized individual quadricep forces upon the patella for Participant 1

Table 5-6: Mass Normalized Rectus Femoris force upon patella at instance of maximum knee extensor moment

Participant	**Max Force (N/Kg)**		
	Model A	**Model B**	**Model C**
1	11.0	9.2	12.3
2	7.3	7.8	8.8
3	8.7	9.2	10.7
4	6.5	7.7	7.4
5	6.8	6.5	7.7
6	5.6	5.0	6.3
7	10.2	9.5	12.0
8	4.3	5.0	5.1
9	4.6	4.9	4.9
10	4.7	5.9	5.6
Average ±SD	7.0±2.2	7.1±1.8	8.1±2.6

Table 5-7: Mass Normalized Vastus Medialis Longus force upon patella at instance of maximum knee extensor moment

Participant	Max Force (N/Kg)		
	Model A	**Model B**	**Model C**
1	7.8	6.5	8.7
2	5.2	5.5	6.2
3	6.2	6.5	7.7
4	4.6	5.5	5.3
5	4.8	4.6	5.5
6	4.0	3.6	4.4
7	7.3	6.7	8.5
8	3.1	3.5	3.7
9	3.3	3.5	3.5
10	3.4	4.2	4.0
Average ±SD	5.0±1.6	5.0±1.2	5.8±1.9

Table 5-8: Mass Normalized Vastus Intermedius force upon patella at instance of maximum knee extensor moment

Participant	Max Force (N/Kg)		
	Model A	**Model B**	**Model C**
1	19.7	16.5	22.0
2	13.1	14.0	15.8
3	14.0	14.7	17.3
4	10.4	12.4	11.9
5	12.3	11.8	13.9
6	10.1	9.0	11.2
7	18.4	17.0	21.5
8	6.9	8.0	8.3
9	7.4	7.9	7.8
10	7.5	9.6	9.0
Average ±SD	12.0±4.2	12.1±3.2	13.9±4.9

Table 5-9: Mass Normalized Vastus Lateralis force upon patella at instance of maximum knee extensor moment

Participant	**Max Force (N/Kg)**		
	Model A	**Model B**	**Model C**
1	19.7	16.4	22.0
2	13.0	14.0	15.7
3	14.3	15,1	17.7
4	10.7	12.7	12.1
5	12.2	11.7	13.8
6	10.0	9.0	11.2
7	18.3	17.0	21.5
8	7.1	8.2	8.5
9	7.6	8.1	8.0
10	7.7	9.7	9.2
Average ±SD	13.9±4.9	11.1±4.4	14.0±4.9

Table 5-10: Mass Normalized Vastus Medialis Oblique force upon patella at instance of maximum knee extensor moment

Participant	**Max Force (N/Kg)**		
	Model A	**Model B**	**Model C**
1	29.3	22.9	34.9
2	11.6	12.6	14.8
3	29.4	20.2	33.5
4	18.3	19.7	21.7
5	5.6	4.6	6.7
6	9.2	7.1	10.2
7	20.8	17.2	24.4
8	0.9	3.1	2.2
9	5.2	5.9	5.6
10	3.3	4.8	4.4
Average ±SD	13.4±10.0	11.8±7.2	15.8±11.5

Table 5-11: Average muscular forces at instance of maximum knee extensor moment

Muscle	Model A	Model B	Model C
Rectus Femoris	7.0±2.2	7.1±1.8	8.1±2.6
Vastus Medialis Longus	5.0±1.6	5.0±1.2	5.8±1.9
Vastus Intermedius	12.0±4.2	12.1±3.2	13.9±4.9
Vastus Lateralis	13.9±4.9	11.1±4.4	14.0±4.9
Vastus Medialis Oblique	13.4±10.0	11.8±7.2	15.8±11.5

The VMO and VL contributed most to the muscular action upon the patella. The VML and RF were the smallest contributors.

5.5. Moments Acting on the Femur.

5.5.1. Graphical Representation

The transverse moments created by the quadriceps upon the femur reflected that the VMO and VML had a negative, and therefore external rotation moment. And the VL and VI had a stronger positive moment and therefore an internal rotation moment. An example of the moments experienced throughout the vertical jump can be seen in Figure 5-5. The graphical representations of the mass normalized moments for each participant can be found in Appendix F.

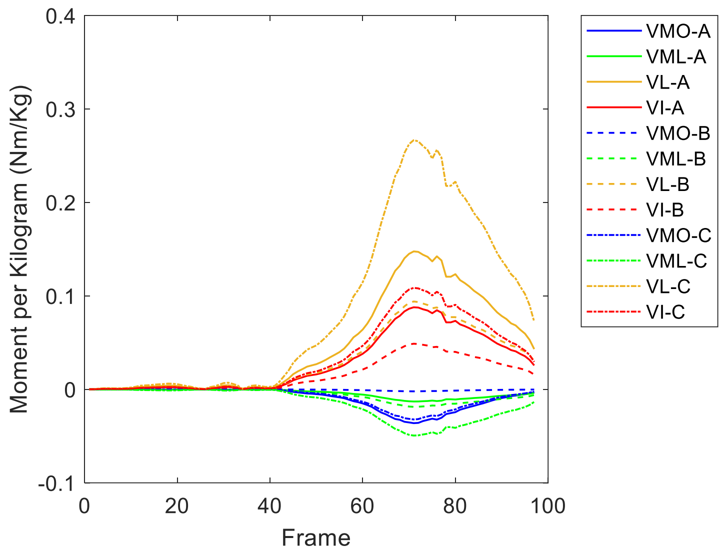


Figure 5-5: Example graph of Participant 1 mass normalized moment acting about the longitudinal axis of the femur.

The moments were calculated using a cylindrical coordinate system. The moment arm, or radius of the cylinder, and a unit vector related to the angle of the departure of the radius from the X-axis (theta). The maximal transverse moments and the associated moment arms and unit vectors for the VMO, VML, VL, and VI are respectively shown below in Table 5-6, Table 5-7, Table 5-8, and Table 5-9

Table 5-12: Mass Normalized maximal Vastus Medialis Oblique moments about femoral longitudinal axis

	Model A			Model B			Model C		
Participant	**Moment Arm (cm)**	**U_θ**	**Moment (Nm/Kg)**	**Moment Arm (cm)**	**U_θ**	**Moment (Nm/Kg)**	**Moment Arm (cm)**	**U_θ**	**Moment (Nm/Kg)**
1	0.98	-0.12	-0.04	0.29	-0.03	-0.00	0.84	-0.11	-0.03
2	1.14	-0.12	-0.02	0.34	-0.03	-0.00	0.97	-0.11	-0.02
3	1.07	-0.12	-0.04	0.32	-0.03	-0.00	0.90	-0.11	-0.03
4	1.05	-0.12	-0.02	0.31	-0.03	-0.00	0.89	-0.11	-0.02
5	1.02	-0.12	-0.01	0.30	-0.03	-0.00	0.86	-0.11	-0.01
6	1.04	-0.12	-0.01	0.31	-0.03	-0.00	0.88	-0.11	-0.01
7	1.12	-0.12	-0.03	0.33	-0.03	-0.00	0.94	-0.11	-0.03
8	1.17	-0.12	-0.00	0.35	-0.03	-0.00	1.00	-0.11	-0.00
9	1.06	-0.12	-0.01	0.31	-0.03	-0.00	0.90	-0.11	-0.01
10	1.04	-0.12	-0.01	0.31	-0.03	-0.00	0.88	-0.11	-0.01
Average ±SD	1.10±0.1	-0.10±0.0	-0.02±0.0	0.32±0.0	-0.03±0.0	-0.0±0.0	0.91±0.0	-0.11±0.0	-0.02±0.0

Table 5-13: Mass Normalized maximal Vastus Medialis Longus moments about femoral longitudinal axis

	Model A			Model B			Model C		
Participant	**Moment Arm (cm)**	**U_θ**	**Moment (Nm/Kg)**	**Moment Arm (cm)**	**U_θ**	**Moment (Nm/Kg)**	**Moment Arm (cm)**	**U_θ**	**Moment (Nm/Kg)**
1	1.96	-0.08	-0.01	2.58	-0.11	-0.19	3.61	-0.15	-0.05
2	2.26	-0.08	-0.01	2.98	-0.11	-0.02	4.18	-0.15	-0.04
3	2.12	-0.08	-0.04	2.12	-0.11	-0.04	3.91	-0.15	-0.05
4	3.91	-0.08	-0.05	2.75	-0.11	-0.01	3.86	-0.15	-0.03
5	2.03	-0.08	-0.01	2.66	-0.11	-0.01	3.73	-0.15	-0.03
6	2.05	-0.08	-0.01	2.71	-0.11	-0.01	3.80	-0.15	-0.03
7	2.20	-0.08	-0.01	2.90	-0.11	-0.02	4.06	-0.15	-0.06
8	2.33	-0.08	-0.01	3.07	-0.11	-0.01	4.3	-0.15	-0.02
9	2.11	-0.08	-0.01	2.77	-0.11	-0.01	3.89	-0.15	-0.02
10	2.06	-0.08	-0.01	2.71	-0.11	-0.01	3.80	-0.15	-0.02
Average ±SD	2.30±0.5	-0.08±0.0	-0.02±0.0	2.73±0.2	-0.11±0.0	-0.03±0.1	3.91±0.2	-0.15±0.0	-0.04±0.0

Table 5-14: Mass Normalized maximal Vastus Lateralis moments about femoral longitudinal axis

	Model A			Model B			Model C		
Participant	**Moment Arm (cm)**	**U_θ**	**Moment (Nm/Kg)**	**Moment Arm (cm)**	**U_θ**	**Moment (Nm/Kg)**	**Moment Arm (cm)**	**U_θ**	**Moment (Nm/Kg)**
1	4.20	0.17	0.15	3.63	0.15	0.09	5.37	0.22	0.27
2	4.85	0.17	0.12	4.20	0.15	0.09	6.21	0.22	0.23
3	4.54	0.17	0.11	3.93	0.15	0.09	5.81	0.22	0.22
4	4.48	0.17	0.08	3.88	0.15	0.08	5.73	0.22	0.15
5	4.34	0.17	0.09	3.76	0.15	0.07	5.55	0.22	0.17
6	4.41	0.17	0.08	3.82	0.15	0.05	5.64	0.22	0.14
7	4.72	0.17	0.17	4.09	0.15	0.11	6.04	0.22	0.31
8	5.00	0.17	0.06	4.33	0.15	0.05	6.39	0.22	0.12
9	4.52	0.17	0.06	3.91	0.15	0.05	5.78	0.22	0.11
10	4.42	0.17	0.06	3.83	0.15	0.06	5.65	0.22	0.12
Average ±SD	4.55±0.2	0.17±0.0	0.10±0.0	3.94±0.2	0.15±0.0	0.07±0.0	5.82±0.3	0.22±0.0	0.18±0.1

Table 5-15: Mass Normalized maximal Vastus Intermedius moments about femoral longitudinal axis

	Model A			Model B			Model C		
Participant	**Moment Arm (cm)**	**U_θ**	**Moment (Nm/Kg)**	**Moment Arm (cm)**	**U_θ**	**Moment (Nm/Kg)**	**Moment Arm (cm)**	**U_θ**	**Moment (Nm/Kg)**
1	3.39	0.13	0.09	2.77	0.10	0.05	3.57	0.13	0.11
2	3.92	0.13	0.07	3.20	0.10	0.05	4.12	0.13	0.09
3	3.66	0.13	0.07	2.99	0.10	0.05	3.86	0.13	0.09
4	3.62	0.13	0.04	2.95	0.10	0.04	3.81	0.13	0.06
5	3.51	0.13	0.06	2.86	0.10	0.04	3.69	0.13	0.07
6	3.56	0.13	0.05	2.91	0.10	0.03	3.75	0.13	0.06
7	3.81	0.13	0.10	3.11	0.10	0.06	4.01	0.13	0.13
8	4.04	0.13	0.04	3.30	0.10	0.03	4.25	0.13	0.05
9	3.65	0.13	0.04	2.98	0.10	0.02	3.84	0.13	0.04
10	3.57	0.13	0.04	2.91	0.10	0.03	3.75	0.13	0.05
Average ±SD	3.67±0.2	0.13±0.0	0.06±0.0	3.00±0.2	0.10±0.0	0.04±0.0	3.87±0.2	0.13±0.0	0.08±0.0

On average, the VL contributed the highest magnitude to the transverse moments of the thigh, followed by the VI, VML, and VMO. Model C showed the highest average values between the three models. The moment arms of the VMO were substantially smaller than the other muscles, potentially contributing to the small magnitude of the VMO transverse moments, as the unit vectors for the VMO were comparable to the VML and VI unit vectors. The VL had the largest unit vector values indicating a high level of horizontal pull.

On average, the VMO made little contribution to the moments in the transverse plane. The VMO in all Model B provided virtually no contribution, and Model A and C provided insignificant contributions. The VML provided more contribution than the VMO, also in the negative direction. On average, the VML in all models produced insignificant contributions in the transverse plane.

The VL had the highest magnitude of transverse moment contribution and was directed positively. All average moments recorded were significant. Interestingly, the VI also made somewhat significant contributions to the positive moment direction, with Model C having the best average contribution.

When the moments of all four quadriceps muscles are combined, the net moment caused by the muscles can be calculated. As previously discussed, there is an overall positive moment evoked by the muscles. The net moment caused by the quadriceps upon the femur about the longitudinal axis can be seen below in Table 5-16: Net moment from quadriceps upon femur about the longitudinal axis at instance of peak knee extensor moment. Model C consistently produced the highest positive moment.

Table 5-16: Net moment from quadriceps upon femur about the longitudinal axis at instance of peak knee extensor moment.

Participant	Net Moment (Nm/Kg)		
	Model A	**Model B**	**Model C**
1	0.18	0.12	0.28
2	0.15	0.12	0.24
3	0.13	0.11	0.23
4	0.10	0.09	0.16
5	0.13	0.09	0.20
6	0.10	0.07	0.16
7	0.20	0.17	0.32
8	0.09	0.07	0.14
9	0.08	0.06	0.12
10	0.08	0.07	0.13
Average ±SD	0.12±0.03	0.10±0.03	0.20±0.06

Chapter 6: Discussion

6.1. Force and Moment Outcomes

6.1.1. Net Joint Moment

The net knee extensor moment measured during the vertical jump did not achieve the maximum values elicited within the isometric knee extension exercise. As seen in Table 5-1, on average, the peak knee extensor moment achieved during the vertical jump is just over half of the peak isometric knee extensor moment value. It is expected that there is an element of co-contraction of the hamstrings happening during the vertical jump, therefore the knee extensor moment likely underestimates the true maximum capacity of the quadriceps [38], [39]. Hamstring involvement creates a knee flexor moment and detracts from the knee extensor moment, creating a lesser knee extensor net joint moment. The isometric test was designed to eliminate the need for balance, or for any gravitational influence because of the ninety-degree knee angle, therefore reducing knee flexor involvement and isolating knee extensors and the associated moment.

Although the net joint moment cannot elucidate the maximum contribution of the knee extensors, it does show the minimum contribution and that the knee extensors prevail over the knee flexors. However, it is assumed that the isometric knee extension test will provide a reliable estimate of the maximum contribution ability of the knee extensors and serves as an upper limit of force potential when considering the knee extensors at the same joint angle as the maximum isometric test during a vertical jump.

6.1.2. Force upon the Leg

The mass normalized resultant force upon the leg was calculated as the force acting through the patellar ligament upon the tibial tuberosity. It was assumed that the vector of the patellar ligament only had Y and Z components as the X component was relatively ineffective and allowed for the use of only the knee extensor moment to calculate the force in the sagittal plane. It can be seen in Table 5-1 that each model retains the same trend throughout the exercise, but the individual values differ between models. Model B consistently produced the highest force per kilogram throughout all trials, speaking to the effect that bone geometry has upon force production.

6.1.3. Forces upon the Patella

The mass normalized directional forces upon the patella were calculated using the line of action of the muscle tendons from the patella to their corresponding bony attachment, as well as the associated unit vectors. Since the resultant PL force was known, the forces of the QT and VMO could be calculated using a system of equations. The X involvement of the muscles varied between models, although there was very little relative X involvement across models. Interestingly, most models returned X values all with negative directionality, indicating that all three muscles were pulling upon the patella in the medial direction. And in the case where the QT or VMO did pull in the positive direction, the contribution was minimal and unmeaningful. Although this is expected of the VMO, it was expected that the other muscles would have to compensate with lateral force to balance the forces upon the patella. This could indicate that there is another force acting upon the patella to prevent medial collapse other than the three muscle tendons studied. This is contrary to the initial hypothesis portrayed in Figure 5-2 that shows the three muscles balancing each other out upon the knee joint center. Other potential contributories could include the lateral cruciate ligament which crosses the knee joint on the lateral side, patellofemoral contact force, and patellofemoral and patellotibial ligaments.

The Y involvement of each muscle was expected to be negative for all muscles as this indicates posterior pull of the patella into the knee joint. The PL and QT produced similar Y forces upon the patella, with the QT typically producing slightly more force. The VMO contributed much less force in the Y direction compared to the PL and QT, which is to be expected given its relatively anterior attachment location upon the femoral condyles, compared to the relatively posterior bony attachments of both the QT and PL. The Y force was the highest among the maximum forces most likely because of its instance during the vertical jump, where a squat position would cause high loads in the Y direction upon the patella with the position of the femur causing the muscles to pull posteriorly on the patella.

The Z involvement of each muscle differed most in this direction. The graphical representation indicated expectations of the PL producing a negative Z force in the inferior direction, and the QT and VMO producing a positive Z force in the superior direction. Given the inferior patellar and superior tibial attachment location of the PL and the superior patellar and inferior femoral attachment location of the QT and VMO, these findings are expected. However, the VMO force dips into the negative near the peak knee extensor moment instance, which can be attributed to the location of the proximal attachment location moving beneath the distal attachment point during the squat. This resulted in the Z force of the VMO being recorded as negative within the tables, as it coincided closely with the lowest and most loaded instance in the vertical jump. Overall, the PL produced more Z force than the QT and VMO combined. The net Z force did not balance out to 0, indicating that there could be another force present upon the patella that prevents substantial inferior movement from the patella.

6.1.4. Individual Quadriceps Forces

The individual quadricep forces were indicative of their relative PCSA contributions. There was a large discrepancy of maximum muscle forces between participants and models. The PCSA calculations were heavily influenced by the total forces calculated from the previous section finding PL, QT, and VMO forces upon the patella. Each participant appeared to have a unique pattern of individual muscle involvement, although the VMO and VL were often the highest contributors, and the VML was the lowest.

The resultant force from the VMO was often calculated to be more than what the PCSA percentage would suggest its contribution should be in comparison to the other quadriceps muscles. The PCSA percentage values used to split the quadriceps forces did not specify the VML and VMO percentage split, resulting in the need to calculate the leftover contribution of the VML from the VMO and QT force. Because of the large relative contribution of the VMO, and the previously known normative PCSA values for the other quadriceps, the VML would often be calculated to have uncharacteristically small PCSA content. Therefore, the force output would also be unexpectedly small.

6.1.5. Moments about the Longitudinal Axis of the Femur

The hypothesis of this project was that the VMO would exert a substantial external rotational force about the longitudinal axis of the femur at the proximal femoral attachment location. Because of the wrapping nature of the quadriceps muscles from the posterior to anterior side of the femur, there is expected to be external rotational pull upon the proximal femoral attachment location of the muscles. The specificity of interest in the VMO stemmed from the oblique orientation of the fibers, and the expectation that this orientation would contribute more rotational force than the longitudinally oriented fibers of the other vasti muscles.

The moments acting about the longitudinal axis of the femur were calculated using a cylindrical coordinate system. The radius of the cylinder (moment arm), and the unit vector of the force contribution perpendicular to the radius vector could be used to calculate the moment acting about the longitudinal axis. A visual representation of the longitudinal axis can be found in Figure 2-1:Femoral geometry and the interpretation of the mechanical axis of the femur corresponding with the longitudinal axis labelled in black, as opposed to the femoral axis labelled in red. in the introduction.

The VL and VI produced internal rotation moments, and the VMO and VML produced external rotation moments. The combined internal rotation moment production exceeded the combined external rotation moment across participants. The average VMO moment showed no meaningful contribution, and the VML showed very minimal contribution.

The lack of moment production by the VMO could be attributed to the substantially smaller moment arm compared to the other quadriceps muscles, with the largest VMO moment arm being less than half of the next smallest moment arm value. The location of the proximal attachment location is much lower on the femoral shaft than the other muscles and is located much closer to the longitudinal axis. The proximal attachment locations for the VML, VI, and VL are higher on the shaft near the femoral head and are laterally located to the proximal end of the longitudinal axis. Although these attachment locations have a similar radial location compared to the shaft of the femur, their proximity to the longitudinal axis creates a larger moment arm about this axis than the VMO.

The VI produced a much higher moment than was expected. Given that the VI is the only vasti muscle that does not wrap around the femur, it was expected that there would be little to no transverse moment interaction. However, when considering the path of the muscle compared to the longitudinal axis, there is lateral involvement from the distal to proximal attachment points. The VI also has a higher-than expected moment arm given its location compared to the longitudinal axis. This combined with the lateral involvement created a substantial internal rotation moment. The VML had a lower-than-expected moment which could be attributed to its small moment arm, being more medial and closer to the origin of the axis. But the VL had a high, yet somewhat expected moment that could be attributed to its high moment arm, as it was most lateral and furthest from the axis.

There was a large difference between some models. Within the models for the VMO, there was no meaningful difference between models as the contribution itself was not meaningful. The VML showed more difference between models, but like the VMO, the contribution itself was not substantial. The VI showed some differences in models, but the VL showed substantial differences in models with the highest average being more than double the smallest average. Since this muscle clearly contributed the most to the rotational moments discussed, it would be expected that it would also demonstrate the highest level of variability.

6.1.6. Relevance of Thigh Moments

Although it is clear that the vasti muscles do in fact exert a rotational moment upon the longitudinal axis of the femur, it was important to consider whether or not the values were meaningful enough to be relevant. Chiu [40], investigated the minimum effect worth detecting a net joint moment. When considering a transverse moment at the hip, the minimum effect worth detecting was found to be 0.06 Nm/Kg, and a larger effect value was found to be 0.12 Nm/Kg which indicated a greater difference. For reference, the minimum and larger effect values for a sagittal hip moment were 0.11 Nm/Kg and 0.25 Nm/kg respectively, and the minimum and larger effect values of a frontal hip moment were 0.11 Nm/Kg and 0.24 Nm/Kg respectively. The transverse moment values are clearly lower than the corresponding values for a sagittal or frontal moment at the hip, showing that a lower relative transverse moment is to be expected. It should be noted that the moments calculated in this thesis are solely from the quadriceps upon the femur, whereas the moments calculated by Chiu were found at the hip joint. Therefore, these values are used as a peripheral reference for the analysis of the values exerted upon the femur.

Although it was determined by the research team that a value of 0.12 Nm/Kg would be considered significant, the contribution compared to a known meaningful moment, such as the knee extensor moment, should be addressed. The expectation was that the knee extensor moment would be substantially larger than the calculated transverse rotational moment as there is still far more movement about the X axis of the knee joint. Upon comparing the knee extensor moment and the calculated net moment upon the femur at the instance of peak knee extensor moment, Model B showed that the average transverse moment was only seven percent of the knee extensor moment, whereas Model C was most substantial with the average transverse moment being worth fifteen percent of the knee extensor moment. Compared to the values published by Chiu where the substantial value of a transverse moment would be worth nearly fifty percent of the corresponding sagittal and frontal moments, this is lower than what would be considered ideal. However, as explained earlier, it is expected that the transverse moment would be lower than the frontal and sagittal moments. Therefore, smaller values of a transverse moment should be considered important despite their smaller values.

The external rotation moment created by the VMO was unsubstantial and did not support the initial hypothesis. However, the net internal rotation moment indicates there may be a natural inclination to internally rotate the femur at the hip during the loading phase of a vertical jump. This femur orientation creates dynamic knee valgus, which requires a hip external rotator net joint moment [40]. Once put into an internal rotation position during the loading phase of the jump, the external rotators are responsible for returning the hip to neutral during the upward loading phase and subsequent jump. As the gluteus maximus is both a hip extensor and external rotator, the action of the vasti to internally rotate the femur may possibly be a strategy to engage the gluteus maximus when large hip extensor moments are required, such as during a maximal vertical jump.

6.2. Importance of the VMO

It was hypothesized that the VMO would have a substantial stabilizing action upon the patella that resisted counter force from the QT and PL upon the patella. However, our findings did not suggest that the purpose of the VMO was to counteract. Instead, we found that the VMO often pulled in the same direction as the QT and the PL in the medial direction. The VMO did pull in the medial direction, which was expected, but what was not expected, was the QT pulling medially rather than laterally, and the PL not then countering the forces produced by the QT and VMO together. Therefore, the three muscles may not stabilize the patella upon the knee in the way that was previously predicted, but whether the VMO is involved in patellar stabilization cannot be ruled out based on these findings alone. Since the patella clearly does not leave the knee joint, the VMO must be responsible for pulling on the patella in a meaningful way that causes ideal movement. It may also help prevent dislocation of the patella upon the knee joint, possibly synergistically with the QT and PL tendons. It is predicted that the VMO is still involved in patellar tracking and stabilization, but in response to forces that were not part of the original hypothesis. Investigation into a lateral force upon the patella that counters the net medial pull from the VMO, QT, and PL should be considered.

6.3. Bone Geometry

It was important to the research team that the different anatomical models were not averaged, with the intention of showing the significance of the difference in anatomy in influencing the forces and moments experienced at each bone. Models are typically averaged to account for variability and to address reliability concerns, but this was not the goal of the study. Forces and moments were calculated individually for each model and displayed graphically to highlight the differences that bone geometry can make despite having the same participant as their jumper. There were meaningful differences between models highlighting the effect that bone geometry has on force production.

6.4. Strengths

The most prominent strength of this study is the method of collection of bone geometry. As discussed in the background, the ability to collect accurate bone geometry is limited with other methods and is dependent on invasive methods. There has never been, to our knowledge, a collection of the surface anatomy of human bone to the extent that was done in this research study. Obtaining the full extent of the bone geometry rather than only coordinates of bony landmarks contributed to the robustness of our model.

Another strength was the development of our own procedures for capturing human bone using photogrammetry. Despite a plethora of documentation on how to use photogrammetry, there is no published evidence of using photogrammetry for this purpose. A methodical procedure was established after multiple iterations. The procedure was also verified for its ability to reflect the geometry accurately [9]. This highlights the novelty of the project.

The moments calculated as acting upon the femur were calculated in a novel way, involving the cylindrical coordinate system. Although use of the cylindrical coordinate system is not novel, using it for the purpose of calculating a moment acting about the rotational axis of the conceptual cylinder highlights how basic mathematical principles can be used in conjunction with anatomy to explain biomechanical motion. Typically, muscles are modelled as going from point to point in a straight line and does not account for wrapping about the bone. Therefore, the prospect of mapping the path of the muscle and extracting the tangential force to the attachment location is not well explored. This was a novel approach to investigating muscular action and should be further cultivated when considering the three-dimensional anatomy of human musculature.

6.5. Limitations

6.5.1. Dissection Limitations

During the dissection of the bones, we were not given permission to use more aggressive methods of removing non-bone tissue such as boiling or chemical usage. Because of this limitation, the recorded bone geometry included trace amounts of non-bone tissue that should not be contributed to the surface anatomy of the bone. Using dissection methods by hand, it was difficult to strip the bones of all their non-bony tissue without compromising the integrity of the true structure of the bone. The femur and leg bones were least affected by this problem, although the connection between the tibia and fibula was a difficult area to clean out. The small size of the patella was the biggest problem with the bone. However, the foot was the most difficult to dissect as we wanted to retain the integrity of the full bone structure without separating each individual bone. Removing the skin and muscle tissue without damaging the ligaments took significant care and patience, and it was important to recognize when tissue must be left behind to preserve the structure.

Another limitation related to the previous limitation, was the inability to collect the exact bone geometry of the foot because of the desire to retain the structure. Unobstructed bone geometry was not included in this study as the inclusion of individual foot bones would have drastically increased the difficulty of dissection, and the ability to match bones to motion capture. The only proven accurate method of capturing motion of the foot is to place markers on the first and fifth metatarsal heads, which is only two out of twenty-six bones in the foot, hence why the rest of the structure was retained. Although there was little emphasis placed on the foot in this study, future research can improve upon this practice as to include proper bone geometry of individual foot bones, and inclusion of the behaviour of a foot as a multi-segment rigid body.

The weakness of this study in direct comparison to other methods of obtaining bone geometry is the fact that our models are collected from cadaveric specimens. This methodology limits our investigation to post-mortem bone models, making it impossible to determine bone geometry and subsequent calculated values for a living person. Since the purpose of the study was to determine *if* bone geometry can change the forces and moments experienced at the joint, and therefore solidify its relevance in the investigation of anatomy, the need to extend our findings to the living population is outside the scope of this investigation. However, if future techniques of imaging permit, the ability to determine an individual's bone geometry and its effect on their movement could drastically improve the sphere of athletic performance and muscle training.

6.5.2. Practical Limitations

An assumption that should be highlighted was the decision to consider the patellar ligament action upon the leg as only acting in the sagittal plane. Given the limited medial-lateral involvement of the ligament, this assumption could be justified in the pursuit of determining the force experienced in the patellar ligament. The motion capture data provides moment values experienced at the knee in all three directions. The force experienced in the sagittal plane can be determined by dividing the knee extensor moment by the calculated moment arm in the sagittal plane. Having a singular force value in the sagittal plane that is considered the total force experienced in the patellar ligament, significantly eases the difficulty of determining the forces experienced at the patella.

It was also assumed that the joint reaction forces between the bones at the knee had moment arm values of zero, which leads to a moment value of 0, and subsequent negligible contribution upon the joint. When considering a sum of moments equation, the expectation of equal and opposite reactions paired with an assumption of a negligible moment arm substantially simplifies the complicated equations needed to determine the forces experienced at the knee.

Despite having access to cadaveric bone geometry, the bone geometry of the participants cannot be known using this method. Therefore, the specific effect of an individual's bone geometry upon their movement cannot be studied. However, the broad finding of the effect of bone geometry on function can still be explored.

Similar to the aforementioned, the exact physiological cross-sectional area (PCSA) of the participants muscle cannot be known without more advanced imaging techniques. The division of PCSA among the quadriceps is based on literature values for the average male and female. Usage of a simple splitting of muscle force based on PCSA could be improved upon by investigating the advantages of optimization and more advanced methods of muscle force calculations.

During the vertical jump, there is co-contraction among the muscles responsible for moving the knee. Although there is a dominant knee extensor moment, the value assessed during the jump can only be assumed to be the minimum extensor moment experienced at the knee, as the reported value is a net value between the knee extensor moment and the knee flexor moment. This principle is also true for knee external rotation and internal rotation, and knee abduction and adduction. Therefore, the forces calculated by use of the knee extensor moment reflect the minimum involvement of the muscles, and not the exact contribution of the muscles. This is addressed by reporting the estimated maximum force that can be produced by the muscle at a given joint angle, based on PCSA, and the force-length relationship. Therefore, it is known that the actual force lies within the window created by the estimated max force and the known minimum force.

A limitation experienced during the motion capture stage was associated with the size of the motion capture markers placed on the patella. It was important to the research team to track the patella more accurately than with a single marker placed in the centre of the patella. There were four markers placed on the patella, of which were relatively large in size compared to the patella itself. Because of the nature of the movement of the patella relative to the skin, there was also a concern for skin motion artifact among the markers. However, the model was successful in reporting realistic values similar to patella models with singular markers, but with more information to inform the calculation of advanced force values.

Wrapping was a major interest in this study and was addressed by using a cylindrical coordinate system to model the muscle as wrapping around a cylinder. Each muscle was given its own coordinate system to wrap around its own unique cylinder related to the distance between the longitudinal axis and the proximal attachment location. Although the wrapping path of the quadriceps does follow a somewhat circular path, a cylinder is generic and does not represent the exact muscular path around the femur. There is future room for improvement in creating a mathematical model that allows for the constraints of the shape of the femur rather than a known geometric shape. Given the complexity of generating such a model, this assumption was made for the purposes of creating this pilot project and evaluating the importance of creation.

Finally, the forces on the patella for participant three were highly inconsistent with the other nine participants. Despite the three other reported sets of data following the trends of the other participants, the X and Y patellar force information appeared to be flipped, while the Z information aligned with the trend of the other participants. The same code was used to calculate the forces for every participant, indicating that information from the participants motion capture data may have errors Motion capture information was verified for incorrect labelling or tracking, and no errors were found. It is expected that the patella was improperly rotated about its Z axis, which would explain why the X and Y information was different, yet the Z information did not appear incorrect. This may have occurred during the transformation of the forces into the patellar coordinate system as the other results did not appear to be substantially different from the other nine participants. Despite nine successful outputs of information, it is clear that there may be an oversight within the code or the rotation matrix exportation that results in the incorrect orientation of the patella, and subsequent issues with calculating the proper force on the patella. Since this only affected ten percent of the results, the values from participant three were unused. However, future users should be aware of potential errors and should always use this code and exportation method among multiple participants to isolate errored results.

6.6. Future Investigations

This project affords a significant foundation for future research. Since this project focused on the lower extremity, there is an obvious ensuing interest in a project that could look at the trunk and upper extremity. Work within the biomechanics laboratory has already begun on creating photogrammetry models of the humerus, radius and ulna, and the hand. However, investigation of the trunk poses unique challenges because of its exponentially greater capacity for movement about joints that are hard to track using motion capture. The structure of the spine, composed of thirty-three vertebrae, of which are especially difficult to track superficially in more than one location, poses challenges that will need advances in biomechanical technologies to accurately assess using our methods.

There is also interest in using these methods to isolate the forces of the medial and posterior thigh muscles upon the knee. However, the quadriceps is unique in its single attachment point upon the tibia, which is different from the individual attachment points of the remaining muscles in the thigh. This detail about the quadriceps allowed significant assumptions that cannot be made about other muscles. Therefore, without the use of optimization methods, further mathematical considerations and advancements would have to be made to accurately isolate the forces within each muscle.

Chapter 7: Implications

7.1. Conclusions

There were three main takeaways from the project. The first being the unexpected medial-lateral forces upon the patella. With the expectation of a balanced medial-lateral pull, the outcome of all the recorded muscle tendons pulling medially upon the patella was surprising. From the perspective of a simple free body diagram, if the VMO, QT, and PL are the only forces acting upon the patella, it would be pulled away from the knee in the medial direction without counter forces. However, the patella obviously does not leave the knee joint at any time during the vertical jump, leading the researchers to believe that there is another unrecorded force upon the patella that could contribute to a lateral counter force. As discussed previously, there are multiple potential sources that could contribute to lateral action on the patella, which should be included in future investigation using this method.

Another takeaway from the project was the finding of meaningful moment values upon the femur exerted by the vasti muscles. As explained in the introduction, the vasti muscles are unique as they wrap around the femur from posterior to anterior, making them prime candidates for rotational pull about the longitudinal axis of the femur. Using a cylindrical coordinate system, the force exerted about the axis at the location of muscular attachment, or the moment about the longitudinal axis, could be calculated. It was hypothesized that the VMO would have a high moment in this direction because of its oblique nature, but this was not the case. The VML and VMO had unmeaningful moments, yet the VL and VI produced substantial moments, that contributed to a net internal rotation moment caused by the vasti muscles. The overall internal rotation moment points to ideal movement mechanics during active movements such as a vertical jump as seen in this study. Although the outcome did not support the hypothesis, the discovery of a substantial transverse moment upon the femur did support the notion that the wrapping of the vasti muscles was relevant to their full function.

The last main takeaway was the importance of bony geometry on force and moment production. The initial motivation for this project was spurred by questioning how important bone geometry was in relation to force production. Upon observation of different femurs, it was clear that despite being the same bone with similar functions, there were still significant differences in the geometry. In seeing how the different bone models influenced the force and moment values throughout ten different participants, it became clear that bone geometry was indeed relevant in influencing movement. There are many reasons as to why the specific bone geometry may have influenced the results, one significant reason being that more substantial bony prominences may contribute to larger moment arms and therefore larger moments. But it is clear that different geometry does have an impact on force production. However, it is unknown at this stage what the explicit reasonings are for the different outcomes. Further investigation into the minutiae of specific bone geometry could be highly beneficial in the scope of biomechanics.

7.2. Significance

The creation of a musculoskeletal model using proper bone geometry holds great significance in the progression of movement analysis and prediction. It improves upon an already well-established area of research that has demonstrated its importance to multiple populations. In the scope of high-performance sports, there are multiple applications of a functional computerized model. This may include the ability to probe for injury or injury potential, based on an athlete's technique. In the scope of the disabled community, it may allow for simulations of assistive devices to aid in the selection of an appropriately suited device. This is more time efficient and functionally effective for people experiencing disability.

The importance of this specific model in relation to previously generated models is the emphasis on accurate bony geometry and realistic muscle lines of action. Acknowledgement of the unique properties of the bones and their effects on force distribution introduces another step towards a fully functioning model. And in contrast to previous models with two-dimensional muscle paths following the straight line from the proximal to distal attachment locations, the wrapping path of the vasti muscles was included. This is highly important for the accurate estimation of directional forces upon the femur. Furthermore, clarity of the function and individuality of the vasti muscles can be beneficial in exploiting its function more effectively in the context of injury prevention and performance.

The findings of a predominant internal rotation moment upon the proximal femur was not expected, as the VMO was expected to produce a substantial external rotation moment. However, this supports the notion that a naturally occurring internal rotation moment when jumping maximally may point to the need for further investigation into the long-standing assumption that external rotation of the femur upon the hip is considered the best practice. This outcome was not expected by the researchers.

Ultimately, the pursuit of uncovering more information about the functioning of the muscles upon their attachment bones is beneficial to many populations. The action of muscles in the X and Y directions are well studied, but the action within the Z direction is often neglected. Given the outcomes of the study, it is clear that action in the Z direction is substantial and deserves recognition. This could influence methods of training that a coach implements to target the previously disregarded potential of a muscle. This can be especially important for sports that require high power and explosiveness, since the information was collected under the premise of maximal force exertion. It is less clear how the information collected can be of benefit to endurance athletes, although knowledge of higher involvement of a muscle can still inform exercise and training method selection.

This work can be an invaluable step in the pursuit of a fully functioning, computerized model of the human lower extremity.

References

[1] Y. Kim, Y. Jung, W. Choi, K. Lee, and S. Koo, "Similarities and differences between musculoskeletal simulations of OpenSim and AnyBody modeling system," *J. Mech. Sci. Technol.*, vol. 32, no. 12, pp. 6037–6044, Dec. 2018, doi: 10.1007/s12206-018-1154-0.

[2] J. Rasmussen, "The AnyBody Modeling System," in *DHM and Posturography*, Elsevier, 2019, pp. 85–96. doi: 10.1016/B978-0-12-816713-7.00008-8.

[3] W. F. Dostal, G. L. Soderberg, and J. G. Andrews, "Actions of Hip Muscles," *Phys. Ther.*, vol. 66, no. 3, pp. 351–359, Mar. 1986, doi: 10.1093/ptj/66.3.351.

[4] S. L. Delp and J. P. Loan, "A computational framework for simulating and analyzing human and animal movement," *Comput. Sci. Eng.*, vol. 2, no. 5, pp. 46–55, Oct. 2000, doi: 10.1109/5992.877394.

[5] A. Seth, M. Sherman, J. A. Reinbolt, and S. L. Delp, "OpenSim: a musculoskeletal modeling and simulation framework for in silico investigations and exchange," *Procedia IUTAM*, vol. 2, pp. 212–232, 2011, doi: 10.1016/j.piutam.2011.04.021.

[6] L. Modenese and J.-B. Renault, "Automatic generation of personalised skeletal models of the lower limb from three-dimensional bone geometries," *J. Biomech.*, vol. 116, p. 110186, Feb. 2021, doi: 10.1016/j.jbiomech.2020.110186.

[7] "Creation of an MR Image • Magnetic Resonance in Medicine – The Basics – by Peter A. Rinck | NMR MR MRI | Essentials, introduction, basic principles, facts, history | The primer of EMRF/TRTF." Accessed: Sep. 08, 2023. [Online]. Available: https://www.magnetic-resonance.org/ch/06-06.html

[8] A. D. P. Bankoff, *Human Musculoskeletal Biomechanics*. Rikeka, Croatia: InTech, 2011.

[9] J. C. Feddema and L. Z. F. Chiu, "Accuracy and repeatability of 3D Photogrammetry to digitally reconstruct bones," *Morphologie*, vol. 108, no. 363, p. 100793, Dec. 2024, doi: 10.1016/j.morpho.2024.100793.

[10] H. Gray, *Gray's Anatomy*, 15th ed. New York: Bounty Books.

[11] J. S. P. Lumley, J. L. Craven, and J. T. Aitken, *Essential Anatomy*, Reprinted. Edinburgh and London: E. & S. Livingstone LTD., 1969.

[12] L. Z. F. Chiu and T. E. Daehlin, “Three-dimensional modelling of human quadriceps femoris forces,” *J. Biomech.*, vol. 120, p. 110347, May 2021, doi: 10.1016/j.jbiomech.2021.110347.

[13] F. Farahmand, W. Sejiavongse, and A. A. Amis, “Quantitative study of the quadriceps muscles and trochlear groove geometry related to instability of the patellofemoral joint,” *J. Orthop. Res.*, vol. 16, no. 1, pp. 136–143, Jan. 1998, doi: 10.1002/jor.1100160123.

[14] H. B. Rajput, “Variation in Morphometry of Vastus Medialis Muscle,” *J. Clin. Diagn. Res.*, 2017, doi: 10.7860/JCDR/2017/29162.10527.

[15] J. Peeler, J. Cooper, M. M. Porter, J. A. Thliveris, and J. E. Anderson, “Structural parameters of the vastus medialis muscle,” *Clin. Anat.*, vol. 18, no. 4, pp. 281–289, May 2005, doi: 10.1002/ca.20110.

[16] H. Toumi, G. Poumarat, M. Benjamin, T. Best, S. F’Guyer, and J. Fairclough, “New Insights into the Function of the Vastus Medialis with Clinical Implications,” *Med. Sci. Sports Exerc.*, vol. 39, no. 7, pp. 1153–1159, Jul. 2007, doi: 10.1249/01.mss.0b013e31804ec08d.

[17] E. Panagiotopoulos, P. Strzelczyk, M. Herrmann, and G. Scuderi, “Cadaveric study on static medial patellar stabilizers: the dynamizing role of the vastus medialis obliquus on medial patellofemoral ligament,” *Knee Surg. Sports Traumatol. Arthrosc.*, vol. 14, no. 1, pp. 7–12, Jan. 2006, doi: 10.1007/s00167-005-0631-z.

[18] T. D. O’Brien, N. D. Reeves, V. Baltzopoulos, D. A. Jones, and C. N. Maganaris, “Muscle–tendon structure and dimensions in adults and children,” *J. Anat.*, vol. 216, no. 5, pp. 631–642, May 2010, doi: 10.1111/j.1469-7580.2010.01218.x.

[19] T. G. Balshaw, T. M. Maden-Wilkinson, G. J. Massey, and J. P. Folland, “The Human Muscle Size and Strength Relationship: Effects of Architecture, Muscle Force, and Measurement Location,” *Med. Sci. Sports Exerc.*, vol. 53, no. 10, pp. 2140–2151, Oct. 2021, doi: 10.1249/MSS.0000000000002691.

[20] B. I. Prilutsky and V. M. Zatsiorsky, “Optimization-Based Models of Muscle Coordination:,” *Exerc. Sport Sci. Rev.*, vol. 30, no. 1, pp. 32–38, Jan. 2002, doi: 10.1097/00003677-200201000-00007.

[21] C. Alessandro, F. O. Barroso, A. Prashara, D. P. Tentler, H.-Y. Yeh, and M. C. Tresch, "Coordination amongst quadriceps muscles suggests neural regulation of internal joint stresses, not simplification of task performance," *Proc. Natl. Acad. Sci.*, vol. 117, no. 14, pp. 8135–8142, Apr. 2020, doi: 10.1073/pnas.1916578117.

[22] J. J. Elias, D. R. Bratton, D. M. Weinstein, and A. J. Cosgarea, "Comparing two estimations of the quadriceps force distribution for use during patellofemoral simulation," *J. Biomech.*, vol. 39, no. 5, pp. 865–872, Jan. 2006, doi: 10.1016/j.jbiomech.2005.01.030.

[23] "Applied Mathematical Programming." Accessed: Sep. 22, 2024. [Online]. Available: https://web.mit.edu/15.053/www/AMP.htm

[24] F. C. Anderson and M. G. Pandy, "Static and dynamic optimization solutions for gait are practically equivalent," *J. Biomech.*, vol. 34, no. 2, pp. 153–161, Feb. 2001, doi: 10.1016/S0021-9290(00)00155-X.

[25] S. A. Roelker, E. J. Caruthers, R. K. Hall, N. C. Pelz, A. M. W. Chaudhari, and R. A. Siston, "Effects of Optimization Technique on Simulated Muscle Activations and Forces," *J. Appl. Biomech.*, vol. 36, no. 4, pp. 259–278, Aug. 2020, doi: 10.1123/jab.2018-0332.

[26] H. Hatze, "The complete optimization of a human motion," *Math. Biosci.*, vol. 28, no. 1–2, pp. 99–135, Jan. 1976, doi: 10.1016/0025-5564(76)90098-5.

[27] M. M. Morrow, J. W. Rankin, R. R. Neptune, and K. R. Kaufman, "A comparison of static and dynamic optimization muscle force predictions during wheelchair propulsion," *J. Biomech.*, vol. 47, no. 14, pp. 3459–3465, Nov. 2014, doi: 10.1016/j.jbiomech.2014.09.013.

[28] H. Faber, A. J. Van Soest, and D. A. Kistemaker, "Inverse dynamics of mechanical multibody systems: An improved algorithm that ensures consistency between kinematics and external forces," *PLOS ONE*, vol. 13, no. 9, p. e0204575, Sep. 2018, doi: 10.1371/journal.pone.0204575.

[29] C. Xu, J. Reifman, M. Baggaley, W. B. Edwards, and G. Unnikrishnan, "Individual Differences in Women During Walking Affect Tibial Response to Load Carriage: The Importance of Individualized Musculoskeletal Finite-Element Models," *IEEE Trans. Biomed. Eng.*, vol. 67, no. 2, pp. 545–555, Feb. 2020, doi: 10.1109/TBME.2019.2917415.

[30] K. Rathnayaka *et al.*, “Quantification of the accuracy of MRI generated 3D models of long bones compared to CT generated 3D models,” *Med. Eng. Phys.*, vol. 34, no. 3, pp. 357–363, Apr. 2012, doi: 10.1016/j.medengphy.2011.07.027.

[31] O. L. Bruce, M. Baggaley, L. Welte, M. J. Rainbow, and W. B. Edwards, “A statistical shape model of the tibia-fibula complex: sexual dimorphism and effects of age on reconstruction accuracy from anatomical landmarks,” *Comput. Methods Biomech. Biomed. Engin.*, vol. 25, no. 8, pp. 875–886, Jun. 2022, doi: 10.1080/10255842.2021.1985111.

[32] E. A. Audenaert, C. Pattyn, G. Steenackers, J. De Roeck, D. Vandermeulen, and P. Claes, “Statistical Shape Modeling of Skeletal Anatomy for Sex Discrimination: Their Training Size, Sexual Dimorphism, and Asymmetry,” *Front. Bioeng. Biotechnol.*, vol. 7, p. 302, Nov. 2019, doi: 10.3389/fbioe.2019.00302.

[33] The MathWorks Inc., *MATLAB version 9.13.0 (R2022b)*. (2022). The MathWorks Inc. [Online]. Available: https://www.mathworks.com

[34] A. Vanezis and A. Lees, “A biomechanical analysis of good and poor performers of the vertical jump,” *Ergonomics*, vol. 48, no. 11–14, pp. 1594–1603, Sep. 2005, doi: 10.1080/00140130500101262.

[35] *Visual 3D*. C-Motion, Germantown, MD.

[36] A. L. Bell, D. R. Pedersen, and R. A. Brand, “A comparison of the accuracy of several hip center location prediction methods,” *J. Biomech.*, vol. 23, no. 6, pp. 617–621, Jan. 1990, doi: 10.1016/0021-9290(90)90054-7.

[37] R. Grelsamer and J. Klein, “The Biomechanics of the Patellofemoral Joint,” *J. Orthop. Sports Phys. Ther.*, vol. 28, no. 5, pp. 286–298, Nov. 1998.

[38] L. Z. F. Chiu, “Biomechanical Methods to Quantify Muscle Effort During Resistance Exercise,” *J. Strength Cond. Res.*, vol. 32, no. 2, pp. 502–513, Feb. 2018, doi: 10.1519/JSC.0000000000002330.

[39] M. A. Bryanton, J. P. Carey, M. D. Kennedy, and L. Z. F. Chiu, “Quadriceps effort during squat exercise depends on hip extensor muscle strategy,” *Sports Biomech.*, vol. 14, no. 1, pp. 122–138, Jan. 2015, doi: 10.1080/14763141.2015.1024716.

[40] L. Z. F. Chiu, "'Knees Out' or 'Knees In'? Volitional Lateral vs. Medial Hip Rotation During Barbell Squats," *J. Strength Cond. Res.*, vol. 38, no. 3, pp. 435–443, Dec. 2023.

[41] Agisoft LLC, *Agisoft Metashape Standard*. (2022).

Appendix A Dissection Procedures

Cadaver specimens that had been previously dissected by kinesiology students were obtained for further investigation. Cadavers had been previously skinned, with muscular, arterial, and nervous structures preserved. There were preliminary standards that each specimen had to meet to be considered for inclusion in the study. The presence of anatomical alterations such as a hip replacement, extreme malformation, or conditions that were not considered representative of the population prevented the cadaver from being included. However, if the condition of one side of the body was such that it did not affect the other side, the opposing side could be considered for usage. The standards for inclusion could change based on what was observed, but anomalies were recorded for transparency. The following table details the cadavers that were used, and the associated anatomical information. Each specimen was from the right side.

The quadriceps, hamstrings, and plantar flexor muscles were measured for later usage in computer-generated models of forces for the purposes of a fully functioning model. The length of the muscle belly, defined as the length of the muscle fibers between the proximal and distal tendons, was measured in centimeters. The length of the proximal and distal tendons was also measured. Both measurements of the tendons and muscle bellies were important to understand the contractile properties of each muscle, as tendon and muscle fibers behaved differently under stress. Certain muscles that possessed wrapping characteristics were measured twice, once on the proximal edge of the muscle, and again on the distal edge of the muscle. This was considered for the vastus medialis and vastus lateralis. The measurements were always repeated by the same person to ensure consistency.

The rectus femoris was measured first. Following measurement, the muscle and surrounding neurovascular structures could be stripped from the cadaver. The remaining three quadriceps muscles had to be retained for later measurement. The cadaver was turned to have the back facing up. The semitendinosus, semimembranosus, biceps femoris short head, and long head were measured. These muscles and surrounding neurovascular structures could be stripped, including the pelvic musculature. The gastrocnemius medial and lateral heads could now be measured. The distal end of the muscle belly measurement finished at the bottom of the triangular shape leading into the tendon. This was in the center of each head, but medial or lateral from the center of the tendon. The tendon of the gastrocnemius started proximally from the top of the diamond leading into the bellies, and towards the calcaneal tuberosity. The soleus could be measured using the underside of the muscle to determine the separation between belly and tendon.

Following these measurements, the lower extremity could be stripped of any structures apart from the remaining quadriceps muscles. The hip could be dislocated to allow better visualization of the vastus muscles. Further dissection could be made to dislocate the hip while taking care not to damage the quadriceps. The femur could be dislocated from the hip without damaging the quadriceps because the vastus muscles all originated from the femur rather than the pelvis. The vastus lateralis and vastus medialis were measured on each side of the muscle to better understand the wrapping that occurred. In addition, the vastus medialis oblique and vastus medialis longus were independently measured. The distinction was determined as the difference in fiber orientation and searching for a natural fascial plane. This line of separation could serve as the superior border for the vastus medialis oblique and the inferior border for the vastus medialis longus. If no fascial plane could be determined, the separation could be assumed to be where there was a significant difference in the direction of muscle fibers. Although this left room for interpretation, a more meticulous procedure could be established upon observation of the muscles. Measurements of the vastus muscles stopped at the border of the patella, as the patellar ligament was measured from the base of the patella. Following measurement, the vastus medialis and vastus lateralis could be dissected back to better expose the vastus intermedius. The final measurement of the vastus intermedius could be taken on the medial and lateral sides.

Following proper measurement of each muscle, the thigh and leg could be stripped of all structures. The femur, tibia, and fibula (leg), patella, and foot could all be dissected and cleaned of their connected tissues. The intention was to retain the foot structure, rather than dissect out each individual bone. Therefore, care was taken to preserve the ligaments that held the bones together. Determination of a standard of cleanliness was based on the material, as the dryness of the tissue affected the ability to clean the bones. This process was repeated for up to ten cadavers, while switching left and right sides, and taking note of the sex of the specimen.

Table A-1: Cadaver Anatomical Properties- Bone.

	Measurement (cm)	Model A	Model B	Model C
Femur	Length (GT to LFE)	41.9	49.9	41.2
	Intercondylar Width	7.9	8.9	9.2
	Femoral Head (A-P)	4.7	5.2	5.9
	Femoral Head (S-I)	4.5	5.2	6.1
Leg	Length (MC-MM)	34.0	43.3	36.5
	Length (FH-MM)	35.0	39.5	39.0
	Intercondylar Width	7.2	8.7	9.2
	Intermalleolar Width	4.4	7.8	7.4
Patella	Height	4.0	4.5	4.9
	Width	4.4	6.0	5.1
Foot	Calcaneus to First MT Head	17.2	18.5	19.2
	Calcaneus to Fifth MT Head	15.5	14.9	16.1

Appendix B Photogrammetry Procedures

Photography

Photogrammetry uses photos taken of a given object at various angles to capture its entire geometry. A Nikon D3500 camera was used to capture the images, and the software, Metashape (Agisoft) [41], was used to create 3D models of various objects [19]. To determine the usefulness of the software, various items were used for the creation of computerized models.

Camera settings such as shutter speed, International Organization for Standardization (ISO), and aperture were adjusted to achieve the best quality image for model construction. A Pentax Spotmeter V light meter was used to manage the variability of choosing camera settings which is created for determining the best camera settings in the current environment. The light meter assesses the lighting conditions and returns an ISO and shutter speed setting that will best capture an image in the current lighting. This allowed for variation in room lighting as the desired outcome is to allow others to regenerate their own models in any environment. However, it is recommended by Metashape (Agisoft) that zooming is not applied, and therefore the physical location of the photographer will be adjusted. This will be categorized as the object taking up most of the image (close), the object taking up half of the image (regular), or the object being less than half the image (far). This was included as a variable to determine the best practice to create optimal reconstructions.

Objects differed in characteristics, including shape, size, colour, and pattern. The chosen software distinguishes object geometry using edges and patterns. Inclusion of a "calibration object" such as a construction triangle was explored to determine if model accuracy and quality was improved. The rationale behind this consideration is to give the software edges and lines as reference points to better distinguish less distinguishable objects. A dodgeball which is uniform in shape, colour, and pattern served as a control to determine the ability of the software in reconstructing indistinguishable objects. A sample of the photos taken of the dodgeball with and without a calibration frame can be seen in Figure B-1. Variable objects included a white shoe, a patterned volleyball, and a dumbbell as seen in Figure B-2.

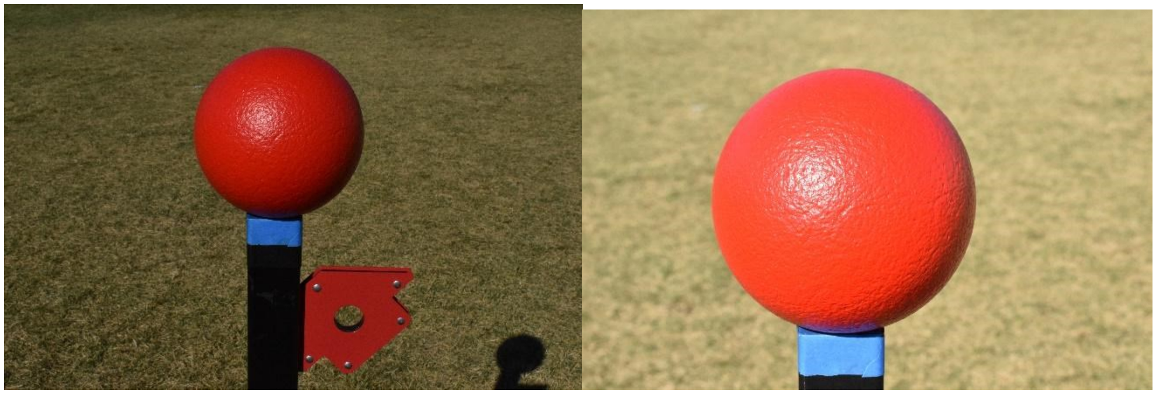


Figure B-1: Sample photos of dodgeball with and without calibration frame.

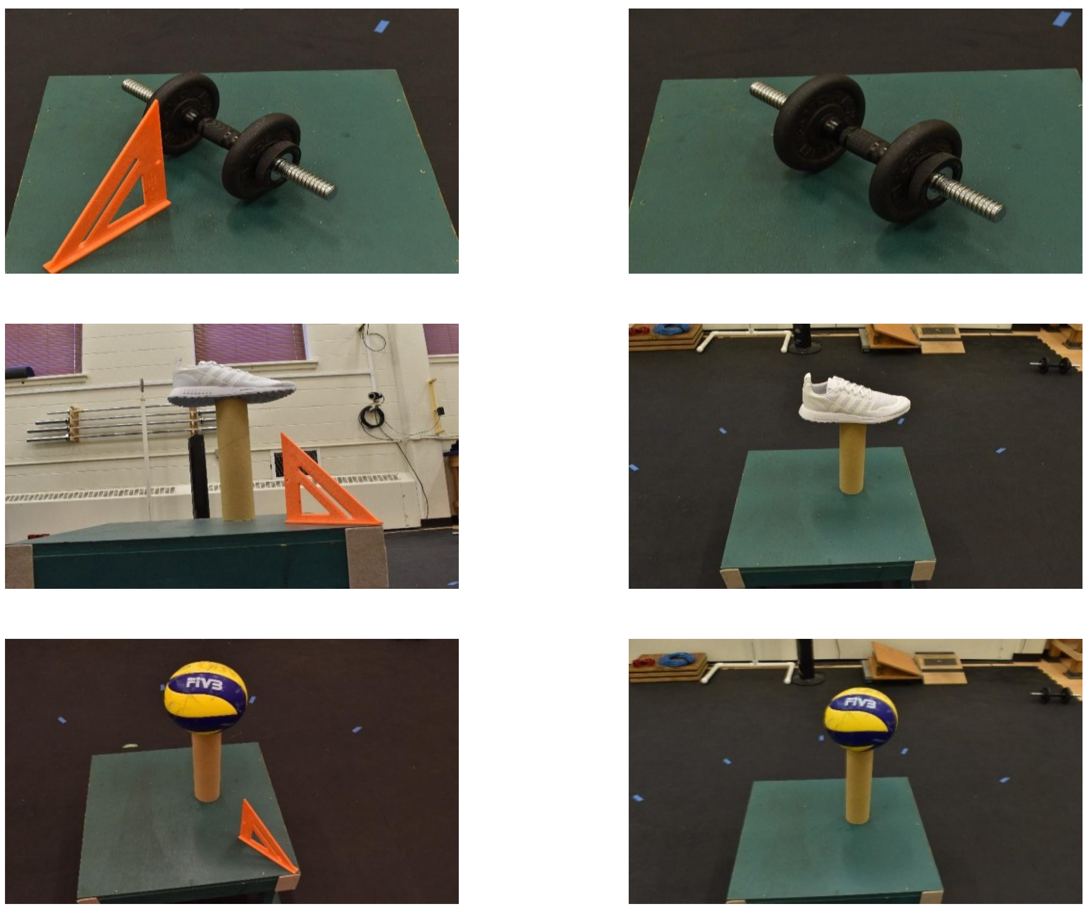


Figure B-2: Sample photos of dumbbell, shoe, and volleyball with and without calibration frames.

Techniques

Photos were taken using the “walk around” method, where the photographer walks around the stationary object. It is suggested that photos be taken to overlap by at least one third with each other. Therefore, a strategy of stepping thirty degrees around the object was used by placing pieces of masking tape on the floor every thirty degrees to retain consistency. The object is rounded three times, once at an angle below the object, once at an angle level with the object, and finally at an angle above the object. This results in thirty-six photographs that capture all angles of the object. Photos in Figure B-1 and Figure B-2 were taken using this strategy.

For the variable objects, photos were taken in high lighting and low lighting conditions. This was to account for how the difference in lighting may affect the quality of reconstructions. Additionally, sets of photos were taken at variable closeness to the object to test how this would affect detailing of the reconstructions. A “normal” closeness is defined as the object taking up fifty percent of the image, “far” being defined as taking up less than fifty percent of the image, and “close” being defined as taking up most of the image. Table B-1 shows the combinations of variables used to reconstruct multiple objects.

Table B-1: Camera Specifications Used.

Object	ISO	Shutter Speed (s)	Aperture	Closeness to Object
Dodgeball-Indoors	400	1/8	11	Regular
Dodgeball- Indoors with Calibration	400	1/8	11	Close
Dodgeball- Indoors with Calibration	400	1/8	11	Regular
Dodgeball- Indoors	400	1/8	11	Far
Dodgeball- Outdoors	1600	1/25	18	Close
Dodgeball- Outdoors with Calibration	1600	1/25	18	Close
Dumbbell- Indoors	400	1/8	11	Close
Dumbbell- Indoors with Calibration	400	1/8	11	Close
Shoe- Indoors	400	1/8	11	Far
Shoe- Indoors with Calibration	400	1/8	11	Far
Shoe- Indoors	400	1/8	11	Regular
Shoe- Indoors with Calibration	400	1/8	11	Regular
Volleyball- Indoors	400	1/8	11	Far
Volleyball- Indoors with Calibration	400	1/10	11	Far
Volleyball- Indoors	400	1/8	11	Close
Volleyball- Indoors with Calibration	400	1/8	11	Close
Volleyball- Indoors	400	1/10	11	Regular
Volleyball- Indoors with Calibration	400	1/10	11	Regular
Dodgeball-Indoors	400	1/8	11	Regular
Dodgeball- Indoors with Calibration	400	1/8	11	Close

Metashape

Once a photo set is obtained, all photos can be inputted into the Metashape user interface. A screenshot of the user interface accompanied by step-by-step instructions on how to achieve the final product can be found in Figure B-3. The software will use shared characteristics among the photo set to determine the focal point of interest. Because of this it is important for distinct geometry to be present in the photo sets, as an object such as the patella that does not have many defining characteristics may be interpreted as an indistinguishable object. Distinct lines and edges are meshed together to create the model, and depth of the image is interpreted from possessing images at all angles.

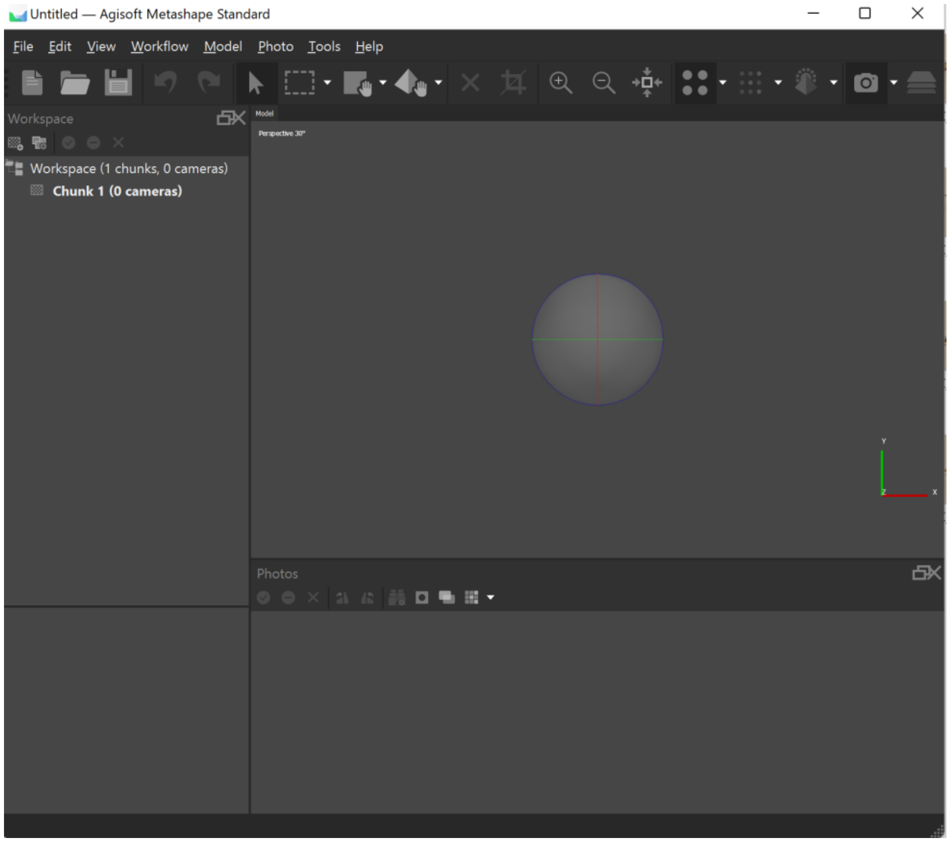


Figure B-3: Metashape user interface.

The reconstruction of the dodgeball created using the set of photos associated with the sample photos in Figure B-1 can be seen in Figure B-4. It is clear that the addition of the calibration frame improves the quality of the reconstruction as evidenced by the lack of "loose pieces" or the speckling around the outer borders of the dodgeball. This method was therefore applied in future applications. In Figure B-5, a schematic of the photo alignment shows how the software can deduce locations of where the photos are taken, resulting in a fully surrounded object and 3D reconstruction.

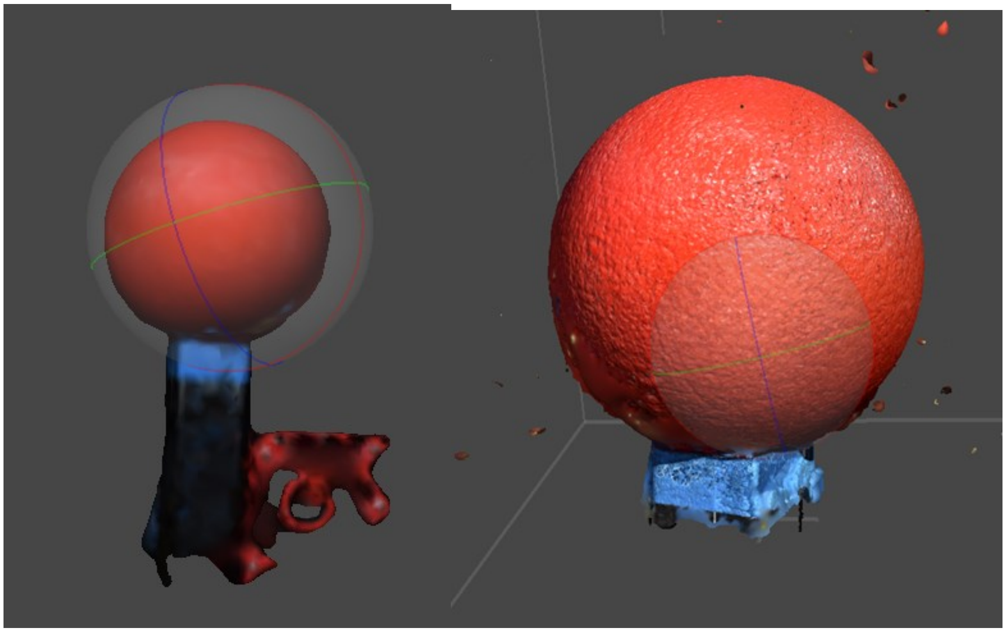


Figure B-4: Sample reconstructions of dodgeball with and without calibration frames.

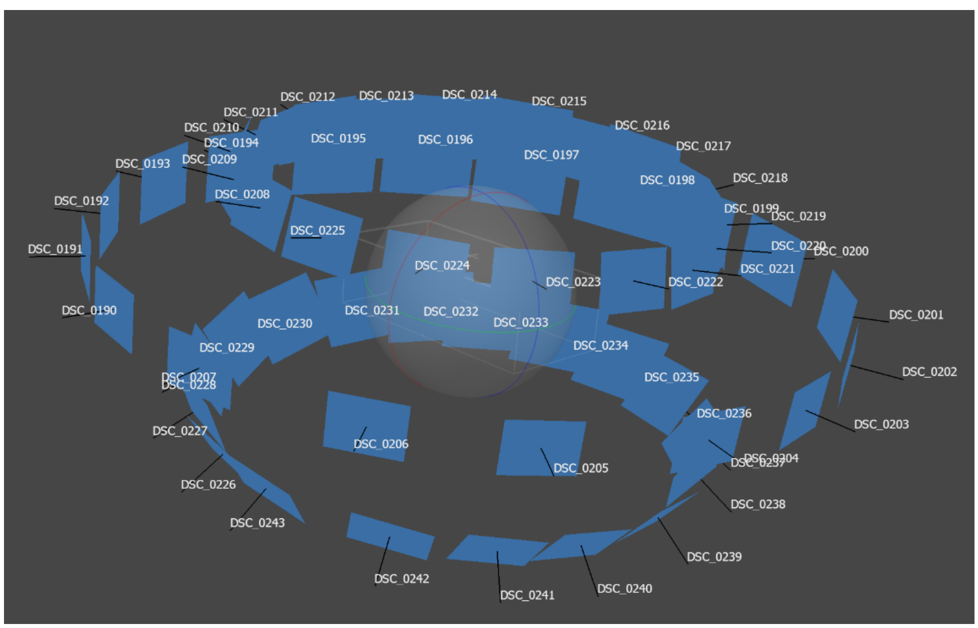


Figure B-5: Photo alignment in reconstruction.

Photogrammetry

Similar to the methods used for the dodgeball and variable objects, photos will be taken of the dissected bones to be used for reconstruction. However, another method in addition to the walk around technique will be tested. This is the turntable technique. The same principles used in the walk around method are considered, but the object is turned instead of the person walking around the object. A turntable would be turned every thirty degrees with a photo taken at each position. Like the previous method, a set of photos will be taken at three different heights to capture all aspects of the object for reconstruction. There are additional factors that must be considered when using this method. The background must be uniform, otherwise the software will see that the object is turning, but the background does not change. Because of this, the object must be displayed against a white wall with no lines or defining features, to fool the software into thinking the object is being walked around.

Reconstructions will be made using both photo taking techniques and validation can be done to determine the best method. To test the validity of the models, they can be compared to the measurements taken of the raw bones. Although the absolute size of the bone cannot be known by the software preliminarily, the relative sizing should be the same if the model has accurately portrayed the bones. Additionally, models can be compared to normative data sets within the age of the cadaver to see if they are within average ranges of relative measurements.

Appendix C Rationale for Digitization Location

Femur

The femoral head is spherical and can be recognized by Matlab as a spherical object. The code was programmed to locate any spherical object within maximum and minimum coordinate constraints. The values of the constraints in all directions were entered and Matlab was able to discern the central location of the femoral head, which was the origin of the femoral coordinate system.

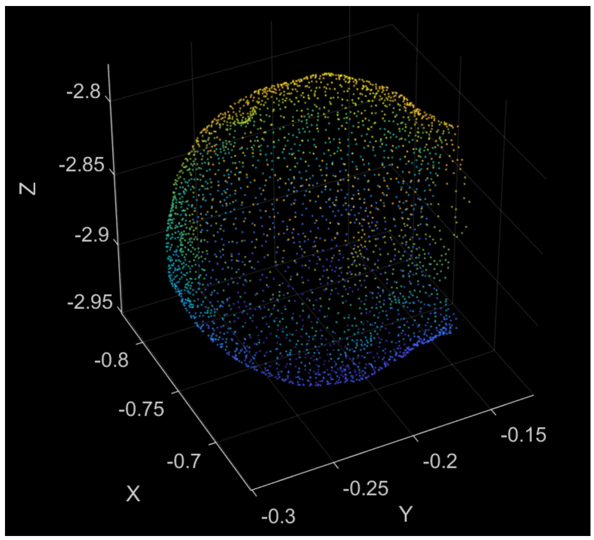


Figure C-1: Cloudpoint of Femoral Head.

The medial and lateral condyles were determined as the most medial and lateral aspects of the condyles at the distal end of the femur. They were also digitized near the center of the medial/lateral surface of the condyles.

Figure C-2: Digitization Locations of Medial Femoral Epicondyle (Left) and Lateral Femoral Epicondyle (Right).

The greater trochanter was digitized at the most lateral aspect of the landmark, to correspond with the lateral location of the motion capture marker in the same place.

Figure C-3: Digitization Location of Femoral Greater Trochanter.

The proximal attachment location of the vastus medialis longus and vastus lateralis were determined as the most superior aspect of the linea aspera, and on their corresponding sides of the ridge. Since the line aspera branches in a Y-shape, the locations were digitized at the points of each of the branch points. The vastus medialis oblique proximal attachment was digitized at the distal end of the line aspera on the medial side of the distal branching of the ridge. The proximal attachment location of the vastus intermedius was digitized at the apex created by the intertrochanteric line and the anterior ridge of the greater trochanter. The distal attachment locations of the vastus medialis longus, vastus lateralis, and vastus intermedius were digitized above the patellar surface after the curvature caused by the condyles evens out, and the femoral width returns to being consistent with the rest of the shaft. This is assumed to be the point where the three vasti muscles coalesce into the quadriceps tendon, which will attach upon the superior patella. It is not assumed that this is where the muscles attach upon the bone. The vastus medialis oblique distal attachment point was digitized on the medial side of the femur at the point where the femur begins to widen. This is considered to be the point of the transition into the vastus medialis oblique tendon that attaches on the medial side of the patella.

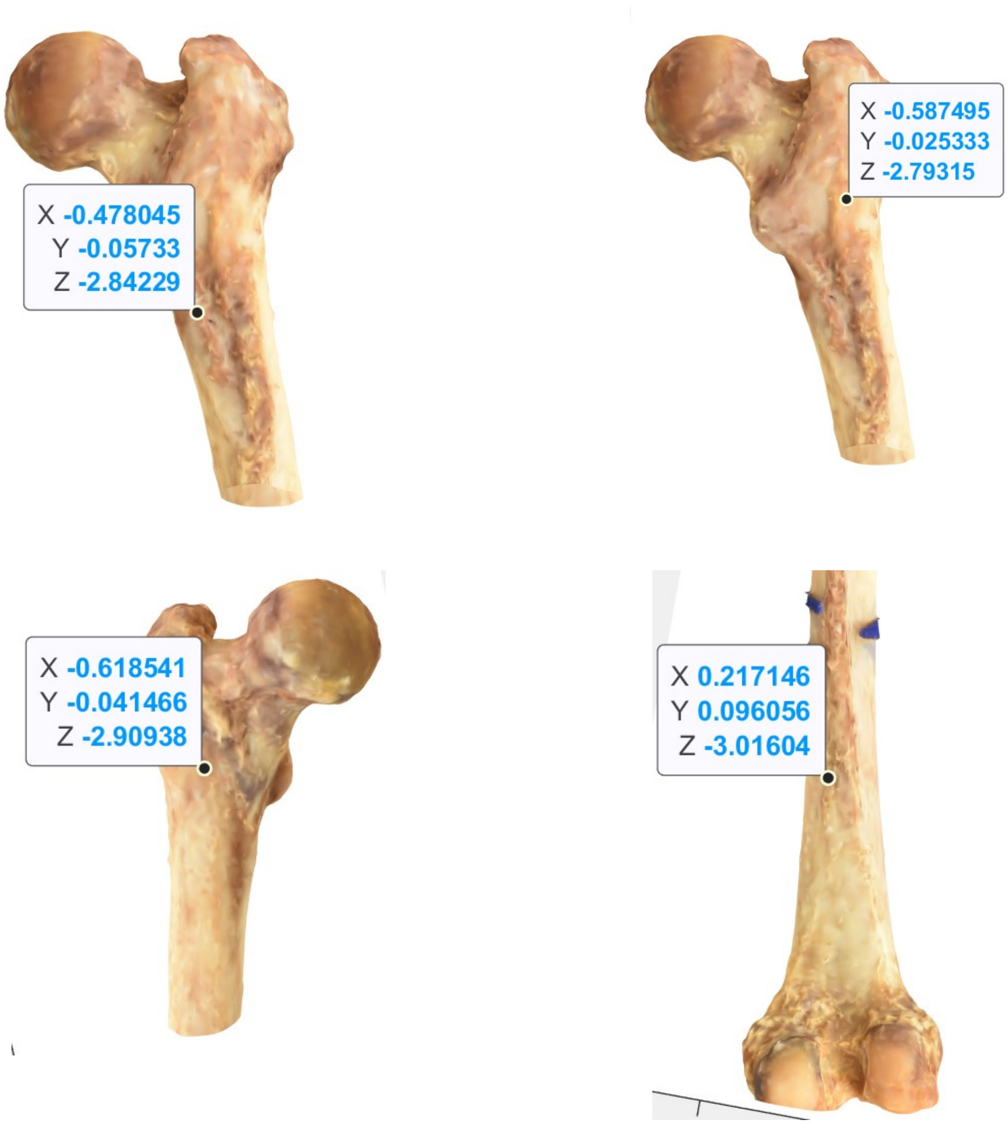


Figure C-4: Digitization locations of Vastus Medialis Longus (top-left), Vastus Lateralis (top-right), Vastus Intermedius (bottom-left), and Vastus Medialis Oblique (bottom-right) proximal attachment locations.

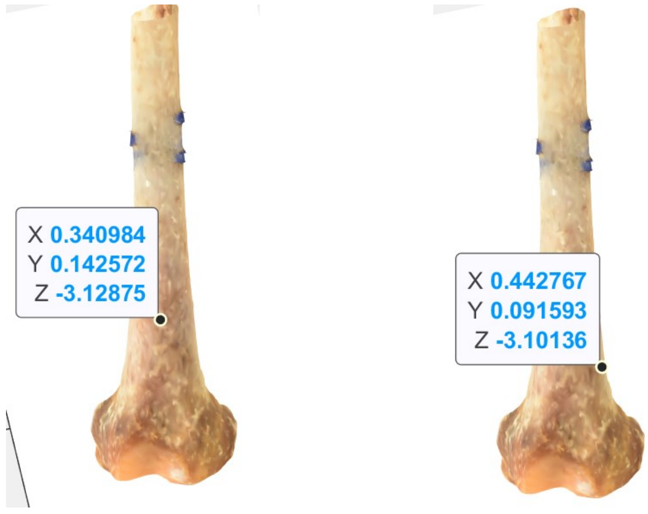


Figure C-5: Digitization locations of distal attachments for Vastus Lateralis, Vastus Intermedius and Vastus Medialis Longus (left), and Vastus Medialis Oblique (right).

Patella

The apex, base, medial facet ridge, lateral facet ridge, and vastus medialis oblique attachment point was digitized on the patella. The base of the patella is on the superior side and is the point of attachment for the quadriceps tendon. The apex was chosen to be the most inferior aspect of the patella, and is the attachment point of the patellar ligament. The medial and lateral facet ridges were chosen together by digitizing them with the combined criteria of being most lateral or medial, whilst being horizontally aligned with one another. Finally, the vastus medialis oblique attachment point was chosen at the corner of the superior-medial edge. This is the attachment point of the vastus medialis oblique tendon. The purpose of including the medial and lateral facet ridges was for the ability to calculate the central location of the patella. This could be done by averaging the positions of the apex, base, and ridges. This is considered the origin of the patellar coordinate system.

Figure C-6: Digitization locations of patella. Medial facet ridge (top left), lateral facet ridge (top right), apex (center left), base (center right), and Vastus Medialis Oblique (bottom center) attachment location.

Leg

The lateral and medial tibial condyles were the most lateral and medial aspects of the condylar surface respectively. The tibial tuberosity was chosen where the bony prominence was most anterior. The medial and lateral malleoli were chosen at their most inferior-medial and inferior-lateral aspects respectively. The middle point between the lateral and medial condyles was considered the origin of the leg coordinate system.

Figure C-7: Digitization locations of lateral tibial condyle (left), medial tibial condyle (center), and tibial tuberosity (right)

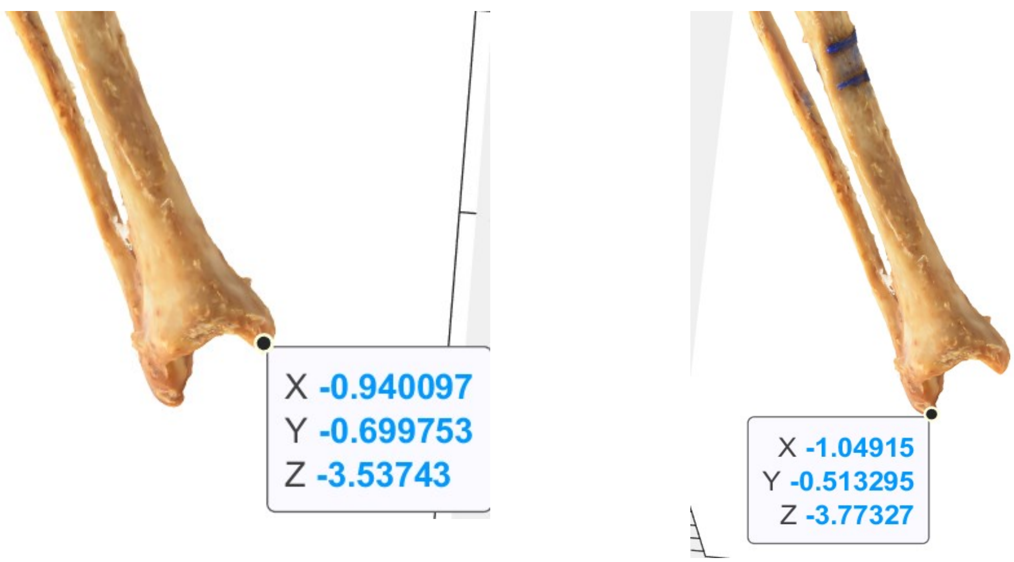


Figure C-8: Digitization locations of medial (left) and lateral (right) malleoli.

Foot

The posterior calcaneus was chosen at the most posterior aspect of the calcaneus bone where the Achilles tendon would ideally attach. The navicular tuberosity was digitized at the most medial aspect of the bony prominence, as was the first metatarsal head. The fifth metatarsal head was digitized at the most lateral aspect of the bony prominence. The point between the malleoli where the talus would connect was considered the origin of the foot coordinate system.

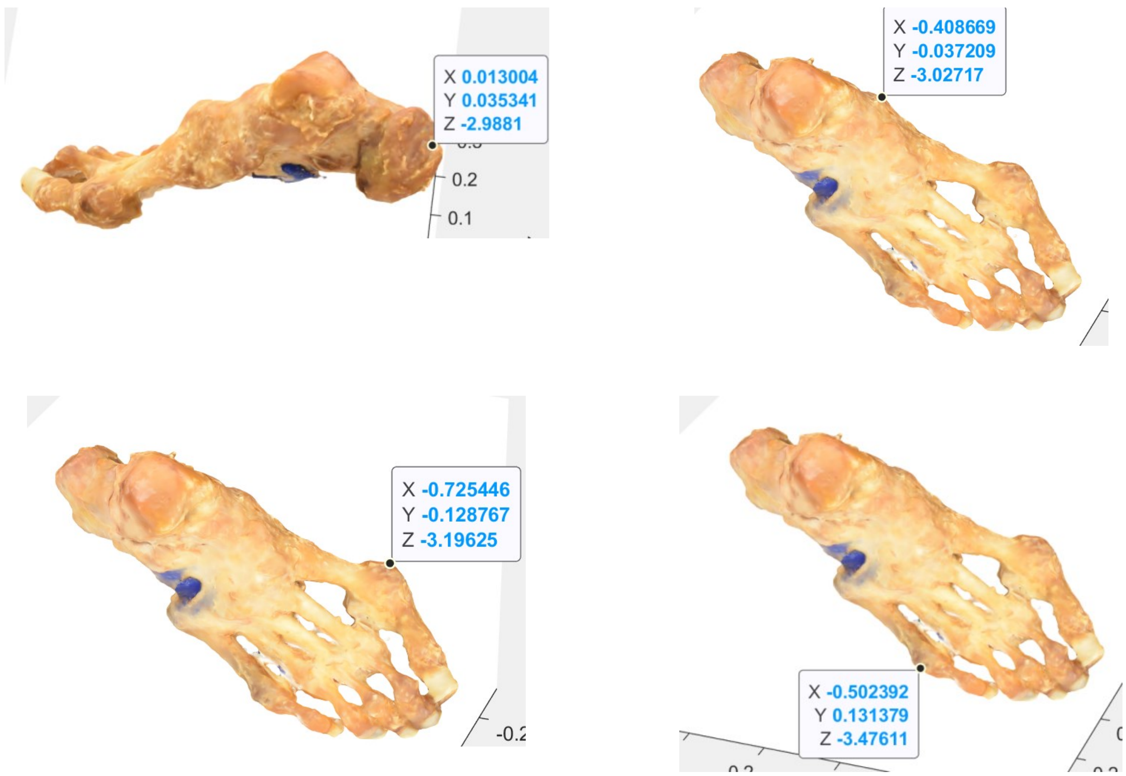


Figure C-9: Digitization locations of calcaneus (top left), navicular tuberosity (top right), first metatarsal head (bottom left) and fifth metatarsal head (bottom right).

Appendix E Marker Placement Locations

Table E-1: Static Marker Placement Locations- Anatomical

Location of Marker Placements
Lateral Malleolus
Medial Malleolus
Head of Fifth Metatarsal
Head of First Metatarsal
Lateral Tibial Condyle
Medial Tibial Condyle
Tibial Tuberosity
Patella- Base
Patella- Medial
Patella- Lateral
Patella- Apex
Medial Femoral Condyle
Lateral Femoral Condyle
Greater Trochanter
Anterior Superior Iliac Spine
Iliac Crest
Posterior Superior Iliac Spine
Lumbar Vertebrae V/ S1
Thigh Cluster
Leg Cluster
Foot Cluster

Table E-2: Dynamic Marker Placement Locations- Anatomical

Location of Marker Placements
Foot cluster
Leg cluster
Thigh cluster
Iliac crest
L5/S1 markers
Patella- Base
Patella- Medial
Patella- Lateral

Appendix F Participant Graphs

Participant 1

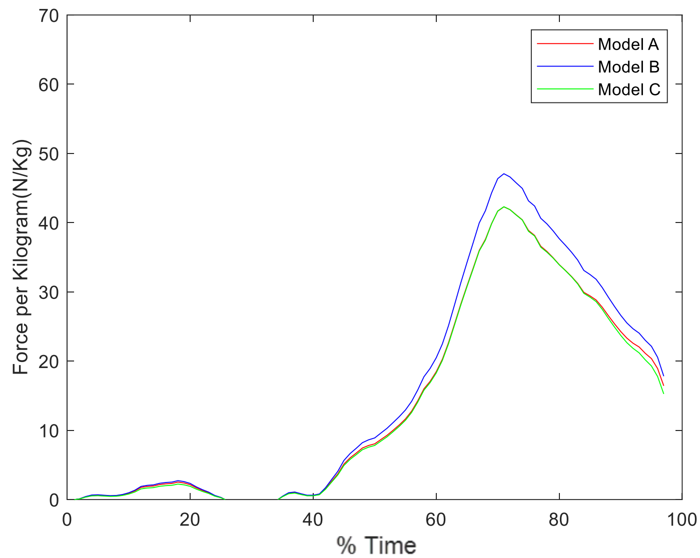


Figure F-1: P1 normalized patellar ligament force upon leg resultant forces.

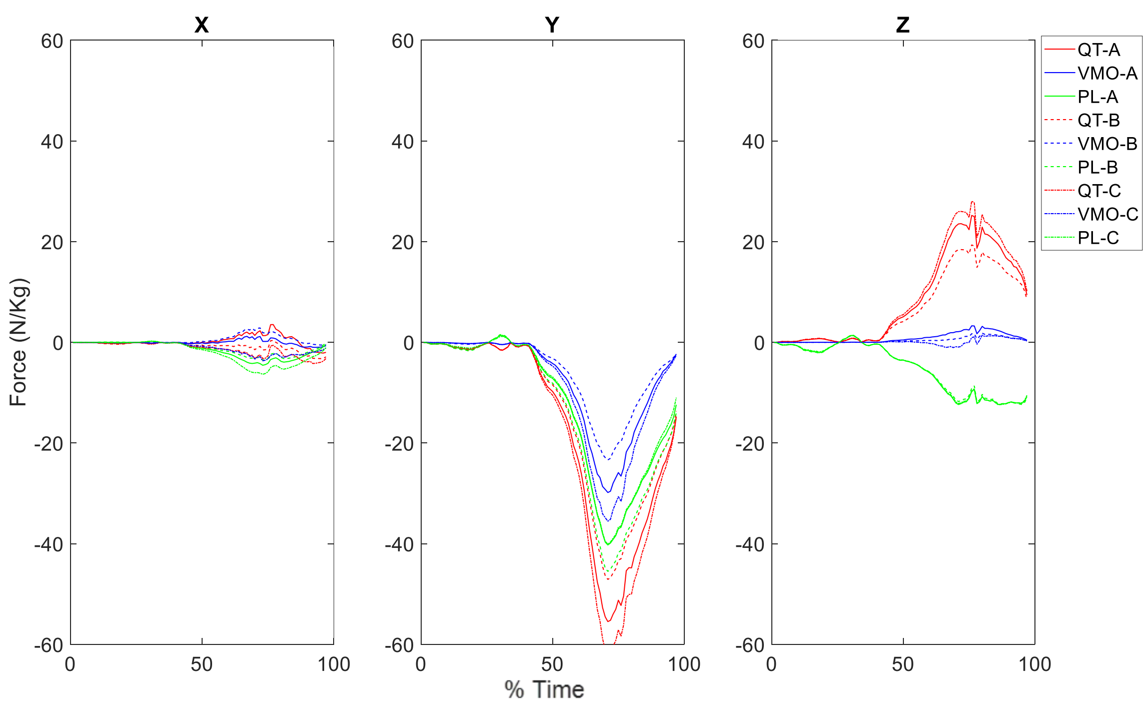


Figure F-2: P1 normalized forces of PL, QT, and VMO upon patella.

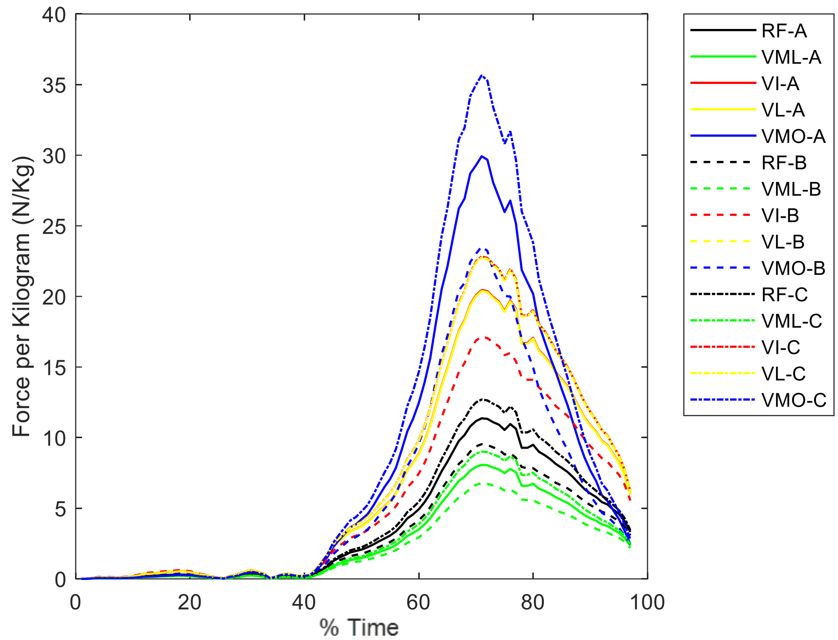


Figure F-3: P1 individual normalized quadricep forces upon patella.

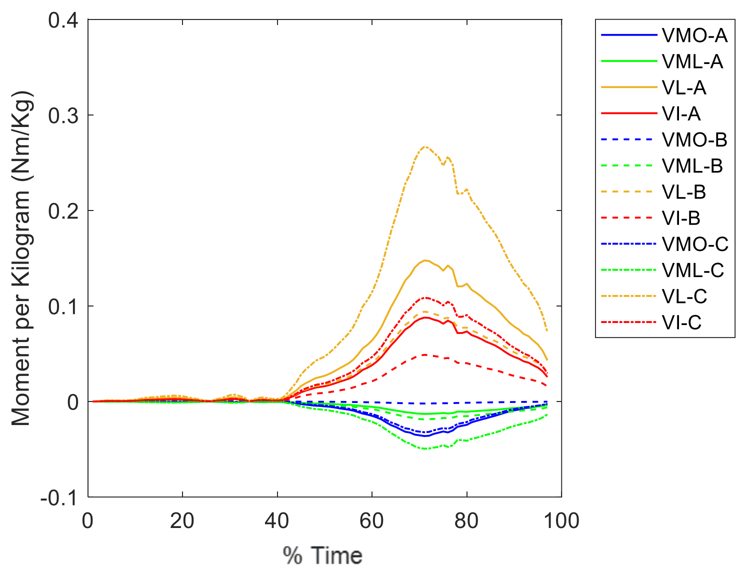


Figure F-4: P1 normalized moments about femoral longitudinal axis .

Participant 2

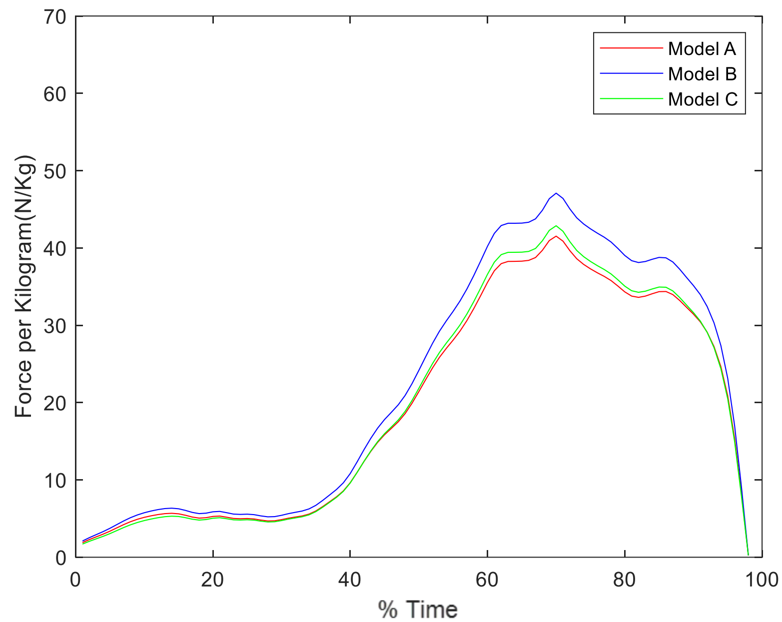


Figure F-5: P2 normalized patellar ligament force upon leg resultant forces.

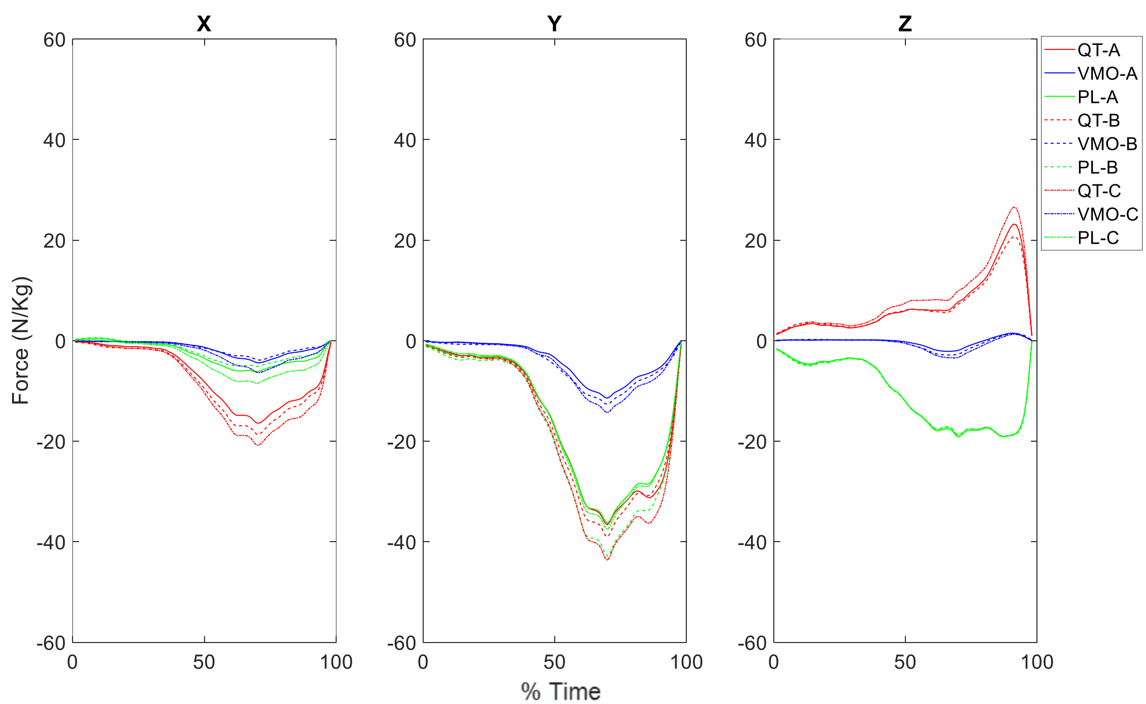


Figure F-6: P2 normalized forces of PL, QT, and VMO upon patella.

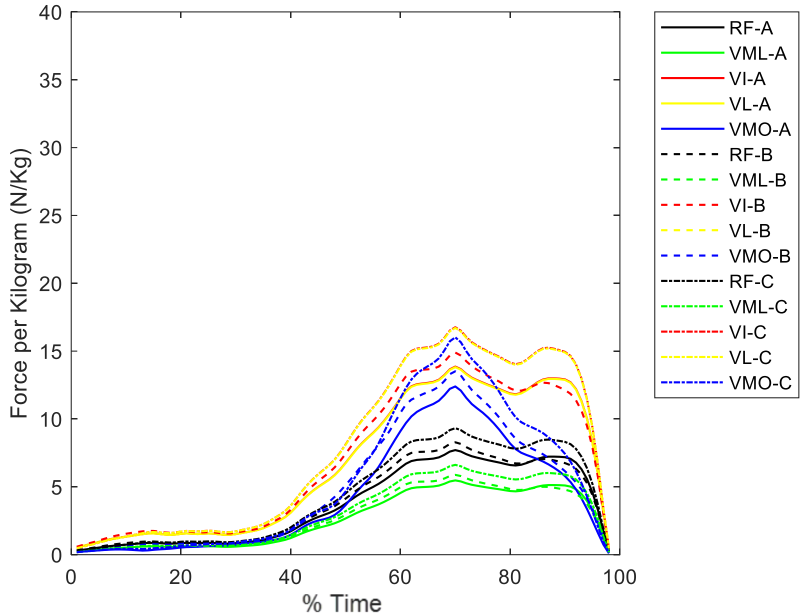


Figure F-7: P2 individual normalized quadricep forces upon patella.

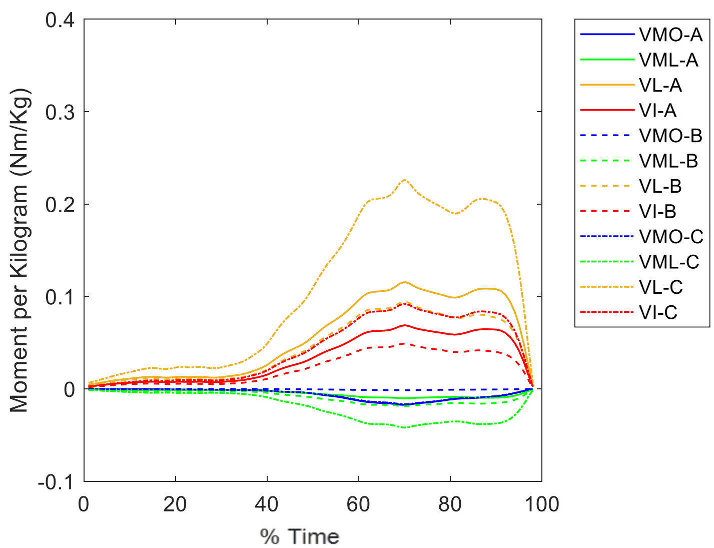


Figure F-8: P2 normalized moments about femoral longitudinal axis .

Participant 3

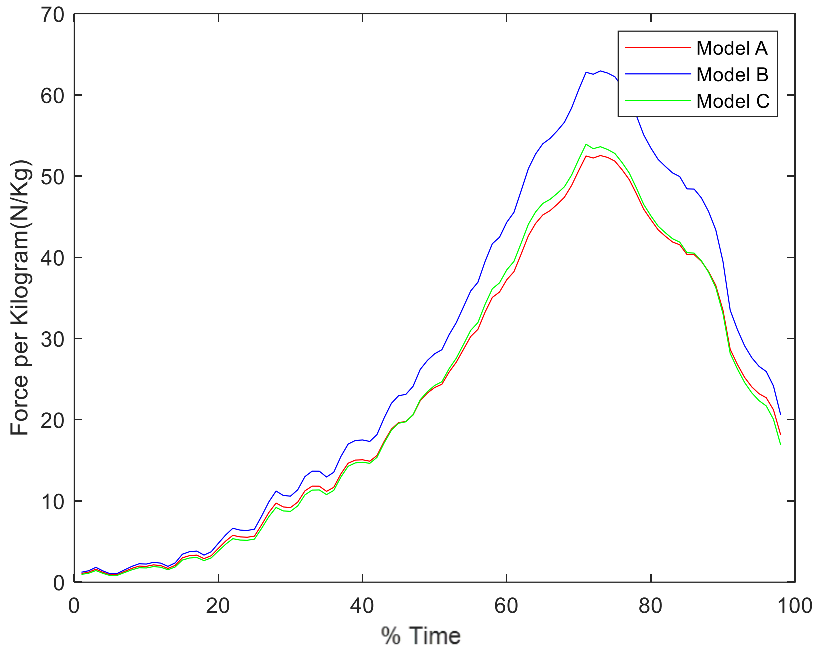


Figure F-9: P3 normalized patellar ligament force upon leg resultant forces.

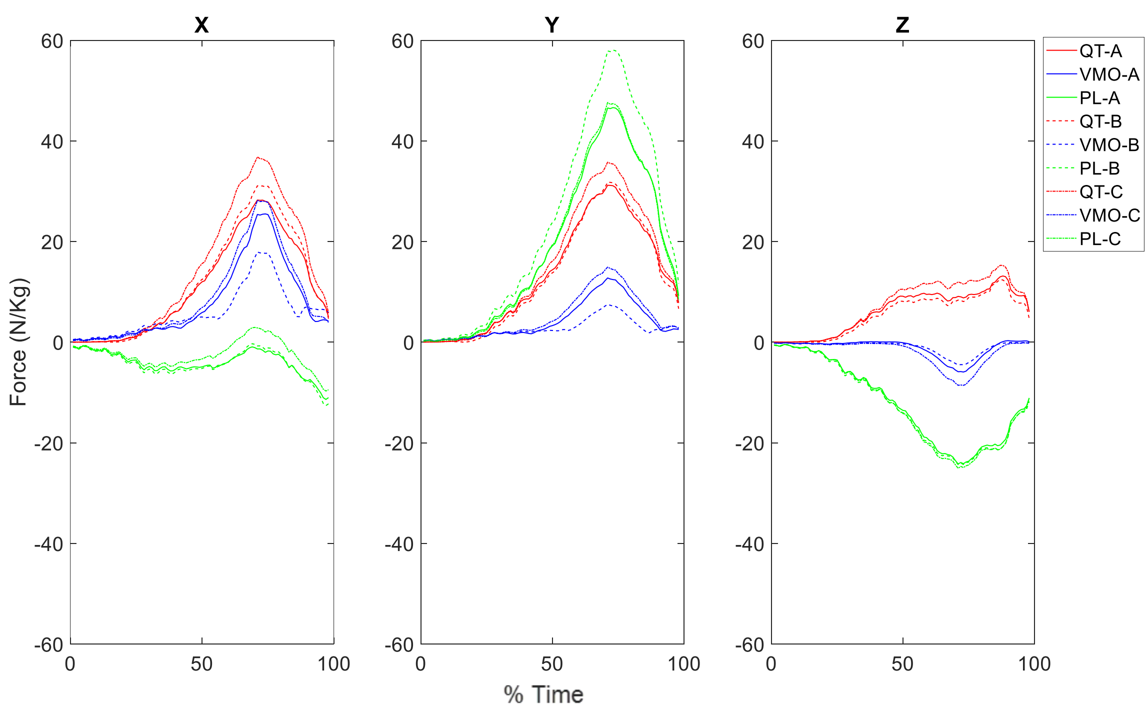


Figure F-10: P3 normalized forces of PL, QT, and VMO upon patella.

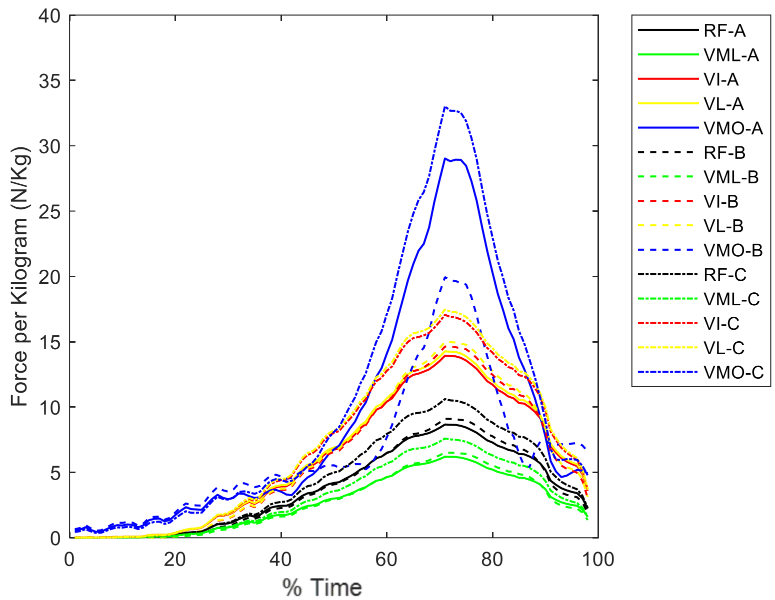


Figure F-11: P3 individual normalized quadricep forces upon patella.

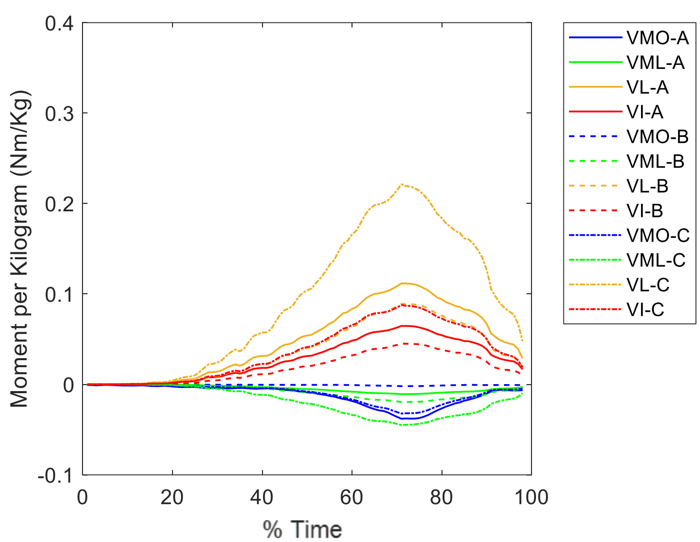


Figure F-12: P3 normalized moments about femoral longitudinal axis .

Participant 4

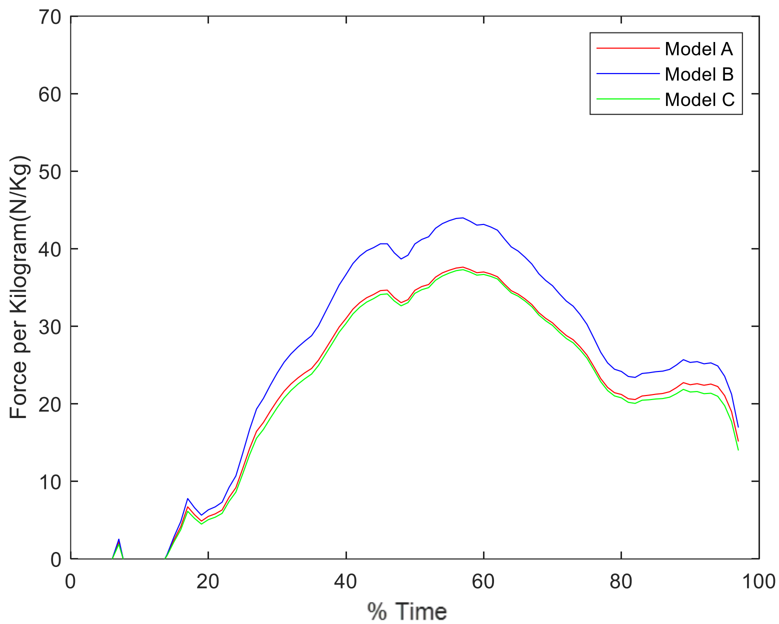


Figure F-13: P4 normalized patellar ligament force upon leg resultant forces.

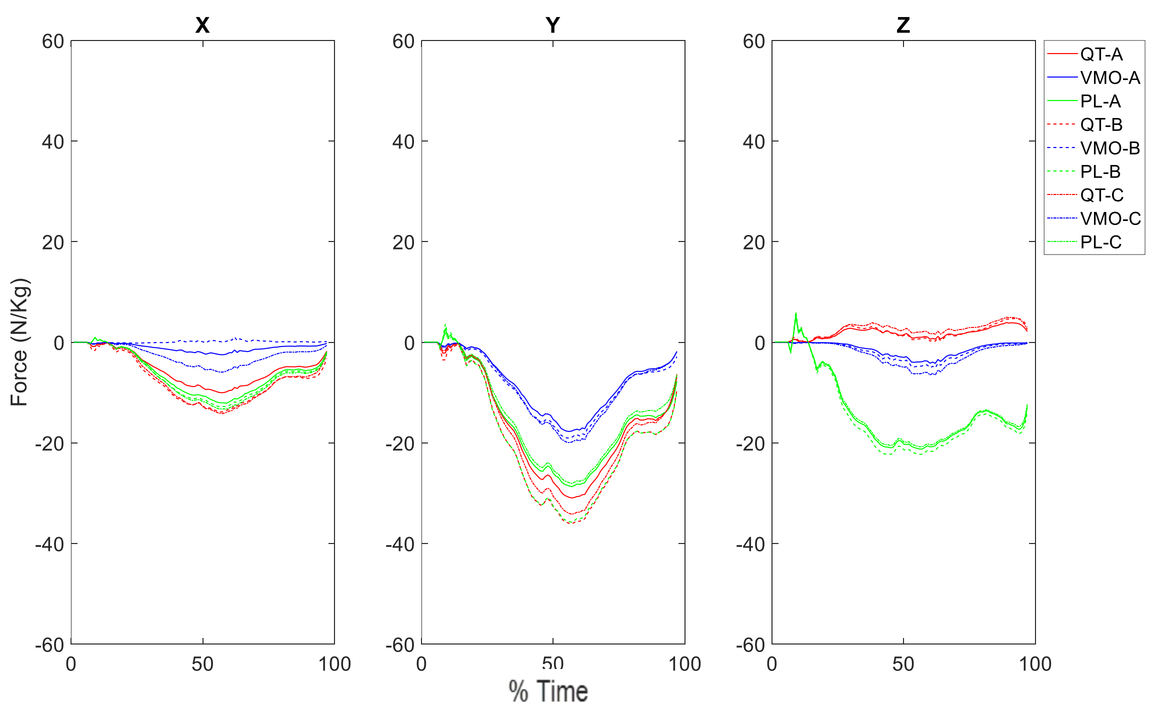


Figure F-14: P4 normalized forces of PL, QT, and VMO upon patella.

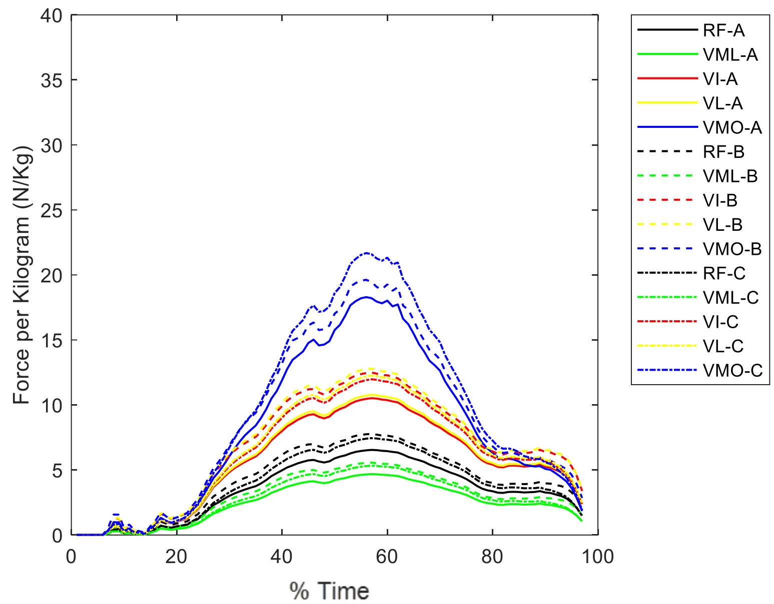


Figure F-15: P4 individual normalized quadricep forces upon patella.

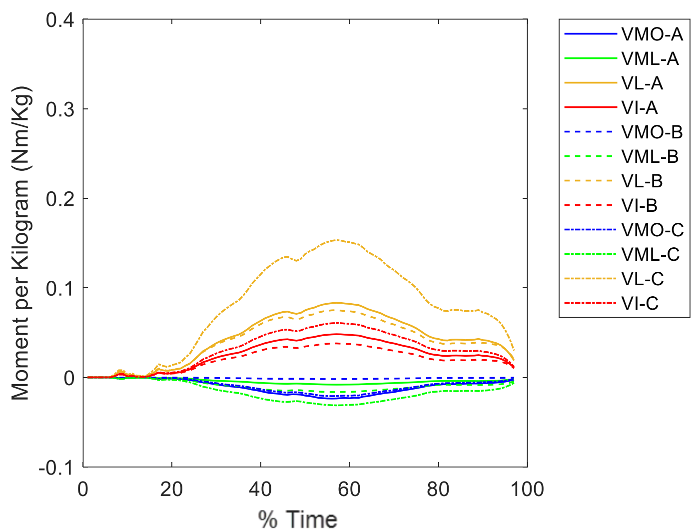


Figure F-16: P4 normalized moments about femoral longitudinal axis .

Participant 5

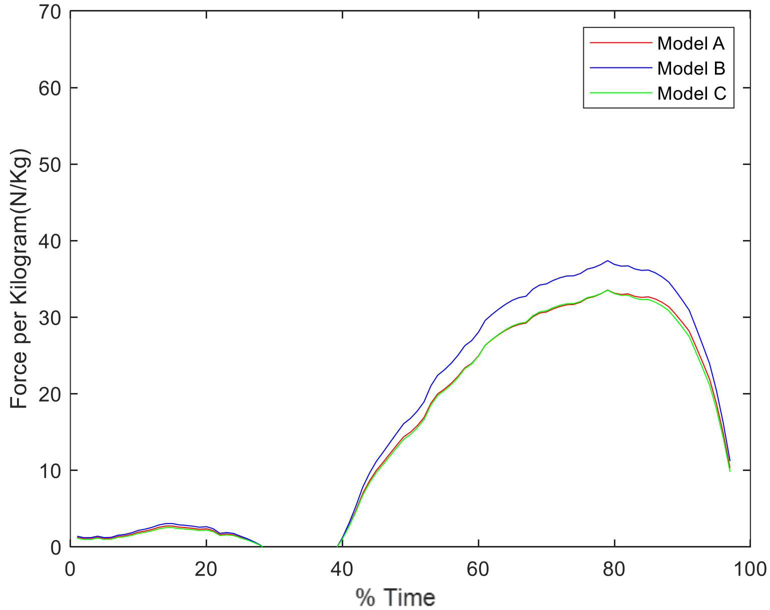


Figure F-17: P5 normalized patellar ligament force upon leg resultant forces.

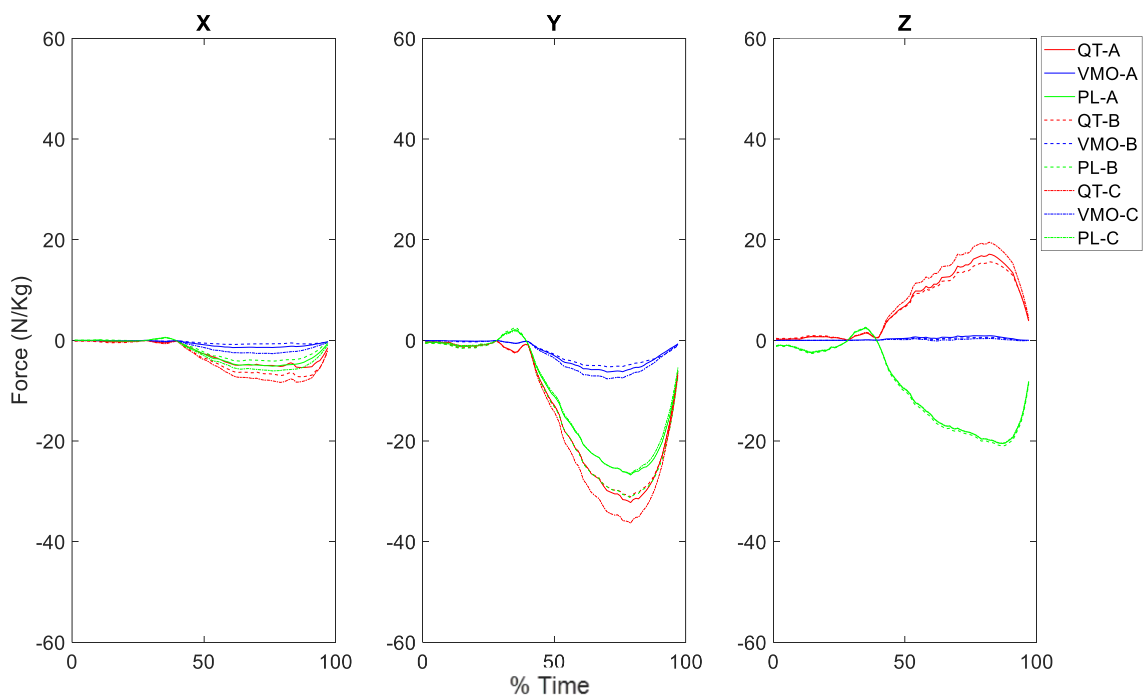


Figure F-18: P5 normalized forces of PL, QT, and VMO upon patella.

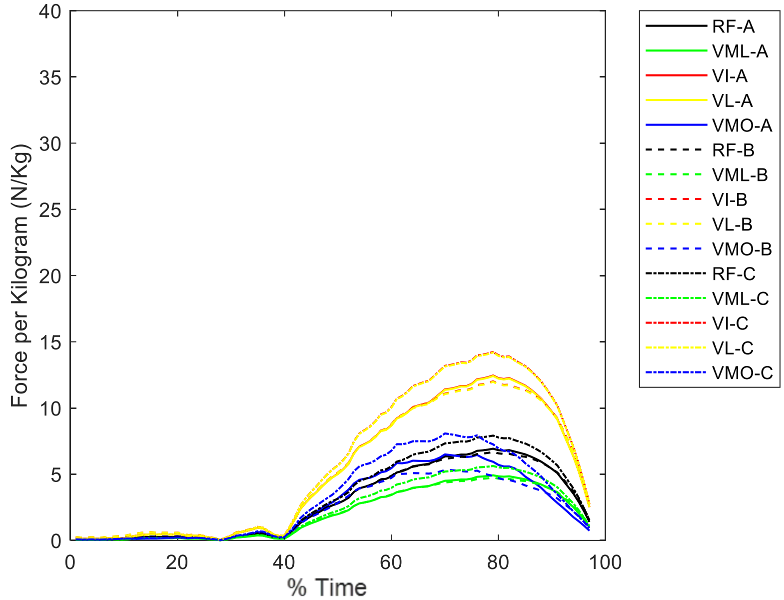


Figure F-19: P5 individual normalized quadricep forces upon patella.

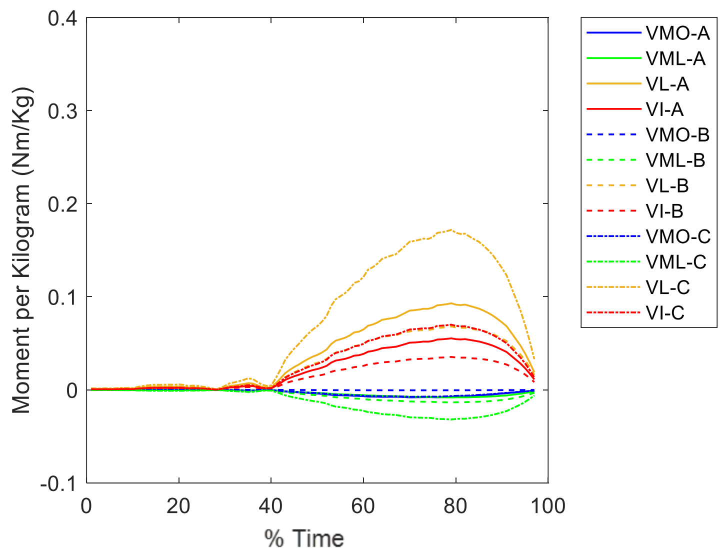


Figure F-20: P5 normalized moments about femoral longitudinal axis .

Participant 6

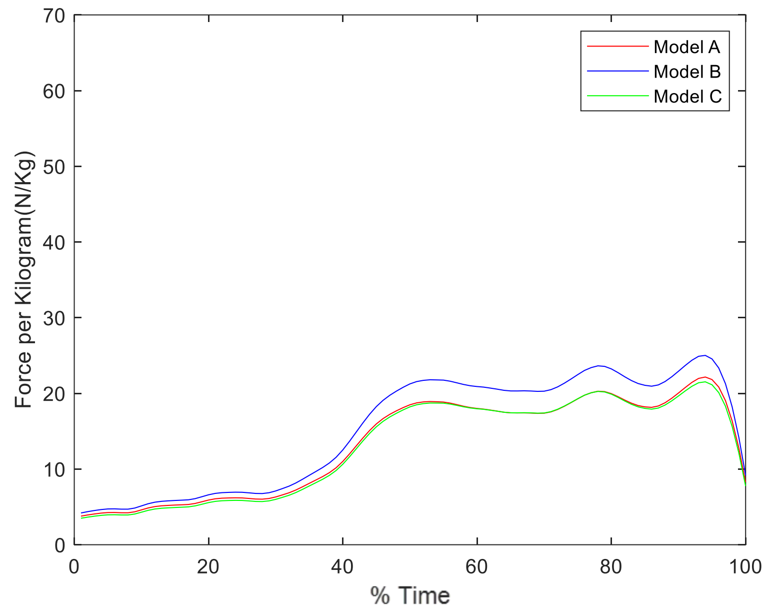


Figure F-21: P6 normalized patellar ligament force upon leg resultant forces.

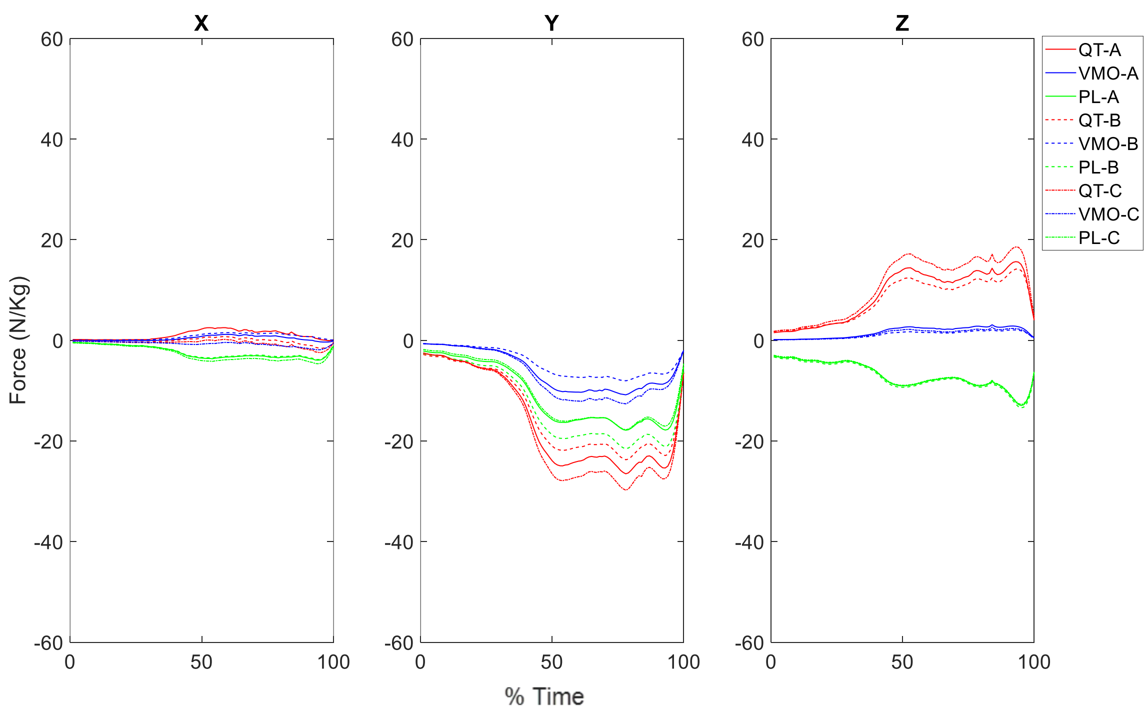


Figure F-22: P6 normalized forces of PL, QT, and VMO upon patella.

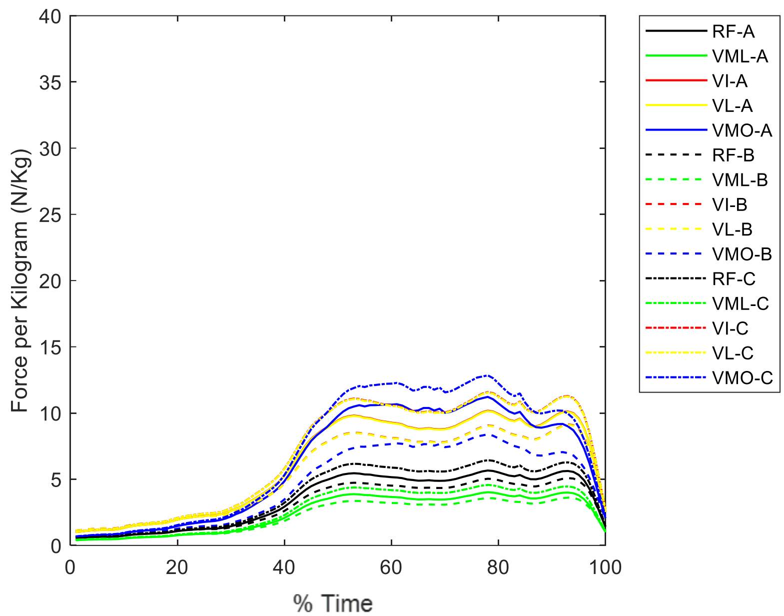


Figure F-23: P6 individual normalized quadricep forces upon patella.

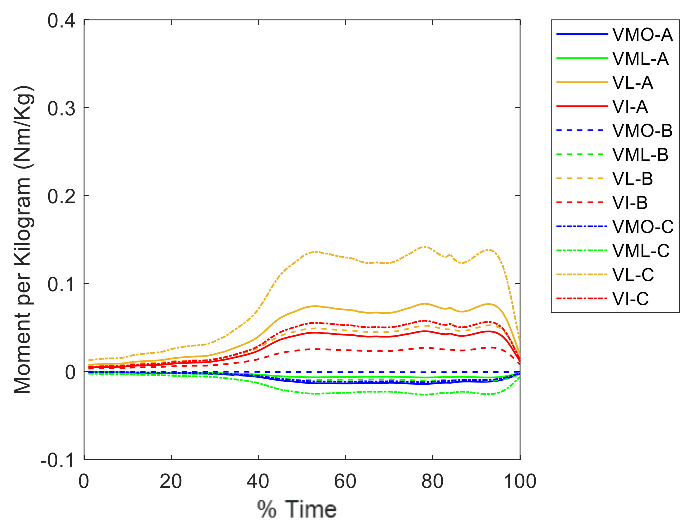


Figure F-24: P6 normalized moments about femoral longitudinal axis .

Participant 7

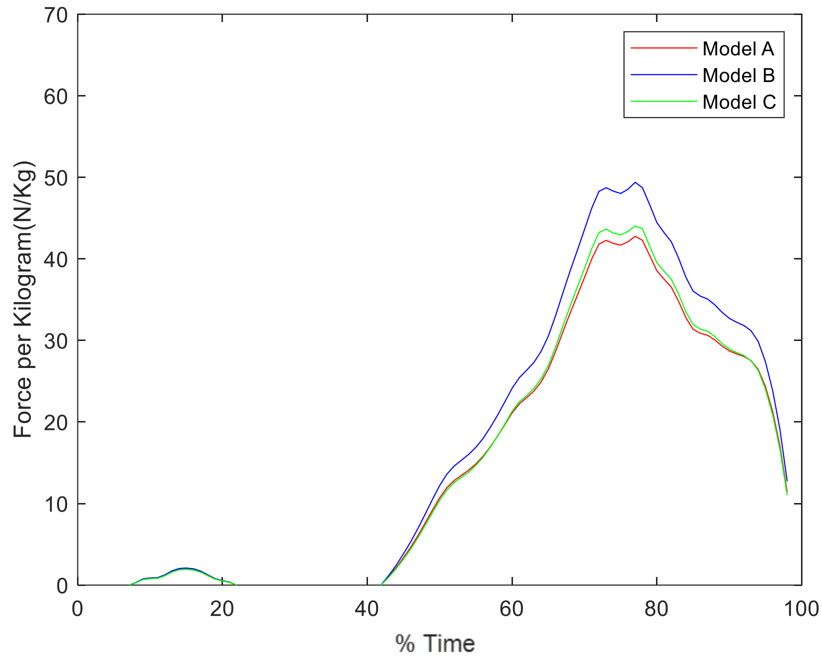


Figure F-25: P7 normalized patellar ligament force upon leg resultant forces.

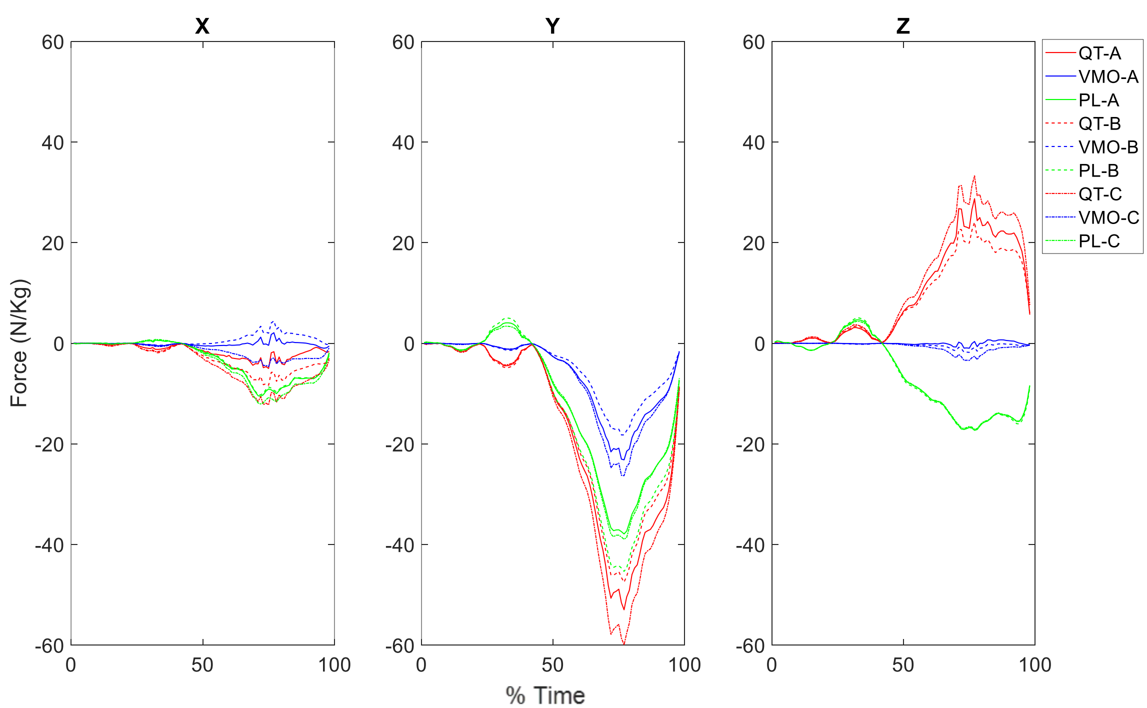


Figure F-26: P7 normalized forces of PL, QT, and VMO upon patella.

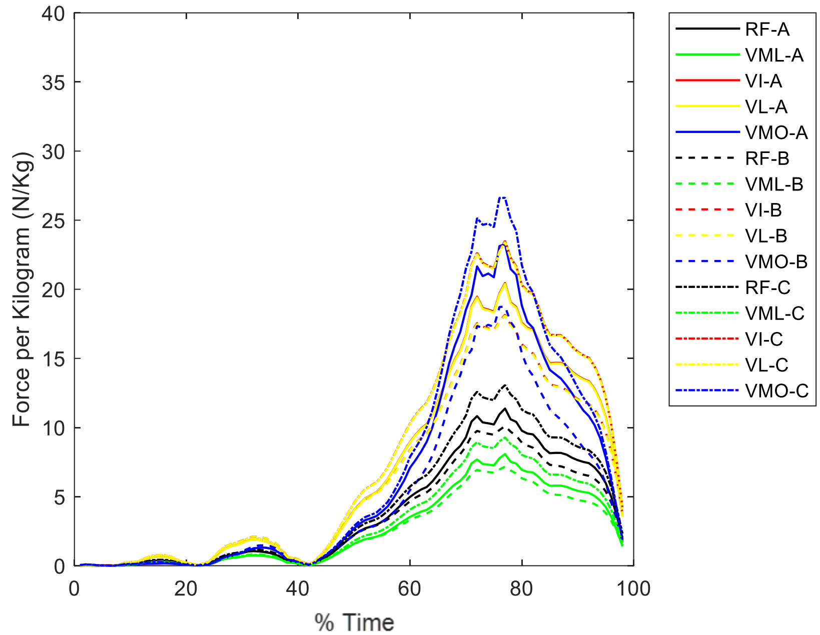


Figure F-27: P7 individual normalized quadricep forces upon patella.

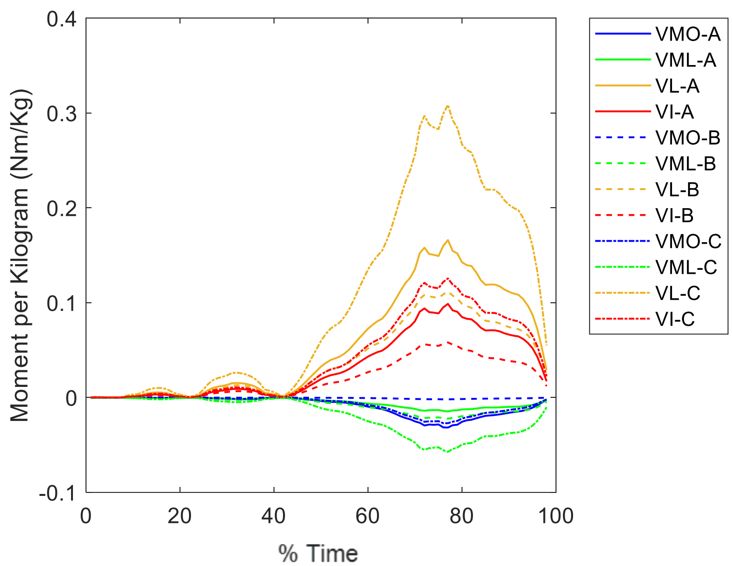


Figure F-28: P7 normalized moments about femoral longitudinal axis .

Participant 8

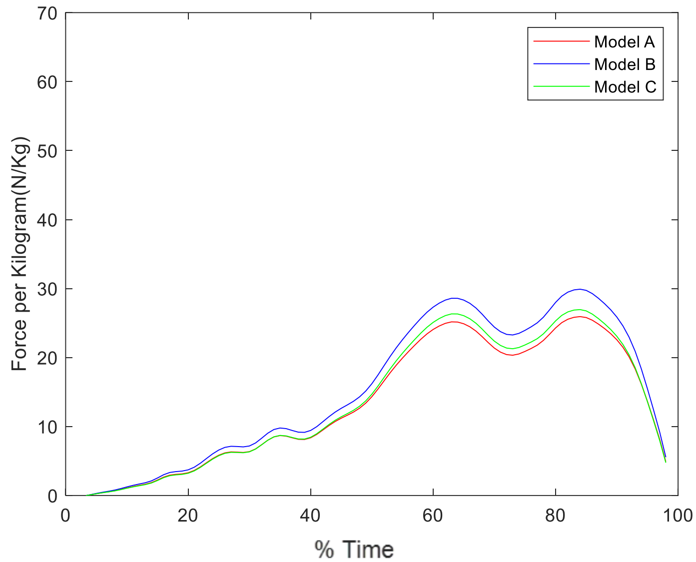


Figure F-29: P8 normalized patellar ligament force upon leg resultant forces.

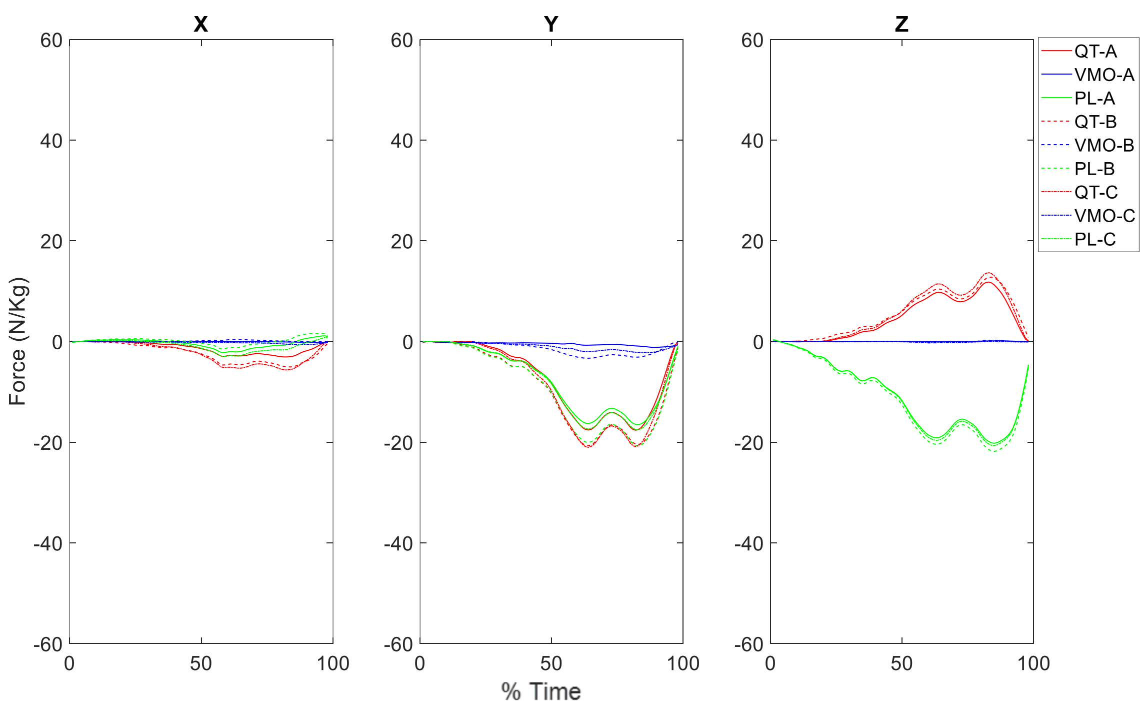


Figure F-30: P8 normalized forces of PL, QT, and VMO upon patella.

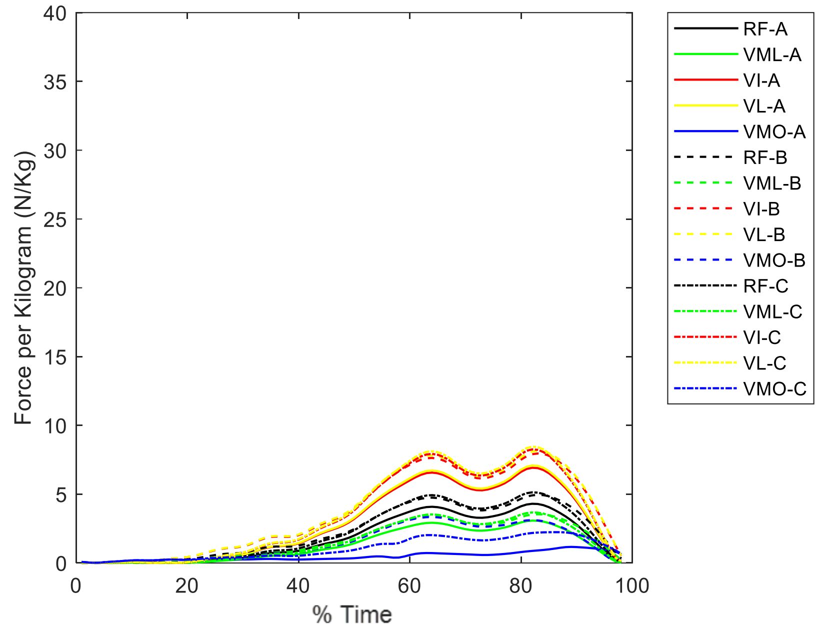


Figure F-31: P8 individual normalized quadricep forces upon patella.

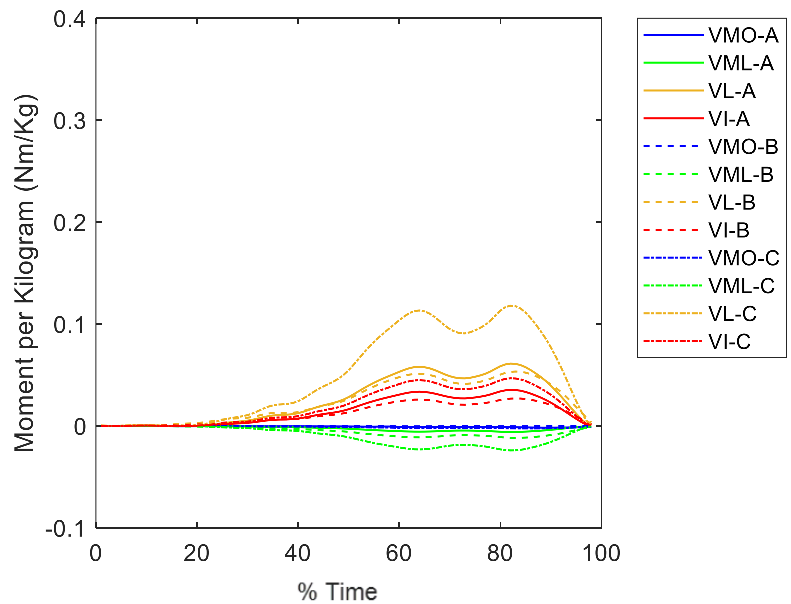


Figure F-32: P8 normalized moments about femoral longitudinal axis .

Participant 9

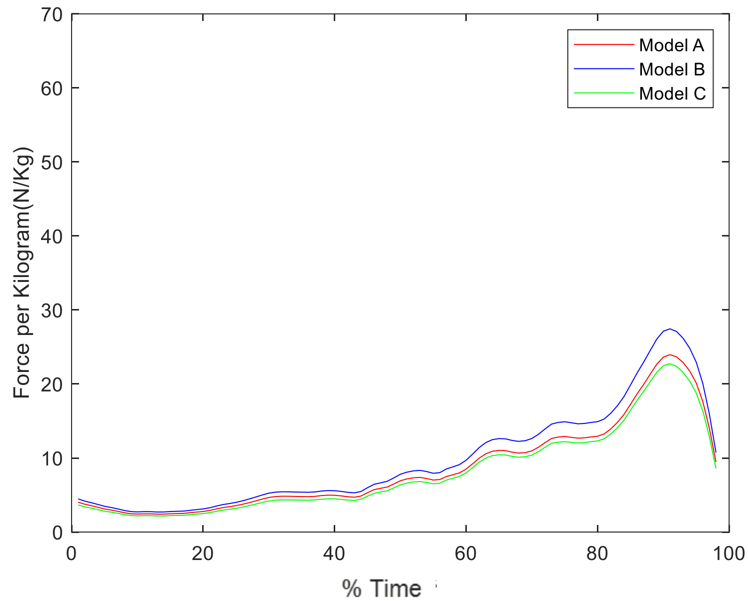


Figure F-33: P9 normalized patellar ligament force upon leg resultant forces.

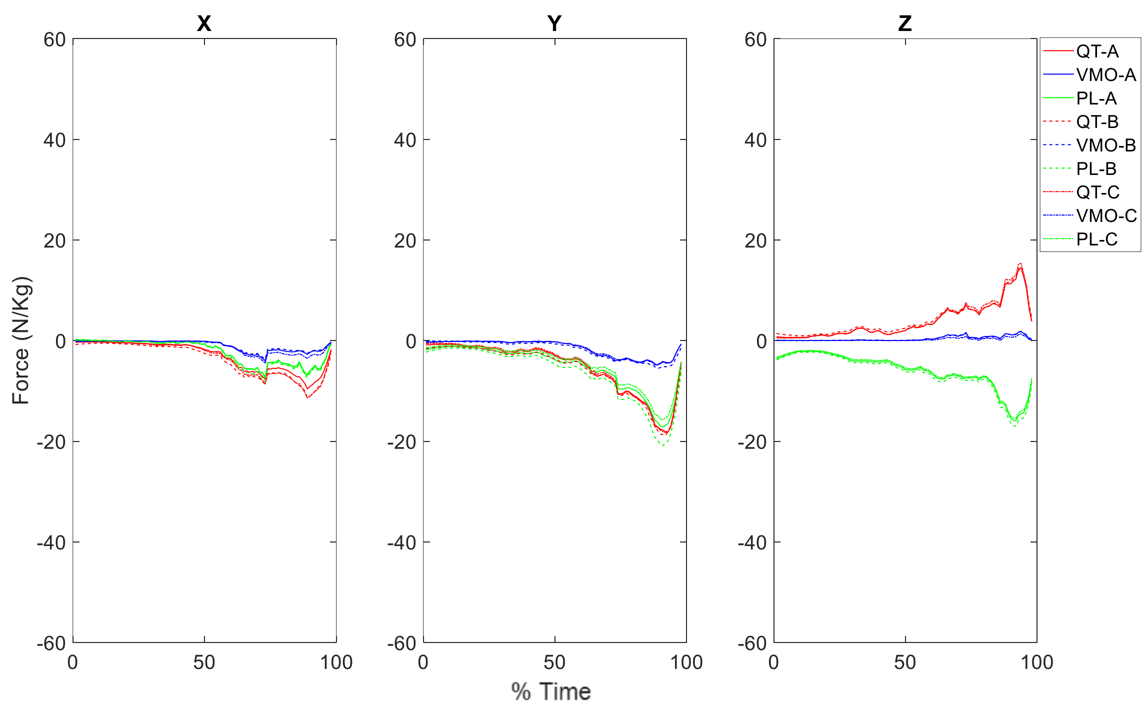


Figure F-34: P9 normalized forces of PL, QT, and VMO upon patella.

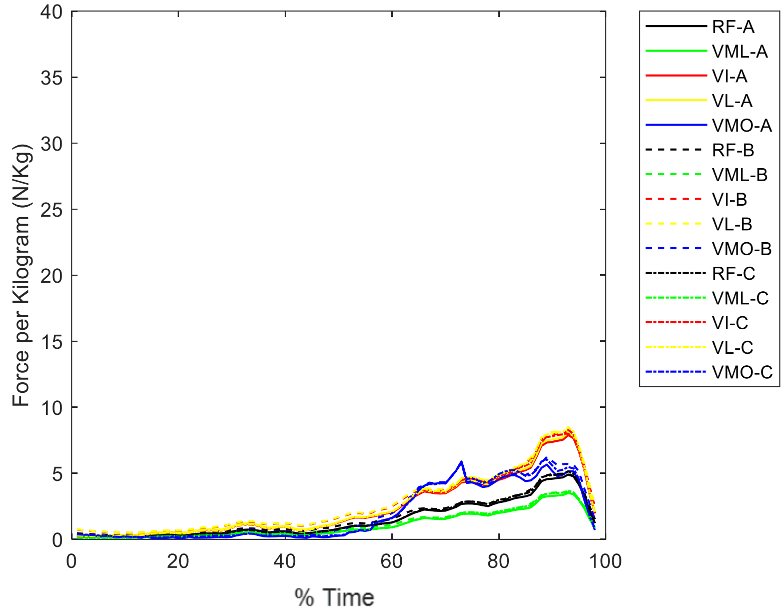


Figure F-35: P9 individual normalized quadricep forces upon patella.

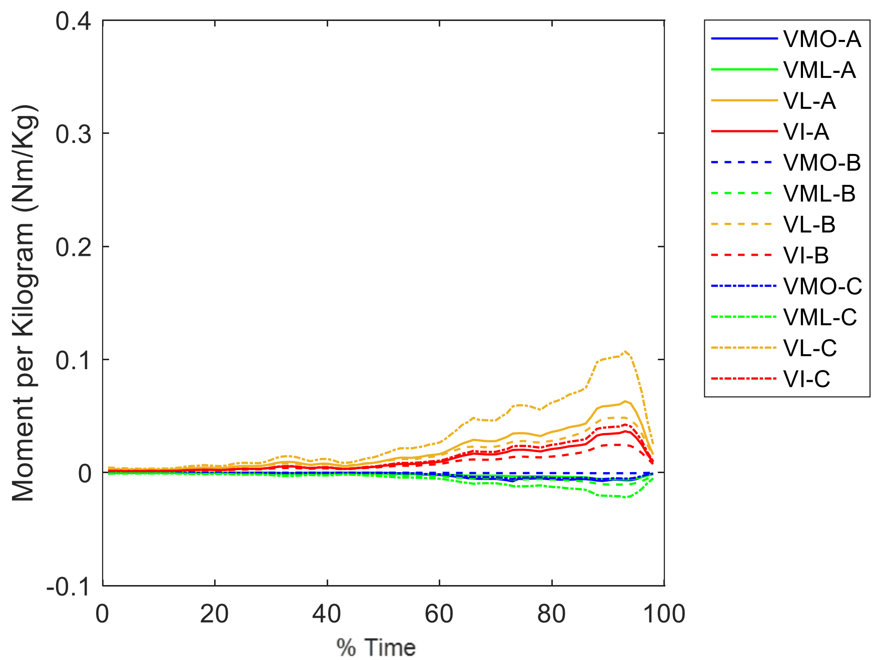


Figure F-36: P9 normalized moments about femoral longitudinal axis .

Participant 10

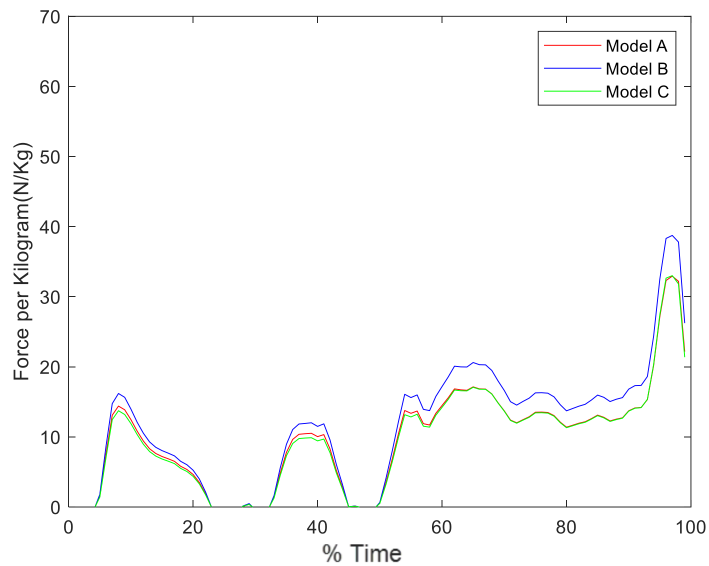


Figure F-37: P10 normalized patellar ligament force upon leg resultant forces.

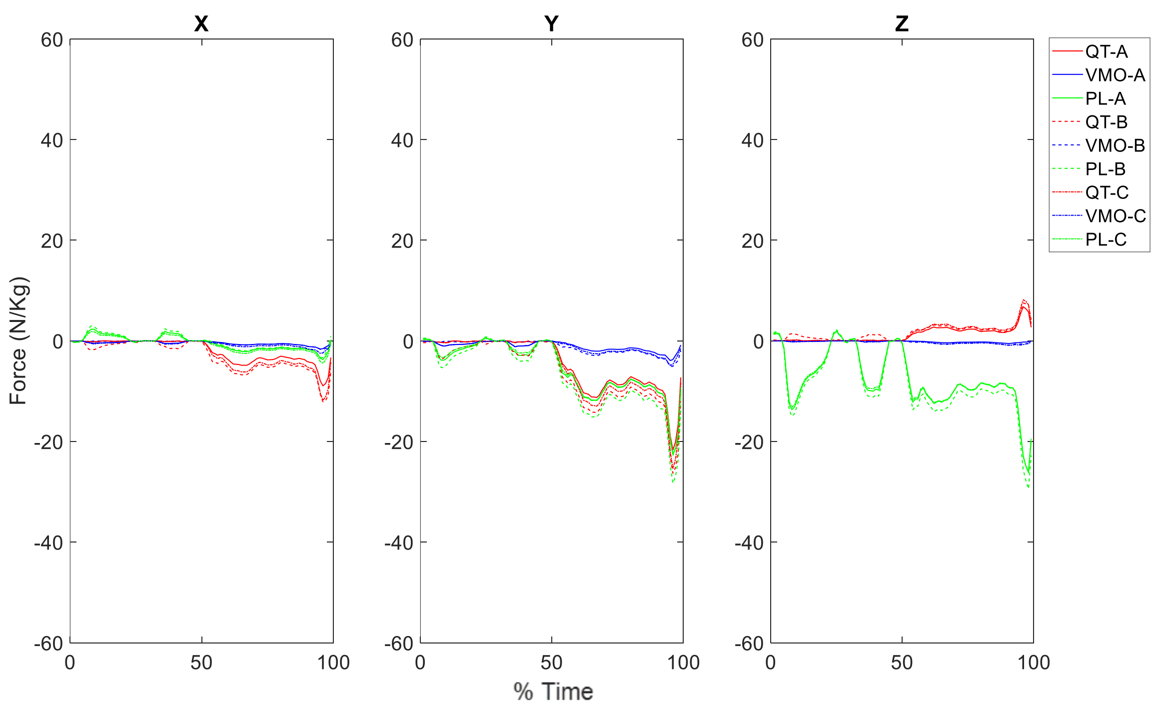


Figure F-38: P10 normalized forces of PL, QT, and VMO upon patella.

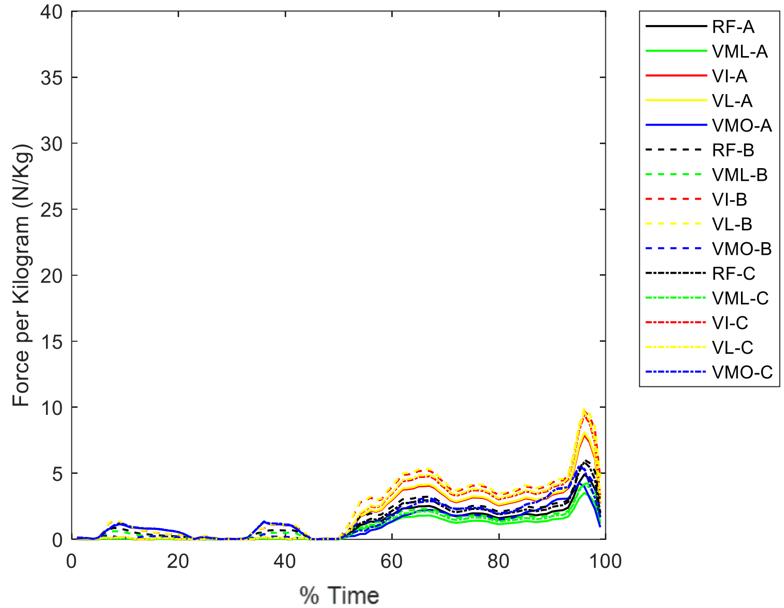


Figure F-39: P10 individual normalized quadricep forces upon patella.

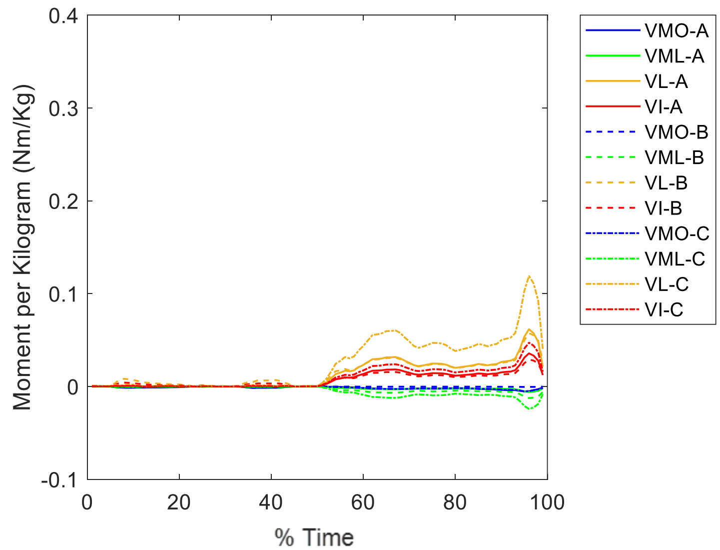


Figure F-40: P10 normalized moments about femoral longitudinal axis .

Appendix G Leg Extension Code

The code is used to extract and amalgamate the moments from the leg extension load cell. Inputs include participant number, a baseline test without contraction, and six leg extensions at 30%, 60%, 90%, and three 100% contractions.

```
% Mark IV Dynamometer Software for Knee Extensor MVIC @ 90 Degrees
%% Measurement information
prompt_participant = 'Enter participant number: ';
P = input(prompt_participant);
prompt_test = 'Select test [1 = Baseline; 2 = Knee Extension]: ';

T = input(prompt_test);
prompt_trial = 'Enter trial number: ';
Tr = input (prompt_trial);

if T == 1
    t_s = 10; % sampling time for baseline recordings

elseif T == 2
    t_s = 20; % sampling time for isometric KE trials
end

fname = sprintf('%s%02d%s','P',P,'.mat');

%% ***CURRENTLY SET FOR DEV1***
% ***IF NOT WORKING, RUN FOLLOWING LINE TO CHECK DEVICE NAME ASSIGNMENT
% daqlist
% This will find DAQ devices and change DAQ name (e.g. Dev1) where necessary
%% Create data acquisition session
daqreset
d_in = daq('ni');
d_in.Rate = 100; % fixed sampling frequency
%d_out = daq('ni');

%% Add channels
% Analog input
ch_in = addinput(d_in,'Dev1',[0 1],'Voltage'); % add 2 channels for Dev1
ch_in(1).Range = [-5.0 5.0]; ch_in(1).TerminalConfig = 'SingleEnded';
ch_in(2).Range = [-5.0 5.0]; ch_in(2).TerminalConfig = 'SingleEnded';

% Analog output to trigger Delsys EMG on Qualisys QTM
%ch_out = addoutput(d_out,'Dev1','ao0','Voltage');
%outScanData = zeros(d.Rate * d.Duration,1); % set voltage for DAQ session
%write(d_out,V_out)

scanData = read(d_in,seconds(t_s));

%% Generate moment data
calibration.R1_m = 118.9095; % slope/calibration coeffficient
calibration.R1_y = 0; % y-intercept
calibration.R2_m = 118.9268; % slope/calibration coeffficient
```

```
calibration.R2_y = 0; % y-intercept

voltage = [seconds(scanData.Time), scanData.Dev1_ai0, scanData.Dev1_ai1];

Data.Voltage(:,1) = voltage(:,1); % Time
Data.Voltage(:,2) = voltage(:,2); % R1
Data.Voltage(:,3) = voltage(:,3); % R2
Data.Moment.Right(:,1) = Data.Voltage(:,1);
Data.Moment.Right(:,2) = calibration.R1_m*Data.Voltage(:,2) + calibration.R1_y;
Data.Moment.Right(:,3) = calibration.R2_m*Data.Voltage(:,3) + calibration.R2_y;
Data.Moment.Right(:,4) = Data.Moment.Right(:,3) - Data.Moment.Right(:,2);

%% Save data
f_yn = isfile(fname);
if f_yn == 1
    load(fname)
end

if T == 1 % for baseline recordings
    baseline(Tr).Data = Data;
    save(fname,'baseline');
elseif T == 2 % for isometric KE trials
    trial(Tr).Data = Data;
    save(fname,'baseline','trial');
    %save(fname,'trial','-append');
end

%% Plot data

t = 0:0.01:(t_s-0.01);

if T == 1
    k = Tr;
    plot(t,baseline(k).Data.Moment.Right(:,2),'k')
    hold on
    plot(t,baseline(k).Data.Moment.Right(:,3),'r')
    hold off
    title('Moment')
    xlabel('Time (s)')
    ylabel('Moment (Nm)')
    ylim([0 250])
    legend('Top','Bottom')
elseif T == 2
    for k = 1:Tr
        if k == 1; c = 'k';
        elseif k == 2; c = 'r';
        elseif k == 3; c = 'g';
        elseif k == 4; c = 'b';
        elseif k == 5; c = 'm';
        elseif k == 6; c = 'c';
        elseif k == 7; c = 'k';
        end
        txt = ['Trial ', num2str(k)];
        plot(t,trial(k).Data.Moment.Right(:,4),c,'DisplayName',txt)
```

```
        title('Moment')
        xlabel('Time (s)')
        ylabel('Moment (Nm)')
        legend()
        hold on
    end
    hold off
    legend show
end

disp('Press <Enter> to continue')
pause
close

clear
clc
```

Appendix H Bone Digitization Code

The following code was used to digitize the bony landmarks on the digital bone models. Inputs include the .obj file and .jpg file for the specified bones. Prompts are included for the bone model set and side of the body to orient the coordinate axis properly. Outputs include the digital locations of the bony landmarks.

```
% Script to identify landmarkcs to set origin and orientation of lower extremity
bones
clear
clc
%% Identify .obj file, texture file, and bone
prompt_1 = "Input Filename ('Filename only w/out file extension'): ";
fname = input(prompt_1);
prompt_2 = "Identify bone (1 = Femur, 2 = Leg, 3 = Foot, 4 = Patella): ";
B = input(prompt_2);
prompt_3 = "Identify side (1 = Left, 2 = Right): ";
side = input(prompt_3);
filename = sprintf('%s%s',fname,'.obj');
texture = sprintf('%s%s',fname,'.jpg'); % May need to change file extension!!!

if B == 1
    bone = 'Femur';
elseif B == 2
    bone = 'Leg';
elseif B == 3
    bone = 'Foot';
elseif B == 4
    bone = 'Patella';
end

%% Open .obj file
v = []; vt = []; vn = []; f.v = []; f.vt = []; f.vn = [];
fid = fopen(filename);
while 1
    tline = fgetl(fid);
    if ~ischar(tline),   break,   end  % exit at end of file
     ln = sscanf(tline,'%s',1); % line type
     %disp(ln)
    switch ln
        case 'v'   % mesh vertexs
            v = [v; sscanf(tline(2:end),'%f')'];
        case 'vt'  % texture coordinate
            vt = [vt; sscanf(tline(3:end),'%f')'];
        case 'vn'  % normal coordinate
            vn = [vn; sscanf(tline(3:end),'%f')'];
        case 'f'   % face definition
            fv = []; fvt = []; fvn = [];
            str = textscan(tline(2:end),'%s'); str = str{1};
```

```
            nf = length(findstr(str{1},'/')); % number of fields with this face
vertices
            [tok str] = strtok(str,'//');     % vertex only
             for k = 1:length(tok) fv = [fv str2num(tok{k})]; end
             if (nf > 0)
             [tok str] = strtok(str,'//');   % add texture coordinates
                 for k = 1:length(tok) fvt = [fvt str2num(tok{k})]; end
             end
             if (nf > 1)
             [tok str] = strtok(str,'//');   % add normal coordinates
                 for k = 1:length(tok) fvn = [fvn str2num(tok{k})]; end
             end
              f.v = [f.v; fv]; f.vt = [f.vt; fvt]; f.vn = [f.vn; fvn];
    end
end
fclose(fid);
% set up matlab object
obj.v = v(:,1:3); obj.vt = vt; obj.vn = vn; obj.f = f; % v(:,1:3) retains columns 1:3
and deletes columns 4:6 which are colours

%% Open texture file
texture = imread(texture);
texture_img = flipdim(texture,1);
[sy sx sz] = size(texture_img);
texture_img =  reshape(texture_img,sy*sx,sz);

% make image 3D if grayscale
if sz == 1
    texture_img = repmat(texture_img,1,3);
end

% select what texture correspond to each vertex according to face
% definition
[vertex_idx fv_idx] = unique(obj.f.v);
texture_idx = obj.f.vt(fv_idx);

x = abs(round(obj.vt(:,1)*(sx-1)))+1;
y = abs(round(obj.vt(:,2)*(sy-1)))+1;
xy = sub2ind([sy sx],y,x);
texture_pts = xy(texture_idx);
tval = double(texture_img(texture_pts,:))/255;

fig = figure;
patch('vertices',obj.v,'faces',obj.f.v,'FaceVertexCData', tval);
xlabel('X')
ylabel('Y')
zlabel('Z')
shading interp
colormap gray(256)
lighting phong
camproj('perspective')
axis square
axis equal
%axis tight
```

```matlab
% If/elseif statements for bones
%% Femur
if B==1
    datacursormode on
    dcm_obj = datacursormode(fig);
    disp('Select Medial Femoral Epicondyle then press <Enter>')
    pause
    vals = getCursorInfo(dcm_obj);
    Landmarks.fem_mep = vals.Position;
    disp('Select Lateral Femoral Epicondyle then press <Enter>')
    pause
    vals = getCursorInfo(dcm_obj);
    Landmarks.fem_lep = vals.Position;
    disp('Select Greater Trochanter then press <Enter>')
    pause
    vals = getCursorInfo(dcm_obj);
    Landmarks.fem_gt = vals.Position;
     disp('Select Middle Superior Shaft then press <Enter>')
    pause
    vals = getCursorInfo(dcm_obj);
    Landmarks.fem_mss = vals.Position;
     disp('Select Middle Inferior Shaft then press <Enter>')
    pause
    vals = getCursorInfo(dcm_obj);
    Landmarks.fem_mis = vals.Position;

    close

    % Generate a point cloud to find the femoral head centroid
    femur_PC = pointCloud(obj.v);
    FHC_ok = 2;
    while FHC_ok == 2
        figure
        pcshow(femur_PC)
        axis on
        xlabel('X')
        ylabel('Y')
        zlabel('Z')
        title("Femur Point Cloud")

        view(0,90)
        prompt_x1 = 'Input X MIN boundary for femoral head: ';
        FH_boundary.x1 = input(prompt_x1);
        prompt_x2 = 'Input X MAX boundary for femoral head: ';
        FH_boundary.x2 = input(prompt_x2);
        prompt_y1 = 'Input Y MIN boundary for femoral head: ';
        FH_boundary.y1 = input(prompt_y1);
        prompt_y2 = 'Input Y MAX boundary for femoral head: ';
        FH_boundary.y2 = input(prompt_y2);
        view(0,0)
        prompt_z1 = 'Input Z MIN boundary for femoral head: ';
        FH_boundary.z1 = input(prompt_z1);
        prompt_z2 = 'Input Z MAX boundary for femoral head: ';
```

```
        FH_boundary.z2 = input(prompt_z2);
        close

        maxDistance = 0.01;
        roi = [FH_boundary.x1,FH_boundary.x2;...
            FH_boundary.y1,FH_boundary.y2;FH_boundary.z1,FH_boundary.z2];
        sampleIndices = findPointsInROI(femur_PC,roi);
        [model_FH,inlierIndices] =
pcfitsphere(femur_PC,maxDistance,SampleIndices=sampleIndices);
        FH_PC = select(femur_PC,inlierIndices);

        figure
        pcshow(FH_PC)
        title('Femoral Head Point Cloud')
        xlabel('X')
        ylabel('Y')
        zlabel('Z')

        prompt_FHC = 'Type 1 if satisfied or 2 to try again: ';
        FHC_ok = input(prompt_FHC);
        close
    end
    Landmarks.fem_hea = model_FH.Center;

    figure
    pcshow(femur_PC)
    xlabel('X')
    ylabel('Y')
    zlabel('Z')
    title('Femur with Landmarks')
    hold on

plot3(Landmarks.fem_hea(1),Landmarks.fem_hea(2),Landmarks.fem_hea(3),'o','Color','r',
'LineWidth',2,'MarkerSize',10)
    hold on

plot3(Landmarks.fem_mep(1),Landmarks.fem_mep(2),Landmarks.fem_mep(3),'o','Color','r',
'LineWidth',2,'MarkerSize',10)
    hold on

plot3(Landmarks.fem_lep(1),Landmarks.fem_lep(2),Landmarks.fem_lep(3),'o','Color','r',
'LineWidth',2,'MarkerSize',10)
    hold on

plot3(Landmarks.fem_gt(1),Landmarks.fem_gt(2),Landmarks.fem_gt(3),'o','Color','r','Li
neWidth',2,'MarkerSize',10)
    hold off
    disp('Press <Enter> to continue')
    pause
    close

    % Describe the XYZ axes as unit vectors
    CS.A = Landmarks.fem_hea; CS.B = Landmarks.fem_mep; CS.C = Landmarks.fem_lep; %
Change landmarks depending on bone
```

```
    CS.Prox = CS.A;
    CS.Dist = CS.B + 0.5*(CS.C - CS.B);
    CS.Ax1.v = CS.Prox - CS.Dist; % vector from distal to proximal
    CS.Ax1.l = norm(CS.Ax1.v);
    CS.Ax1.u = CS.Ax1.v ./ CS.Ax1.l;
    CS.Ax2.v = CS.Prox - CS.B; % vector from B to proximal
    CS.Ax3.v = CS.Prox - CS.C; % vector from C to proximal
    if side == 1 % Left, or else will have a left-hand CS
        % calculate the Y-axis, then x-axis for TF1 on left side
        CS.Ax3_Ax1.v = cross(CS.Ax3.v,CS.Ax2.v);
        CS.Ax3_Ax2.l = norm(CS.Ax3_Ax2.v);
        CS.Ax3_Ax2.u = CS.Ax3_Ax2.v ./ CS.Ax3_Ax2.l;
        CS.X = cross(CS.Ax3_Ax2.u,CS.Ax1.u);
        CS.Y = CS.Ax3_Ax2.u;
    elseif side == 2 % Right
        % calculate the Y-axis, then x-axis for TF1 on right side
        CS.Ax2_Ax3.v = cross(CS.Ax2.v,CS.Ax3.v);
        CS.Ax2_Ax3.l = norm(CS.Ax2_Ax3.v);
        CS.Ax2_Ax3.u = CS.Ax2_Ax3.v ./ CS.Ax2_Ax3.l;
        CS.TF1.X = cross(CS.Ax2_Ax3.u,CS.Ax1.u);
        CS.TF1.Y = CS.Ax2_Ax3.u;
    end
    CS.TF1.Z = CS.Ax1.u;
    % Specify the origin for the technical frames
    CS.TF1.origin = CS.Prox;
    % Define LCS for global to local and vice versa transforms
    Axis.TF1.G2L = [CS.TF1.X; CS.TF1.Y; CS.TF1.Z];
    Axis.TF1.L2G = transpose(Axis.TF1.G2L);
    % Define the origin position and orientation
    Transform.position = CS.TF1.origin;
    Transform.rotation(1) = rad2deg(atan2(-Axis.TF1.L2G(2,3),Axis.TF1.L2G(3,3)));
    Transform.rotation(2) = rad2deg(asin(Axis.TF1.L2G(1,3)));
    Transform.rotation(3) = rad2deg(atan2(-Axis.TF1.L2G(1,2),Axis.TF1.L2G(1,1)));

    figure
    pcshow(femur_PC)
    xlabel('X')
    ylabel('Y')
    zlabel('Z')
    title('Femur Model')
    hold on

plot3(CS.TF1.origin(1),CS.TF1.origin(2),CS.TF1.origin(3),'o','Color','r','LineWidth',
2,'MarkerSize',10)
    hold on
    plot3([CS.TF1.origin(1) CS.TF1.origin(1)+(2*Axis.TF1.G2L(1,1))],...
        [CS.TF1.origin(2) CS.TF1.origin(2)+(2*Axis.TF1.G2L(1,2))],...
        [CS.TF1.origin(3)
CS.TF1.origin(3)+(2*Axis.TF1.G2L(1,3))],'Color','r','LineWidth',2)
    hold on
    plot3([CS.TF1.origin(1) CS.TF1.origin(1)+(2*Axis.TF1.G2L(2,1))],...
        [CS.TF1.origin(2) CS.TF1.origin(2)+(2*Axis.TF1.G2L(2,2))],...
        [CS.TF1.origin(3)
CS.TF1.origin(3)+(2*Axis.TF1.G2L(2,3))],'Color','g','LineWidth',2)
```

```
    hold on
    plot3([CS.TF1.origin(1) CS.TF1.origin(1)+(2*Axis.TF1.G2L(3,1))],...
        [CS.TF1.origin(2) CS.TF1.origin(2)+(2*Axis.TF1.G2L(3,2))],...
        [CS.TF1.origin(3)
CS.TF1.origin(3)+(2*Axis.TF1.G2L(3,3))],'Color','b','LineWidth',2)
    hold off

    disp('Press <Enter> to continue')
    pause
    close

    %save(fname, 'Landmarks', 'Axis', 'Transform')%Origins and insertions

%% Leg
elseif B==2
    datacursormode on
    dcm_obj = datacursormode(fig);
    disp('Select Medial Tibial Condyle then press <Enter>')
    pause
    vals = getCursorInfo(dcm_obj);
    Landmarks.leg_mc = vals.Position;
    disp('Select Lateral Tibial Condyle then press <Enter>')
    pause
    vals = getCursorInfo(dcm_obj);
    Landmarks.leg_lc = vals.Position;
    disp('Select Intercondylar Eminence then press <Enter>')
    pause
    vals = getCursorInfo(dcm_obj);
    Landmarks.leg_ice = vals.Position;
    disp('Select Tibial Tuberosity then press <Enter>')
    pause
    vals = getCursorInfo(dcm_obj);
    Landmarks.leg_tub = vals.Position;
    disp('Select Fibular Head then press <Enter>')
    pause
    vals = getCursorInfo(dcm_obj);
    Landmarks.leg_fh = vals.Position;
    disp('Select Medial Malleolus then press <Enter>')
    pause
    vals = getCursorInfo(dcm_obj);
    Landmarks.leg_mm = vals.Position;
    disp('Select Lateral Malleolus then press <Enter>')
    pause
    vals = getCursorInfo(dcm_obj);
    Landmarks.leg_lm = vals.Position;

    close

    leg_PC = pointCloud(obj.v);
    figure
    pcshow(leg_PC)
    xlabel('X')
    ylabel('Y')
    zlabel('Z')
```

```
    title('Leg with Landmarks')
    hold on

plot3(Landmarks.leg_mc(1),Landmarks.leg_mc(2),Landmarks.leg_mc(3),'o','Color','r','Li
neWidth',2,'MarkerSize',10)
    hold on

plot3(Landmarks.leg_lc(1),Landmarks.leg_lc(2),Landmarks.leg_lc(3),'o','Color','r','Li
neWidth',2,'MarkerSize',10)
    hold on

plot3(Landmarks.leg_ice(1),Landmarks.leg_ice(2),Landmarks.leg_ice(3),'o','Color','r',
'LineWidth',2,'MarkerSize',10)
    hold on

plot3(Landmarks.leg_tub(1),Landmarks.leg_tub(2),Landmarks.leg_tub(3),'o','Color','r',
'LineWidth',2,'MarkerSize',10)
    hold on

plot3(Landmarks.leg_fh(1),Landmarks.leg_fh(2),Landmarks.leg_fh(3),'o','Color','r','Li
neWidth',2,'MarkerSize',10)
    hold on

plot3(Landmarks.leg_mm(1),Landmarks.leg_mm(2),Landmarks.leg_mm(3),'o','Color','r','Li
neWidth',2,'MarkerSize',10)
    hold on

plot3(Landmarks.leg_lm(1),Landmarks.leg_lm(2),Landmarks.leg_lm(3),'o','Color','r','Li
neWidth',2,'MarkerSize',10)
    hold off
    disp('Press <Enter> to continue')
    pause
    close

    % Describe the XYZ axes as unit vectors
    CS.A = Landmarks.leg_mc; CS.B = Landmarks.leg_lc; CS.C = Landmarks.leg_tub; ...
        CS.D = Landmarks.leg_mm; CS.E = Landmarks.leg_lm; % Change landmarks
depending on bone
    CS.Prox = CS.A + 0.5*(CS.B - CS.A);
    CS.Dist = CS.D + 0.5*(CS.E - CS.D);
    CS.Ax1.v = CS.Prox - CS.Dist; % vector from distal to proximal (all TF)
    CS.Ax1.l = norm(CS.Ax1.v);
    CS.Ax1.u = CS.Ax1.v ./ CS.Ax1.l;
    % NEED TO ADD LEFT/RIGHT
    if side==1 % left side
        CS.Ax2.v = CS.A - CS.B; % vector from B to A (TF1)
        CS.Ax4.v = CS.D - CS.E; % vector from E to D (TF3)
    elseif side==2
        CS.Ax2.v = CS.B - CS.A; % vector from A to B (TF1)
        CS.Ax4.v = CS.E - CS.D; % vector from D to E (TF3)
    end
    CS.Ax3.v = CS.Prox - CS.C; % vector from C to Proximal (TF2)
    % TF1
    CS.Ax1_Ax2.v = cross(CS.Ax1.v,CS.Ax2.v);
```

```
    CS.Ax1_Ax2.l = norm(CS.Ax1_Ax2.v);
    CS.Ax1_Ax2.u = CS.Ax1_Ax2.v ./ CS.Ax1_Ax2.l;
    CS.TF1.X = cross(CS.Ax1_Ax2.u,CS.Ax1.u);
    CS.TF1.Y = CS.Ax1_Ax2.u;
    CS.TF1.Z = CS.Ax1.u;
    % TF2
    CS.Ax1_Ax3.v = cross(CS.Ax1.v,CS.Ax3.v);
    CS.Ax1_Ax3.l = norm(CS.Ax1_Ax3.v);
    CS.Ax1_Ax3.u = CS.Ax1_Ax3.v ./ CS.Ax1_Ax3.l;
    CS.TF2.X = CS.Ax1_Ax3.u;
    CS.TF2.Y = cross(CS.Ax1.u,CS.TF2.X); %htere is some typo here that we need to
address
    CS.TF2.Z = CS.Ax1.u;
    % TF3
    CS.Ax1_Ax4.v = cross(CS.Ax1.v,CS.Ax4.v);
    CS.Ax1_Ax4.l = norm(CS.Ax1_Ax4.v);
    CS.Ax1_Ax4.u = CS.Ax1_Ax4.v ./ CS.Ax1_Ax4.l;
    CS.TF3.X = cross(CS.Ax1_Ax4.u,CS.Ax1.u);
    CS.TF3.Y = CS.Ax1_Ax4.u;
    CS.TF3.Z = CS.Ax1.u;
    % Specify the origin for the technical frames
    CS.TF1.origin = CS.Prox;
    CS.TF2.origin = CS.Prox;
    CS.TF3.origin = CS.Dist;

    % Define LCS for global to local and vice versa transforms
    Axis.TF1.G2L = [CS.TF1.X; CS.TF1.Y; CS.TF1.Z];
    Axis.TF1.L2G = transpose(Axis.TF1.G2L);
    Axis.TF2.G2L = [CS.TF2.X; CS.TF2.Y; CS.TF2.Z];
    Axis.TF2.L2G = transpose(Axis.TF2.G2L);
    Axis.TF3.G2L = [CS.TF3.X; CS.TF3.Y; CS.TF3.Z];
    Axis.TF3.L2G = transpose(Axis.TF3.G2L);
    % Define the origin position and orientation
    Transform.position = CS.TF1.origin;
    Transform.rotation(1) = rad2deg(atan2(-Axis.TF1.L2G(2,3),Axis.TF1.L2G(3,3)));
    Transform.rotation(2) = rad2deg(asin(Axis.TF1.L2G(1,3)));
    Transform.rotation(3) = rad2deg(atan2(-Axis.TF1.L2G(1,2),Axis.TF1.L2G(1,1)));

    figure
    pcshow(leg_PC)
    xlabel('X')
    ylabel('Y')
    zlabel('Z')
    title('Leg Model')
    hold on

plot3(CS.TF1.origin(1),CS.TF1.origin(2),CS.TF1.origin(3),'o','Color','r','LineWidth',
2,'MarkerSize',10)
    hold on
    plot3([CS.TF1.origin(1) CS.TF1.origin(1)+(2*Axis.TF1.G2L(1,1))],...
        [CS.TF1.origin(2) CS.TF1.origin(2)+(2*Axis.TF1.G2L(1,2))],...
        [CS.TF1.origin(3)
CS.TF1.origin(3)+(2*Axis.TF1.G2L(1,3))],'Color','r','LineWidth',2)
    hold on
```

```
    plot3([CS.TF1.origin(1) CS.TF1.origin(1)+(2*Axis.TF1.G2L(2,1))],...
        [CS.TF1.origin(2) CS.TF1.origin(2)+(2*Axis.TF1.G2L(2,2))],...
        [CS.TF1.origin(3)
CS.TF1.origin(3)+(2*Axis.TF1.G2L(2,3))],'Color','g','LineWidth',2)
    hold on
    plot3([CS.TF1.origin(1) CS.TF1.origin(1)+(2*Axis.TF1.G2L(3,1))],...
        [CS.TF1.origin(2) CS.TF1.origin(2)+(2*Axis.TF1.G2L(3,2))],...
        [CS.TF1.origin(3)
CS.TF1.origin(3)+(2*Axis.TF1.G2L(3,3))],'Color','b','LineWidth',2)
    hold on
    plot3([CS.TF2.origin(1) CS.TF2.origin(1)+(2*Axis.TF2.G2L(1,1))],...
        [CS.TF2.origin(2) CS.TF2.origin(2)+(2*Axis.TF2.G2L(1,2))],...
        [CS.TF2.origin(3)
CS.TF2.origin(3)+(2*Axis.TF2.G2L(1,3))],'Color','m','LineWidth',2)
    hold on
    plot3([CS.TF2.origin(1) CS.TF2.origin(1)+(2*Axis.TF2.G2L(2,1))],...
        [CS.TF2.origin(2) CS.TF2.origin(2)+(2*Axis.TF2.G2L(2,2))],...
        [CS.TF2.origin(3)
CS.TF2.origin(3)+(2*Axis.TF2.G2L(2,3))],'Color','k','LineWidth',2)
    hold on

plot3(CS.TF3.origin(1),CS.TF3.origin(2),CS.TF3.origin(3),'o','Color','r','LineWidth',
2,'MarkerSize',10)
    hold on
    plot3([CS.TF3.origin(1) CS.TF3.origin(1)+(2*Axis.TF3.G2L(1,1))],...
        [CS.TF3.origin(2) CS.TF3.origin(2)+(2*Axis.TF3.G2L(1,2))],...
        [CS.TF3.origin(3)
CS.TF3.origin(3)+(2*Axis.TF3.G2L(1,3))],'Color','r','LineWidth',2)
    hold on
    plot3([CS.TF3.origin(1) CS.TF3.origin(1)+(2*Axis.TF3.G2L(2,1))],...
        [CS.TF3.origin(2) CS.TF3.origin(2)+(2*Axis.TF3.G2L(2,2))],...
        [CS.TF3.origin(3)
CS.TF3.origin(3)+(2*Axis.TF3.G2L(2,3))],'Color','g','LineWidth',2)
    hold on
    plot3([CS.TF3.origin(1) CS.TF3.origin(1)+(2*Axis.TF3.G2L(3,1))],...
        [CS.TF3.origin(2) CS.TF3.origin(2)+(2*Axis.TF3.G2L(3,2))],...
        [CS.TF3.origin(3)
CS.TF3.origin(3)+(2*Axis.TF3.G2L(3,3))],'Color','b','LineWidth',2)
    hold off

    disp('Press <Enter> to continue')
    pause
    close

    %save(fname, 'Landmarks', 'Axis', 'Transform')

elseif B==3
    datacursormode on
    dcm_obj = datacursormode(fig);
    disp('Select Posterior Calcaneus then press <Enter>')
    pause
    vals = getCursorInfo(dcm_obj);
    Landmarks.ft_pst = vals.Position;
    disp('Select Navicular Tuberosity then press <Enter>')
```

```
    pause
    vals = getCursorInfo(dcm_obj);
    Landmarks.ft_tub = vals.Position;
    disp('Select 1st Metatarsal Head then press <Enter>')
    pause
    vals = getCursorInfo(dcm_obj);
    Landmarks.ft_mh1 = vals.Position;
    disp('Select 5th Metatarsal Head then press <Enter>')
    pause
    vals = getCursorInfo(dcm_obj);
    Landmarks.ft_mh5 = vals.Position;

    close

    foot_PC = pointCloud(obj.v);
    figure
    pcshow(foot_PC)
    xlabel('X')
    ylabel('Y')
    zlabel('Z')
    title('Foot with Landmarks')
    hold on

plot3(Landmarks.ft_pst(1),Landmarks.ft_pst(2),Landmarks.ft_pst(3),'o','Color','r','Li
neWidth',2,'MarkerSize',10)
    hold on

plot3(Landmarks.ft_tub(1),Landmarks.ft_tub(2),Landmarks.ft_tub(3),'o','Color','r','Li
neWidth',2,'MarkerSize',10)
    hold on

plot3(Landmarks.ft_mh1(1),Landmarks.ft_mh1(2),Landmarks.ft_mh1(3),'o','Color','r','Li
neWidth',2,'MarkerSize',10)
    hold on

plot3(Landmarks.ft_mh5(1),Landmarks.ft_mh5(2),Landmarks.ft_mh5(3),'o','Color','r','Li
neWidth',2,'MarkerSize',10)
    hold off
    disp('Press <Enter> to continue')
    pause
    close

    % Describe the XYZ axes as unit vectors
    CS.A = Landmarks.ft_pst; CS.B = Landmarks.ft_tub; CS.C = Landmarks.ft_mh1; ...
        CS.D = Landmarks.ft_mh5;% Change landmarks depending on bone
    CS.Prox = CS.A;
    CS.Dist = CS.C + 0.5*(CS.D - CS.C);
    CS.Ax1.v = CS.Prox - CS.Dist; % vector from distal to proximal (all TF)
    CS.Ax1.l = norm(CS.Ax1.v);
    CS.Ax1.u = CS.Ax1.v ./ CS.Ax1.l;
    if side==1 % left side
        CS.Ax2.v = CS.C - CS.D; % vector from D to C (TF1)
    elseif side==2 % right side
        CS.Ax2.v = CS.D - CS.C; % vector from C to D (TF1)
```

```
    end
    % TF1
    CS.Ax1_Ax2.v = cross(CS.Ax1.v,CS.Ax2.v);
    CS.Ax1_Ax2.l = norm(CS.Ax1_Ax2.v);
    CS.Ax1_Ax2.u = CS.Ax1_Ax2.v ./ CS.Ax1_Ax2.l;
    CS.TF1.X = cross(CS.Ax1_Ax2.u,CS.Ax1.u);
    CS.TF1.Y = CS.Ax1_Ax2.u;
    CS.TF1.Z = CS.Ax1.u;
    % Specify the origin for the technical frames
    CS.TF1.origin = CS.Prox;

    % Define LCS for global to local and vice versa transforms
    Axis.TF1.G2L = [CS.TF1.X; CS.TF1.Y; CS.TF1.Z];
    Axis.TF1.L2G = transpose(Axis.TF1.G2L);
    % Define the origin position and orientation
    Transform.position = CS.TF1.origin;
    Transform.rotation(1) = rad2deg(atan2(-Axis.TF1.L2G(2,3),Axis.TF1.L2G(3,3)));
    Transform.rotation(2) = rad2deg(asin(Axis.TF1.L2G(1,3)));
    Transform.rotation(3) = rad2deg(atan2(-Axis.TF1.L2G(1,2),Axis.TF1.L2G(1,1)));

    figure
    pcshow(foot_PC)
    xlabel('X')
    ylabel('Y')
    zlabel('Z')
    title('Foot Model')
    hold on

plot3(CS.TF1.origin(1),CS.TF1.origin(2),CS.TF1.origin(3),'o','Color','r','LineWidth',
2,'MarkerSize',10)
    hold on
    plot3([CS.TF1.origin(1) CS.TF1.origin(1)+(2*Axis.TF1.G2L(1,1))],...
        [CS.TF1.origin(2) CS.TF1.origin(2)+(2*Axis.TF1.G2L(1,2))],...
        [CS.TF1.origin(3)
CS.TF1.origin(3)+(2*Axis.TF1.G2L(1,3))],'Color','r','LineWidth',2)
    hold on
    plot3([CS.TF1.origin(1) CS.TF1.origin(1)+(2*Axis.TF1.G2L(2,1))],...
        [CS.TF1.origin(2) CS.TF1.origin(2)+(2*Axis.TF1.G2L(2,2))],...
        [CS.TF1.origin(3)
CS.TF1.origin(3)+(2*Axis.TF1.G2L(2,3))],'Color','g','LineWidth',2)
    hold on
    plot3([CS.TF1.origin(1) CS.TF1.origin(1)+(2*Axis.TF1.G2L(3,1))],...
        [CS.TF1.origin(2) CS.TF1.origin(2)+(2*Axis.TF1.G2L(3,2))],...
        [CS.TF1.origin(3)
CS.TF1.origin(3)+(2*Axis.TF1.G2L(3,3))],'Color','b','LineWidth',2)
    hold off

    disp('Press <Enter> to continue')
    pause
    close

    %save(fname, 'Landmarks', 'Axis', 'Transform')

elseif B==4
```

```
    datacursormode on
    dcm_obj = datacursormode(fig);
    disp('Select Medial Patella - Facet Ridge then press <Enter>')
    pause
    vals = getCursorInfo(dcm_obj);
    Landmarks.pat_mf = vals.Position;
    disp('Select Lateral Patella - Facet Ridge then press <Enter>')
    pause
    vals = getCursorInfo(dcm_obj);
    Landmarks.pat_lf = vals.Position;
    disp('Select Patellar Base then press <Enter>')
    pause
    vals = getCursorInfo(dcm_obj);
    Landmarks.pat_bas = vals.Position;
    disp('Select Patellar Apex press <Enter>')
    pause
    vals = getCursorInfo(dcm_obj);
    Landmarks.pat = vals.Position;

    close

    patella_PC = pointCloud(obj.v);
    figure
    pcshow(patella_PC)
    xlabel('X')
    ylabel('Y')
    zlabel('Z')
    title('Patella with Landmarks')
    hold on

plot3(Landmarks.pat_mf(1),Landmarks.pat_mf(2),Landmarks.pat_mf(3),'o','Color','r','Li
neWidth',2,'MarkerSize',10)
    hold on

plot3(Landmarks.pat_lf(1),Landmarks.pat_lf(2),Landmarks.pat_lf(3),'o','Color','r','Li
neWidth',2,'MarkerSize',10)
    hold on

plot3(Landmarks.pat_bas(1),Landmarks.pat_bas(2),Landmarks.pat_bas(3),'o','Color','r',
'LineWidth',2,'MarkerSize',10)
    hold on

plot3(Landmarks.pat(1),Landmarks.pat(2),Landmarks.pat(3),'o','Color','r','LineWidth',
2,'MarkerSize',10)
    hold off
    disp('Press <Enter> to continue')
    pause
    close

    % Describe the XYZ axes as unit vectors
    CS.A = Landmarks.pat_bas; CS.B = Landmarks.pat_mf; CS.C = Landmarks.pat_lf; ...
        CS.D = Landmarks.pat;% Change landmarks depending on bone
    CS.Mid = CS.B + 0.5*(CS.C - CS.B);
    CS.Dist = CS.D;
```

```
    CS.Ax1.v = CS.Mid - CS.Dist; % vector from distal to middle (all TF)
    CS.Ax1.l = norm(CS.Ax1.v);
    CS.Ax1.u = CS.Ax1.v ./ CS.Ax1.l;
    if side==1 % left side
        CS.Ax2.v = CS.B - CS.C; % vector from C to B (TF1)
    elseif side==2 % right side
        CS.Ax2.v = CS.C - CS.B; % vector from B to C (TF1)
    end
    % TF1
    CS.Ax1_Ax2.v = cross(CS.Ax1.v,CS.Ax2.v);
    CS.Ax1_Ax2.l = norm(CS.Ax1_Ax2.v);
    CS.Ax1_Ax2.u = CS.Ax1_Ax2.v ./ CS.Ax1_Ax2.l;
    CS.TF1.X = cross(CS.Ax1_Ax2.u,CS.Ax1.u);
    CS.TF1.Y = CS.Ax1_Ax2.u;
    CS.TF1.Z = CS.Ax1.u;
    % Specify the origin for the technical frames
    CS.TF1.origin = CS.Mid;

    % Define LCS for global to local and vice versa transforms
    Axis.TF1.G2L = [CS.TF1.X; CS.TF1.Y; CS.TF1.Z];
    Axis.TF1.L2G = transpose(Axis.TF1.G2L);
    % Define the origin position and orientation
    Transform.position = CS.TF1.origin;
    Transform.rotation(1) = rad2deg(atan2(-Axis.TF1.L2G(2,3),Axis.TF1.L2G(3,3)));
    Transform.rotation(2) = rad2deg(asin(Axis.TF1.L2G(1,3)));
    Transform.rotation(3) = rad2deg(atan2(-Axis.TF1.L2G(1,2),Axis.TF1.L2G(1,1)));

    figure
    pcshow(patella_PC)
    xlabel('X')
    ylabel('Y')
    zlabel('Z')
    title('Patella Model')
    hold on

plot3(CS.TF1.origin(1),CS.TF1.origin(2),CS.TF1.origin(3),'o','Color','r','LineWidth',
2,'MarkerSize',10)
    hold on
    plot3([CS.TF1.origin(1) CS.TF1.origin(1)+(2*Axis.TF1.G2L(1,1))],...
        [CS.TF1.origin(2) CS.TF1.origin(2)+(2*Axis.TF1.G2L(1,2))],...
        [CS.TF1.origin(3)
CS.TF1.origin(3)+(2*Axis.TF1.G2L(1,3))],'Color','r','LineWidth',2)
    hold on
    plot3([CS.TF1.origin(1) CS.TF1.origin(1)+(2*Axis.TF1.G2L(2,1))],...
        [CS.TF1.origin(2) CS.TF1.origin(2)+(2*Axis.TF1.G2L(2,2))],...
        [CS.TF1.origin(3)
CS.TF1.origin(3)+(2*Axis.TF1.G2L(2,3))],'Color','g','LineWidth',2)
    hold on
    plot3([CS.TF1.origin(1) CS.TF1.origin(1)+(2*Axis.TF1.G2L(3,1))],...
        [CS.TF1.origin(2) CS.TF1.origin(2)+(2*Axis.TF1.G2L(3,2))],...
        [CS.TF1.origin(3)
CS.TF1.origin(3)+(2*Axis.TF1.G2L(3,3))],'Color','b','LineWidth',2)
    hold off
```

```
    disp('Press <Enter> to continue')
    pause
    close

   % save(fname, 'Landmarks', 'Axis', 'Transform')
end
```

Appendix I Bone Orientation Code

The following code includes the methods of orienting the femur bone into a femoral coordinate system and establishing coordinates for vasti attachment points. Outputs include the rotation matrix used to transform the rigid body, and the landmarks for the attachment locations.

```
% Script to orient lower extremity bones
%% Identify .obj file, texture file, and bone
prompt_1 = "Input Filename ('Filename only w/out file extension'): ";
fname = input(prompt_1);
prompt_2 = "Identify side (1 = Left, 2 = Right): ";
side = input(prompt_2);
filename = sprintf('%s%s',fname,'.obj');
texture = sprintf('%s%s',fname,'.jpg'); % May need to change file extension!!!

%% Open .obj file
v = []; vt = []; vn = []; f.v = []; f.vt = []; f.vn = [];
fid = fopen(filename);
while 1
    tline = fgetl(fid);
    if ~ischar(tline),   break,   end  % exit at end of file
     ln = sscanf(tline,'%s',1); % line type
     %disp(ln)
    switch ln
        case 'v'   % mesh vertexs
            v = [v; sscanf(tline(2:end),'%f')'];
        case 'vt'  % texture coordinate
            vt = [vt; sscanf(tline(3:end),'%f')'];
        case 'vn'  % normal coordinate
            vn = [vn; sscanf(tline(3:end),'%f')'];
        case 'f'   % face definition
            fv = []; fvt = []; fvn = [];
            str = textscan(tline(2:end),'%s'); str = str{1};
           nf = length(findstr(str{1},'/')); % number of fields with this face
vertices
           [tok str] = strtok(str,'//');     % vertex only
            for k = 1:length(tok) fv = [fv str2num(tok{k})]; end
            if (nf > 0)
            [tok str] = strtok(str,'//');   % add texture coordinates
                for k = 1:length(tok) fvt = [fvt str2num(tok{k})]; end
            end
            if (nf > 1)
            [tok str] = strtok(str,'//');   % add normal coordinates
                for k = 1:length(tok) fvn = [fvn str2num(tok{k})]; end
            end
             f.v = [f.v; fv]; f.vt = [f.vt; fvt]; f.vn = [f.vn; fvn];
    end
end
fclose(fid);
% set up matlab object
obj.v = v(:,1:3); obj.vt = vt; obj.vn = vn; obj.f = f; % v(:,1:3) retains columns 1:3
and deletes columns 4:6 which are colours
```

```
%% Open .mat file
filename2 = sprintf('%s%s',fname,'.mat');
load(filename2);

%% Open texture file
texture = imread(texture);
texture_img = flipdim(texture,1);
[sy sx sz] = size(texture_img);
texture_img =  reshape(texture_img,sy*sx,sz);

% make image 3D if grayscale
if sz == 1
    texture_img = repmat(texture_img,1,3);
end

% select what texture correspond to each vertex according to face
% definition
[vertex_idx fv_idx] = unique(obj.f.v);
texture_idx = obj.f.vt(fv_idx);

x = abs(round(obj.vt(:,1)*(sx-1)))+1;
y = abs(round(obj.vt(:,2)*(sy-1)))+1;
xy = sub2ind([sy sx],y,x);
texture_pts = xy(texture_idx);
tval = double(texture_img(texture_pts,:))/255;

%% Transform
translation_A = -Transform.position;
rotationAngles_B = -Transform.rotation;
obj_A.v = obj.v + translation_A;

obj_B.v = obj_A.v * Axis.LCS.L2G;

% Scaling
scale.recon_vec = Landmarks.gt - Landmarks.lfc;
scale.recon_l = norm(scale.recon_vec);
scale_prompt_actual = "Input Distance Between Landmarks (m): "; %0.367 for
Femur_Right
scale.actual_l = input(scale_prompt_actual);
scale.factor = scale.actual_l / scale.recon_l;

% Final transform is scaling
sx = scale.factor;
sy = scale.factor;
sz = scale.factor;
[tx,ty,tz] = deal(0,0,0);
A = [sx 0 0 tx; 0 sy 0 ty; 0 0 sz tz; 0 0 0 1];
tform_C = affinetform3d(A);
obj_C.v = obj_B.v .* [sx sy sz];
obj_C.f = obj.f;

fig = figure;
patch('vertices',obj_C.v,'faces',obj_C.f.v,'FaceVertexCData', tval);
```

```
xlabel('X')
ylabel('Y')
zlabel('Z')
shading interp
colormap gray(256)
lighting phong
camproj('perspective')
view(180,0)
axis square
axis equal
%axis tight

%% Digitize Coordinates
dig_cont = 1; % 1 - continue to digitize; 2 - finished digitizing
while dig_cont == 1
    datacursormode on
    dcm_obj = datacursormode(fig);
    prompt_muscle = 'Select muscle 1: VL, 2: VML, 3: VMO, 4: VI: ';
    m_id = input(prompt_muscle);
    if m_id == 1
        m_n = 'VL';
    elseif m_id == 2
        m_n = 'VML';
    elseif m_id == 3
        m_n = 'VMO';
    elseif m_id == 4
        m_n = 'VI';
    end
    prompt_pd = 'Select 1 for proximal or 2 for distal: ';
    m_loc = input(prompt_pd);
    if m_loc == 1
        m_pd = 'P';
    elseif m_loc == 2
        m_pd = 'D';
    end

    disp('Select anatomical landmark then press <Enter>')
    pause
    vals = getCursorInfo(dcm_obj);
    LM.muscle.(m_n).(m_pd) = vals.Position;

    prompt_digitize = 'Type 1 to continue digitizing or 2 finish: ';
    dig_cont = input(prompt_digitize);
end
close

%% Transform Landmark Coordinates
% Translation
Landmarks2.A.mfc = Landmarks.mfc + translation_A;
Landmarks2.A.lfc = Landmarks.lfc + translation_A;
Landmarks2.A.gt = Landmarks.gt + translation_A;
Landmarks2.A.fhc= Landmarks.fhc + translation_A;
% Rotation
Landmarks2.B.mfc = Landmarks2.A.mfc * Axis.LCS.L2G;
```

```
Landmarks2.B.lfc = Landmarks2.A.lfc * Axis.LCS.L2G;
Landmarks2.B.gt = Landmarks2.A.gt * Axis.LCS.L2G;
Landmarks2.B.fhc = Landmarks2.A.fhc * Axis.LCS.L2G;
% Scale
LM.bone.mfc = Landmarks2.B.mfc .* [sx sy sz];
LM.bone.lfc = Landmarks2.B.lfc .* [sx sy sz];
LM.bone.gt = Landmarks2.B.gt .* [sx sy sz];
LM.bone.fhc = Landmarks2.B.fhc .* [sx sy sz];

%% Plot Landmarks
figure
patch('vertices',obj_C.v,'faces',obj_C.f.v,'FaceVertexCData', tval);
xlabel('X')
ylabel('Y')
zlabel('Z')
shading interp
colormap gray(256)
lighting phong
camproj('perspective')
view(180,0)
axis square
axis equal
title('Femur with Landmarks')
hold on
plot3(LM.bone.mfc(1),LM.bone.mfc(2),LM.bone.mfc(3),'*','Color','r','LineWidth',2,'Mar
kerSize',10)
hold on
plot3(LM.bone.lfc(1),LM.bone.lfc(2),LM.bone.lfc(3),'o','Color','r','LineWidth',2,'Mar
kerSize',10)
hold on
plot3(LM.bone.gt(1),LM.bone.gt(2),LM.bone.gt(3),'o','Color','r','LineWidth',2,'Marker
Size',10)
hold on
plot3(LM.muscle.VL.P(1),LM.muscle.VL.P(2),LM.muscle.VL.P(3),'o','Color','b','LineWidt
h',2,'MarkerSize',10)
hold on
plot3(LM.muscle.VL.D(1),LM.muscle.VL.D(2),LM.muscle.VL.D(3),'o','Color','b','LineWidt
h',2,'MarkerSize',10)
hold on
plot3(LM.muscle.VML.P(1),LM.muscle.VML.P(2),LM.muscle.VML.P(3),'o','Color','y','LineW
idth',2,'MarkerSize',10)
hold on
plot3(LM.muscle.VML.D(1),LM.muscle.VML.D(2),LM.muscle.VML.D(3),'o','Color','y','LineW
idth',2,'MarkerSize',10)
hold on
plot3(LM.muscle.VMO.P(1),LM.muscle.VMO.P(2),LM.muscle.VMO.P(3),'o','Color','g','LineW
idth',2,'MarkerSize',10)
hold on
plot3(LM.muscle.VMO.D(1),LM.muscle.VMO.D(2),LM.muscle.VMO.D(3),'o','Color','g','LineW
idth',2,'MarkerSize',10)
hold on
plot3(LM.muscle.VI.P(1),LM.muscle.VI.P(2),LM.muscle.VI.P(3),'o','Color','m','LineWidt
h',2,'MarkerSize',10)
hold on
```

```
plot3(LM.muscle.VI.D(1),LM.muscle.VI.D(2),LM.muscle.VI.D(3),'o','Color','m','LineWidt
h',2,'MarkerSize',10)
hold off
disp('Press <Enter> to continue')
pause
close

%% Define Ellipses
%Vastus Lateralis
bone_dim.VL.centre.proximal = zeros(1,3);
%bone_dim.VL.centre.distal = zeros(1,3);
bone_PC.total = pointCloud(obj_C.v);
bone_PC.VL_P = obj_C.v(obj_C.v(:,3) > LM.muscle.VL.P(3) - 0.005 & obj_C.v(:,3) <
LM.muscle.VL.P(3) + 0.005,:);
bone_dim.VL.anterior.proximal = max(bone_PC.VL_P(:,2));
bone_dim.VL.r.prox_y = 0.5*(bone_dim.VL.anterior.proximal - LM.muscle.VL.P(2));
bone_dim.VL.centroid.proximal = mean(bone_PC.VL_P);
%bone_dim.VL.centre.proximal(1) = LM.muscle.VL.P(2) + bone_dim.VL.r.prox_y;
bone_PC.VL_D = obj_C.v(obj_C.v(:,3) > LM.muscle.VL.D(3) - 0.005 & obj_C.v(:,3) <
LM.muscle.VL.D(3) + 0.005,:);
bone_dim.VL.posterior.distal = min(bone_PC.VL_D(:,2));
bone_dim.VL.r.dist_y = 0.5*(LM.muscle.VL.D(2) - bone_dim.VL.posterior.distal);
bone_dim.VL.centroid.distal = mean(bone_PC.VL_D);
%bone_dim.VL.centre.distal(1) = LM.muscle.VL.D(2) + bone_dim.VL.r.dist_y;

%Vastus Medialis Longus
bone_dim.VML.centre.proximal = zeros(1,3);
%bone_dim.VML.centre.distal = zeros(1,3);
bone_PC.total = pointCloud(obj_C.v);
bone_PC.VML_P = obj_C.v(obj_C.v(:,3) > LM.muscle.VML.P(3) - 0.005 & obj_C.v(:,3) <
LM.muscle.VML.P(3) + 0.005,:);
bone_dim.VML.anterior.proximal = max(bone_PC.VML_P(:,2));
bone_dim.VML.r.prox_y = 0.5*(bone_dim.VML.anterior.proximal - LM.muscle.VML.P(2));
bone_dim.VML.centroid.proximal = mean(bone_PC.VML_P);
%bone_dim.VML.centre.proximal(1) = LM.muscle.VML.P(2) + bone_dim.VML.r.prox_y;
bone_PC.VML_D = obj_C.v(obj_C.v(:,3) > LM.muscle.VML.D(3) - 0.005 & obj_C.v(:,3) <
LM.muscle.VML.D(3) + 0.005,:);
bone_dim.VML.posterior.distal = min(bone_PC.VML_D(:,2));
bone_dim.VML.r.dist_y = 0.5*(LM.muscle.VML.D(2) - bone_dim.VML.posterior.distal);
bone_dim.VML.centroid.distal = mean(bone_PC.VML_D);
%bone_dim.VML.centre.distal(1) = LM.muscle.VML.D(2) + bone_dim.VML.r.dist_y;

%Vastus Medialis Oblique
bone_dim.VMO.centre.proximal = zeros(1,3);
%bone_dim.VMO.centre.distal = zeros(1,3);
bone_PC.total = pointCloud(obj_C.v);
bone_PC.VMO_P = obj_C.v(obj_C.v(:,3) > LM.muscle.VMO.P(3) - 0.005 & obj_C.v(:,3) <
LM.muscle.VMO.P(3) + 0.005,:);
bone_dim.VMO.anterior.proximal = max(bone_PC.VMO_P(:,2));
bone_dim.VMO.r.prox_y = 0.5*(bone_dim.VMO.anterior.proximal - LM.muscle.VMO.P(2));
bone_dim.VMO.centroid.proximal = mean(bone_PC.VMO_P);
%bone_dim.VMO.centre.proximal(1) = LM.muscle.VMO.P(2) + bone_dim.VMO.r.prox_y;
bone_PC.VMO_D = obj_C.v(obj_C.v(:,3) > LM.muscle.VMO.D(3) - 0.005 & obj_C.v(:,3) <
LM.muscle.VMO.D(3) + 0.005,:);
```

```
bone_dim.VMO.posterior.distal = min(bone_PC.VMO_D(:,2));
bone_dim.VMO.r.dist_y = 0.5*(LM.muscle.VMO.D(2) - bone_dim.VMO.posterior.distal);
bone_dim.VMO.centroid.distal = mean(bone_PC.VMO_D);
%bone_dim.VMO.centre.distal(1) = LM.muscle.VMO.D(2) + bone_dim.VMO.r.dist_y;

%Vastus Intermedius
bone_dim.VI.centre.proximal = zeros(1,3);
%bone_dim.VI.centre.distal = zeros(1,3);
bone_PC.total = pointCloud(obj_C.v);
bone_PC.VI_P = obj_C.v(obj_C.v(:,3) > LM.muscle.VI.P(3) - 0.005 & obj_C.v(:,3) <
LM.muscle.VI.P(3) + 0.005,:);
bone_dim.VI.anterior.proximal = max(bone_PC.VI_P(:,2));
bone_dim.VI.r.prox_y = 0.5*(bone_dim.VI.anterior.proximal - LM.muscle.VI.P(2));
bone_dim.VI.centroid.proximal = mean(bone_PC.VI_P);
%bone_dim.VI.centre.proximal(1) = LM.muscle.VI.P(2) + bone_dim.VI.r.prox_y;
bone_PC.VI_D = obj_C.v(obj_C.v(:,3) > LM.muscle.VI.D(3) - 0.005 & obj_C.v(:,3) <
LM.muscle.VI.D(3) + 0.005,:);
bone_dim.VI.posterior.distal = min(bone_PC.VI_D(:,2));
bone_dim.VI.r.dist_y = 0.5*(LM.muscle.VI.D(2) - bone_dim.VI.posterior.distal);
bone_dim.VI.centroid.distal = mean(bone_PC.VI_D);
%bone_dim.VI.centre.distal(1) = LM.muscle.VI.D(2) + bone_dim.VI.r.dist_y;
% ADD TRANSFORM TO SHAFT AXIS?
if side == 1
   %Vastus Lateralis
    bone_dim.VL.medial.proximal = max(bone_PC.VL_P(:,1));
    bone_dim.VL.lateral.proximal = min(bone_PC.VL_P(:,1));
    bone_dim.VL.r.prox_x = 0.5*(bone_dim.VL.medial.proximal -
bone_dim.VL.lateral.proximal);
    %bone_dim.VL.centre.proximal(2) = bone_dim.VL.lateral.proximal +
bone_dim.VL.r.prox_x;
    bone_dim.VL.medial.distal = max(bone_PC.VL_D(:,1));
    bone_dim.VL.lateral.distal = min(bone_PC.VL_D(:,1));
    bone_dim.VL.r.dist_x = 0.5*(bone_dim.VL.medial.distal -
bone_dim.VL.lateral.distal);
    %bone_dim.VL.centre.distal(2) = bone_dim.VL.lateral.distal +
bone_dim.VL.r.dist_x;

    %Vastus Medialis Longus
    bone_dim.VML.medial.proximal = min(bone_PC.VML_P(:,1));
    bone_dim.VML.lateral.proximal = max(bone_PC.VML_P(:,1));
    bone_dim.VML.r.prox_x = 0.5*(bone_dim.VML.medial.proximal -
bone_dim.VML.lateral.proximal);
    %bone_dim.VML.centre.proximal(2) = bone_dim.VML.lateral.proximal +
bone_dim.VML.r.prox_x;
    bone_dim.VML.medial.distal = min(bone_PC.VML_D(:,1));
    bone_dim.VML.lateral.distal = max(bone_PC.VML_D(:,1));
    bone_dim.VML.r.dist_x = 0.5*(bone_dim.VML.medial.distal -
bone_dim.VML.lateral.distal);
    %bone_dim.VML.centre.distal(2) = bone_dim.VML.lateral.distal +
bone_dim.VML.r.dist_x;
elseif side == 2
   %Vastus Lateralis
    bone_dim.VL.medial.proximal = min(bone_PC.VL_P(:,1));
    bone_dim.VL.lateral.proximal = max(bone_PC.VL_P(:,1));
```

```
    bone_dim.VL.r.prox_x = 0.5*(bone_dim.VL.lateral.proximal -
bone_dim.VL.medial.proximal);
    %bone_dim.VL.centre.proximal(2) = bone_dim.VL.medial.proximal +
bone_dim.VL.r.prox_x;
    bone_dim.VL.medial.distal = min(bone_PC.VL_D(:,1));
    bone_dim.VL.lateral.distal = max(bone_PC.VL_D(:,1));
    bone_dim.VL.r.dist_x = 0.5*(bone_dim.VL.lateral.distal -
bone_dim.VL.medial.distal);
    %bone_dim.VL.centre.distal(2) = bone_dim.VL.medial.distal + bone_dim.VL.r.dist_x;

    %Vastus Medialis Longus
     bone_dim.VML.medial.proximal = min(bone_PC.VML_P(:,1));
    bone_dim.VML.lateral.proximal = max(bone_PC.VML_P(:,1));
    bone_dim.VML.r.prox_x = 0.5*(bone_dim.VML.lateral.proximal -
bone_dim.VML.medial.proximal);
    %bone_dim.VML.centre.proximal(2) = bone_dim.VML.medial.proximal +
bone_dim.VML.r.prox_x;
    bone_dim.VML.medial.distal = min(bone_PC.VML_D(:,1));
    bone_dim.VML.lateral.distal = max(bone_PC.VML_D(:,1));
    bone_dim.VML.r.dist_x = 0.5*(bone_dim.VML.lateral.distal -
bone_dim.VML.medial.distal);
    %bone_dim.VML.centre.distal(2) = bone_dim.VML.medial.distal +
bone_dim.VML.r.dist_x;
end
%bone_dim.VL.centre.proximal(3) = LM.muscle.VL.P(3);
%bone_dim.VL.centre.distal(3) = LM.muscle.VL.D(3);

t = -pi:0.01:pi;

%Vastus Lateralis
VL_a.proximal = bone_dim.VL.r.prox_x;
VL_b.proximal = bone_dim.VL.r.prox_y;
VL_x0_P = bone_dim.VL.centroid.proximal(1);
VL_y0_P = bone_dim.VL.centroid.proximal(2);
VL_ell.proximal.x = VL_x0_P + VL_a.proximal*cos(t);
VL_ell.proximal.y = VL_y0_P + VL_b.proximal*sin(t);
VL_ell.proximal.z = zeros(1,size(t,2));
VL_ell.proximal.z(1:size(t,2)) = bone_dim.VL.centroid.proximal(3);

VL_a.distal = bone_dim.VL.r.dist_x;
VL_b.distal = bone_dim.VL.r.dist_y;
VL_x0_D = bone_dim.VL.centroid.distal(1);
VL_y0_D = bone_dim.VL.centroid.distal(2);
VL_ell.distal.x = VL_x0_D + VL_a.distal*cos(t);
VL_ell.distal.y = VL_y0_D + VL_b.distal*sin(t);
VL_ell.distal.z = zeros(1,size(t,2));
VL_ell.distal.z(1:size(t,2)) = bone_dim.VL.centroid.distal(3);

%Mastus Medialis Longus
VML_a.proximal = bone_dim.VML.r.prox_x;
VML_b.proximal = bone_dim.VML.r.prox_y;
VML_x0_P = bone_dim.VML.centroid.proximal(1);
VML_y0_P = bone_dim.VML.centroid.proximal(2);
VML_ell.proximal.x = VML_x0_P + VML_a.proximal*cos(t);
```

```
VML_ell.proximal.y = VML_y0_P + VML_b.proximal*sin(t);
VML_ell.proximal.z = zeros(1,size(t,2));
VML_ell.proximal.z(1:size(t,2)) = bone_dim.VML.centroid.proximal(3);

VML_a.distal = bone_dim.VML.r.dist_x;
VML_b.distal = bone_dim.VML.r.dist_y;
VML_x0_D = bone_dim.VML.centroid.distal(1);
VML_y0_D = bone_dim.VML.centroid.distal(2);
VML_ell.distal.x = VML_x0_D + VML_a.distal*cos(t);
VML_ell.distal.y = VML_y0_D + VML_b.distal*sin(t);
VML_ell.distal.z = zeros(1,size(t,2));
VML_ell.distal.z(1:size(t,2)) = bone_dim.VML.centroid.distal(3);

%% Plot Ellipses
figure
patch('vertices',obj_C.v,'faces',obj_C.f.v,'FaceVertexCData', tval);
xlabel('X')
ylabel('Y')
zlabel('Z')
shading interp
colormap gray(256)
lighting phong
camproj('perspective')
view(180,0)
axis square
axis equal
title('Femur with Ellipses')
hold on
plot3(VL_ell.proximal.x,VL_ell.proximal.y,VL_ell.proximal.z,'Color','b','LineWidth',1
.5)
hold on
plot3(VL_ell.distal.x,VL_ell.distal.y,VL_ell.distal.z,'Color','b','LineWidth',1.5)
hold on
plot3(LM.muscle.VL.P(1),LM.muscle.VL.P(2),LM.muscle.VL.P(3),'o','Color','b','LineWidt
h',2,'MarkerSize',10)
hold on
plot3(LM.muscle.VL.D(1),LM.muscle.VL.D(2),LM.muscle.VL.D(3),'o','Color','b','LineWidt
h',2,'MarkerSize',10)
hold on
plot3(VML_ell.proximal.x,VML_ell.proximal.y,VML_ell.proximal.z,'Color','r','LineWidth
',1.5)
hold on
plot3(VML_ell.distal.x,VML_ell.distal.y,VML_ell.distal.z,'Color','r','LineWidth',1.5)
hold on
plot3(LM.muscle.VML.P(1),LM.muscle.VML.P(2),LM.muscle.VML.P(3),'o','Color','r','LineW
idth',2,'MarkerSize',10)
hold on
plot3(LM.muscle.VML.D(1),LM.muscle.VML.D(2),LM.muscle.VML.D(3),'o','Color','r','LineW
idth',2,'MarkerSize',10)
hold off
%disp('Press <Enter> to continue')
%pause
%close
```

```
save(fname, 'LM', 'Axis', 'Transform')
```

Appendix J Moment Calculation Code

The following code contains the methods of obtaining the forces and moments at the patella and femur.

Explanation of Code

The code begins with three prompts for input. The first designates which bone model will be used to model the movements of the motion capture data. Three bone models were collected, each of which has unique bone geometry. The purpose of including multiple cadaveric bone models is to determine if varying bone geometry will cause differences in the forces and moments experienced at the knee joint. For this project, only the right side was studied for mathematical simplicity. Coordinates of digitized landmarks were loaded into the code, which aligned with the bony landmarks tracked during motion capture.

The second prompt asks for participant number. Each participant was given a numerical code. Input of the numerical code attached to a participant cues the loading of motion capture, and maximum leg extension data, each in separate Matlab files. The motion capture data includes the proximal and distal end positions of the femur, leg, and patella. There is also information for the segment and joint angles for the hip, knee, and ankle, and segment angles for the patella. Finally, joint moments experienced at the hip, knee, and ankle were collected.

The proximal end position of the femur was specified as the center of the femoral head, which is calculated in Visual 3D as halfway between the greater trochanter and the center of the pelvis. The distal end of the femur is halfway between the lateral and medial femoral epicondyles. The proximal end position of the leg is considered halfway between the lateral and medial tibial condyles, and the distal end of the leg is considered halfway between the medial and lateral malleoli. The proximal end of the patella is considered to be the location of the base, and the distal end of the patella is considered to be the apex. The joint centers of the hip and knee were defined as the head of the femur and the midpoint between the femur distal end position and the leg proximal end position respectively. Relative joint angles are determined as the angle between the selected proximal and distal segment, whereas absolute segment angles are determined as the angle of the segment in relation to the global coordinate system of the recorded space. All angles and moments were collected in all three coordinate axes.

The inclusion of sex-specific coding is significant as the literature suggests that the architecture of human musculature differs between sexes[18]. This is most relevant in the relative makeup of the quadriceps muscle, and the contribution of each individual muscle to the force production of the entire quadriceps. As such, each participant was given a secondary numerical code to denote their sex, with '1' referring to males, and '2' referring to females. Depending on the sex, the percentages of physiological cross-sectional area in relation to the total as found in O'Brien et. al (2010) are applied.

Each bone model segment had to be transformed to match the size and location of the motion capture data. First, the corresponding proximal and distal end positions of the bone models had to be established to match that of the motion capture segments. The positions were found in the same way that the motion capture locations were found. Using these locations, the length of each bone segment was found, and a scaling factor was calculated by dividing the motion capture length by the bone model length. A scaling factor was calculated for each segment, and the bone models were multiplied by the scaling factors to resize the object to match the motion capture segments. Once this scaling factor was determined, all relevant landmarks, such as quadriceps attachment points on the bone, were also multiplied by the scaling factor to match their parent bone.

However, each bone model was oriented to their own local coordinate systems, meaning that although their size now matched that of the motion capture data, the bone location and orientation was not aligned with the motion capture data. Rigid body transformation to align bone landmarks with bone orientation during each frame in the vertical jump was performed by using the location of the bone's proximal endpoint (e.g. bone origin) and the rotation matrix describing the bone's orientation. These data were exported from Visual 3D.

```
%%
clear



prompt_1 = "Identify File (1= R2, 2=B3, 3= O3): ";
Colour = input (prompt_1);
% prompt_2 = "Identify bone (1 = Femur, 2 = Leg, 3 = Foot, 4 = Patella): ";
```

```
% B = input(prompt_2);
prompt_3 = "Identify Participant:  ";
Participant = input (prompt_3);
% prompt_4= "Identify Trial:  ";
%  D = input (prompt_4);

if Colour ==1



         load R2F_R_oriented.mat
         thigh.LM=LM;


 load R_2SR.mat
 shank.LM=Landmarks;
 shank.Axis=Axis;
 shank.Transform=Transform;

 load R_2FoR.mat
 foot.LM=Landmarks;
 foot.Axis=Axis;
 foot.Transform=Transform;

 load R2P_R.mat
 patella.LM=Landmarks;
 patella.Axis=Axis;
 patella.Transform=Transform;


elseif Colour == 2



         load B3F_R_Oriented.mat
    thigh.LM=LM;
 load B3S_R.mat
  shank.LM=Landmarks;
   shank.Axis=Axis;
 shank.Transform=Transform;
 load B3Fo_R.mat

  foot.LM=Landmarks;
 foot.Axis=Axis;
 foot.Transform=Transform;
 load B3P_R.mat
patella.LM=Landmarks;
 patella.Axis=Axis;
 patella.Transform=Transform;

elseif Colour == 3
```

```
         load O3F_R_oriented.mat
       thigh.LM=LM;
 load O3S_R.mat
    shank.LM=Landmarks;
     shank.Axis=Axis;
 shank.Transform=Transform;
 load O3Fo_R.mat
    foot.LM=Landmarks;
    foot.LM=Landmarks;
 foot.Axis=Axis;
 foot.Transform=Transform;
 load O3P_R.mat
patella.LM=Landmarks;
 patella.Axis=Axis;
 patella.Transform=Transform;
end


if Participant == 1  %Participant
load P1.mat
load p1_locations.mat
load p1_tt.mat
load P01.mat
E=1;

elseif Participant == 2
 load P2.mat
 load p2_locations.mat
 load p2_tt.mat
 load P02.mat
 E=1;

elseif Participant == 3
load P3.mat
load p3_locations.mat
load P03.mat
E=2;

elseif Participant == 4 %MF
    load p4.mat
    load P04.mat
    E=2;
elseif Participant == 5
    load p5.mat
    load P05.mat
    E=1;
   elseif Participant == 6
    load p6.mat
    load P06.mat
    E=1;

    elseif Participant == 7
    load p7.mat
    load P07.mat
```

```
    E=2;
    elseif Participant == 8
    load p8.mat
    load P08.mat
    E=2;
elseif Participant == 9
    load p9.mat
    load P09.mat
    load P9_rot.mat
   mass=78;



    E=1;
   D=5;

   KJangle=115;
    elseif Participant == 10
    load p10.mat
    load P010.mat
    load P10_rot.mat

mass=71.4;
    E=1;
    D=4;
    KJangle=115;
    elseif Participant == 11
    load p11.mat
    load P011.mat
  load P11_rot.mat
    mass=72.3;
    E=2;
    D=4;
    KJangle=115;
    elseif Participant == 12
    load p12.mat
    load P012.mat
   load P12_rot.mat
   mass=69.8;
    E=2;
    D=1;
    KJangle=115;
    elseif Participant == 13
    load p13.mat
    load P013.mat
 load P13_rot.mat
      mass=78.2;
    E=1;
    D=5;
    KJangle=115;
      elseif Participant == 14
    load p14.mat
    load P014.mat
  load P14_rot.mat
```

```
   mass=62;
    E=1;
    D=1;
    KJangle=115;
      elseif Participant == 15
    load p15.mat
    load P015.mat
load P15_rot.mat
    mass=63.9;
    E=1;
    D=5;
    KJangle=122;
    %CONFIRM INFORMATION WITH LAST PARTICIPANTS
      elseif Participant == 16
    load p16.mat
    load P016.mat
load P16_rot.mat
    mass=95;
    E=2;
    D=3;
    KJangle=120;
     elseif Participant == 17
    load p17.mat
    load P017.mat
   load P17_rot.mat
     mass=65.1;
    E=2;
    D=1;
    KJangle=115;
     elseif Participant == 18
    load p18.mat
    load P018.mat
load P18_rot.mat
   mass=51.9;
    E=2;
    D=1;
KJangle=108;


end




%%

% thigh.LM.muscle.VL.D(:, 1) = thigh.LM.muscle.VL.D(:, 1) + 0.05;
% %shank.LM.leg_tub(:,2)=shank.LM.leg_tub(:,2)+0.03;
% shank.LM.leg_tub(:,3)=shank.LM.leg_tub(:,3)-0.03;
if D == 1  %trial
    PEP_TH=PEP_TH{1,1};
    DEP_TH=DEP_TH{1,1};
    PEP_SH=PEP_SH{1,1};
    DEP_SH=DEP_SH{1,1};
```

```
    PEP_PAT=PEP_PAT{1,1};
    DEP_PAT=DEP_PAT{1,1};
    Ankle_JA=Ankle_JA{1,1};
    Ankle_MO=Ankle_MO{1,1};
    Ankle_SA=Ankle_SA{1,1};
     Knee_JA=Knee_JA{1,1};
    Knee_MO=Knee_MO{1,1};
    Knee_SA=Knee_SA{1,1};
     Hip_JA=Hip_JA{1,1};
    Hip_MO=Hip_MO{1,1};
    Hip_SA=Hip_SA{1,1};
    Patella_SA=Patella_SA{1,1};

elseif D == 2
    PEP_TH=PEP_TH{2,1};
     DEP_TH=DEP_TH{2,1};
      PEP_SH=PEP_SH{2,1};
    DEP_SH=DEP_SH{2,1};
     PEP_PAT=PEP_PAT{2,1};
    DEP_PAT=DEP_PAT{2,1};
     Ankle_JA=Ankle_JA{2,1};
    Ankle_MO=Ankle_MO{2,1};
    Ankle_SA=Ankle_SA{2,1};
     Knee_JA=Knee_JA{2,1};
    Knee_MO=Knee_MO{2,1};
    Knee_SA=Knee_SA{2,1};
     Hip_JA=Hip_JA{2,1};
    Hip_MO=Hip_MO{2,1};
    Hip_SA=Hip_SA{2,1};
    Patella_SA=Patella_SA{2,1};

    elseif D == 3
    PEP_TH=PEP_TH{3,1};
      DEP_TH=DEP_TH{3,1};
        PEP_SH=PEP_SH{3,1};
    DEP_SH=DEP_SH{3,1};
     PEP_PAT=PEP_PAT{3,1};
    DEP_PAT=DEP_PAT{3,1};
     Ankle_JA=Ankle_JA{3,1};
    Ankle_MO=Ankle_MO{3,1};
    Ankle_SA=Ankle_SA{3,1};
     Knee_JA=Knee_JA{3,1};
    Knee_MO=Knee_MO{3,1};
    Knee_SA=Knee_SA{3,1};
     Hip_JA=Hip_JA{3,1};
    Hip_MO=Hip_MO{3,1};
    Hip_SA=Hip_SA{3,1};
    Patella_SA=Patella_SA{3,1};

    elseif D == 4
    PEP_TH=PEP_TH{4,1};
    DEP_TH=DEP_TH{4,1};
     PEP_SH=PEP_SH{4,1};
    DEP_SH=DEP_SH{4,1};
```

```
     PEP_PAT=PEP_PAT{4,1};
    DEP_PAT=DEP_PAT{4,1};
     Ankle_JA=Ankle_JA{4,1};
    Ankle_MO=Ankle_MO{4,1};
    Ankle_SA=Ankle_SA{4,1};
     Knee_JA=Knee_JA{4,1};
    Knee_MO=Knee_MO{4,1};
    Knee_SA=Knee_SA{4,1};
     Hip_JA=Hip_JA{4,1};
    Hip_MO=Hip_MO{4,1};
    Hip_SA=Hip_SA{4,1};
    Patella_SA=Patella_SA{4,1};

    elseif D == 5
    PEP_TH=PEP_TH{5,1};
     DEP_TH=DEP_TH{5,1};
      PEP_SH=PEP_SH{5,1};
    DEP_SH=DEP_SH{5,1};
     PEP_PAT=PEP_PAT{5,1};
    DEP_PAT=DEP_PAT{5,1};
     Ankle_JA=Ankle_JA{5,1};
    Ankle_MO=Ankle_MO{5,1};
    Ankle_SA=Ankle_SA{5,1};
     Knee_JA=Knee_JA{5,1};
    Knee_MO=Knee_MO{5,1};
    Knee_SA=Knee_SA{5,1};
     Hip_JA=Hip_JA{5,1};
    Hip_MO=Hip_MO{5,1};
    Hip_SA=Hip_SA{5,1};
    Patella_SA=Patella_SA{5,1};

end

%%



%%
 thigh.segmentangle = [Hip_SA(:,1), Hip_SA(:,2), Hip_SA(:,3)];
    thigh.jointangle= Hip_JA;
    shank.segmentangle = [Knee_SA(:,1), Knee_SA(:,2), Knee_SA(:,3)];
    shank.jointangle=180-abs(Knee_JA);
    patella.segmentangle=[Patella_SA(:,1), Patella_SA(:,2), Patella_SA(:,3)];
    shank.knee.moment=Knee_MO;
if E==1
    load male_muscle.mat
    thigh.VML.PCSA=25.35;
    thigh.VMO.PCSA=25.35;
    thigh.VI.PCSA=64.2;
    thigh.VL.PCSA=64;
    thigh.RF.PCSA=35.7;
    thigh.Q.PCSA=214.6;
    thigh.QT.PCSA=thigh.VL.PCSA+thigh.VI.PCSA+thigh.RF.PCSA+thigh.VML.PCSA;
    thigh.QT.VLPercent=thigh.VL.PCSA/thigh.QT.PCSA;
```

```
    thigh.QT.VIPercent=thigh.VI.PCSA/thigh.QT.PCSA;
    thigh.QT.RFPercent=thigh.RF.PCSA/thigh.QT.PCSA;
    thigh.QT.VMLPercent=thigh.VML.PCSA/thigh.QT.PCSA;
    thigh.VML.percent=thigh.VML.PCSA/thigh.Q.PCSA;
    thigh.VMO.percent=[thigh.VMO.PCSA/thigh.Q.PCSA, thigh.VMO.PCSA/thigh.Q.PCSA,
thigh.VMO.PCSA/thigh.Q.PCSA];
    thigh.VI.percent=thigh.VI.PCSA/thigh.Q.PCSA;
    thigh.VL.percent=thigh.VL.PCSA/thigh.Q.PCSA;
    thigh.RF.percent=thigh.RF.PCSA/thigh.Q.PCSA;

thigh.QT.percent=thigh.VML.percent+thigh.VI.percent+thigh.RF.percent+thigh.VL.percent
;

patella.PL.percent=[thigh.VML.percent+thigh.VL.percent+thigh.VI.percent+thigh.RF.perc
ent, thigh.VML.percent+thigh.VL.percent+thigh.VI.percent+thigh.RF.percent,
thigh.VML.percent+thigh.VL.percent+thigh.VI.percent+thigh.RF.percent];
  thigh.Q.MaxForce=25*thigh.Q.PCSA;
thigh.VML.MaxForce=25*thigh.VML.PCSA;
thigh.VMO.MaxForce=25*thigh.VMO.PCSA;
thigh.VL.MaxForce=25*thigh.VL.PCSA;
thigh.VI.MaxForce=25*thigh.VI.PCSA;
thigh.RF.MaxForce=25*thigh.RF.PCSA;



elseif E == 2

    load female_muscle.mat
    thigh.VL.PCSA=46.6;
    thigh.VML.PCSA=20.25;
    thigh.VMO.PCSA=20.25;
    thigh.VI.PCSA=45.5;
    thigh.RF.PCSA=28.3;
    thigh.Q.PCSA=170.9;
    thigh.QT.PCSA=thigh.VL.PCSA+thigh.VI.PCSA+thigh.RF.PCSA+thigh.VML.PCSA;
    thigh.QT.VLPercent=thigh.VL.PCSA/thigh.QT.PCSA;
    thigh.QT.VIPercent=thigh.VI.PCSA/thigh.QT.PCSA;
    thigh.QT.RFPercent=thigh.RF.PCSA/thigh.QT.PCSA;
    thigh.QT.VMLPercent=thigh.VML.PCSA/thigh.QT.PCSA;
      thigh.VML.percent=thigh.VML.PCSA/thigh.Q.PCSA;
    thigh.VMO.percent=[thigh.VMO.PCSA/thigh.Q.PCSA, thigh.VMO.PCSA/thigh.Q.PCSA,
thigh.VMO.PCSA/thigh.Q.PCSA];
    thigh.VI.percent=thigh.VI.PCSA/thigh.Q.PCSA;
    thigh.VL.percent=thigh.VL.PCSA/thigh.Q.PCSA;
    thigh.RF.percent=thigh.RF.PCSA/thigh.Q.PCSA;

thigh.QT.percent=thigh.VML.percent+thigh.VI.percent+thigh.RF.percent+thigh.VL.percent
;

patella.PL.percent=[thigh.VML.percent+thigh.VL.percent+thigh.VI.percent+thigh.RF.perc
ent, thigh.VML.percent+thigh.VL.percent+thigh.VI.percent+thigh.RF.percent,
thigh.VML.percent+thigh.VL.percent+thigh.VI.percent+thigh.RF.percent];
    thigh.Q.MaxForce=25*thigh.Q.PCSA;
thigh.VML.MaxForce=25*thigh.VML.PCSA;
```

```
thigh.VMO.MaxForce=25*thigh.VMO.PCSA;
thigh.VL.MaxForce=25*thigh.VL.PCSA;
thigh.VI.MaxForce=25*thigh.VI.PCSA;
thigh.RF.MaxForce=25*thigh.RF.PCSA;


end
f=length(thigh.jointangle);
 thigh.rotation=thigh_rot{1,1};
    shank.rotation=shank_rot{1,1};
    patella.rotation=pat_rot{1,1};



% Sample data for illustration: replace this with your actual rotation values
numFrames = f;
rotationValues = rand(numFrames, 9); % Example: random values for 307 frames

% Initialize a 3D array to store 3x3 rotation matrices for each frame
thigh.rotationmatrix = zeros(3, 3, numFrames);
shank.rotationmatrix=zeros(3, 3, numFrames);
patella.rotationmatrix=zeros(3, 3, numFrames);

% Loop through each frame and reshape the corresponding row into a 3x3 matrix
for frame = 1:numFrames
    thigh.rotationmatrix(:, :, frame) = reshape(thigh.rotation(frame, :), [3, 3]);
    shank.rotationmatrix(:, :, frame) = reshape(shank.rotation(frame, :), [3, 3]);
    patella.rotationmatrix(:, :, frame) = reshape(patella.rotation(frame, :), [3,
3]);
end


% if  B == 1

%Loren Suggestion
%  transformed_pep= (Axis.LCS.G2L(Landmarks.gt-Transform.position))
% Scaled_pep= transformed_pep.*SF_TH
% To transform the coordinates, you would use: Transformed_coordinates = (R *
(Landmark_Coordinates - Position)')'
% where R is the G2L rotation matrix; make sure to include the transpose functions.
%
% Then, you will need to scale the coordinates, so: Scaled_coordinates =
Transformed_coordinates * scaling_factor
% where the scaling factor can be a single value, e.g. scaling_factor = 0.5, or an
array [0.5 0.5 0.5]
thigh.ST=25;
thigh.pep=thigh.LM.bone.fhc;
thigh.dep=(thigh.LM.bone.lfc+thigh.LM.bone.mfc)/2;
thigh.length=norm (thigh.dep-thigh.pep);
Digital_Length_TH=norm (DEP_TH(1,:)-PEP_TH(1,:));
SF_TH=Digital_Length_TH/thigh.length;
thigh.pep=thigh.pep*SF_TH;
thigh.dep=thigh.dep*SF_TH;
thigh.length=norm(thigh.dep-thigh.pep);
```

```
shank.LM.leg_mc = (shank.LM.leg_mc - shank.Transform.position)*shank.Axis.TF1.L2G;
shank.LM.leg_lc = (shank.LM.leg_lc - shank.Transform.position)*shank.Axis.TF1.L2G;
shank.LM.leg_ice = (shank.LM.leg_ice - shank.Transform.position)*shank.Axis.TF1.L2G;
shank.LM.leg_tub = (shank.LM.leg_tub - shank.Transform.position)*shank.Axis.TF1.L2G;
shank.LM.leg_fh = (shank.LM.leg_fh - shank.Transform.position)*shank.Axis.TF1.L2G;
shank.LM.leg_mm = (shank.LM.leg_mm - shank.Transform.position)*shank.Axis.TF1.L2G;
shank.LM.leg_lm = (shank.LM.leg_lm - shank.Transform.position)*shank.Axis.TF1.L2G;
shank.LM.pep=(shank.LM.leg_mc+shank.LM.leg_lc)/2;
shank.LM.dep=(shank.LM.leg_mm+shank.LM.leg_lm)/2;
shank.length=norm (shank.LM.dep-shank.LM.pep);
Digital_Length_SH=norm (DEP_SH(1,:)-PEP_SH(1,:));
SF_SH=Digital_Length_SH/shank.length;
shank.pep=shank.LM.pep*SF_SH;
shank.dep=shank.LM.dep*SF_SH;
shank.tt=shank.LM.leg_tub*SF_SH;
shank.length=norm(shank.dep-shank.pep);

patella.LM.pat_mf = (patella.LM.pat_mf -
patella.Transform.position)*patella.Axis.TF1.L2G;
patella.LM.pat_lf = (patella.LM.pat_lf -
patella.Transform.position)*patella.Axis.TF1.L2G;
patella.LM.pat_bas = (patella.LM.pat_bas -
patella.Transform.position)*patella.Axis.TF1.L2G;
patella.LM.pat = (patella.LM.pat - patella.Transform.position)*patella.Axis.TF1.L2G;
patella.LM.pat_VMO = (patella.LM.pat_VMO -
patella.Transform.position)*patella.Axis.TF1.L2G;
patella.centre=(patella.LM.pat+patella.LM.pat_bas+patella.LM.pat_lf+patella.LM.pat_mf
)/4;

correction=-patella.LM.pat_bas;

patella.LM.pat_mf=patella.LM.pat_mf+correction;
patella.LM.pat_lf=patella.LM.pat_lf+correction;
patella.LM.pat_bas=patella.LM.pat_bas+correction;
patella.LM.pat=patella.LM.pat+correction;
patella.LM.pat_VMO=patella.LM.pat_VMO+correction;
patella.centre=patella.centre+correction;


patella.height=norm(patella.LM.pat_bas-patella.LM.pat);
Digital_height=norm(PEP_PAT(1,:)- DEP_PAT(1,:));
SF_P=Digital_height/patella.height;

patella.base= patella.LM.pat_bas*SF_P;
patella.apex= patella.LM.pat*SF_P;
patella.lf= patella.LM.pat_lf*SF_P;
patella.mf= patella.LM.pat_mf*SF_P;
patella.VMO=patella.LM.pat_VMO*SF_P;
patella.centre= patella.centre*SF_P; %multiply by scaling factor


% DtoMC_translation_vector=PEP_TH(1,:)-scaled_thigh.pep; %always have to translate
the digital bone by this value because this is from digital to mocap position
```

```
%
% thigh.pep_original=scaled_thigh.pep+DtoMC_translation_vector;
% thigh.dep_original=scaled_thigh.dep+DtoMC_translation_vector;



    thigh.PEP = PEP_TH;
    thigh.DEP = DEP_TH;
    shank.PEP = PEP_SH;
    shank.DEP = DEP_SH;
    patella.PEP=PEP_PAT;
    patella.DEP=DEP_PAT;
    patella.CENTRE=(patella.PEP+patella.DEP)/2;
    %will have to add in corresponding patella and foot ones.


thigh.VL.P = thigh.LM.muscle.VL.P*SF_TH;
thigh.VL.D= thigh.LM.muscle.VL.D*SF_TH;
thigh.VML.P= thigh.LM.muscle.VML.P*SF_TH;
thigh.VML.D=thigh.LM.muscle.VML.D*SF_TH;
thigh.VI.P=thigh.LM.muscle.VI.P*SF_TH;
thigh.VI.D=thigh.LM.muscle.VI.D*SF_TH;
thigh.VMO.P=thigh.LM.muscle.VMO.P*SF_TH;
thigh.VMO.D=thigh.LM.muscle.VMO.D*SF_TH;
%%
%%
f = size(thigh.PEP,1); % find the number of frames
rotm.thigh=zeros(3,3,f);
rotm.shank=zeros(3,3,f);
rotm.patella=zeros(3,3,f);

% create transformation matrix
% tform = rigidtform3d(angle,translation)
%rotm = eul2rotm(eul,sequence)


    for s = 1:f % use loop to create transformation matrix for each frame
        tform.thigh(s,:) = rigidtform3d(thigh.segmentangle(s,:),thigh.PEP(s,:));
        tform.shank(s,:) = rigidtform3d(shank.segmentangle(s,:),shank.PEP(s,:));
        tform.patella(s,:) =
rigidtform3d(patella.segmentangle(s,:),patella.CENTRE(s,:));
      % tform.thigh(s,1).R=eul2rotm(thigh.segmentangle(s,:), "XYZ");
      % tform.shank(s,1).R=eul2rotm(shank.segmentangle(s,:), "ZYX");
      % rotm.patella(:,:,s)=eul2rotm(patella.segmentangle(s,:), "ZYX");

    end


% random coordinates; replace here with structures with the coordinates you want to
transform
% e.g. thigh.distal, thigh.VL, shank.distal, shank.TT, etc.

    %
```

```
    f = size(thigh.PEP,1);
    % thigh.VL.P = zeros(f,3); % create an array the size of the trial
    for s = 1:f % use loop to apply transform for each frame

        thigh.pep_tform(s,:)= PEP_TH(s,:);
        thigh.dep_tform(s,:)= (thigh.rotationmatrix(:,:,s) * thigh.dep')' +
PEP_TH(s,:);
        thigh.VL.P_tform(s,:) = (thigh.rotationmatrix(:,:,s) * thigh.VL.P')' +
PEP_TH(s,:);
        thigh.VL.D_tform(s,:) = (thigh.rotationmatrix(:,:,s) * thigh.VL.D')' +
PEP_TH(s,:);
        thigh.VML.P_tform(s,:) = (thigh.rotationmatrix(:,:,s) * thigh.VML.P')' +
PEP_TH(s,:);
        thigh.VML.D_tform(s,:) = (thigh.rotationmatrix(:,:,s) * thigh.VML.D')' +
PEP_TH(s,:);
        thigh.VMO.P_tform(s,:) = (thigh.rotationmatrix(:,:,s) * thigh.VMO.P')' +
PEP_TH(s,:);
        thigh.VMO.D_tform(s,:) = (thigh.rotationmatrix(:,:,s) * thigh.VMO.D')' +
PEP_TH(s,:);
        thigh.VI.P_tform(s,:) = (thigh.rotationmatrix(:,:,s) * thigh.VI.P')' +
PEP_TH(s,:);
        thigh.VI.D_tform(s,:) = (thigh.rotationmatrix(:,:,s) * thigh.VI.D')' +
PEP_TH(s,:);

        shank.pep_tform(s,:) = PEP_SH(s,:);
        shank.dep_tform(s,:) = (shank.rotationmatrix(:,:,s)*shank.dep')'+PEP_SH(s,:);
        shank.tt_tform(s,:) = (shank.rotationmatrix(:,:,s) * shank.tt')' +
PEP_SH(s,:);


        patella.base_tform(s,:) = PEP_PAT(s,:);
        patella.apex_tform(s,:) = DEP_PAT(s,:);
        patella.lf_tform(s,:) = (patella.rotationmatrix(:,:,s) * patella.lf')' +
PEP_PAT(s,:);
        patella.mf_tform(s,:) = (patella.rotationmatrix(:,:,s) * patella.mf')' +
PEP_PAT(s,:);
        patella.centre_tform(s,:) = (patella.rotationmatrix(:,:,s) *
patella.centre')' + PEP_PAT(s,:);
        patella.VMO_tform(s,:)=(patella.rotationmatrix(:,:,s) * patella.VMO')' +
PEP_PAT(s,:);
    end
% patella.base_tform(:,2) = patella.base_tform(:,2)-0.03;
%         patella.apex_tform(:,2) =  patella.apex_tform(:,2)-0.03;
%         patella.lf_tform(:,2) =  patella.lf_tform(:,2)-0.03;
%         patella.mf_tform(:,2) =  patella.mf_tform(:,2)-0.03;
%         patella.centre_tform(:,2) =  patella.centre_tform(:,2)-0.03;
%         patella.VMO_tform(:,2)= patella.VMO_tform(:,2)-0.03;
     %Angles


%%

%Total force experienced at joint angles (x).
a=zeros (f,1);
```

```
b=zeros(f,1);
c=zeros(f,1);
shank.PL.Forceyz=zeros (f,1);
Phi=zeros (f,1);
Theta=zeros(f,1);
shank.PL.i=zeros (f,1);
shank.PL.j=zeros (f,1);
shank.PL.k=zeros (f,1);
shank.PL.ij=zeros (f,1);
shank.PL.jk=zeros (f,1);
shank.PL.ki=zeros (f,1);
patella.PL.i=zeros (f,1);
patella.PL.j=zeros (f,1);
patella.PL.k=zeros (f,1);
patella.PL.ij=zeros (f,1);
patella.PL.jk=zeros (f,1);
patella.PL.ki=zeros (f,1);
 patella.PL.Forcexy= zeros (f,1);
  patella.PL.Forceyz= zeros (f,1);
   patella.PL.Forcezx= zeros (f,1);
 shank.PL.rxy=zeros (f,1);
 shank.PL.ryz=zeros (f,1);
 shank.PL.rzx=zeros (f,1);
     shank.PL.ryz=zeros (f,1);
     shank.PL.rzx=zeros (f,1);
      shank.PL.Theta.yz=zeros (f,1);
   shank.PL.Forceyz=zeros (f,1);
patella.PL.Forcey= zeros (f,1);
patella.PL.Forcez= zeros (f,1);
patella.PL.M=zeros (f,1);
patella.QT.Forcex= zeros (f,1);
patella.QT.Forcey= zeros (f,1);
patella.QT.Forcez= zeros (f,1);
patella.QT.M=zeros (f,1);
patella.VMOVec.Forcex= zeros (f,1);
patella.VMOVec.Forcey= zeros (f,1);
patella.VMOVec.Forcez= zeros (f,1);
patella.VMOVec.M=zeros (f,1);
%%
      % PL Action on Shank

for s = 1:f
    shank.PL.r(s,:) = shank.tt_tform(s,:) - shank.pep_tform(s,:); % PL moment arm
vector
    shank.PL.FVec(s,:) = patella.apex_tform(s,:) - shank.tt_tform(s,:); % PL force
vector
    shank.PL.rxy(s)=norm([shank.PL.r(s,1) shank.PL.r(s,2)]);
      shank.PL.ryz(s)=norm([shank.PL.r(s,2) shank.PL.r(s,3)]);
        shank.PL.rzx(s)=norm([shank.PL.r(s,3) shank.PL.r(s,1)]);
    % I believe both of the above are in the lab coordinate system, in
    % which case we will also need to transform them into the shank
    % coordinate system (so we are getting the shank sagittal plane)
    % before running the next lines in the for loop
    shank.PL.Funit(s,1) = shank.PL.FVec(s,1)/norm(shank.PL.FVec(s,:));
```

```
    shank.PL.Funit(s,2) = shank.PL.FVec(s,2)/norm(shank.PL.FVec(s,:));
    shank.PL.Funit(s,3) = shank.PL.FVec(s,3)/norm(shank.PL.FVec(s,:));
    shank.PL.MAV(s,:) =
(cross(shank.PL.r(s,:),shank.PL.Funit(s,:)))*tform.shank(s,1).R'; % caculate "moment
arm vector"
    shank.PL.force(s,:) = shank.knee.moment(s,1) / shank.PL.MAV(s,1); % use NJM X and
MAV in column 1 (YZ plane) to calculate force

    % shank.PL.normalforce(s,:)=norm(shank.PL.force(s,:));
end
shank.PL.forcey=shank.PL.force.*shank.PL.Funit(:,2);
    shank.PL.forcez=shank.PL.force.*shank.PL.Funit(:,3);

      %%


% %%
     %Patellar Ligament action on patella
     patella.PL.rxy=zeros(f,1);
  patella.PL.ryz=zeros(f,1);
  patella.PL.rzx=zeros(f,1);
 for s=1:f
       patella.PL.r(s,:)= patella.apex_tform (s,:)-patella.centre_tform(s,:);%correct
       patella.PL.rxy(s)=sqrt(patella.PL.r(s,1)^2+patella.PL.r(s,2)^2);
       patella.PL.ryz(s)=sqrt(patella.PL.r(s,2)^2+patella.PL.r(s,3)^2);
        patella.PL.rzx(s)=sqrt(patella.PL.r(s,3)^2+patella.PL.r(s,1)^2);
        patella.PL.FVec(s,:)=shank.tt_tform(s,:)-patella.apex_tform(s,:);
        patella.PL.i(s)=patella.PL.FVec(s,1)/norm(patella.PL.FVec(s,:));
        patella.PL.j(s)= patella.PL.FVec(s,2)/norm(patella.PL.FVec(s,:));
        patella.PL.k(s)= patella.PL.FVec(s,3)/norm( patella.PL.FVec(s,:));
        patella.PL.ij(s)=norm([patella.PL.i(s) patella.PL.j(s)]);
        patella.PL.jk(s)=norm([patella.PL.j(s) patella.PL.k(s)]);
        patella.PL.ki(s)=norm([patella.PL.k(s) patella.PL.i(s)]);
        patella.PL.ijk(s,:)=[patella.PL.i(s), patella.PL.j(s), patella.PL.k(s)];
   patella.PL.u= norm(patella.PL.ijk);
patella.PL.Force(s,:)=[shank.PL.force(s)*patella.PL.i(s),shank.PL.force(s)*patella.PL
.j(s), shank.PL.force(s)*patella.PL.k(s)];
patella.PL.Forcexy(s)=sqrt(patella.PL.Force(s,1)^2+patella.PL.Force(s,2)^2);
patella.PL.Forceyz(s)=sqrt(patella.PL.Force(s,2)^2+patella.PL.Force(s,3)^2);
patella.PL.Forcezx(s)=sqrt(patella.PL.Force(s,3)^2+patella.PL.Force(s,1)^2);



 end
  patella.PL.Theta.xy=zeros(f,1);
  patella.PL.Theta.yz=zeros(f,1);
  patella.PL.Theta.zx=zeros(f,1);
%Theta for Mz


%%

 %Quadriceps Tendon
 patella.QT.rxy=zeros(f,1);
```

```
  patella.QT.ryz=zeros(f,1);
  patella.QT.rzx=zeros(f,1);
  patella.QT.i=zeros(f,1);
  patella.QT.j=zeros(f,1);
  patella.QT.k=zeros(f,1);
   patella.QT.ij=zeros(f,1);
  patella.QT.jk=zeros(f,1);
  patella.QT.ki=zeros(f,1);



for s=1:f
      patella.QT.r(s,:)= patella.base_tform(s,:)-patella.centre_tform(s,:);
      patella.QT.rxy(s)=sqrt(patella.QT.r(s,1)^2+patella.QT.r(s,2)^2);
       patella.QT.ryz(s)=sqrt(patella.QT.r(s,2)^2+patella.QT.r(s,3)^2);
        patella.QT.rzx(s)=sqrt(patella.QT.r(s,3)^2+patella.QT.r(s,1)^2);
        patella.QT.FVec(s,:)=thigh.VL.D_tform(s,:)-patella.base_tform(s,:);
        patella.QT.i(s)=patella.QT.FVec(s,1)/norm( patella.QT.FVec(s,:));
        patella.QT.j(s)= patella.QT.FVec(s,2)/norm( patella.QT.FVec(s,:));
        patella.QT.k(s)= patella.QT.FVec(s,3)/norm( patella.QT.FVec(s,:));
         patella.QT.ij(s)=norm([patella.QT.i(s) patella.QT.j(s)]);
        patella.QT.jk(s)=norm([patella.QT.j(s) patella.QT.k(s)]);
        patella.QT.ki(s)=norm([patella.QT.k(s) patella.QT.i(s)]);
        patella.QT.ijk(s,:)=[patella.QT.i(s), patella.QT.j(s), patella.QT.k(s)];

   patella.QT.u(s)= norm(patella.QT.ijk(s,:));
end


patella.QT.Theta.xy=zeros(f,1);
  patella.QT.Theta.yz=zeros(f,1);
  patella.QT.Theta.zx=zeros(f,1);


    %%
  %VMO

 patella.VMOVec.rxy=zeros(f,1);
  patella.VMOVec.ryz=zeros(f,1);
  patella.VMOVec.rzx=zeros(f,1);
  patella.VMOVec.i=zeros(f,1);
  patella.VMOVec.j=zeros(f,1);
  patella.VMOVec.k=zeros(f,1);
   patella.VMOVec.ij=zeros(f,1);
  patella.VMOVec.jk=zeros(f,1);
  patella.VMOVec.ki=zeros(f,1);

for s=1:f
    patella.VMOVec.r(s,:)= patella.VMO_tform(s,:)-patella.centre_tform(s,:);
    patella.VMOVec.rxy(s)=sqrt(patella.VMOVec.r(s,1)^2+patella.VMOVec.r(s,2)^2);
       patella.VMOVec.ryz(s)=sqrt(patella.VMOVec.r(s,2)^2+patella.VMOVec.r(s,3)^2);
        patella.VMOVec.rzx(s)=sqrt(patella.VMOVec.r(s,3)^2+patella.VMOVec.r(s,1)^2);
        patella.VMOVec.FVec(s,:)=thigh.VMO.D_tform(s,:)-patella.VMO_tform(s,:);
        patella.VMOVec.i(s)=patella.VMOVec.FVec(s,1)/norm( patella.VMOVec.FVec(s,:));
```

```
        patella.VMOVec.j(s)= patella.VMOVec.FVec(s,2)/norm(
patella.VMOVec.FVec(s,:));
        patella.VMOVec.k(s)= patella.VMOVec.FVec(s,3)/norm(
patella.VMOVec.FVec(s,:));
         patella.VMOVec.ij(s)=norm([patella.VMOVec.i(s) patella.VMOVec.j(s)]);
        patella.VMOVec.jk(s)=norm([patella.VMOVec.j(s) patella.VMOVec.k(s)]);
        patella.VMOVec.ki(s)=norm([patella.VMOVec.k(s) patella.VMOVec.i(s)]);
        patella.VMOVec.ijk(s,:)=[patella.VMOVec.i(s), patella.VMOVec.j(s),
patella.VMOVec.k(s)];
   patella.VMOVec.u(s)= norm(patella.VMOVec.ijk(s,:));
end
%% VMO XY
clear A B C

dot_AB_BC=zeros(f,1);
magnitude_AB=zeros(f,1);
magnitude_BC=zeros(f,1);
cos_angle=zeros(f,1);
angle_rad=zeros(f,1);
angle_deg=zeros(f,1);
patella.VMOVec.Theta.xy=zeros(f,1);
patella.VMOVec.Theta.yz=zeros(f,1);
patella.VMOVec.Theta.zx=zeros(f,1);

for s=1:f
% Define the coordinates of the points
A(s,:) = [patella.centre_tform(s,1) patella.centre_tform(s,2)];
B(s,:)= [patella.VMO_tform(s,1), patella.VMO_tform(s,2)];
C(s,:) = [thigh.VMO.D_tform(s,1),thigh.VMO.D_tform(s,2) ];

% Define the vectors AB and BC
AB(s,:) = B(s,:) - A(s,:);
BC(s,:) =  B(s,:)-C(s,:);

% Calculate the dot products
dot_AB_BC(s) = dot(AB(s,:), BC(s,:));

% Calculate the magnitudes of the vectors
magnitude_AB(s) = norm(AB(s,:));
magnitude_BC(s) = norm(BC(s,:));

% Calculate the cosine of the angle between the vectors using the dot product formula
cos_angle(s) = dot_AB_BC(s) / (magnitude_AB(s) * magnitude_BC(s));

% Calculate the angle in radians
angle_rad(s) = acos(cos_angle(s));

% Convert the angle to degrees
patella.VMOVec.Theta.xy(s) = rad2deg(angle_rad(s));
end

for s=1:f
    if patella.VMOVec.Theta.xy(s)>90
        patella.VMOVec.Theta.xy(s)=180-patella.VMOVec.Theta.xy(s);
```

```
    end
end

for s=1:f
% Define the coordinates of the points
A(s,:) = [patella.centre_tform(s,2) patella.centre_tform(s,3)];
B(s,:)= [patella.VMO_tform(s,2), patella.VMO_tform(s,3)];
C(s,:) = [thigh.VMO.D_tform(s,2),thigh.VMO.D_tform(s,3) ];

% Define the vectors AB and BC
AB(s,:) = B(s,:) - A(s,:);
BC(s,:) =  B(s,:)-C(s,:);

% Calculate the dot products
dot_AB_BC(s) = dot(AB(s,:), BC(s,:));

% Calculate the magnitudes of the vectors
magnitude_AB(s) = norm(AB(s,:));
magnitude_BC(s) = norm(BC(s,:));

% Calculate the cosine of the angle between the vectors using the dot product formula
cos_angle(s) = dot_AB_BC(s) / (magnitude_AB(s) * magnitude_BC(s));

% Calculate the angle in radians
angle_rad(s) = acos(cos_angle(s));

% Convert the angle to degrees
patella.VMOVec.Theta.yz(s) = rad2deg(angle_rad(s));
end

for s=1:f
    if patella.VMOVec.Theta.yz(s)>90
        patella.VMOVec.Theta.yz(s)=180-patella.VMOVec.Theta.yz(s);
    end
end

for s=1:f
% Define the coordinates of the points
A(s,:) = [patella.centre_tform(s,3) patella.centre_tform(s,1)];
B(s,:)= [patella.VMO_tform(s,3), patella.VMO_tform(s,1)];
C(s,:) = [thigh.VMO.D_tform(s,3),thigh.VMO.D_tform(s,1) ];

% Define the vectors AB and BC
AB(s,:) = B(s,:) - A(s,:);
BC(s,:) =  B(s,:)-C(s,:);

% Calculate the dot products
dot_AB_BC(s) = dot(AB(s,:), BC(s,:));

% Calculate the magnitudes of the vectors
magnitude_AB(s) = norm(AB(s,:));
magnitude_BC(s) = norm(BC(s,:));

% Calculate the cosine of the angle between the vectors using the dot product formula
```

```
cos_angle(s) = dot_AB_BC(s) / (magnitude_AB(s) * magnitude_BC(s));

% Calculate the angle in radians
angle_rad(s) = acos(cos_angle(s));

% Convert the angle to degrees
patella.VMOVec.Theta.zx(s) = rad2deg(angle_rad(s));
end

for s=1:f
    if patella.VMOVec.Theta.zx(s)>90
        patella.VMOVec.Theta.zx(s)=180-patella.VMOVec.Theta.zx(s);
    end
end

%% QT
for s=1:f
% Define the coordinates of the points
A(s,:) = [patella.centre_tform(s,1) patella.centre_tform(s,2)];
B(s,:)= [patella.base_tform(s,1), patella.base_tform(s,2)];
C(s,:) = [thigh.VL.D_tform(s,1),thigh.VL.D_tform(s,2) ];

% Define the vectors AB and BC
AB(s,:) = B(s,:) - A(s,:);
BC(s,:) =  B(s,:)-C(s,:);

% Calculate the dot products
dot_AB_BC(s) = dot(AB(s,:), BC(s,:));

% Calculate the magnitudes of the vectors
magnitude_AB(s) = norm(AB(s,:));
magnitude_BC(s) = norm(BC(s,:));

% Calculate the cosine of the angle between the vectors using the dot product formula
cos_angle(s) = dot_AB_BC(s) / (magnitude_AB(s) * magnitude_BC(s));

% Calculate the angle in radians
angle_rad(s) = acos(cos_angle(s));

% Convert the angle to degrees
patella.QT.Theta.xy(s) = rad2deg(angle_rad(s));
end

for s=1:f
    if patella.QT.Theta.xy(s)>90
        patella.QT.Theta.xy(s)=180-patella.QT.Theta.xy(s);
    end
end

for s=1:f
% Define the coordinates of the points
A(s,:) = [patella.centre_tform(s,2) patella.centre_tform(s,3)];
B(s,:)= [patella.base_tform(s,2), patella.base_tform(s,3)];
C(s,:) = [thigh.VL.D_tform(s,2),thigh.VL.D_tform(s,3) ];
```

```
% Define the vectors AB and BC
AB(s,:) = B(s,:) - A(s,:);
BC(s,:) =  B(s,:)-C(s,:);

% Calculate the dot products
dot_AB_BC(s) = dot(AB(s,:), BC(s,:));

% Calculate the magnitudes of the vectors
magnitude_AB(s) = norm(AB(s,:));
magnitude_BC(s) = norm(BC(s,:));

% Calculate the cosine of the angle between the vectors using the dot product formula
cos_angle(s) = dot_AB_BC(s) / (magnitude_AB(s) * magnitude_BC(s));

% Calculate the angle in radians
angle_rad(s) = acos(cos_angle(s));

% Convert the angle to degrees
patella.QT.Theta.yz(s) = rad2deg(angle_rad(s));
end

for s=1:f
    if patella.QT.Theta.yz(s)>90
        patella.QT.Theta.yz(s)=180-patella.QT.Theta.yz(s);
    end
end

for s=1:f
% Define the coordinates of the points
A(s,:) = [patella.centre_tform(s,3) patella.centre_tform(s,1)];
B(s,:)= [patella.base_tform(s,3), patella.base_tform(s,1)];
C(s,:) = [thigh.VL.D_tform(s,3),thigh.VL.D_tform(s,1) ];

% Define the vectors AB and BC
AB(s,:) = B(s,:) - A(s,:);
BC(s,:) =  B(s,:)-C(s,:);

% Calculate the dot products
dot_AB_BC(s) = dot(AB(s,:), BC(s,:));

% Calculate the magnitudes of the vectors
magnitude_AB(s) = norm(AB(s,:));
magnitude_BC(s) = norm(BC(s,:));

% Calculate the cosine of the angle between the vectors using the dot product formula
cos_angle(s) = dot_AB_BC(s) / (magnitude_AB(s) * magnitude_BC(s));

% Calculate the angle in radians
angle_rad(s) = acos(cos_angle(s));

% Convert the angle to degrees
patella.QT.Theta.zx(s) = rad2deg(angle_rad(s));
end
```

```
for s=1:f
    if patella.QT.Theta.zx(s)>90
        patella.QT.Theta.zx(s)=180-patella.QT.Theta.zx(s);
    end
end

%% PL
for s=1:f
% Define the coordinates of the points
A(s,:) = [patella.centre_tform(s,1) patella.centre_tform(s,2)];
B(s,:)= [patella.apex_tform(s,1), patella.apex_tform(s,2)];
C(s,:) = [shank.tt_tform(s,1),shank.tt_tform(s,2) ];

% Define the vectors AB and BC
AB(s,:) = B(s,:) - A(s,:);
BC(s,:) =  B(s,:)-C(s,:);

% Calculate the dot products
dot_AB_BC(s) = dot(AB(s,:), BC(s,:));

% Calculate the magnitudes of the vectors
magnitude_AB(s) = norm(AB(s,:));
magnitude_BC(s) = norm(BC(s,:));

% Calculate the cosine of the angle between the vectors using the dot product formula
cos_angle(s) = dot_AB_BC(s) / (magnitude_AB(s) * magnitude_BC(s));

% Calculate the angle in radians
angle_rad(s) = acos(cos_angle(s));

% Convert the angle to degrees
patella.PL.Theta.xy(s) = rad2deg(angle_rad(s));
end

for s=1:f
    if patella.PL.Theta.xy(s)>90
        patella.PL.Theta.xy(s)=180-patella.PL.Theta.xy(s);
    end
end

for s=1:f
% Define the coordinates of the points
A(s,:) = [patella.centre_tform(s,2) patella.centre_tform(s,3)];
B(s,:)= [patella.apex_tform(s,2), patella.apex_tform(s,3)];
C(s,:) = [shank.tt_tform(s,2),shank.tt_tform(s,3) ];

% Define the vectors AB and BC
AB(s,:) = B(s,:) - A(s,:);
BC(s,:) =  B(s,:)-C(s,:);

% Calculate the dot products
dot_AB_BC(s) = dot(AB(s,:), BC(s,:));
```

```
% Calculate the magnitudes of the vectors
magnitude_AB(s) = norm(AB(s,:));
magnitude_BC(s) = norm(BC(s,:));

% Calculate the cosine of the angle between the vectors using the dot product formula
cos_angle(s) = dot_AB_BC(s) / (magnitude_AB(s) * magnitude_BC(s));

% Calculate the angle in radians
angle_rad(s) = acos(cos_angle(s));

% Convert the angle to degrees
patella.PL.Theta.yz(s) = rad2deg(angle_rad(s));
end

for s=1:f
    if patella.PL.Theta.yz(s)>90
        patella.PL.Theta.yz(s)=180-patella.PL.Theta.yz(s);
    end
end

for s=1:f
% Define the coordinates of the points
A(s,:) = [patella.centre_tform(s,3) patella.centre_tform(s,1)];
B(s,:)= [patella.apex_tform(s,3), patella.apex_tform(s,1)];
C(s,:) = [shank.tt_tform(s,3),shank.tt_tform(s,1) ];

% Define the vectors AB and BC
AB(s,:) = B(s,:) - A(s,:);
BC(s,:) =  B(s,:)-C(s,:);

% Calculate the dot products
dot_AB_BC(s) = dot(AB(s,:), BC(s,:));

% Calculate the magnitudes of the vectors
magnitude_AB(s) = norm(AB(s,:));
magnitude_BC(s) = norm(BC(s,:));

% Calculate the cosine of the angle between the vectors using the dot product formula
cos_angle(s) = dot_AB_BC(s) / (magnitude_AB(s) * magnitude_BC(s));

% Calculate the angle in radians
angle_rad(s) = acos(cos_angle(s));

% Convert the angle to degrees
patella.PL.Theta.zx(s) = rad2deg(angle_rad(s));
end

for s=1:f
    if patella.PL.Theta.zx(s)>90
        patella.PL.Theta.zx(s)=180-patella.PL.Theta.zx(s);
    end
end
patella.PL.MAV(:,1) = patella.PL.ryz.*sind(patella.PL.Theta.yz);
patella.PL.MAV(:,2) = patella.PL.rzx.*sind(patella.PL.Theta.zx);
```

```
patella.PL.MAV(:,3) = patella.PL.rxy.*sind(patella.PL.Theta.xy);
patella.QT.MAV(:,1) = patella.QT.ryz.*sind(patella.QT.Theta.yz);
patella.QT.MAV(:,2) = patella.QT.rzx.*sind(patella.QT.Theta.zx);
patella.QT.MAV(:,3) = patella.QT.rxy.*sind(patella.QT.Theta.xy);
patella.VMOVec.MAV(:,1) = patella.VMOVec.ryz.*sind(patella.VMOVec.Theta.yz);
patella.VMOVec.MAV(:,2) = patella.VMOVec.rzx.*sind(patella.VMOVec.Theta.zx);
patella.VMOVec.MAV(:,3) = patella.VMOVec.rxy.*sind(patella.VMOVec.Theta.xy);

   VMOVec.yz=zeros(f,1);
VMOVec.zx=zeros(f,1);
VMOVec.xy=zeros(f,1);

QT.yz=zeros(f,1);
QT.zx=zeros(f,1);
QT.xy=zeros(f,1);
PL.yz=zeros(f,1);
PL.zx=zeros(f,1);
PL.xy=zeros(f,1);


for s=1:f
    VMOVec.xy(s)=patella.VMOVec.MAV(s,3)*patella.VMOVec.ij(s);
    VMOVec.yz(s)=patella.VMOVec.MAV(s,1)*patella.VMOVec.jk(s);
    VMOVec.zx(s)=patella.VMOVec.MAV(s,2)*patella.VMOVec.ki(s);
end
for s=1:f
   QT.xy(s)=patella.QT.MAV(s,3)*patella.QT.ij(s);
    QT.yz(s)=patella.QT.MAV(s,1)*patella.QT.jk(s);
    QT.zx(s)=patella.QT.MAV(s,2)*patella.QT.ki(s);
    PL.xy(s)=patella.PL.MAV(s,3)*patella.PL.Forcexy(s);
    PL.yz(s)=patella.PL.MAV(s,1)*patella.PL.Forceyz(s);
    PL.zx(s)=patella.PL.MAV(s,2)*patella.PL.Forcezx(s);


end


 %%
 %substitution equation
 patella.QT.TotalForce=zeros(f,1);
 patella.VMOVec.TotalForce=zeros(f,1);
%  thigh.VMO.percent=zeros(f,1);
%  PCSATotal=zeros(f,1);
%  thigh.VL.PCSA=zeros(f,1);
% thigh.VI.PCSA=zeros(f,1);
% thigh.RF.PCSA=zeros(f,1);
% thigh.VMO.PCSA=zeros(f,1);
% thigh.VML.percent=zeros(f,1);
% thigh.VML.PCSA=zeros(f,1);
% TEST=zeros(f,1);
 value1=VMOVec.yz.*PL.xy./VMOVec.xy-PL.yz;
 value2=QT.yz-VMOVec.yz.*QT.xy./VMOVec.xy;

 for s=1:f
```

```
  patella.QT.TotalForce(s)=abs(value1(s)/value2(s));
patella.VMOVec.TotalForce(s)=abs((-PL.xy(s,:)-
QT.xy(s,:)*patella.QT.TotalForce(s,:))/VMOVec.xy(s,:));
 end


%%
%splitting of forces into individual components
    patella.VMOVec.force(:,1)=abs(patella.VMOVec.TotalForce).*patella.VMOVec.i(:,1);
    patella.VMOVec.force(:,2)=abs(patella.VMOVec.TotalForce).*patella.VMOVec.j(:,1);
    patella.VMOVec.force(:,3)=abs(patella.VMOVec.TotalForce).*patella.VMOVec.k(:,1);
     patella.QT.force(:,1)=abs(patella.QT.TotalForce).*patella.QT.i(:,1);
    patella.QT.force(:,2)=abs(patella.QT.TotalForce).*patella.QT.j(:,1);
    patella.QT.force(:,3)=abs(patella.QT.TotalForce).*patella.QT.k(:,1);
    patella.PL.force=patella.PL.Force;

for s=1:f


patella.VMOVec.forcetransformed(s,:)=(patella.rotationmatrix(:,:,s)'*patella.VMOVec.f
orce(s,:)');

patella.QT.forcetransformed(s,:)=(patella.rotationmatrix(:,:,s)'*patella.QT.force(s,:
)');

patella.PL.forcetransformed(s,:)=(patella.rotationmatrix(:,:,s)'*patella.PL.force(s,:
)');
end

%moment created by each muscle
    VMOVec.moment(:,1)=VMOVec.yz.*patella.VMOVec.TotalForce;
    VMOVec.moment(:,2)=VMOVec.zx.*patella.VMOVec.TotalForce;
    VMOVec.moment(:,3)=VMOVec.xy.*patella.VMOVec.TotalForce;
    QT.moment(:,1)=QT.yz.*patella.QT.TotalForce;
    QT.moment(:,2)=QT.zx.*patella.QT.TotalForce;
    QT.moment(:,3)=QT.xy.*patella.QT.TotalForce;
    PL.moment(:,1)=PL.yz;
    PL.moment(:,2)=PL.zx;
    PL.moment(:,3)=PL.xy;
%sum of moments
MOMENTx=(QT.yz.*patella.QT.TotalForce)+(VMOVec.yz.*patella.VMOVec.TotalForce)+PL.yz;%
should come out to 0
MOMENTy=(QT.zx.*patella.QT.TotalForce)+(VMOVec.zx.*patella.VMOVec.TotalForce)+PL.zx;
MOMENTz=(QT.xy.*patella.QT.TotalForce)+(VMOVec.xy.*patella.VMOVec.TotalForce)+PL.xy;

thigh.VML.TotalForce=patella.QT.TotalForce.*thigh.QT.VMLPercent;
thigh.VI.TotalForce=patella.QT.TotalForce.*thigh.QT.VIPercent;
thigh.VL.TotalForce=patella.QT.TotalForce.*thigh.QT.VLPercent;
thigh.RF.TotalForce=patella.QT.TotalForce.*thigh.QT.RFPercent;
thigh.VML.Force=patella.QT.force(:,:)*thigh.QT.VMLPercent;
thigh.VI.Force=patella.QT.force(:,:)*thigh.QT.VIPercent;
thigh.VL.Force=patella.QT.force(:,:)*thigh.QT.VLPercent;
thigh.RF.Force=patella.QT.force(:,:)*thigh.QT.RFPercent;
```

```
% Calculate the absolute differences
differences = abs(shank.jointangle(:,1) - KJangle);

% Find the index of the minimum difference
[~, index] = min(differences);
%%
%Have to find the proportions of force before I can put them in here.
%Step 3- Cylindrical Coordinate System
%VML
%Coordinates in thigh system will be designated as THIGH
for s=1:f

THIGH.VL.D(s,:)=(thigh.VL.D_tform(s,:)-
thigh.pep_tform(s,:))*thigh.rotationmatrix(:,:,s);
THIGH.VL.P(s,:)=(thigh.VL.P_tform(s,:)-
thigh.pep_tform(s,:))*thigh.rotationmatrix(:,:,s);
THIGH.VML.D(s,:)=(thigh.VML.D_tform(s,:)-
thigh.pep_tform(s,:))*thigh.rotationmatrix(:,:,s);
THIGH.VML.P(s,:)=(thigh.VML.P_tform(s,:)-
thigh.pep_tform(s,:))*thigh.rotationmatrix(:,:,s);
THIGH.VI.D(s,:)=(thigh.VI.D_tform(s,:)-
thigh.pep_tform(s,:))*thigh.rotationmatrix(:,:,s);
THIGH.VI.P(s,:)=(thigh.VI.P_tform(s,:)-
thigh.pep_tform(s,:))*thigh.rotationmatrix(:,:,s);
THIGH.VMO.D(s,:)=(thigh.VMO.D_tform(s,:)-
thigh.pep_tform(s,:))*thigh.rotationmatrix(:,:,s);
THIGH.VMO.P(s,:)=(thigh.VMO.P_tform(s,:)-
thigh.pep_tform(s,:))*thigh.rotationmatrix(:,:,s);

end
%% VML
C_height=THIGH.VML.P(:,3)-THIGH.VML.D(:,3);
C_length=[THIGH.VML.P(:,1) THIGH.VML.P(:,2) C_height];
r=sqrt(C_length(:,1).^2+C_length(:,2).^2);
C_vector=sqrt(C_length(:,1).^2+C_length(:,2).^2+C_length(:,3).^2);
i=C_length(:,1)./abs(C_vector);
j=C_length(:,2)./abs(C_vector);
k=C_length(:,3)./abs(C_vector);
theta=atan2d(C_length(:,2),C_length(:,1));
Utheta=-(i.*cosd(theta)+j.*sind(theta));
Ur=-i.*sind(theta)+j.*cos(theta);
Uz=k;
for s=1:f
 thigh.VML.Momentz(s,:)=r(s)*Utheta(s,:)*thigh.VML.TotalForce(s,:);

 % how do I find the X and Y moments for the femur here?

end
% disp('VML Max Moment')
% disp(max(abs(thigh.VML.Momentz)))

    %% VL
    C_height=THIGH.VL.P(:,3)-THIGH.VL.D(:,3);
```

```
C_length=[THIGH.VL.P(:,1) THIGH.VL.P(:,2) C_height];
r=sqrt(C_length(:,1).^2+C_length(:,2).^2);
C_vector=sqrt(C_length(:,1).^2+C_length(:,2).^2+C_length(:,3).^2);
i=C_length(:,1)./abs(C_vector);
j=C_length(:,2)./abs(C_vector);
k=C_length(:,3)./abs(C_vector);
theta=atan2d(C_length(:,2),C_length(:,1));
Utheta=i.*cosd(theta)+j.*sind(theta);
Ur=-i.*sind(theta)+j.*cos(theta);
Uz=k;

for s=1:f
 thigh.VL.Momentz(s,:)=r(s)*Utheta(s,:)*thigh.VL.TotalForce(s,:);

end
% disp('VL Max Moment')
% disp(max(abs(thigh.VL.Momentz)))
    %% VI
C_height=THIGH.VI.P(:,3)-THIGH.VI.D(:,3);
C_length=[THIGH.VI.P(:,1) THIGH.VI.P(:,2) C_height];
r=sqrt(C_length(:,1).^2+C_length(:,2).^2);
C_vector=sqrt(C_length(:,1).^2+C_length(:,2).^2+C_length(:,3).^2);
i=C_length(:,1)./abs(C_vector);
j=C_length(:,2)./abs(C_vector);
k=C_length(:,3)./abs(C_vector);
theta=atan2d(C_length(:,2),C_length(:,1));
Utheta=i.*cosd(theta)+j.*sind(theta);
Ur=-i.*sind(theta)+j.*cos(theta);
Uz=k;
for s=1:f
 thigh.VI.Momentz(s,:)=r(s)*Utheta(s,:)*thigh.VI.TotalForce(s,:);

end
% disp('VI Max Moment')
% disp(max(abs(thigh.VI.Momentz)))
    %% VMO
C_height=THIGH.VMO.P(:,3)-THIGH.VMO.D(:,3);
C_length=[THIGH.VMO.P(:,1) THIGH.VMO.P(:,2) C_height];
r=sqrt(C_length(:,1).^2+C_length(:,2).^2);
C_vector=sqrt(C_length(:,1).^2+C_length(:,2).^2+C_length(:,3).^2);
i=C_length(:,1)./abs(C_vector);
j=C_length(:,2)./abs(C_vector);
k=C_length(:,3)./abs(C_vector);
theta=atan2d(C_length(:,2),C_length(:,1));
Utheta=-(i.*cosd(theta)+j.*sind(theta));
Ur=-i.*sind(theta)+j.*cos(theta);
Uz=k;
for s=1:f
 thigh.VMO.Momentz(s,:)=r(s)*Utheta(s,:)*patella.VMOVec.TotalForce(s,:);

end
%
% disp('VMO Max Moment')
% disp(max(abs(thigh.VMO.Momentz)))
```

```
%%



Moment.one=resample(trial(1).Data.Moment.Right(:,4), 1000, 200);
Moment.two=resample(trial(2).Data.Moment.Right(:,4), 1000, 200);
Moment.three=resample(trial(3).Data.Moment.Right(:,4), 1000, 200);
Moment.four=resample(trial(4).Data.Moment.Right(:,4), 1000, 200);
Moment.five=resample(trial(5).Data.Moment.Right(:,4), 1000, 200);
Moment.six=resample(trial(6).Data.Moment.Right(:,4), 1000, 200);
%%
%This occurs at 90 degrees
Moment.max= max([max(Moment.four), max(Moment.five), max(Moment.six)]);
Moment.proportion.one=(max(Moment.one))/Moment.max;
Moment.proportion.two=max(Moment.two)/Moment.max;
Moment.proportion.three=max(Moment.three)/Moment.max;
%Degrees are measured per individual. Moment is in the x, so we are
%dividing by the ryz.
% Use interpolation to find the moment arm at the desired angle
MA = interp1(shank.jointangle(:,1), shank.PL.MAV(:,1), KJangle);
Force=Moment.max/MA;

PCSATotal=max(thigh.VL.TotalForce+thigh.VI.TotalForce+thigh.RF.TotalForce+thigh.VML.T
otalForce+patella.VMOVec.TotalForce)./25;
thigh.VL.PCSA=max(thigh.VL.TotalForce)/25;
thigh.VI.PCSA=max(thigh.VI.TotalForce)/25;
thigh.RF.PCSA=max(thigh.RF.TotalForce)/25;
thigh.VMO.PCSA=max(patella.VMOVec.TotalForce./25);
thigh.VMO.percent=thigh.VMO.PCSA./PCSATotal;
thigh.VML.percent=(PCSATotal-thigh.VL.PCSA-thigh.VI.PCSA-thigh.RF.PCSA-
thigh.VMO.PCSA)/PCSATotal;
thigh.VML.PCSA=PCSATotal.*(thigh.VML.percent);
TEST=(thigh.VML.PCSA+thigh.VL.PCSA+thigh.VI.PCSA+thigh.VMO.PCSA+thigh.RF.PCSA)/PCSATo
tal;
% Find the indices of elements that equal the specified value


% % Display the index
% disp('Index of the closest value:');
% disp(index);
%
% % Display the closest value
% disp('Closest value:');
% disp(shank.jointangle(index,1));
% disp(patella.QT.TotalForce(index,1)+patella.VMOVec.TotalForce(index,1));
variable=Force/(patella.QT.TotalForce(index,1)+patella.VMOVec.TotalForce(index,1));
EstimatedForce=(patella.QT.TotalForce+patella.VMOVec.TotalForce)*variable;

QTProp=patella.QT.TotalForce./(patella.QT.TotalForce+patella.VMOVec.TotalForce);
VMOProp=patella.VMOVec.TotalForce./(patella.QT.TotalForce+patella.VMOVec.TotalForce);
Estimated.VMO=[EstimatedForce.*VMOProp.*patella.VMOVec.i
EstimatedForce.*VMOProp.*patella.VMOVec.j EstimatedForce.*VMOProp.*patella.VMOVec.k];
```

```
Estimated.QT=[EstimatedForce.*QTProp.*patella.QT.i
EstimatedForce.*QTProp.*patella.QT.j EstimatedForce.*QTProp.*patella.QT.k];
Estimated.VL=Estimated.QT.*thigh.QT.VLPercent;
Estimated.VI=Estimated.QT.*thigh.QT.VIPercent;
Estimated.VML=Estimated.QT.*thigh.QT.VMLPercent;
Estimated.RF=Estimated.QT.*thigh.QT.RFPercent;

% Display the indices

JumpForceKJ.VL=thigh.VL.Force(index,:);
JumpForceKJ.VML=thigh.VML.Force(index,:);
JumpForceKJ.VI=thigh.VI.Force(index,:);
JumpForceKJ.RF=thigh.RF.Force(index,:);
JumpForceKJ.VMO=patella.VMOVec.force(index,:);
JumpForceKJ.Total=JumpForceKJ.VL+JumpForceKJ.VI+JumpForceKJ.VML+JumpForceKJ.RF+JumpFo
rceKJ.VMO;
%next we want to divide up the forces into the muscles


%save P9EstimationOrange Estimated

%This is only at the maximum space and only in the yz plane
% FPL=MaxMoment/rKJC;


clearvars -except r Utheta mass PCSATotal Colour Participant THIGH thigh shank
patella Moment f MOMENTx MOMENTy MOMENTz PL QT VMOVec DEP_TH DEP_SH DEP_PAT PEP_TH
PEP_SH PEP_PAT Hip_SA Force KJangle JumpForceKJ E


%
% Define the folder path where you want to save the variables
folderPath = 'C:\MATLAB\Bone Modeling\figures';

% Check if the folder exists, if not, create it
if ~exist(folderPath, 'dir')
   mkdir(folderPath);
end
if Colour==1
% Define the file name for the .mat file
if Participant ==9
P9.Red = 'P9.Red.mat';
% Construct the full file path
filePath = fullfile(folderPath, P9.Red);

% Save all variables in the current workspace to the specified file
save(filePath);

elseif Participant ==10
   P10.Red='P10.Red.mat' ;
   % Construct the full file path
filePath = fullfile(folderPath, P10.Red);

% Save all variables in the current workspace to the specified file
```

```
save(filePath);

   elseif Participant ==11
P11.Red = 'P11.Red.mat';
% Construct the full file path
filePath = fullfile(folderPath, P11.Red);

% Save all variables in the current workspace to the specified file
save(filePath);

elseif Participant ==12
P12.Red = 'P12.Red.mat';
% Construct the full file path
filePath = fullfile(folderPath, P12.Red);

% Save all variables in the current workspace to the specified file
save(filePath);

elseif Participant ==13
P13.Red = 'P13.Red.mat';
% Construct the full file path
filePath = fullfile(folderPath, P13.Red);

% Save all variables in the current workspace to the specified file
save(filePath);

elseif Participant ==14
P14.Red = 'P14.Red.mat';
% Construct the full file path
filePath = fullfile(folderPath, P14.Red);

% Save all variables in the current workspace to the specified file
save(filePath);

elseif Participant ==15
P15.Red = 'P15.Red.mat';
% Construct the full file path
filePath = fullfile(folderPath, P15.Red);

% Save all variables in the current workspace to the specified file
save(filePath);

elseif Participant ==16
P16.Red = 'P16.Red.mat';
% Construct the full file path
filePath = fullfile(folderPath, P16.Red);

% Save all variables in the current workspace to the specified file
save(filePath);

elseif Participant ==17
P17.Red = 'P17.Red.mat';
% Construct the full file path
filePath = fullfile(folderPath, P17.Red);
```

```
% Save all variables in the current workspace to the specified file
save(filePath);

elseif Participant ==18
P18.Red = 'P18.Red.mat';
% Construct the full file path
filePath = fullfile(folderPath, P18.Red);

% Save all variables in the current workspace to the specified file
save(filePath);

end
elseif Colour==2


if Participant ==9
P9.Blue = 'P9.Blue.mat';
% Construct the full file path
filePath = fullfile(folderPath, P9.Blue);

% Save all variables in the current workspace to the specified file
save(filePath);

elseif Participant ==10
   P10.Blue='P10.Blue.mat' ;
% Construct the full file path
filePath = fullfile(folderPath, P10.Blue);

% Save all variables in the current workspace to the specified file
save(filePath);

elseif Participant ==11
P11.Blue = 'P11.Blue.mat';
% Construct the full file path
filePath = fullfile(folderPath, P11.Blue);

% Save all variables in the current workspace to the specified file
save(filePath);

elseif Participant ==12
P12.Blue = 'P12.Blue.mat';
% Construct the full file path
filePath = fullfile(folderPath, P12.Blue);

% Save all variables in the current workspace to the specified file
save(filePath);

elseif Participant ==13
P13.Blue = 'P13.Blue.mat';
% Construct the full file path
filePath = fullfile(folderPath, P13.Blue);

% Save all variables in the current workspace to the specified file
```

```
save(filePath);

elseif Participant ==14
P14.Blue = 'P14.Blue.mat';
% Construct the full file path
filePath = fullfile(folderPath, P14.Blue);

% Save all variables in the current workspace to the specified file
save(filePath);

elseif Participant ==15
P15.Blue = 'P15.Blue.mat';
% Construct the full file path
filePath = fullfile(folderPath, P15.Blue);

% Save all variables in the current workspace to the specified file
save(filePath);

elseif Participant ==16
P16.Blue = 'P16.Blue.mat';
% Construct the full file path
filePath = fullfile(folderPath, P16.Blue);

% Save all variables in the current workspace to the specified file
save(filePath);

elseif Participant ==17
P17.Blue = 'P17.Blue.mat';
% Construct the full file path
filePath = fullfile(folderPath, P17.Blue);

% Save all variables in the current workspace to the specified file
save(filePath);

elseif Participant ==18
P18.Blue = 'P18.Blue.mat';
% Construct the full file path
filePath = fullfile(folderPath, P18.Blue);

% Save all variables in the current workspace to the specified file
save(filePath);

end
elseif Colour==3
    if Participant ==9
P9.Orange = 'P9.Orange.mat';
% Construct the full file path
filePath = fullfile(folderPath, P9.Orange);

% Save all variables in the current workspace to the specified file
save(filePath);

elseif Participant ==10
   P10.Orange='P10.Orange.mat' ;
```

```
% Construct the full file path
filePath = fullfile(folderPath, P10.Orange);

% Save all variables in the current workspace to the specified file
save(filePath);

    elseif Participant ==11
P11.Orange = 'P11.Orange.mat';
% Construct the full file path
filePath = fullfile(folderPath, P11.Orange);

% Save all variables in the current workspace to the specified file
save(filePath);

    elseif Participant ==12
P12.Orange = 'P12.Orange.mat';
% Construct the full file path
filePath = fullfile(folderPath, P12.Orange);

% Save all variables in the current workspace to the specified file
save(filePath);

    elseif Participant ==13
P13.Orange = 'P13.Orange.mat';
% Construct the full file path
filePath = fullfile(folderPath, P13.Orange);

% Save all variables in the current workspace to the specified file
save(filePath);

    elseif Participant ==14
P14.Orange = 'P14.Orange.mat';
% Construct the full file path
filePath = fullfile(folderPath, P14.Orange);

% Save all variables in the current workspace to the specified file
save(filePath);

    elseif Participant ==15
P15.Orange = 'P15.Orange.mat';
% Construct the full file path
filePath = fullfile(folderPath, P15.Orange);

% Save all variables in the current workspace to the specified file
save(filePath);

    elseif Participant ==16
P16.Orange = 'P16.Orange.mat';
% Construct the full file path
filePath = fullfile(folderPath, P16.Orange);

% Save all variables in the current workspace to the specified file
save(filePath);
```

```
    elseif Participant ==17
P17.Orange = 'P17.Orange.mat';
% Construct the full file path
filePath = fullfile(folderPath, P17.Orange);

% Save all variables in the current workspace to the specified file
save(filePath);

  elseif Participant ==18
P18.Orange = 'P18.Orange.mat';
% Construct the full file path
filePath = fullfile(folderPath, P18.Orange);

% Save all variables in the current workspace to the specified file
save(filePath);
    end
end
%%
```

Appendix K Data Management Plan

3D Musculoskeletal Modeling of the Human Lower Extremity-Thesis

A Data Management Plan created using DMP Assistant

Creator: Jasmine Feddema

Data Manager: Lauryn Tremblay, Claire Hopkins, Jane Hopkins, Shelby Schmidt

Project Administrator: Dr. Loren Chiu

Affiliation: University of Alberta

Template: University of Alberta Template

Project abstract:

It is known that the quadriceps extend the leg at the thigh, and the thigh when the leg is fixed. But the wrapping of the vasti muscles around the femur suggests a transverse plane action upon the bone. A muscle of particular importance is the vastus medialis, which can be separated into the vastus medialis longus (VML) and the vastus medialis oblique (VMO). These findings stem from force analysis of the muscle but ignore the specific femoral geometry. To determine the action of these muscles in further detail, the bone's geometry must be known. The general shape of the femur directing medially from the hip to the knee, accompanied by parallel acting muscles of the vasti, already indicates that a transverse action of the vasti can be implemented at the knee. However, further knowledge of the specificity of bone geometry may inform a more complete understanding of the function. To investigate this problem, bones and muscle measurement parameters will be obtained from cadaveric specimens. Three-dimensional (3D) models of human bones from the specimens will be developed using photogrammetry software. The measurements of muscular parameters will inform mapping of the vasti muscles onto the generated 3D model. Muscle measurements combined with bony geometry will allow us to conduct force analysis of the vasti muscles. The goal of the project is to improve upon previous computerized models and elucidate the function of the VMO muscle upon the knee joint. Furthermore, it may aid in the treatment of knee pathologies.

Identifier: 10891

Start date: 01-09-2022

End date: 01-01-2024

Last modified: 30-03-2023

Data Management

Data Collection

What types of data will you collect, create, acquire and/or record?

Photogrammetry models of bones will be created using previously taken images of the bones. Three-dimensional models will be inputted into a computer automated design program. Muscular measurements of cadavers will be used to map musculature onto the geometry of the bone and create animated movement. Force generation of the individual muscles and their directions will be recorded.

What file formats will your data be collected in? Will these formats allow for data re-use, sharing and long-term access to the data?

File formats of photogrammetry models will be saved in a Metashape Project file type, requiring access to the software to open the files. Force models will be saved within the chosen computer program, requiring access to the program for access to the models created. For the purposes of the project, it is impossible to save files publicly, creating barriers to re-use. However, images of the created models will be made accessible in the written output.

If data is collected using laptops or mobile devices, please explain how you will securely store and transfer the data.

N/A

How much data do you anticipate collecting? Include an estimate of how much storage space you will require (in megabytes, gigabytes, terabytes). This estimate should also take into account storage space required for file versioning, backups, and the growth rate over time.

The photos taken and reconstructions developed will require large amounts of computer space. It is anticipated that more than fifty gigabytes of storage will be needed to comfortably contain all the data. An external, one terabyte, hard drive will be obtained to satisfy this requirement. In the future, more space may be needed, making a dual external hard drive base connector ideal for containment of information.

Are there any existing data that you can re-use? If so, please explain how you will obtain that data and integrate it into your research project.

Existing photos taken of cadaveric specimens and various objects will be used to create the protocols necessary for generation of digital models, and subsequently the models themselves. Data was created within the current lab, so will be easily obtained.

What conventions and procedures will you use to structure, name and version control your files to ensure that your data is well-organized?

The data will be organized by specimen number, bone name, and date if multiple models of the same bone are created. If the models are created on the same day, defining characteristics will be added to differentiate the files. Photo ranges will be documented upon capture to ensure proper collection of photos into appropriate folders. Photos will be offloaded into the computer external hard drive immediately following capture to prevent confusion with photo ranges.

Documentation and Metadata

What documentation will be needed for the data to be read and interpreted correctly in the future? This includes study-level documentation, data-level description, and any other contextual information required to make the data usable by other researchers.

The methods for procurement of the data will be described in full. The purpose of the study is to make the procedures for development of models accessible to anybody, therefore the detailed stepwise procedure will be published.

Please list the metadata standard and tools you will use to document and describe your data. If there is not an appropriate standard, please explain how you will ensure consistency in your documentation.

Unsure currently

How will you make sure that documentation is created or captured consistently throughout your project?

Documentation procedures are agreed upon by all members associated with the research group, and are followed. Frequent checks are employed upon the documentation to ensure consistency among files.

Storage and Backup

How will your data be stored and backed up during your research project?

Data will be stored upon a one terabyte external hard drive to allow for the transfer of data between computers.

How will you ensure that sensitive data is stored securely and only accessible to the research team during the research project?

Sensitive data will be stored on a hard drive in a locked office only accessible by members of the research team.

Preservation

Which data are selected for preservation and access will depend on potential reuse value, whether there are obligations to either retain or destroy data, and the resources required to properly curate the data and ensure that it remains usable in the future. In some circumstances, it may be feasible to preserve all versions of the data (e.g. raw, processed, analyzed, final), but in others, it may be preferable to keep only selected data (e.g. transcripts instead of audio interviews).

All three-dimensional models will be preserved. Photographs previously stored on the hard drive may be destroyed if usable 3D models have been created and are deemed successful.

At the end of your research project, where will you deposit your data for long-term preservation and access?

Data will remain on the hard drive for continuation of the project into new iterations.

Please describe how you will prepare the data for preservation and access, including any necessary procedures for data cleaning, normalization or de-identification. Explain how you will prevent data from being lost while processing and converting files.

Unsure currently.

Data Sharing and Reuse

What data will you be sharing and in what form? (e.g., raw, processed, analyzed, final). Consider which data may need to be shared in order to meet institutional or funding requirements, and which data may be restricted because of confidentiality/privacy issues.

Photos, reconstructions, and models will be shared to demonstrate outcomes. Necessary ethics requirements have been met to share information. Any information pertaining to the cadaveric specimens other than sex and side of dissection will be confidential.

How will you be sharing your data? (e.g., institutional repository, a specialized data archive, project website, informal/on-request sharing). Include a brief description of any resources needed to share your data (equipment, systems, expertise, etc.).

Data will be shared in the form of a Master Thesis manuscript. To see the data created, the user would most likely need access to the computer programs that the file originates from.

Please describe whether there will be any restrictions placed on your data when they are made available and who may access them. If data is not openly available, describe the process for gaining access.

Given the sensitivity of using human remains, permission will be required for users to access data directly. Users will be required to contact the research group for permission, and state how they plan to use this information. In the case of educational use, the likelihood of approval will be much higher than for personal interest. However, the process of how to create their own data will be made readily available as per the goals of the project.

What type of end-user license will you include with your data? Please include a copy of this license with your Data Management Plan.

Unsure currently

Responsibilities and Resources

Who will be responsible for data management during the project? (i.e., during collection, processing, analysis, documentation). Identify staff and organizational roles and their responsibilities for carrying out the data management plan (DMP), including time allocations and training requirements.

I will be primarily responsible for management of the data. The principal organizer, Dr. Loren Chiu, may also contribute to data collection and documentation. Specially chosen undergraduate students will also be involved in data collection, with one specific student trained to document information using the protocols developed by the research team. Chosen undergraduate students have an education in anatomy, biomechanics, and basic research protocols to be considered apt for the position.

What will happen when personnel changes occur or if the principal investigator leaves the institution before the project has concluded?

There is intent for future graduate students to continue the work started here, and progress to other parts of the body. New personnel will be updated on the current data management plan and encouraged to continue working within those procedures.

Who will be responsible for data sharing and preservation after the project has concluded? Indicate the party who will have primary responsibility for how the data will persist over time when the original personnel have moved on.

Dr. Loren Chiu will have primary responsibility for the data collected and may continue the project using the collected data.

What resources will you require to implement your plan? Will extra people, time or hardware, storage be required? How much will this cost (estimation)?

This project is unfunded at this time and works to implement procedures with minimum costs. The use of Metashape software costs $60. The labour of undergraduate students is voluntary.

And the external hard drive may cost $70. The chosen computer program will be free of charge. In preparation of other costs associated, the project budget is $200.

Model representation - "Force Model of the Quadriceps "

Three-dimensional model of the quadriceps upon the lower extremity bones. Force value inputs allow the model to be automated, with muscles acting upon the bones according to the force-length relationship and joint angle.

Text - "3D Musculoskeletal Modeling of the Human Lower Extremity"

Thesis manuscript for the purposes of obtaining a graduate degree in science with specialization in kinesiology, anatomy, and biomechanics.

Planned research output details.

Title	Type	Anticipated release date	Initial access level	Intended repository(ies)	Anticipated file size	License	Metadata standard(s)	May contain sensitive data?	May contain PII?
Lump Sum Model	Image	Unspecified	Open	None specified		None specified	None specified	No	No
Force Model of the Quadriceps	Model representation	Unspecified	Open	None specified		None specified	None specified	Yes	No
3D Musculoskeletal Modeling of the Human Lower Ext ...	Text	Unspecified	Open	None specified		None specified	None specified	Yes	No

Appendix L Recruitment Materials

L-1. Recruitment Flyer shared with varsity and faculty communities.

UNIVERSITY OF ALBERTA
FACULTY OF KINESIOLOGY, SPORT, AND RECREATION

VERTICAL JUMPING RESEARCH

This study is being conducted by researchers at the University of Alberta to assess the kinematics and kinetics of the joints when vertical jumping

Males and females may be eligible to participate if they are:

- Between 18 and 35 years old
- Participate in competitive or recreational sports or performance activities involving running and/or jumping

Participation would require:

- One visit to the Sports Biomechanics Laboratory
- The visit will take up to 1 hour

For more information, contact:
Jasmine Feddema, BSc [jcfeddem@ualberta.ca]

Pro00135286
Document Version: September 29, 2023

L-2. Example script for participant recruitment via email or presentation

Email subject: Vertical Jumping Research – Sports Biomechanics Lab

“Hello, we are conducting research to examine joint kinematics and kinetics during vertical jumps. The project title is “3D Musculoskeletal Modelling of the Human Lower Extremity” and has been approved by the University of Alberta Research Ethics Board [ID Pro00135286] The principal investigator for this research is Jasmine Feddema, MSc student, Faculty of Kinesiology, Sport, and Recreation. We are looking for volunteers to participate in this research study. To be eligible, individuals should be between 18 and 35, and physically active, and participating in sports or similar performance activities that involve running and/or jumping. Individuals who have current lower extremity or back injuries cannot participate. Study participants will visit the Sport Biomechanics Laboratory where jumping tasks will be performed and biomechanical measurements taken. One visit is required that will take up to 1 hour.

If you are interested or would like more information, please contact Jasmine Feddema at jcfeddem@ualberta.ca”